Web Knowledge Governance: Legal Knowledge Representation and Automated Compliance Checking

Cumulative Habilitation Thesis
to obtain the license to teach in the subject Business Informatics
(Wirtschaftsinformatik)

Sabrina Kirrane

Department of Information Systems and Operations Management
Vienna University of Economics and Business

Dated: April 28, 2023

Contents

I Preface

II Publications

Part I

Preface

1. Introduction

The world wide web, often simply referred to as the web, has evolved from a read-only medium for information dissemination to a ubiquitous information and communication platform that supports interaction and collaboration internationally. Although the web was originally designed for humans, the web community continuously enhances and extends web standards and technologies such that computer programs (e.g., search engines, services, bots, etc.) can support humans when it comes to navigating and leveraging web resources.

A specific sub field of web research, known as the semantic web, strives towards a web of data that can be exploited by computer programs, referred to as intelligent software web agents, that carry out data-centric tasks on behalf of humans. In this context, agents are goal oriented computer programs that possess the intelligence necessary to autonomously navigate the web in order to perform tasks on behalf of their owners. Although the intelligent software web agents vision has yet to be realised, the community has made significant advances in terms of the standards, tools, and technologies that facilitate the provision and consumption of machine-readable data and knowledge on the web. However, there is a major open challenge when it comes to ensuring that intelligent software web agents are aware of and adhere to various policies and norms. In this context, policies denote data access and usage restrictions, while norms refer to legal requirements that need to be fulfilled. In order to address this gap, there is a need for machine readable legal knowledge representation as well as automated enforcement and compliance checking.

In the following, we provide a high-level overview of the state of the art, with respect to the semantic web, legal knowledge representation, and automated compliance checking, which serves to set the scene for the work described in this cumulative habilitation thesis.

Semantic Web. Semantic web standards, tools, and technologies enable machines to make use of web resources thanks to a semantically rich data model (i.e., the Resource Description Framework (RDF) [59]), modelling languages (i.e., Resource Description Framework Schema (RDFS) [17] and the Web Ontology Language (OWL) [47]), various ontologies and vocabularies that support data integration and exchange (cf., [29, 22]), as well as service discovery and composition (cf., [89, 88]). There is a large body of work which demonstrates how ontologies and/or rules can be used to encode knowledge and support reasoning (cf., [31, 40, 61, 76]) in a variety of domains. Additionally, researchers have demonstrated how various ontology learning techniques can be used to enhance manually crafted ontologies (cf., [91, 77]). In a comprehensive survey on ontology learning techniques and applications, Asim et al. [5] highlight that ontology learning has benefited from a variety of research fields, namely natural language processing, machine learning, information retrieval, data mining, and knowledge representation.

Legal Knowledge Representation. When it comes to the legal domain specifically, researchers have proposed cross-domain ontologies that can be used to encode legal text in a machine-readable format using LegalRuleML [14] and adaptations thereof (cf., [6, 71]). Others focus on facilitating legal document indexing and search using the European Law Identifier (ELI)[1] and the European Case Law Identifier (ECLI)[2] (cf., [68, 21]), or bridging the gap between European Union (EU) and

[1] ELI, `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52012XG1026(01)`

[2] ECLI, `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52011XG0429(01)`

member state legal terminology (cf., [3, 12]). There has also been some work on encoding legal provisions using description logics (cf., [41]). Additionally, researchers have explored the potential of legal ontology design patterns for legal knowledge modelling (cf., [42, 82, 57]). Besides these cross-domain activities, there has also been various domain specific initiatives. For instance, the ELI ontology has be extended in order to facilitate the encoding of the text of the General Data Protection Regulation (GDPR)[3] (cf., [75]). While, others have focused specifically on modelling privacy policies (cf., [69, 72]). The Open Digital Rights Language (ODRL)[4], which is a W3C recommendation, has gained a lot of traction in recent years in terms of intellectual property rights management (cf., [74, 85, 46, 64]). Additionally, the ODRL model and vocabularies have been extended in order to model contracts [45], personal data processing consent (cf., [32]), and data protection regulatory requirements (cf., [73, 26]). There has also been some work on automatically extracting rights and conditions from textual documents (cf., [20, 19]) or extracting important information from legal cases (cf., [92, 65]).

Automated Compliance Checking. Many of the existing legal knowledge representation works focus specifically on encoding knowledge without proposing any compliance checking approaches. In the case of ELI and ECLI specifically researchers primarily focuses on making data available via search engines and software services (cf., [68, 21]). In contrast, LegalRuleML researchers have proposed automated compliance approaches based on auditing (cf., [30, 72]) and business process compliance checking (cf., [73, 8]). Others propose approaches for translating LegalRuleML into defeasible logic such that it can be interpreted by a generic defeasible logic reasoner (cf., [56]. In turn, Governatori et al. [44] shows how LegalRuleML together with Semantic technologies is used for business process regulatory compliance checking based on a rule based logic combining defeasible and deontic logic. One of the advantages of description logic based approaches, when it comes to consistency and compliance checking, is that they are able to leverage generic reasoners, such as Pellet[5] (cf., [41]). Although there are presently no ODRL specific reasoning engines, researchers have demonstrated how ODRL can be translated into rules that can be processed by answer set programming (ASP) [7] solvers such as Clingo [43] (cf., [46, 26]). Additionally, there have been several custom applications that are designed to support ODRL enforcement or compliance checking, such as a license-based search engine [64]; a generalised contract schema and role based access control enforcement mechanism [45]; and an access request matching and authorisation algorithm [32].

The work on web knowledge governance described in this cumulative habilitation thesis has led to the following major research outputs: (i) a better understanding of semantic web research and the gaps that need to filled in order to realise the original intelligent software web agent vision; (ii) approaches for norm encoding and automated compliance checking in line with requirements stipulated in the GDPR; (iii) techniques that cater for automated legal knowledge extraction, encoding and integration that could be used to assess legal compliance more generally; (iv) an approach for assessing the effectiveness of explainable artificial intelligence (xAI) tools that can be used to ensure xAI is usable; and (v) technical measures to support the data integrity and confidentiality principle of the GDPR.

[3] GDPR, https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32016R0679&qid=1681238509224

[4] ODRL, https://www.w3.org/TR/odrl-model/

[5] Pellet, https://github.com/stardog-union/pellet

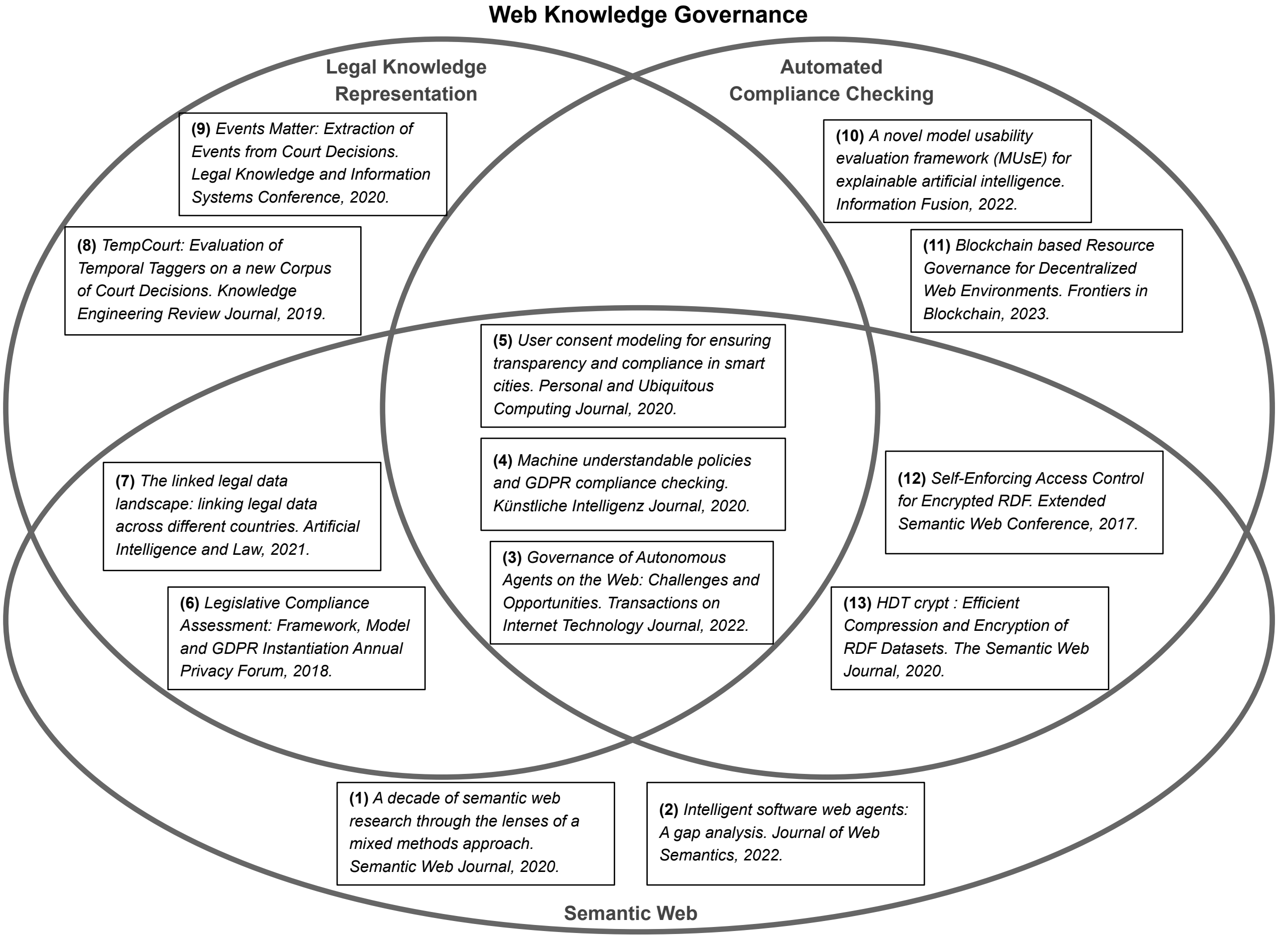

Web Knowledge Governance
Legal Knowledge Representation
Automated Compliance Checking
Semantic Web
(9) Events Matter: Extraction of Events from Court Decisions. Legal Knowledge and Information Systems Conference, 2020.
(8) TempCourt: Evaluation of Temporal Taggers on a new Corpus of Court Decisions. Knowledge Engineering Review Journal, 2019.
(10) A novel model usability evaluation framework (MUsE) for explainable artificial intelligence. Information Fusion, 2022.
(11) Blockchain based Resource Governance for Decentralized Web Environments. Frontiers in Blockchain, 2023.
(5) User consent modeling for ensuring transparency and compliance in smart cities. Personal and Ubiquitous Computing Journal, 2020.
(4) Machine understandable policies and GDPR compliance checking. Künstliche Intelligenz Journal, 2020.
(7) The linked legal data landscape: linking legal data across different countries. Artificial Intelligence and Law, 2021.
(12) Self-Enforcing Access Control for Encrypted RDF. Extended Semantic Web Conference, 2017.
(3) Governance of Autonomous Agents on the Web: Challenges and Opportunities. Transactions on Internet Technology Journal, 2022.
(6) Legislative Compliance Assessment: Framework, Model and GDPR Instantiation Annual Privacy Forum, 2018.
(13) HDT crypt : Efficient Compression and Encryption of RDF Datasets. The Semantic Web Journal, 2020.
(1) A decade of semantic web research through the lenses of a mixed methods approach. Semantic Web Journal, 2020.
(2) Intelligent software web agents: A gap analysis. Journal of Web Semantics, 2022.

Figure 2.1: Thematic overview of the articles comprising this habilitation thesis.

2. Summary of the Publications

This habilitation thesis comprises a collection of articles relating to the semantic web, legal knowledge representation, and compliance automation that collectively fit under the umbrella web knowledge governance. A high-level thematic overview of the articles presented herein and their relationship to one another is depicted in *Figure* 2.1.

Paper (1) entitled *"A decade of semantic web research through the lenses of a mixed methods approach"* [51] explores the evolution of the semantic web community. In this paper, we collate and analyse topics and trends over a ten year period from 2006 to 2015 using: (i) a top-down qualitative analysis of the predominant semantic web seminal papers [10, 11, 33]; and (ii) a bottom-up data-driven quantitative analysis of our 10-year corpus using three different topic extraction approaches (Rexplore [70], Saffron [63], PoolParty [83]). Our analysis uncovers evidence of some broadly postulated trends within the community: (i) topics such as linked data and open data have increased in importance over the years; and (ii) topics such as semantic web, web services, and ontology matching are declining. Interestingly, despite the original semantic web vision whereby a web of machine-readable data would be exploited by software web agents that would carry out data-centric tasks on behalf of humans, topics in relation to intelligent software agents and multi-agent systems did not feature prominently in the analysed semantic web corpus. Also, although topics in relation to privacy, trust, and security are considered important by experts they are only weakly represented in the predominant topic lists produced by Poolparty and Rexplore and were absent from the Saffron list.

Paper (2) entitled *"Intelligent software web agents: A gap analysis"* [50] digs deeper into the status quo of intelligent software agent research by using an integrative literature review [90, 87] in order to better understand the state of the art and how the various proposals relate to one another. Inspired by the performance measure, environment, actuators, and sensors (PEAS) assessment criteria proposed by Russel and Norvig [80], we propose an agent task environment requirements assessment framework and use it to perform a detailed analysis of the original software web agent use case scenario [10]. In addition, we introduce a web based hybrid agent architecture, which grounds our gap analysis. Our research shows that over the years researchers have proposed various ontologies and vocabularies that can be used to model agent knowledge and to describe services in a manner that facilitates both discovery and composition. However, the suitability of the various proposals from both a practical and a performance/scalability perspective have yet to be determined. Additionally, although behavioural functions (i.e., benevolence, rationality, and mobility) and code of conduct functions (i.e., identification, security, privacy, trust, and ethics) are often mentioned in the early years of the semantic web, when it comes to the development of standards, tools, and technologies specifically for intelligent software agents these topics have received very little attention.

Paper (3) entitled *"Governance of autonomous agents on the web: challenges and opportunities"* [48], which was one of the major outputs of the *"Autonomous Agents on the Web"* Dagstuhl seminar, provides a blueprint for controlling software web agent behaviour and facilitating multi-agent system governance. The paper brings together different perspectives from three distinct yet overlapping research communities: semantic web and linked data; web architecture and the web of things; as well as autonomous agents and multi-agent systems. Inspired by first class abstractions commonly used to model multi-agent systems (cf. [13]), we propose a conceptual framework that serves to define the role played by various norms, policies and preferences. The effectiveness of the proposed conceptual framework is demonstrated with the help of a motivating vaccination roll-out use case scenario. Additionally, we identify several research challenges and opportunities in a broad research roadmap concerning the governance of collaborating software web agents. In essence, this roadmap calls for research into: incorporating norm governance approaches into architectures and standards; relating norms and interaction protocols; norm based reasoning in light of norm emergence and inconsistency; and cautiously advancing tools and frameworks into practice.

Paper (4) entitled *"Machine understandable policies and GDPR compliance checking"* [16] investigates how semantic web based ontologies and reasoners can be used to facilitate norm governance. More specifically we investigate how machine understandable policies and automated compliance checking can be used to demonstrate compliance with legal obligations encoded in regulations, such as the GDPR. The article describes the various challenges faced by software engineers when it comes to automating legal compliance checking brought about by the connectedness of articles, paragraphs, and points, as well as the pivotal role played by legal interpretation when it comes to legal reasoning in general. Based on a detailed analysis of the GDPR we propose an approach for encoding and reasoning over well understood concepts and relations. We introduce the SPECIAL usage and business policy languages based on a fragment (profile) of OWL2 [47] called $\mathscr{PL}$ (policy logic) [15], as well as their respective grammars. Additionally, we propose two different compliance checking approaches that facilitate: (i) ensuring that data processing complies with the data subjects consent; and (ii) verifying if business processes comply with legal obligations. Initial performance results without any optimisation or parallelism can support approximately 6000 compliance checks per second and more than 518 million checks per day.

Paper (5) entitled *"User consent modeling for ensuring transparency and compliance in smart cities"* [37] extends the SPECIAL usage policy language vocabularies introduced in paper (4) in order to cater for various smart mobility use case scenarios. Following ontology engineering best practices [84] general cyber physical social system (CPSS) concepts are modelled in a SPECIAL-CPSS core ontology and smart mobility use case scenario terms are modelled in use case specific extensions. The core ontology is developed using the systematic mapping study methodology [52] with a specific focus on papers describing concrete CPSSs. Additionally, we propose a practical workflow that can be used by software engineers to define consent and data usage policies for various CPSSs. Both the vocabularies and the workflow are validated by demonstrating how they can be used to construct usage policies according to the SPECIAL usage policy language specification. The proposed vocabularies and workflow are particularly suitable for guiding the development of policies for complex systems, such as the one described in paper (3), that include services, things, and agents.

Paper (6) entitled *"Legislative compliance assessment: framework, model and GDPR instantiation"* [2] also focuses on demonstrating the potential of semantic web technologies when it comes to norm governance. In particular, we tackle problems brought about by the presence of opening clauses (that enable member states to introduce more restrictive obligations via local legislation) or domain specific legislation (that partially overlaps with the GDPR) that play a pivotal role in domain agnostic EU regulations, such as the GDPR. In such cases it is not possible to determine compliance with the legislation by simply encoding the requirements stipulated in a single regulation, but rather there is a need to consider the interplay between different legislations. Our proposal includes a flexible and modular legislative compliance assessment framework, which can be used to encode multiple legislations, as well as different legal interpretations by decoupling the encoding of legal requirements from the compliance system. Additionally, we extend the Open Digital Rights Language (ODRL) [78] policy class with chapter, article, and paragraph subclasses, and the constraint class with feature, discretional, and dispensation subclasses. The effectiveness of the proposed model is evaluated via the the PriWUcy compliance system that can be used by companies to perform a question and answer based manual GDPR compliance assessment.

Paper (7) entitled *"The linked legal data landscape: linking legal data across different countries"* [39] further broadens our work on legal knowledge representation (papers 3-6) by exploring automated techniques for making legal knowledge pertaining to both legislation and court cases accessible to machines. In particular, we propose a legal knowledge graph creation methodology that can be used to transform structured and unstructured legal data into legal knowledge graphs that can be easily linked across different EU member states. Our knowledge graph, which is strongly routed in the ELI and ECLI standardisation initiatives, is populated with data and metadata from the Austrian legal information system and concepts automatically extracted from Austrian legal documents. Considering the structured nature of legal documents, our overall results look very promising with F1 scores (i.e., the harmonic mean of the precision and recall) ranging from .89 for literature references to 1.0 for legal rule references. However, our experiments demonstrate that when it comes to the extraction of legal entities from text there is no one size fits all, with a rule based (Java Annotation Pattern Engine (JAPE) [24]) approach being particularly suitable for extracting case and court references, a probability based approach (*Conditional Random Fields (CRF)* [55]) being effective when it comes to identifying literature and law gazette mentions, and deep learning outperforming the others when it comes to contributor extraction (*Bidirectional Encoder Representations from Transformers (BERT)* [27]) and legal provision detection (*DistilBERT* [81]). Additionally, we perform a comparative assessment of the accessibility of machine-readable legal data across all member states and use the knowledge gained to define a roadmap for a truly interconnected EU wide legal knowledge graph including, but not limited to, increasing information provision and facilitating access via licensing and access policies.

Paper (8) entitled *"TempCourt: evaluation of temporal taggers on a new corpus of court decisions"* [66] is the first of two papers that dig deeper into legal information provision via legal knowledge extraction. When it comes to text relating to legal cases, temporal information plays a major role as it is needed both to form a chronological order of events (i.e., a timeline) and to determine the applicable law. In this paper, we examine the particularities of temporal annotation in the legal domain, including the different court case decision structures adopted by the European Court of Human Rights (ECHR), the European Court of Justice (ECJ), and the United States Supreme Court (USC). Considering the particularities of legal text, we are especially interested in misleading or mistaken temporal expressions in references; and deficiencies in the TimeML[1] temporal annotation standard when it comes to specific legal temporal expressions. In order to evaluate the effectiveness of existing temporal taggers, we create two gold standards: (i) a generic gold standard called StandardTimeML; and (ii) a domain focused gold standard called LegalTimeML. We subsequently evaluate 10 openly available taggers (4 rule based, 1 machine learning based, and 5 hybrid). Our evaluations shows that GUTime [58] (hybrid), TERNIP [67] (rule based), SUTime [23] (rule based) and HeidelTime [86] (rule based) are the best performing temporal taggers, however results vary depending on the corpus and the setting (strict versus lenient). Finally, we conclude our analysis by identifying commonly occurring issues faced by various taggers used in conjunction with legal text.

Paper (9) entitled *"Events matter: extraction of events from court decisions"* [38] builds on paper (8) by performing a comparison of different approaches to automatically extract events and their components (i.e., who; did what; when). The comparative analysis is performed over a set of 30 decisions from the ECHR that were manually annotated by two legal experts. We subsequently used the gold standard to compare the performance of various event extraction tools: rule based (JAPE [24]), probability based (CRF [55]), and deep learning based (Flair [4], BERT [27], DistilBERT [81]). The evaluation shows that the probabilistic CRF extraction is best at identifying what happened, while BERT [27] excels when is comes to extracting the when and who aspects of an event. Our analysis shows that DistilBERT [81] out performs the other approaches when it comes to classifying events in terms of procedure or circumstance. Finally, in order to demonstrate the applicability of event extraction from legal text, we propose a prototypical web interface that displays court decisions according to their extracted timelines.

Paper (10) entitled *"A novel model usability evaluation framework (MUsE) for explainable artificial intelligence"* [28] is particularly timely considering the proposed EU law on artificial intelligence[2] calls for transparent and explainable artificial intelligence (xAI)[53] systems. The overarching goal of this paper is to better understand the effectiveness of the popular Local Interpretable Model-Agnostic Explanations (LIME)[79] xAI framework. In particular, we demonstrate how LIME can be used to better understand and compare the classification (whether on not it will rain) of a particular observation based on different machine learning algorithms (i.e., logistic regression, decision tree, random forest, and XGBoost). In order to assess the understandability of the output produced by LIME we conducted a usability study. Half of the participants had prior knowledge of machine learning in general and classification in particular, and the other half did not. None of the participants had any prior experience with LIME. In order to examine the usability of LIME, more generally, we developed a usability assessment framework, Model Usability Evaluation (MUsE), derived from the ISO 9241-11:2018 standard. Based on our evaluation, we concluded that LIME is particularly suitable for those who are familiar with machine learning classification. Additionally, we highlight the need for data visualisations with better support for global interpretability and contextualised accuracy and reliability insights that limit the potential for negative consequences. xAI is extremely important when it comes to explaining algorithms, such as those employed in papers (8) and (9), however LIME is

[1] TimeML, `http://www.timeml.org`

[2] Proposed EU law on AI, `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52021PC0206`

not yet mature enough for application in multidisciplinary domains, where explanations need to be interpreted by non-technical stakeholders.

Paper (11) entitled *"Blockchain based Resource Governance for Decentralized Web"* [9] investigates how usage policies can be enforced after access to data has been granted. The work helps data owners to ensure the protection of their rights concerning personal data under GDPR or intellectual property under the EU copyright directive[3] via technical means. Additionally, the work is motivated by various technical decentralization initiatives (e.g., Solid[4], Digi.me[5], and ActivityPub[6]) that aim to give data owners more control over their data, and to disrupt the data monopoly by big tech companies. We introduce a resource governance (ReGov) conceptual framework and an instantiation thereof that combines blockchain applications [62] and trusted execution environments [60] in order to facilitate usage control in decentralized web environments. The proposed ReGov framework assumes a network of peers that publish and consume data with some usage terms and conditions. In turn one or more governance ecosystems are responsible for resource indexing (data and associated policies) and policy governance (continuous monitoring of compliance). Our particular instantiation of the ReGov framework includes: (i) an Ethereum Virtual Machine (EVM) blockchain [18] that hosts the DTindexing and DTobligations smart contracts; and (ii) an Intel SGX [93] trusted execution environment that hosts the policy aware data consumption module. The evaluation involves a detailed analysis of concrete requirements derived from our data market motivating scenario and an assessment of the security, privacy, and affordability aspects of the proposed ReGov instantiation. Although the costs are quite high and are highly dependent on market prices, smart contract optimisations and different blockchain implementations could be employed in order to reduce costs.

Paper (12) entitled *"Self-enforcing access control for encrypted RDF"* [35] is the first of two papers that dig deeper into supporting the data integrity and confidentiality principle of the GDPR via technical means. In particular, we demonstrate how predicate-based encryption (PBE) [49], which we refer to as functional encryption (in order to avoid confusion with RDF predicates), can be used to facilitate fine-grained access control based on triple patterns over encrypted RDF datasets. Essentially, each ciphertext is associated with a (secret) attribute vector and each decryption key corresponds to a vector that is incorporated into its respective boolean function. In order to verify whether an encrypted triple can be decrypted with a given decryption key, we compute the inner-product of the two vectors. In essence, decryption keys can decrypt all triples that satisfy their inherent triple pattern (i.e., one query key can open multiple locks). In order to enhance query performance we compare two different indexing strategies: 3-indexes [54] (i.e., `SPO`, `POS`, and `OSP`) and vertical partitioning [1] (i.e., indexing `SO` per predicate). Additionally, in order to obfuscate the structure of the encrypted data we include dummy ciphers that are indistinguishable from real hashes and ciphers. The results of our performance evaluation indicate that although batch decryption is relatively slow results can be served incrementally.

Paper (13) entitled *"HDTcrypt: Compression and encryption of RDF datasets"* [36] also investigates how existing encryption techniques can be used to protect sensitive data encoded as RDF, however the focus is on sharing large RDF datasets securely as opposed to querying encrypted data. Considering the volume of data that could potentially need to be stored by individual agents or exchanged between collaborating agents we explore various strategies for extending HDT (Header Dictionary Triples) [34], a compressed serialization format for RDF Graphs, with encryption mechanisms. In HDT the header component holds metadata needed for discovery, the dictionary component

[3] EU copyright directive, https://eur-lex.europa.eu/eli/dir/2019/790/oj

[4] Solid, `https://solidproject.org/about`

[5] Digi.me, `https://digi.me/what-is-digime/`

[6] ActivityPub, `https://activitypub.rocks/`

holds a mapping from RDF graph terms to ids, and the triple component holds an id based representation of the graph structure. HDTcrypt encrypts the dictionary and triple components using the advanced encryption standard (AES) [25]. Given a knowledge graph composed of different access restricted subgraphs of a dataset, we investigate four alternative partitioning strategies with different space/performance tradeoffs: HDT$_{crypt-A}$ constructs separate HDT components for each graph in the dataset; HDT$_{crypt-B}$ splits the graphs in the dataset according to their canonical partition and constructs separate HDT components for each subgraph; HDT$_{crypt-C}$ creates a canonical partition of terms, such that dictionaries do not contain duplicates; and HDT$_{crypt-D}$ creates a canonical partition of triples and terms, such that triples and dictionaries do not contain duplicates. Experiments show that the proposed partitioning strategies achieve space savings (26-31%) over the compression baseline and are comparable in terms of query performance. Although approach HDT$_{crypt-C}$ is consistently the best in terms of compression, when it comes to querying HDT$_{crypt-A}$ and HDT$_{crypt-B}$ outperform HDT$_{crypt-C}$, which incurs additional overhead as the dictionaries and triples need to be integrated in order to support querying.

3. Fulfilment of the Requirements

The habilitation guide[1] of the Department of Information Systems and Operations at the Vienna University of Economics and Business stipulates that a cumulative habilitation thesis should be composed of at least five excellent academic articles that are thematically related. It is expected that the articles are published (or accepted for publication) in well-known and highly-ranked scientific journals. In order to ensure scientific excellent and to attest to the novelty and expected scientific impact of the work, these articles should have gone through a rigorous review process. In order to demonstrate that the habilitation candidate is capable of conducting independent scientific work, one of the journal articles must be a single author publication. If an article has several authors the habilitation candidate is expected to clearly articulate their specific contribution to the article. Although the department has compiled a list of excellent academic journals that are relevant for the department it is not deemed to be a complete list and thus the habilitation thesis can also contain other highly ranked scientific journals. A maximum of two journal articles can be substituted by excellent conference papers, under the assumption that the conference is organised by major discipline-specific associations, that have a rigorous review processes, and an acceptance rate of less that 30%. Three conference articles are needed to substitute one journal article. It is worth noting that there is no department conference list.

The candidates cumulative habilitation thesis fulfils all requirements. In short, the thesis is composed of ten journal articles, one of them being a single author paper, and three conference publications. In all cases the contributor roles taxonomy (CRediT)[2] is used in order to clearly articulate the habilitation candidates contribution to these articles.

This ten journal articles were published in nine prominent scientific journals. The metrics below were taken from the Scimago[3] and Resurchify[4] journal rankings. The impact score is similar to the impact factor, however the former is calculated based on Scopus data and the latter based on Web of Science data. The metrics relating to the Frontiers in Blockchain journal, which is a new journal, were obtained from Exaly[5] as there was no information in either Scimago or Resurchify.

- Information Fusion, Elsevier, Quartile Q1, h-index=120, Impact Factor=17.564, Impact Score=18.92, SJR=4.557.
- ACM Transactions on Internet Technology, ACM, Quartile Q1, h-index=59, Impact Factor=3.989, Impact Score=3.99, SJR=1.175.
- Semantic Web, IOS Press, Quartile Q1, h-index=45, Impact Factor=3.105, Impact Score=3. 59, SJR=1.242, on the department journal list.
- Artificial Intelligence and Law, Springer, Quartile Q1, h-index=41, Impact Factor=2.723, Impact Score=4.94, SJR=1.316.

[1] https://www.wu.ac.at/fileadmin/wu/d/ipm/web_files/HRL_IPM_v2016_en_fin.pdf

[2] https://www.elsevier.com/authors/policies-and-guidelines/credit-author-statement

[3] https://www.scimagojr.com/

[4] https://www.resurchify.com/

[5] https://exaly.com/

- Web Semantics, Elsevier, Quartiles Q1 & Q2, h-index=85, Impact Factor=2.77, Impact Score=3.68, SJR=0.98, on the department journal list.
- Personal and Ubiquitous Computing, Springer, Quartiles Q1 & Q2, h-index=91, Impact Factor=3.006, Impact Score=3.06, SJR=0.615, on the department journal list.
- The Knowledge Engineering Review, Cambridge University Press, Quartile Q2, h-index=64, Impact Factor=2.016, Impact Score=2.39, SJR=0.681.
- KI-Künstliche Intelligenz, Springer, Quartile Q2, h-index=23, Impact Score=1.91, SJR=0.745.
- Frontiers in Blockchain, Frontiers, Quartile Pending, h-index=12, Impact Factor=3.

This cumulative habilitation thesis is composed of three publications in prominent discipline-specific conferences (legal informatics, privacy, and semantic web) that nicely complement the habilitation candidates journal publications.

- The 33rd International Conference on Legal Knowledge and Information Systems (JURIX 2020). Acceptance rate 23.5%.
- The 7th Annual Privacy Forum (APF 2018). Acceptance rate 22%.
- The 14th Extended Semantic Web Conference (ESWC 2017). Acceptance rate 28%.

In the following, we use the CRediT roles in order to clarify the habilitation candidates contribution to the articles that comprise this cumulative habilitation thesis. Additionally, we provide a short justification for inclusion of each article.

1. **Kirrane, S.**, Sabou, M., Fernández, J.D., Osborne, F., Robin, C., Buitelaar, P., Motta, E. and Polleres, A., 2020. A decade of Semantic Web research through the lenses of a mixed methods approach. Semantic Web, 11(6), pp.979-1005. Quartile Q1, h-index=45, Impact Factor=3.105, Impact Score=3. 59, SJR=1.242. `https://doi.org/10.3233/SW-200371`.

   Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Software, Validation, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization, and Project administration.

   Justification for inclusion: The habilitation candidate is the first and main author of this article. She was responsible for co-ordinating the writing process, took care of one of the natural language processing experiments, and was involved in the qualitative analysis as well as its write-up.

2. **Kirrane, S.**, 2021. Intelligent software web agents: A gap analysis. Journal of Web Semantics, 71, p.100659. Quartiles Q1 & Q2, h-index=85, Impact Factor=2.77, Impact Score=3.68, SJR=0.98. `https://doi.org/10.1016/j.websem.2021.100659`.

   Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization, Project administration, and Funding acquisition.

   Justification for inclusion: A requirement of the Habilitation is that at least one article be a single-authored article.

3. Kampik, T., Mansour, A., Boissier, O., **Kirrane, S.**, Padget, J., Payne, T.R., Singh, M.P., Tamma, V. and Zimmermann, A., 2022. Governance of Autonomous Agents on the Web: Challenges and Opportunities. ACM Transactions on Internet Technology, 22(4), pp.1-31. Quartile Q1, h-index=59, Impact Factor=3.989, Impact Score=3.99, SJR=1.175. `https://doi.org/10.1145/3507910`.

   Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Formal analysis, Investigation, Data Curation, Writing - Original Draft, and Writing - Review & Editing.

   Justification for inclusion: This work is an output from the Autonomous Agents on the Web Dagstuhl seminar and a follow-up to paper (2). The habilitation candidates expertise in norms and policies was instrumental to the working group and to the development of the ideas presented in this paper.

4. Bonatti, P.A., **Kirrane, S.**, Petrova, I.M. and Sauro, L., 2020. Machine understandable policies and GDPR compliance checking. KI-Künstliche Intelligenz, 34, pp.303-315. Quartile Q2, h-index=23, Impact Score=1.91, SJR=0.745. `https://doi.org/10.1007/s13218-020-00677-4`.

   Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Formal analysis, Investigation, Writing - Original Draft, Writing - Review & Editing, Project administration, and Funding acquisition.

   Justification for inclusion: The habilitation candidate made a significant intellectual contribution to the SPECIAL policy language and log vocabulary, as well as the compliance checking approaches presented in this article.

5. Fernández, J.D., Sabou, M., **Kirrane, S.**, Kiesling, E., Ekaputra, F.J., Azzam, A. and Wenning, R., 2020. User consent modeling for ensuring transparency and compliance in smart cities. Personal and Ubiquitous Computing, 24, pp.465-486. Quartiles Q1 & Q2, h-index=91, Impact Factor=3.006, Impact Score=3.06, SJR=0.615. `https://doi.org/10.1007/s00779-019-01330-0`.

   Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Formal analysis, Investigation, Writing - Original Draft, and Writing - Review & Editing.

   Justification for inclusion: This is an important extension of paper (4). The habilitation candidates expertise in the SPECIAL policy language and log vocabulary, as well as automated compliance checking were instrumental to developing the extensions presented in this article.

6. Agarwal, S., Steyskal, S., Antunovic, F., **Kirrane, S.**, 2018. Legislative Compliance Assessment: Framework, Model and GDPR Instantiation. Proceedings of the 7th Annual Privacy Forum (APF 2018). Acceptance rate 22%. `https://doi.org/10.1007/978-3-030-02547-2_8`.

   Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Writing - Review & Editing, and Supervision.

   Justification for inclusion: The habilitation candidate was the main supervisor of this work, and made a significant intellectual contribution to the legislative compliance assessment framework and ODRL extensions presented in this paper. Although this article is a conference paper, the Annual Privacy Forum is a long-standing conference organised by the European Union Agency for Cybersecurity (ENISA) and the European Commissions Directorate-General for Communications Networks, Content and Technology (DG CONNECT). The acceptance rate in 2018, which was based on 3 or more reviews, was 22%.

7. Filtz, E., **Kirrane, S.** and Polleres, A., 2021. The linked legal data landscape: linking legal data across different countries. Artificial Intelligence and Law, 29(4), pp.485-539. Quartile Q1, h-index=41, Impact Factor=2.723, Impact Score=4.94, SJR=1.316. `https://doi.org/10.1007/s10506-021-09282-8`.

   Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Writing - Review & Editing, and Supervision.

   Justification for inclusion: The habilitation candidate was one of two supervisors of this work. Her expertise in legal knowledge representation and natural language processing were instrumental to the development of the legal knowledge graph, automated extraction techniques, and analysis presented in this article.

8. Navas-Loro, M., Filtz, E., Rodríguez-Doncel, V., Polleres, A. and **Kirrane, S.**, 2019. TempCourt: evaluation of temporal taggers on a new corpus of court decisions. The Knowledge Engineering Review, 34, p.e24. Quartile Q2, h-index=64, Impact Factor=2.016, Impact Score=2.39, SJR=0.681. `https://doi.org/10.1017/S0269888919000195`.

   Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Writing - Review & Editing, and Supervision.

   Justification for inclusion: The habilitation candidate was the main supervisor of the collaboration between Wirtschaftsuniversität Wien and Universidad Politécnica de Madrid that led to the publication of this article. Her expertise in scientific research and natural language processing were instrumental to the development of the ideas presented in this article.

9. Filtz, E., Navas-Loro, M., Santos, C., Polleres, A. and **Kirrane, S.**, 2020. Events Matter: Extraction of Events from Court Decisions. Proceedings of the 33rd International Conference on Legal Knowledge and Information Systems (JURIX), 2020. Acceptance rate 23.5%. `https://ebooks.iospress.nl/doi/10.3233/FAIA200847`.

   Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Writing - Review & Editing, and Supervision.

   Justification for inclusion: This article is an important extension of paper (8). Although this article is a conference publication, the International Conferences on Legal Knowledge and Information Systems is the predominant publishing venue for legal informatics scholars. The acceptance rate for full research papers in 2020, which was based on 3 or more reviews, was 23.5%.

10. Dieber, J. and **Kirrane, S.**, 2022. A novel model usability evaluation framework (MUsE) for explainable artificial intelligence. Information Fusion, 81, pp.143-153. Quartile Q1, h-index=120, Impact Factor=17.564, Impact Score=18.92, SJR=4.557. `https://doi.org/10.1016/j.inffus.2021.11.017`.

    Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Writing - Review & Editing, and Supervision.

    Justification for inclusion: The habilitation candidate was the supervisor of this work. Her expertise in conducting scientific research and her knowledge of data science were instrumental to the development of the ideas presented in this article.

11. Basile, D., Di Ciccio, C., Goretti, V., **Kirrane, S.**, 2023. Blockchain based Resource Governance for Decentralized Web Environments. Frontiers in Blockchain, 2023 (In-press). Quartile Pending, h-index=12, Impact Factor=3. `https://www.frontiersin.org/articles/10.3389/fbloc.2023.1141909`.

Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Writing - Original Draft, Writing - Review & Editing, and Supervision.

Justification for inclusion: The habilitation candidate was one of two supervisors of this work. In addition, she motivated the focus on usage control and governance, which are at the core of the proposed framework and instantiation.

12. Fernández, J.D., **Kirrane, S.**, Polleres, A. and Steyskal, S., 2020. HDTcrypt: Compression and encryption of RDF datasets. Semantic Web, 11(2), pp.337-359. Quartile Q1, h-index=45, Impact Factor=3.105, Impact Score=3. 59, SJR=1.242. `https://doi.org/10.3233/SW-180335`.

    Justification for inclusion: The habilitation candidate contributed her expertise in data integrity and confidentiality to this article, and made a significant intellectual contribution to all aspects of the paper.

    Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Formal analysis, Investigation, Writing - Original Draft, Writing - Review & Editing, Supervision, Project administration, and Funding acquisition.

13. Fernández, J.D., **Kirrane, S.**, Polleres, A. and Steyskal, S., 2017. Self-Enforcing Access Control for Encrypted RDF. Proceedings of the 14th Extended Semantic Web Conference (ESWC 2017). Acceptance rate 28%. `https://doi.org/10.1007/978-3-319-58068-5_37`.

    Habilitation Candidate CRediT roles: Conceptualisation, Methodology, Formal analysis, Investigation, Writing - Original Draft, Writing - Review & Editing, Supervision, Project administration, and Funding acquisition

    Justification for inclusion: This article nicely complements paper (12). The habilitation candidate contributed her expertise in data integrity and confidentiality to this article, made a significant intellectual contribution to all aspects of the paper, and presented the work at the Extended Semantic Web Conference. Although this is a conference paper, this long-standing conference is one of two core semantic web publishing outlets. The acceptance rate in 2018, which was based on 3 or more reviews, was 28%.

Bibliography

[1] D. J. Abadi, A. Marcus, S. R. Madden, and K. Hollenbach. Scalable semantic web data management using vertical partitioning. In *Proceedings of Very Large Data Bases*, pages 411–422, 2007.

[2] S. Agarwal, S. Steyskal, F. Antunovic, and S. Kirrane. Legislative compliance assessment: Framework, model and GDPR instantiation. In *Privacy Technologies and Policy - 6th Annual Privacy Forum, APF 2018, Barcelona, Spain, June 13-14, 2018, Revised Selected Papers*, volume 11079 of *Lecture Notes in Computer Science*, pages 131–149. Springer, 2018. doi: 10.1007/978-3-030-02547-2_8.

[3] G. Ajani, G. Boella, L. D. Caro, L. Robaldo, L. Humphreys, S. Praduroux, P. Rossi, and A. Violato. The european taxonomy syllabus: A multi-lingual, multi-level ontology framework to untangle the web of european legal terminology. *Applied Ontology*, 11(4):325–375, 2016. doi: 10.3233/AO-170174.

[4] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, and R. Vollgraf. Flair: An easy-to-use framework for state-of-the-art nlp. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics (demonstrations)*, pages 54–59, 2019.

[5] M. N. Asim, M. Wasim, M. U. G. Khan, W. Mahmood, and H. M. Abbasi. A survey of ontology learning techniques and applications. *Database*, 2018, 2018.

[6] T. Athan, H. Boley, G. Governatori, M. Palmirani, A. Paschke, and A. Wyner. Oasis legalruleml. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Law*, ICAIL '13, page 3–12, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450320801. doi: 10.1145/2514601.2514603.

[7] C. Baral. *Knowledge Representation, Reasoning and Declarative Problem Solving*. CUP, 2003.

[8] C. Bartolini, A. Calabró, and E. Marchetti. Enhancing business process modelling with data protection compliance: An ontology-based proposal. In *ICISSP*, pages 421–428, 2019.

[9] D. Basile, C. Di Ciccio, V. Goretti, and S. Kirrane. Blockchain based resource governance for decentralized web environments. *Frontiers in Blockchain, 2023 (In-press)*, 2023. URL `https://doi.org/10.3233/FAIA200847`.

[10] T. Berners-Lee, J. Hendler, O. Lassila, et al. The semantic web. *Scientific American*, 284(5): 28–37, 2001.

[11] A. Bernstein, J. A. Hendler, and N. F. Noy. A new look at the semantic web. *Commun. ACM*, 59(9):35–37, 2016. doi: 10.1145/2890489.

[12] G. Boella, L. D. Caro, and V. Leone. Semi-automatic knowledge population in a legal document management system. *Artif. Intell. Law*, 27(2):227–251, 2019. doi: 10.1007/s10506-018-9239-8.

[13] O. Boissier, R. H. Bordini, J. Hübner, and A. Ricci. *Multi-agent oriented programming: programming multi-agent systems using JaCaMo*. MIT Press, Cambridge, 2020.

[14] H. Boley, A. Paschke, and O. Shafiq. Ruleml 1.0: the overarching specification of web rules. In *International Workshop on Rules and Rule Markup Languages for the Semantic Web*, pages 162–178. Springer, 2010.

[15] P. A. Bonatti. Fast compliance checking in an OWL2 fragment. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 1746–1752. ijcai.org, 2018. doi: 10.24963/ijcai.2018/241.

[16] P. A. Bonatti, S. Kirrane, I. M. Petrova, and L. Sauro. Machine understandable policies and GDPR compliance checking. *Künstliche Intell.*, 34(3):303–315, 2020. doi: 10.1007/s13218-020-00677-4.

[17] D. Brickley and R. Guha. Rdf schema 1.1. *W3C recommendation*, 2014.

[18] V. Buterin et al. A next-generation smart contract and decentralized application platform. *white paper*, 3(37):2–1, 2014.

[19] E. Cabrio, A. P. Aprosio, and S. Villata. These are your rights - A natural language processing approach to automated RDF licenses generation. In *The Semantic Web: Trends and Challenges - 11th International Conference, ESWC 2014, Anissaras, Crete, Greece, May 25-29, 2014. Proceedings*, volume 8465 of *Lecture Notes in Computer Science*, pages 255–269. Springer, 2014. doi: 10.1007/978-3-319-07443-6_18.

[20] C. Cardellino, S. Villata, L. A. Alemany, and E. Cabrio. Information extraction with active learning: A case study in legal text. In *Computational Linguistics and Intelligent Text Processing: 16th International Conference, CICLing 2015, Cairo, Egypt, April 14-20, 2015, Proceedings, Part II 16*, pages 483–494. Springer, 2015.

[21] I. Chalkidis, C. Nikolaou, P. Soursos, and M. Koubarakis. Modeling and querying greek legislation using semantic web technologies. In *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I*, pages 591–606, 2017. doi: 10.1007/978-3-319-58068-5_36.

[22] M. Challenger, B. T. Tezel, O. F. Alaca, B. Tekinerdogan, and G. Kardas. Development of semantic web-enabled bdi multi-agent systems using sea_ml: An electronic bartering case study. *Applied Sciences*, 8(5):688, 2018.

[23] A. X. Chang et al. Sutime: A library for recognizing and normalizing time expressions. In *Proceedings of LREC 2012*, 2012.

[24] H. Cunningham, H. Cunningham, D. Maynard, D. Maynard, V. Tablan, and V. Tablan. Jape: a java annotation patterns engine, 1999.

[25] J. Daemen and V. Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer, 2013. ISBN 3540425802.

[26] M. De Vos, S. Kirrane, J. Padget, and K. Satoh. Odrl policy modelling and compliance checking. In *Rules and Reasoning: Third International Joint Conference, RuleML+ RR 2019, Bolzano, Italy, September 16–19, 2019, Proceedings 3*, pages 36–51. Springer, 2019.

[27] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1423.

[28] J. Dieber and S. Kirrane. A novel model usability evaluation framework (muse) for explainable artificial intelligence. *Inf. Fusion*, 81:143–153, 2022. doi: 10.1016/j.inffus.2021.11.017.

[29] A. Dimou, M. Vander Sande, P. Colpaert, R. Verborgh, E. Mannens, and R. Van de Walle. Rml: A generic language for integrated rdf mappings of heterogeneous data. *Ldow*, 1184, 2014.

[30] J. Dimyadi, G. Governatori, and R. Amor. Evaluating legaldocml and legalruleml as a standard for sharing normative information in the aec/fm domain. In *Proceedings of the Joint Conference on Computing in Construction (JC3)*, volume 1, pages 637–644. Heriot-Watt University, Edinburgh, UK. Heraklion, Greece, 2017.

[31] J. S. Dong, Y. Feng, Y.-F. Li, C. K.-Y. Tan, B. Wadhwa, and H. H. Wang. Bowl: augmenting the semantic web with beliefs. *Innovations in Systems and Software Engineering*, 11(3), 2015.

[32] B. Esteves, H. J. Pandit, and V. Rodríguez-Doncel. Odrl profile for expressing consent through granular access control policies in solid. In *2021 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pages 298–306. IEEE, 2021.

[33] L. Feigenbaum, I. Herman, T. Hongsermeier, E. Neumann, and S. Stephens. The semantic web in action. *Scientific American*, 297(6):90–97, 2007.

[34] J. Fernández, M. Martínez-Prieto, C. Gutiérrez, A. Polleres, and M. Arias. Binary RDF Representation for Publication and Exchange. *J. Web Semant.*, 19:22–41, 2013.

[35] J. D. Fernández, S. Kirrane, A. Polleres, and S. Steyskal. Self-enforcing access control for encrypted RDF. In *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I*, volume 10249 of *Lecture Notes in Computer Science*, pages 607–622, 2017. doi: 10.1007/978-3-319-58068-5_37.

[36] J. D. Fernández, S. Kirrane, A. Polleres, and S. Steyskal. Hdtcrypt: Compression and encryption of RDF datasets. *Semantic Web*, 11(2):337–359, 2020. doi: 10.3233/SW-180335.

[37] J. D. Fernández, M. Sabou, S. Kirrane, E. Kiesling, F. J. Ekaputra, A. Azzam, and R. Wenning. User consent modeling for ensuring transparency and compliance in smart cities. *Pers. Ubiquitous Comput.*, 24(4):465–486, 2020. doi: 10.1007/s00779-019-01330-0.

[38] E. Filtz, M. Navas-Loro, C. Santos, A. Polleres, and S. Kirrane. Events matter: Extraction of events from court decisions. In *Legal Knowledge and Information Systems - JURIX 2020: The Thirty-third Annual Conference, Brno, Czech Republic, December 9-11, 2020*, volume 334 of *Frontiers in Artificial Intelligence and Applications*, pages 33–42. IOS Press, 2020. doi: 10.3233/FAIA200847.

[39] E. Filtz, S. Kirrane, and A. Polleres. The linked legal data landscape: linking legal data across different countries. *Artif. Intell. Law*, 29(4):485–539, 2021. doi: 10.1007/s10506-021-09282-8.

[40] N. Fornara, A. Chiappa, and M. Colombetti. Using semantic web technologies and production rules for reasoning on obligations and permissions. In *International Conference on Agreement Technologies*, pages 49–63. Springer, 2018.

[41] E. Francesconi. A description logic framework for advanced accessing and reasoning over normative provisions. *Artificial intelligence and Law*, 22(3):291–311, 2014.

[42] A. Gangemi. Design patterns for legal ontology constructions. In *Proceedings of the 2nd Workshop on Legal Ontologies and Artificial Intelligence Techniques June 4th, 2007, Stanford University, Stanford, CA, USA*, volume 321 of *CEUR Workshop Proceedings*, pages 65–85. CEUR-WS.org, 2007. URL `http://ceur-ws.org/Vol-321/paper4.pdf`.

[43] M. Gebser, R. Kaminski, B. Kaufmann, and T. Schaub. Clingo = ASP + control: Preliminary report. *CoRR*, abs/1405.3694, 2014.

[44] G. Governatori, M. Hashmi, H.-P. Lam, S. Villata, and M. Palmirani. Semantic business process regulatory compliance checking using LegalRuleML. In *European Knowledge Acquisition Workshop*, 2016.

[45] S. Guth, G. Neumann, and M. Strembeck. Experiences with the enforcement of access rights extracted from odrl-based digital contracts. In *Proceedings of the 3rd ACM workshop on Digital rights management*, pages 90–102, 2003.

[46] G. Havur, S. Steyskal, O. Panasiuk, A. Fensel, V. Mireles, T. Pellegrini, T. Thurner, A. Polleres, and S. Kirrane. Automatic license compatibility checking. In *SEMANTiCS Posters&Demos*, 2019.

[47] P. Hitzler, M. Krötzsch, B. Parsia, P. F. Patel-Schneider, S. Rudolph, et al. Owl 2 web ontology language primer. *W3C recommendation*, 2009.

[48] T. Kampik, A. Mansour, O. Boissier, S. Kirrane, J. Padget, T. R. Payne, M. P. Singh, V. Tamma, and A. Zimmermann. Governance of autonomous agents on the web: Challenges and opportunities. *ACM Transactions on Internet Technology*, 22(4):1–31, 2022.

[49] J. Katz, A. Sahai, and B. Waters. Predicate Encryption Supporting Disjunctions, Polynomial Equations, and Inner Products. *J. Cryptology*, 26(2):191–224, 2013.

[50] S. Kirrane. Intelligent software web agents: A gap analysis. *J. Web Semant.*, 71:100659, 2021. doi: 10.1016/j.websem.2021.100659.

[51] S. Kirrane, M. Sabou, J. D. Fernández, F. Osborne, C. Robin, P. Buitelaar, E. Motta, and A. Polleres. A decade of semantic web research through the lenses of a mixed methods approach. *Semantic Web*, 11(6):979–1005, 2020. doi: 10.3233/SW-200371.

[52] B. A. Kitchenham, D. Budgen, and O. Pearl Brereton. Using mapping studies as the basis for further research - A participant-observer case study. *Information and Software Technology*, 53 (6):638–651, jun 2011.

[53] P. W. Koh and P. Liang. Understanding black-box predictions via influence functions, 2017.

[54] G. Ladwig and A. Harth. CumulusRDF: linked data management on nested key-value stores. In *Proceedings of Scalable Semantic Web Knowledge Base Systems*, page 30, 2011.

[55] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001*, pages 282–289. Morgan Kaufmann, 2001.

[56] H.-P. Lam and M. Hashmi. Enabling reasoning with LegalRuleML. *Theory and Practice of Logic Programming*, 19(1):1–26, 2019.

[57] V. Leone. *Legal knowledge extraction in the data protection domain based on ontology design patterns*. PhD thesis, University of Luxembourg, Luxembourg City, Luxembourg, 2021. URL `http://orbilu.uni.lu/handle/10993/47854`.

[58] I. Mani et al. Robust temporal processing of news. In *Proceedings of the 38th annual meeting on ACL*, pages 69–76. ACL, 2000.

[59] F. Manola, E. Miller, and B. McBride. Rdf 1.1 concepts and abstract syntax. *W3C recommendation*, 2014.

[60] B. McGillion, T. Dettenborn, T. Nyman, and N. Asokan. Open-tee–an open virtual trusted execution environment. In *2015 IEEE Trustcom/BigDataSE/ISPA*, volume 1, pages 400–407. IEEE, 2015.

[61] Z. Ming, G. Wang, Y. Yan, J. Dal Santo, J. K. Allen, and F. Mistree. An ontology for reusable and executable decision templates. *Journal of Computing and Information Science in Engineering*, 17(3), 2017.

[62] D. Mohanty. Ethereum for architects and developers. *Apress Media LLC, California*, pages 14–15, 2018.

[63] F. Monaghan, G. Bordea, K. Samp, and P. Buitelaar. Exploring your research: Sprinkling some saffron on semantic web dog food. In *Semantic Web Challenge at the International Semantic Web Conference*, volume 117, pages 420–435. Citeseer, 2010.

[64] B. Moreau, P. Serrano-Alvarado, M. Perrin, and E. Desmontils. A license-based search engine. In *The Semantic Web: ESWC 2019 Satellite Events: ESWC 2019 Satellite Events, Portorož, Slovenia, June 2–6, 2019, Revised Selected Papers 16*, pages 130–135. Springer, 2019.

[65] M. Navas-Loro and C. Santos. Events in the legal domain: first impressions. In *TERECOM@JURIX*, pages 45–57, 2018.

[66] M. Navas-Loro, E. Filtz, V. Rodríguez-Doncel, A. Polleres, and S. Kirrane. Tempcourt: evaluation of temporal taggers on a new corpus of court decisions. *Knowl. Eng. Rev.*, 34:e24, 2019. doi: 10.1017/S0269888919000195.

[67] C. Northwood. TERNIP: temporal expression recognition and normalisation in Python. Master's thesis, University of Sheffield, 2010.

[68] A. Oksanen, M. Tamper, J. Tuominen, E. Mäkelä, A. Hietanen, and E. Hyvönen. Semantic finlex: Transforming, publishing, and using finnish legislation and case law as linked open data on the web. *Knowledge of the Law in the Big Data Age*, 317:212–228, 2019.

[69] A. Oltramari, D. Piraviperumal, F. Schaub, S. Wilson, S. Cherivirala, T. B. Norton, N. C. Russell, P. Story, J. R. Reidenberg, and N. M. Sadeh. Privonto: A semantic framework for the analysis of privacy policies. *Semantic Web*, 9(2):185–203, 2018. doi: 10.3233/SW-170283.

[70] F. Osborne and E. Motta. Rexplore: Unveiling the dynamics of scholarly data. In *IEEE/ACM Joint Conference on Digital Libraries, JCDL 2014, London, United Kingdom, September 8-12, 2014*, pages 415–416. IEEE Computer Society, 2014. doi: 10.1109/JCDL.2014.6970202.

[71] M. Palmirani, G. Governatori, A. Rotolo, S. Tabet, H. Boley, and A. Paschke. LegalRuleML: XML-based rules and norms. In *International Workshop on Rules and Rule Markup Languages for the Semantic Web*, pages 298–312. Springer, 2011.

[72] M. Palmirani, M. Martoni, A. Rossi, C. Bartolini, and L. Robaldo. Pronto: Privacy ontology for legal reasoning. In *Electronic Government and the Information Systems Perspective - 7th International Conference, EGOVIS 2018, Regensburg, Germany, September 3-5, 2018, Proceedings*, volume 11032 of *Lecture Notes in Computer Science*, pages 139–152. Springer, 2018. doi: 10.1007/978-3-319-98349-3_11.

[73] M. Palmirani, M. Martoni, A. Rossi, C. Bartolini, and L. Robaldo. Legal ontology for modelling gdpr concepts and norms. In *Legal Knowledge and Information Systems*, pages 91–100. IOS Press, 2018.

[74] O. Panasiuk, S. Steyskal, G. Havur, A. Fensel, and S. Kirrane. Modeling and reasoning over data licenses. In *European Semantic Web Conference*, pages 218–222. Springer, 2018.

[75] H. J. Pandit, K. Fatema, D. O'Sullivan, and D. Lewis. Gdprtext - GDPR as a linked data resource. In *The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings*, pages 481–495, 2018. doi: 10.1007/978-3-319-93417-4_31.

[76] H. Pham and D. Stacey. Practical goal-based reasoning in ontology-driven applications. In *KEOD*, pages 99–109, 2011.

[77] E. Puerto, J. Aguilar, et al. Automatic learning of ontologies for the semantic web: Experiment lexical learning. *Respuestas*, 17(2):5–12, 2012.

[78] S. V. Renato Iannella. ODRL 2.0 core model. Specification, available at `https://www.w3.org/TR/odrl-model/`, W3C, February 2018.

[79] M. Ribeiro, S. Singh, and C. Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 97–101, 02 2016. doi: 10.18653/v1/N16-3020.

[80] S. Russel and P. Norvig. *Artificial intelligence: a modern approach*. Pearson Education Limited, 2013.

[81] V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019. URL `http://arxiv.org/abs/1910.01108`.

[82] C. Santos, C. Pruski, M. D. Silveira, V. Rodríguez-Doncel, A. Gangemi, L. van der Torre, and P. Casanovas. Complaint ontology pattern - COP. In *Advances in Ontology Design and Patterns [revised and extended versions of the papers presented at the 7th edition of the Workshop on Ontology and Semantic Web Patterns, WOP@ISWC 2016, Kobe, Japan, 18th October 2016]*, volume 32 of *Studies on the Semantic Web*, pages 69–83. IOS Press, 2016. doi: 10.3233/978-1-61499-826-6-69.

[83] T. Schandl and A. Blumauer. Poolparty: SKOS thesaurus management utilizing linked data. In *The Semantic Web: Research and Applications, 7th Extended Semantic Web Conference, ESWC 2010, Heraklion, Crete, Greece, May 30 - June 3, 2010, Proceedings, Part II*, volume 6089 of *Lecture Notes in Computer Science*, pages 421–425. Springer, 2010. doi: 10.1007/978-3-642-13489-0_36.

[84] A. Scherp, C. Saathoff, T. Franz, and S. Staab. Designing core ontologies. *Appl. Ontol.*, 6(3): 177–221, Aug. 2011. ISSN 1570-5838.

[85] S. Shakeri, V. Maccatrozzo, L. Veen, R. Bakhshi, L. Gommans, C. De Laat, and P. Grosso. Modeling and matching digital data marketplace policies. In *2019 15th International Conference on eScience (eScience)*, pages 570–577. IEEE, 2019.

[86] J. Strötgen et al. Temporal tagging on different domains: Challenges, strategies, and gold standards. In *Proceedings of LREC 2012*, volume 12, pages 3746–3753, 2012.

[87] R. J. Torraco. Writing integrative literature reviews: Guidelines and examples. *Human resource development review*, 4(3):356–367, 2005.

[88] K. Venkatachalam, N. Karthikeyan, and S. Kannimuthu. Comprehensive survey on semantic web service discovery and composition. *Adv Nat Appl Sci AENSI Publ*, 10(5):32–40, 2016.

[89] H. H. Wang, N. Gibbins, T. Payne, A. Patelli, and Y. Wang. A survey of semantic web services formalisms. *Concurrency and Computation: Practice and Experience*, 27(15):4053–4072, 2015.

[90] R. Whittemore and K. Knafl. The integrative review: updated methodology. *Journal of advanced nursing*, 52(5):546–553, 2005.

[91] W. Y. Wong. *Learning lightweight ontologies from text across different domains using the web as background knowledge*. University of Western Australia, 2009.

[92] A. Z. Wyner and W. Peters. Lexical semantics and expert legal knowledge towards the identification of legal case factors. In *JURIX*, volume 10, pages 127–136, 2010.

[93] W. Zheng, Y. Wu, X. Wu, C. Feng, Y. Sui, X. Luo, and Y. Zhou. A survey of intel sgx and its applications. *Frontiers of Computer Science*, 15:1–15, 2021.

Part II

Publications

1. A decade of semantic web research through the lenses of a mixed methods approach

Bibliographic Information

Kirrane, S., Sabou, M., Fernández, J.D., Osborne, F., Robin, C., Buitelaar, P., Motta, E. and Polleres, A., 2020. A decade of Semantic Web research through the lenses of a mixed methods approach. Semantic Web, 11(6), pp.979-1005.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Software, Validation, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization, and Project administration.

Copyright Notice

Undefined 0 (2023) 1–0
IOS Press

A decade of Semantic Web research through the lenses of a mixed methods approach

Editor(s): Christoph Schlieder, University of Bamberg, Germany
Solicited review(s): Yingjie Hu, University of California, Santa Barbara, USA; two anonymous reviewers

Sabrina Kirrane [a], Marta Sabou [b], Javier D. Fernández [a], Francesco Osborne [c], Cécile Robin [d], Paul Buitelaar [d], Enrico Motta [c], and Axel Polleres [a]

[a] *Vienna University of Economics and Business, Austria*
E-mail: firstname.lastname@wu.ac.at
[b] *Vienna University of Technology, Austria*
E-mail:firstname.lastname@ifs.tuwien.ac.at
[c] *Knowledge Media institute (KMi), The Open University, UK*
E-mail:firstname.lastname@open.ac.uk
[d] *Insight Centre for Data Analytics, National University of Ireland, Galway, Ireland*
E-mail: firstname.lastname@insight-centre.org

Abstract. The identification of research topics and trends is an important scientometric activity, as it can help guide the direction of future research. In the Semantic Web area, initially topic and trend detection was primarily performed through qualitative, *top-down* style approaches, that rely on expert knowledge. More recently, data-driven, *bottom-up* approaches have been proposed that offer a quantitative analysis of the evolution of a research domain. In this paper, we aim to provide a broader and more complete picture of Semantic Web topics and trends by adopting a *mixed methods* methodology, which allows for the combined use of both qualitative and quantitative approaches. Concretely, we build on a qualitative analysis of the main seminal papers, which adopt a top-down approach, and on quantitative results derived with three bottom-up data-driven approaches (Rexplore, Saffron, PoolParty), on a corpus of Semantic Web papers published between 2006 and 2015. In this process, we both use the latter for "fact-checking" on the former and also to derive key findings in relation to the strengths and weaknesses of top-down and bottom-up approaches to research topic identification. Although we provide a detailed study on the past decade of Semantic Web research, the findings and the methodology are relevant not only for our community but beyond the area of the Semantic Web to other research fields as well.

Keywords: Research Topics, Research Trends, Linked Data, Semantic Web, Scientometrics

1. Introduction

The term scientometrics is an all encompassing term used for an emerging field of research that analyses and measures science, technology research and innovation [21]. Although the term scientometrics is a broad term, in this paper, we focus on one particular sub field of scientometrics that uses topic analysis to identify trends in a scientific domain over time [17]. Understanding topics and subsequently predicting trends in research domains are important tasks for researchers and represent vital functions in the life of a research community. Overviews of present and past topics and trends provide important lessons of how research interests evolve and allow research communities to better plan future work, whereas visions of future topics can inspire and channel the work of a research community.

Considering the critical role played by topic and trend analysis when it comes to identifying underrepresented and emerging research topics, it is not surprising that there have been a number of works from

Semantic Web researchers that take an introspective view of the community. Several papers endeavor to predict Semantic Web research topics and trends [1,2], or as the research advanced over the years, to analyse topics and trends within the community [15,19]. In parallel, several researchers [5,22,27,30,31,34] are actively working on tools and techniques that can be used to automatically uncover research topics and trends from scientific publications.

Most of the trend prediction/analysis papers in the Semantic Web area [1,2,15] adopt a *top-down* approach that primarily relies on the knowledge, intuition and insights of experts in the field. While undoubtedly these are very valuable assets, trend-papers that purely follow this approach risk focusing on major topics and trends alone while overlooking under-represented or emerging topics and trends. These shortcomings could potentially be addressed by (semi-) automatic, data-driven approaches, which identify research topics and trends in a *bottom-up* fashion from large corpora.

The primary goal of this paper is to provide a more complete picture of Semantic Web topics and trends in the last decade by relying on both top-down and bottom-up approaches. Our hypothesis being that *there is a high correlation between expert driven and data driven topic and trend analyses, however by combining both approaches it is possible to gain additional, valuable insights with respect to the Semantic Web research domain*. Starting from this hypothesis, we devise two primary research questions:

(1) Is it possible to identify the predominant Semantic Web research topics using both expert based predictions and topic and trend identification tools?
(2) What are the strengths and weaknesses of expert-driven and data-driven topic and trend identification methods?

In order to answer the aforementioned research questions we adopt a *mixed methods* research methodology [25], which involves the combination of quantitative and qualitative research methods, in order to gain better insights into Semantic Web topics and trends. Concretely, our study comprises three core tasks.

– Firstly, in a qualitative study we converge the findings of three top-down style seminal papers [1,2,15] at different points in time, into a unified *Research Landscape*.
– Secondly, we employ three alternative data-driven quantitative approaches in order to uncover topics and trends from a corpus of Semantic Web publications in a bottom-up fashion.
– Thirdly, we compare and contrast the topics derived from both the expert analysis and the data driven approaches, in order to provide a more holistic picture of Semantic Web research.

In order to enable the Semantic Web community to further build upon the results of our study, additional information about the resources described in this paper are available via `https://doi.org/10.5281/zenodo.1492693`.

The remainder of the paper is structured as follows: *Section* 2 provides an overview of existing work on automatic topic and trend analysis in the Semantic Web community. *Section* 3 describes the mixed methods methodology that guided our analysis. *Section* 4 provides a snapshot of the Semantic Web research community based on the observations of several domain specific experts [1,2,15]. This is followed by the presentation of the topic analysis of papers published in the main Semantic Web publishing venues over a 10 year period from 2006 to 2015 in *Section* 5. A discussion on the findings of our analysis is presented in *Section* 6. Finally, *Section* 7 concludes the paper and presents directions for future work.

2. Related Work

The analysis presented in this paper is situated within the field of *Scientometrics*, defined by Leydesdorff and Milojević [26] as the *"quantitative study of science, communication in science, and science policy"*. Although this research field is closely related to *Bibliometrics* (i.e., the application of statistical methods to books and other media of communication), and *Informetrics* (i.e., the study of the information phenomena), these terms are not necessarily synonymous [21]. In this section, we examine approaches for detecting and analyzing research topics, as a specific task within the Scientometrics landscape, with a primary focus on the contributions from the Semantic Web community.

Detecting topics that accurately represent a collection of documents is an important task that has attracted considerable attention in recent years leading to a variety of relevant approaches from different media sources, such as news articles [12], social networks [7], blogs [29], emails [28], to name but a few. A classical way to model the topics of a document is to extract a list of significant terms [6] (e.g., using tf-idf) and to cluster them [39]. Another common solution is the adoption of probabilistic topic models, such as Latent Dirichlet Allocation (LDA) [3] or Probabilistic Latent Semantic Analysis (pLSA) [20]. However, these generic approaches suffer from a number of limitations that often hinder their application for the task of detecting scientific topics. Firstly, they produce unlabeled bags of words that are often difficult to associate with distinct research areas. Secondly, the number of topics to be extracted needs to be known a priori. Finally, using such methods it is not possible to distinguish research areas from other kinds of topics contained in a document.

Therefore, several approaches were proposed to specifically address the problem of detecting research topics. For instance, Morinaga et al [28] present a method that exploits a Finite Mixture Model to detect research topics and to track the emergence of new topics. Derek et al [13] developed an approach that matches scientific articles with a manually curated taxonomy of topics that is used to analyse topics across different timescales. Chavalarias et al [8] propose a tool known as CorText that can be used to extract a list of n-grams from scientific literature and to perform clustering analysis in order to discover patterns in the evolution of scientific knowledge.

Topics can also be identified and analyzed with methods for bibliometric mapping, which focus on generating spatial representations of the interaction between disciplines, papers, and authors. In the last years we saw the emergence of several relevant tools, which leverage a variety of techniques, such as bibliographic coupling and co-author, co-citation, and co-word analysis. CiteSpace [9] is a long running application for identifying trends and patterns in scientific literature that can identify emerging topics by combining co-citation analysis and burst detection [24]. SciMAT[11] is an advanced science mapping analysis tool that incorporates several algorithms and measures and covers all the steps in the bibliometric mapping workflow. VOSViewer [41] is another well-known software for constructing and analyzing bibliographic networks. Jo et al [23] present a relevant approach that detects topics by combining distributions of terms with the citation graph related to publications containing these terms. A detailed comparison of several such tools can be found in [10].

Public tools for the exploration of research data usually identify research areas by using keywords as proxies (e.g., DBLP++ [14], Scival[1]), adopting probabilistic topic models (e.g., aMiner [40]) or exploiting handcrafted classifications (ACM[2], Microsoft Academic Search[3]).

However, all these solutions suffer from some limitations. For example, keywords are unstructured and usually noisy, since they include terms that are not research topics. In addition, the quality of keywords assigned to a paper varies a lot according to the authors and the venues. Probabilistic topic models produce bags of words that are often not easy to map to commonly known research areas within the community. Finally, handcrafted classifications are expensive to build, requiring multiple expertise, and tend to age very quickly, especially in a rapidly evolving field such as Computer Science.

The Semantic Web Community has also produced a number of tools and techniques that use semantic technologies for detecting and analyzing research topics. For instance, Bordea and Buitelaar [5] demonstrate how expertise topics extraction (with ranking and filtering) along with researcher relevance scoring can be used to build expert profiles for the task of expert finding. In a related work, Monaghan et al. [27] present their expertise finding platform Saffron based on the same principles, and demonstrate how it can be used to link expertise topics, researchers and publications, based on their analysis of the Semantic Web Dog Food (SWDF) corpus. The data is further enhanced with URIs and expertise topic descriptions from DBpedia and related information from the Linked Open Data (LOD) cloud. An alternative approach is adopted by the Rexplore system [31], an environment for exploring and making sense of scholarly data that integrates statistical analysis, semantic technologies, and visual analytics. Rexplore builds on Klink-2 [30], an algorithm which combines semantic technologies, machine learning and knowledge from external sources (e.g., the LOD cloud, web pages, calls for papers) to automatically generate large-scale ontologies of research areas. The resulting ontology is used to semantically enhance a variety of data mining and information extraction techniques, and to improve search and visual analytics. Hu et al. [22] demonstrate how Semantic Web technologies can be used in order to support scientometrics over articles and data submitted to the Semantic Web Journal as part of their open review process. Towards this end the authors provide external ac-

[1] http://www.elsevier.com/solutions/scival

[2] https://www.acm.org/publications/class-2012

[3] http://academic.research.microsoft.com/

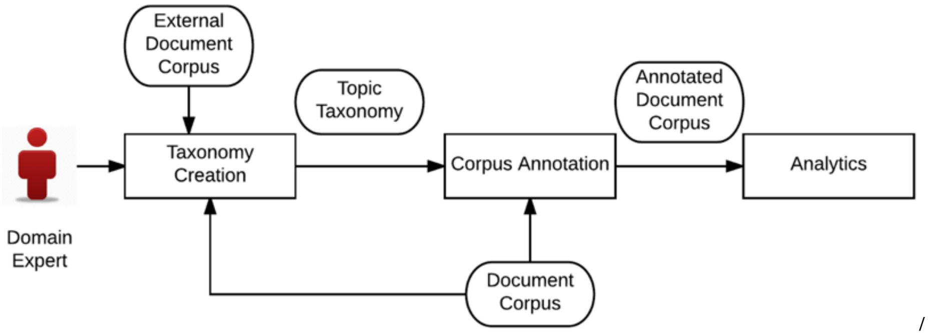


Fig. 1. Conceptual overview of topics detection approaches: main steps and data sources

cess to their semantified dataset, which is also linked to external datasets such as DBpedia and the Semantic Web Dog Food corpus. On top of this data they provide several interactive visualizations that can be used to explore the data, ranging from general statistics to depicting collaborative networks. Whereas Parinov and Kogalovsky [34] describe the Socionet research information system that focuses on linking research objects in general and research outputs in particular, the authors argue that information inferred from the semantic linkage of research objects and actors can be used to derive new scientometric metrics.

An interesting case of data-driven analysis is that reported in Glimm and Stuckenschmidt [19], looking back at the last 15 years of Semantic Web research through the lens of papers published at ISWC conferences from 2002 to 2014. The authors adopt an empirical approach to better understand the topics and trends within the Semantic Web community, in which they identify 12 key topics that describe Semantic Web research and then manually classify papers published in ISWC conference proceedings according to these topics. This work can also be categorized as a data-driven analysis of research topics and trends, which was performed completely manually.

Although data-driven approaches have been evaluated on their own, to date there is a lack of works that compare and contrast existing approaches, or indeed evaluate them with respect to expert-driven approaches. This paper fills this gap by adopting a holistic approach to topic and trend analysis, by analyzing the results of three expert-based and data-driven topic-detection approaches in the context of Semantic Web research.

3. Background and Methodology

In order to gain a better understanding of the topics and trends in the Semantic Web community over a ten year period from 2006-2015, we adopt a *mixed methods* approach to topic extraction and analysis, which combines both expert-based and data-driven approaches. According to Leech and Onwuegbuzie [25], the mixed methods research methodology involves the combination of quantitative and qualitative research methods in order to gain knowledge about some phenomenon under investigation. The mixed methods approach that guided the work carried out in this study is illustrated in *Figure* 2.

3.1. Seminal paper qualitative topic analysis

The goal of the qualitative analysis of the seminal papers was primarily to identify research topics mentioned in [1,2,15]. The work was conducted in a two step process. In `Step 1` each paper was read by three of the authors of this paper who were each tasked with identifying technical research topics mentioned in the three seminal papers (e.g., ontology, OWL). To keep the analysis as objective as possible, the authors extracted the exact wording used in the papers instead of using synonyms more familiar to them. Following on from this, the authors grouped extracted keywords into broader topic areas (e.g., ontologies and modeling, logic and reasoning). In order to reduce any bias, in `Step 2` the results of the aforementioned analysis were discussed and aligned during a consensus workshop. Where disagreement occurred with respect to the grouping of keywords the seminal papers were consulted in order to better understand the context of the topic, such that it was possible to reach consensus as to its categorization. The final outcome of the qualitative analysis is the unified *Research Landscape*, shown in *Table* 2 and discussed in detail in *Section* 4.

3.2. Semantic Web publications quantitative topic analysis

Rather than using a single topic and trend identification tool in `Step 3` we elected to perform the analysis of a corpus of Semantic Web publications with three different tools (i.e., PoolParty[4], Rexplore[5], and Saffron[6]), such that we could compare and contrast the results obtained via the different tools.

Semantic Web Venues (SWVs) corpus: The corpus, which was analyzed by each of the tools, comprises papers from five enduring international publishing venues dedicated to Semantic Web research, namely: the International Semantic Web Conference (ISWC), the Extended Semantic Web Conference (ESWC), the SEMANTiCS conference, the Semantic Web Journal (SWJ) and the Journal of Web Semantics (JWS), over a 10 year period from 2006 to 2015 inclusive. These publishing venues were chosen as they are dedicated to Semantic Web research and have been running continuously for several years. Although, the SEMANTiCS conference was traditionally seen as a more business oriented event, it also has a strong academic component, with high overlap between the organizing and program committee members and the various committees and boards of the other publishing venues. The corpus contained 2,045 papers in total (1,472 conference papers and 573 journal papers). For ease of readability this corpus is simply referred to as the SWVs corpus in the rest of the paper.

A conceptual topic extraction and analysis workflow: Generally speaking, the typical topic extraction and analysis workflow, as depicted in *Figure* 1, is composed of the following sequential steps:

Taxonomy creation involves the creation of a `topic taxonomy` that guides the analysis process. In practice, this step can be achieved manually by domain experts, or automatically with the taxonomy being learned either from the `document corpus` of interest or from a larger `external document corpus`.

Corpus Annotation concerns the annotation of the `document corpus` in terms of the taxonomy topics. Different annotation approaches range from manually assigning each paper in a corpus to the most representative topics, annotating the document abstracts with the relevant topics, or annotating the entire text of the paper based on a topic list or hierarchy.

Analytics refers to various analytical activities that can be conducted over the `annotated document corpus`. For instance, document classification, trend detection, expert profiling and recommendations.

Data-driven topic extraction and analysis tools: Although all three tools conducted their analysis over the same corpus, each of them employ different approaches to topic extraction. An overview of the approaches adopted by PoolParty, Rexplore, and Saffron with respect to the main steps depicted in *Figure* 1 is summarized in *Table* 1 and described below:

PoolParty is a semantic technology suite that supports the creation and maintenance of thesauri by domain experts [38]. Although PoolParty is a commercial product, a free version, which was made available in the context of the PROPEL project[7] [16], was used to perform the analysis described in this paper. In the case of the analysis described in this paper the taxonomy was created from conference and journal metadata (i.e., call for papers, sessions, tracks, special issues etc.), which have been manually curated by experts from the Semantic Web community (i.e., conference organizing committee and editorial board members). In order to reduce the potential for bias during the taxonomy construction, the classification, which was performed in the context of the PROPEL project, was collectively performed by five Semantic Web experts. The topic frequency analysis was subsequently conducted by PoolParty over the full text of the research articles from the SWVs corpus, without any parameterization.

Rexplore is an interactive environment for exploring scholarly data that leverages data mining, semantic technologies and visual analytics techniques [31]. In the context of this paper, we used Rexplore for tagging research papers with relevant research topics from the Computer Science Ontology (CSO) [35], an existing ontology of research areas that was automatically generated from a large computer science corpus. The ap-

[4] PoolParty, https://www.PoolParty.biz/system-architecture/

[5] Rexplore, http://skm.kmi.open.ac.uk/rexplore/

[6] Saffron, http://saffron.insight-centre.org/

[7] PROPEL, https://www.linked-data.at/

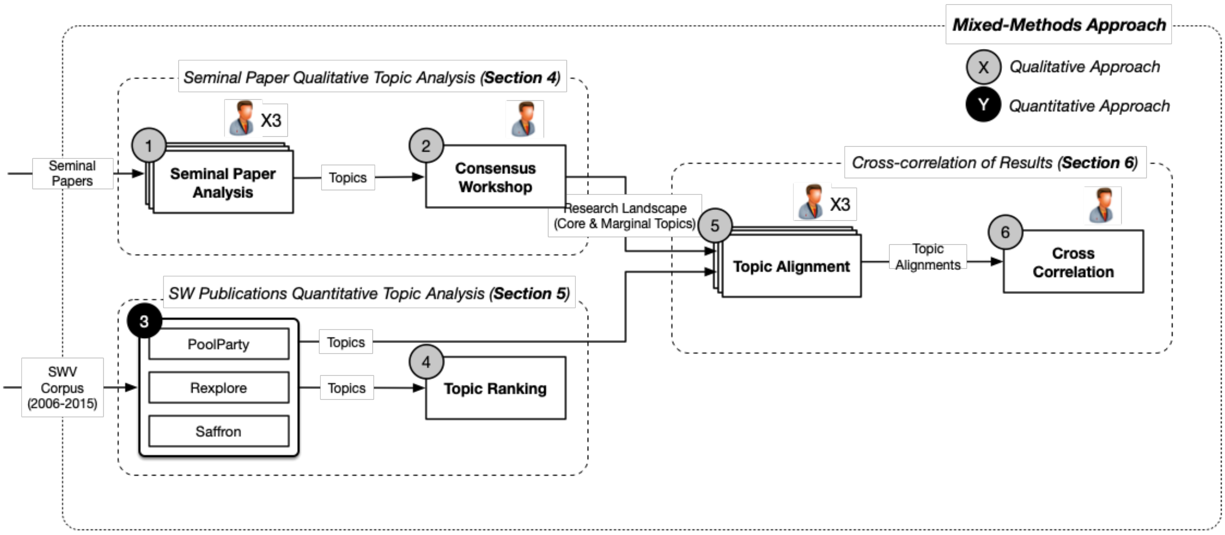


Fig. 2. Overview of the mixed methods-based methodology.

Table 1

Comparison of the methods and data sets used by various topic and trend analysis tools.

Tool	Taxonomy Creation	Topic Taxonomy	Document Corpus	Corpus Annotation	Topic Analysis	Other Analytics
PoolParty	Manual	Fairly broad/deep	SWVs 2006-2015	Automatic (full-text)	Topic frequency in text	Taxonomy extension
Rexplore	Automatic from broader external corpus	17K topics in CS, 96 topic in SW, 9 levels deep	SWVs 2006-2015 Scopus 2006-2015	Automatic (abstracts, titles, keywords)	Number of papers and citations associated with a topic	Taxonomy learning, expert profiling
Saffron	Automatic from the document corpus	Fairly broad/deep	SWVs 2006-2015	Automatic (full-text)	Topic frequency and semantic relatedness	Taxonomy learning, expert finding, document classification and search

proach for tagging the publications, which took into consideration their title, keywords, abstract and citations, is a slight variation of the method adopted by Springer Nature for characterizing semi-automatically their Computer Science proceedings [32]. The analysis involved the generation of statistical information based both on the number of papers and the citations associated with a topic. No special parameterization was used by the Rexplore in the context of this study. Rexplore was applied both on the SWVs corpus and on a more comprehensive dataset including 32,431 publications associated to the Semantic Web. The aim of this additional analysis was to assess if the set of papers published in the main venues present a different topic distribution than the set of all papers about the Semantic Web.

Saffron is a topic and taxonomy extraction tool whose main applications include expert finding, document classification and search [27]. In the context of this paper, we used Saffron's Natural Language Processing (NLP) techniques to extract domain-specific terms based solely on the full text of articles in the SWVs corpus, and a novel taxonomy generation algorithm that uses a global generality measure to direct the edges from generic concepts to more specific ones, in order to construct a topical hierarchy. Additional details on the algorithms used for term (topic) extraction and for extraction of a topic taxonomy can be found in [4]. The topic frequency and relatedness analysis was conducted automatically by Saffron over the SWVs corpus without the need of any additional corpora. Based on previous studies conducted by the Saffron team, in terms of parameterization the taxonomy was limited to 500 topics and topics that appear in at least 3 papers.

In `Step 4` we performed a syntactic analysis of the top forty topics extracted by each of the data-driven tools. Both singular and plural representations of a topic were treated as the same topic. Additionally, topics with a high syntactic correlation were treated as the same, for instance *knowledge base* and *knowledge based systems*. A detailed description of the respective analysis performed by PoolParty, Rexplore and Saffron and the cross correlation of topics is presented in *Section* 5.

3.3. Cross correlation of results

The final stage of our analysis involved the alignment of the topics identified by Rexplore, PoolParty and Saffron with the *Research Landscape* topics emerging from the analysis of the seminal papers. In `Step 5` the output of each of the three data-driven approaches was mapped by one of the authors of this paper to the topics of the *Research Landscape*. The principles used to guide the mapping process, which involved a combination of syntactic and semantic matching, can be summarized as follows:

Exact syntactic match: is the most straightforward case as topics that have exactly the same label (e.g., *Linked Data*) are already aligned.

Partial syntactic match: refers to cases where two topics have similar but not exactly matching labels, however clearly refer to the same body of research. For instance, *Description Logics* is a subtopic of *Logic and Reasoning*.

Semantic match: denotes topics that have syntactically completely disjoint labels but they are semantically related. Links between syntactically different labels are often recorded in our extended *Research Landscape* document, where several keywords were assigned to a larger overlapping topic. For example, we assigned keywords such as SPARQL to the *Query Languages* topic.

No match: is used to represent topics identified by the data-driven approaches that are completely new and cannot be related to any of the topics of the *Research Landscape*.

In order to reduce any bias, in `Step 6` individual topic alignments were cross-checked by the two additional authors and further discussed during an analysis and cross-correlation workshop. The results of this workshop are depicted in *Tables* 3- 6 and further discussed in *Section* 6.

4. Seminal Papers Topic and Trend Analysis

In the Semantic Web area, a handful of well-known papers identify research topics and discuss trends within the community [1,2,15]. Some of these papers predict future topics [1,2], while others reflect on research topics in the past years or in the present [2,15].

4.1. The seminal papers

At the turn of the millennium (2001), Berners-Lee et al. [1] coined the term "Semantic Web" and set a research agenda for a multidisciplinary research field around a handful of topics.

Six years later, Feigenbaum et al. [15] analyzed the uptake of Semantic Web technologies in various domains as of 2007. In doing so, they provided a picture of the technologies available at that time as well as the main challenges that these technologies could solve. The authors took a reflective rather than predictive stance in their work. On the 15-year anniversary of the Semantic Web community, Bernstein et al. [2] provide their vision of research beyond 2016 by grounding their predictions in an overview of past and present research. Therefore, their paper is both reflective of past/present work and predictive in terms of future research.

Each of the vision papers mentioned above are primarily based on the expert knowledge of the authors and reflect their views, without aiming to be complete. Our objective is to use the topics identified in these seminal papers as a baseline for a comparison with the output of the three data-centric topic identification methods discussed in this paper. Note that, unlike in information retrieval research, the proposed *Research Landscape* (cf. *Table* 2) is by no means an absolute gold-standard that should be achieved, but rather acts as an intuitive comparison basis for understanding the strengths and weaknesses of expert-driven versus data-driven topic identification methods.

4.2. Core topics from the seminal papers

After manually annotating research topics discussed in each of the seminal papers, we aligned the identified topics across papers, and observed eleven *core research topics* that are mentioned by two or three of the seminal papers (cf. *Table* 2). All three papers agree on the following eight core research topics:

Knowledge representation languages and standards, such as XML, RDF and a so-called Seman-

Table 2

Research Landscape: Core and Marginal topics discussed in the seminal papers. Topics in () were only intuitively mentioned.

	Berners-Lee et al. [1] Future	Feigenbaum et al. [15] Past (2000-2007)	Bernstein et al. [2] Past (2000-2016)	Bernstein et al. [2] Future from 2016
Core topics	knowledge representation languages and standards	knowledge representation languages and standards	knowledge representation languages and standards	representing lightweight semantics
	ontologies and modeling, taxonomies, vocabularies	ontologies and modeling, taxonomies, vocabularies	ontologies and modeling, (PR) knowledge graphs	-
	logic and reasoning	logic and reasoning	logic and reasoning	-
	search and question answering	(ranking)	(PR) question answering systems	-
	(data integration)	(ontology matching)	(PR) needs-based, lightweight data integration	integration of heterogeneous data
	proof & trust	privacy, trust, access control	personal information, privacy	trust & data provenance (representation, assessment)
	databases	semantic web databases	database management systems	-
	decentralization	(decentralization)	vastly distributed heterogeneous data	(decentralization)
	(machine learning, prediction, analysis, automatic report)	knowledge extraction and discovery	latent semantics, knowledge acquisition, ontology learning	-
	-	query language (SPARQL)	developing efficient query mechanisms	-
	-	(linked data, DBpedia)	(PR) linked data (open government data), (social data)	-
Marginal topics	intelligent software agents	-	multilingual intelligent agents	-
	(Internet of Things)	-	-	high volume and velocity of data, e.g., streaming & sensor data
	-	(scalability, efficiency, robust semantic approaches)	-	scale changes drastically
	(semantic web services)	-	-	-
	-	visualization	-	-
	-	change management and propagation	-	-
	-	(social semantic web, FOAF)	-	
	-	-	-	data quality, e.g., representation, assessment

tic Web language, were considered crucial to enabling the vision of intelligent software agents by Berners-Lee et al. [1]. Work on the development of web-based knowledge representation languages (now also including OWL) continued over the next 7 years [15]. By 2016 this was seen as a core line of research extending also to the standardisation of representation languages for services [2]. As for the future, Bernstein et al. [2] predict that knowledge representation research will focus on representing lightweight semantics, dealing with diverse knowledge representation formats and developing knowledge languages and architectures for an increasingly mobile and app-based Web.

Knowledge structures and modeling. Berners-Lee et al. [1] consider knowledge structures such as ontologies, taxonomies and vocabularies as essential components of the Semantic Web. Follow up papers confirm active research on the creation of ontologies [2,15]. While, Bernstein et al. [2] introduce knowledge graphs as novel knowledge representation structures.

Logic and Reasoning. Berners-Lee et al. [1] assumed that inference rules and expressive rule languages would enable logic-based automated reasoning on the Semantic Web. Their prediction was abundantly confirmed in follow-up papers: Feigenbaum et al. [15] reporting work on the development of inference engines for reasoning by 2007; and Bernstein et al. [2] confirming work on developing tractable and efficient reasoning mechanisms.

Search, retrieval, ranking, and question answering. Besides intelligent agents, Berners-Lee et al. [1] predicted that search and question answering programs would also benefit from the Semantic Web. In 2007, Feigenbaum et al. [15] indirectly refer to this topic in the context of ranking, however this research topic becomes increasingly important according to Bernstein et al. [2] who describe work on question answering systems based on semantic markup and linked data from the Web (e.g., IBM Watson).

Matching and Data Integration. Ontology matching and data integration were already intuitively mentioned, but not concretely named, by Berners-Lee et al. [1]. Data integration played an important role in many commercial applications developed up until 2007 and opened up the need for change management and change propagation across integrated data sets [15]. By 2016, a new trend towards needs-based, lightweight data integration is observed [2]. For the future, Bernstein et al. [2] discuss the need to integrate heterogeneous data as part of the broader topic of data management.

Privacy, Trust, Security, and Provenance. Berners-Lee et al. [1] envision proofs and digital signatures as key aspects of the Semantic Web in order to enable more trustworthy data exchange and the topic of privacy was also mentioned in 2007 [15]. According to Bernstein et al. [2] future work should focus on the representation and assessment of provenance information, as part of the broader topic of data management.

Semantic Web Databases. Similarly to Berners-Lee et al. [1], Feigenbaum et al. [15] discuss research topics around the development of Semantic Web tools as instrumental for commercial uptake, especially ontology editors (e.g., Protégé) and Semantic Web databases (e.g., triple stores). According to Bernstein et al. [2] many of these tools evolved into commercial tools by 2016.

Distribution, decentralization, and federation. Berners-Lee et al. [1] envisioned that the Semantic Web would be as decentralized as possible, bringing new interesting possibilities at the cost of losing consistency. Feigenbaum et al. [15] exemplified one of these novel scenarios by mentioning FOAF as an example of a decentralized social-networking system. Bernstein et al. [2] commented on this topic briefly, confirming that modern semantic approaches already integrate distributed sources in a lightweight fashion, even if the ontologies are contradictory.

Besides the aforementioned core topics, three important topics were not predicted by Berners-Lee et al. [1], but were mentioned by the other two papers:

Knowledge extraction, discovery and acquisition. In 2007, Feigenbaum et al. [15] hint at this topic with terms such as machine learning, prediction and analysis. Automatic knowledge acquisition was boosted by more powerful statistical and machine learning approaches as well as improved computational resources [2]. For the future, Bernstein et al. [2] identify a need for new techniques to extract latent, evidence-based models (ontology learning), to approximate correctness and to reason over automatically extracted ontologies/knowledge structures. An increasing importance is given to using crowdsourcing for capturing collective wisdom and complementing traditional knowledge extraction techniques.

Query Languages and Mechanisms. By 2007, research also focused on the development of query languages, most notably SPARQL [15] and developing efficient query mechanisms [2].

Linked Data. By mentioning DBpedia, Feigenbaum et al. [15] intuitively pointed to the future research topic of Linked Data. This topic became well established by 2016 and a new wave of structured data available on the web (e.g., open government data, social data) further extended research on the Linked (open) Data topic [2].

4.3. Marginal topics from the seminal papers

Our analysis also identified several marginal topics, mentioned by two of the seminal papers (*Table* 2), as follows:

Intelligent software agents. The underpinning theme of Berners-Lee et al. [1]'s vision paper was *intelligent software agents* that would provide advanced functionality to users by being able to access the meaning of Semantic Web data. Interestingly, this topic has not been mentioned until recently, when Bernstein et al. [2] discuss work on training conversational intelligent agents based on multilingual textual data on the web.

Internet Of Things. The application of Semantic Web to physical objects within the context of the future Internet Of Things (IoT) was intuitively mentioned by Berners-Lee et al. [1]. This topic was not mentioned by any of the follow-up papers, even thought it is considered to play an important role in the future. Indeed, Bernstein et al. [2] predict that dealing with high volume and velocity data will be necessary due to the increased number of streaming data sources from sensors and the IoT. They envision techniques for the selection of streaming data (data triage), for decision-making on streaming sensor data as well as the integration of streaming sensor data with high quality semantic data.

Scalability, efficiency and robustness. Feigenbaum et al. [15] position *scalability, efficiency and robust semantic approaches* as key factors needed to address Semantic Web challenges, in particular integration, knowledge management and decision support. In turn, Bernstein et al. [2] recognize that new research is needed given that the *scale changes drastically*.

Semantic Web Services. Berners-Lee et al. [1] also envisioned the applicability of Semantic Web technologies for advertising and discovering web-services.

Human-Computer Interaction. Feigenbaum et al. [15] mention visualization as features of user-centric applications.

Change management and propagation. Feigenbaum et al. [15] mention or hint that *change management and change propagation* across integrated data sets is needed to accompany data integration research.

Social semantic web. Although predicting future trends was not their explicit goal, by mentioning FOAF Feigenbaum et al. [15] intuitively pointed to the future research topic on the *Social Semantic Web*.

Data quality Under the heading of data management, Bernstein et al. [2] group work on data integration, data provenance and new technologies that should allow representing and assessing *data quality*, such as task-focused quality evaluation (e.g., is a resource of sufficient quality for a task?).

4.4. Trends

Although the seminal papers focus primarily on research topic identification, they also offer some hints on the way these topics evolve over time (i.e., trends).

In 2001, Berners-Lee et al. [1], used a fictitious scenario to describe a vision of a web of data that can be exploited by *intelligent software agents* that carry out data centric tasks on behalf of humans. Additionally the paper identifies the infrastructure necessary to realize this vision focusing on four broad areas of research, namely: *expressing meaning*, *knowledge representation*, *ontologies* and *intelligent software agents*.

In 2007, Feigenbaum et al. [15] reflected on the ideas presented in [1] and highlighted that although the original autonomous agent vision was far from being realized, the technologies themselves were proving to be highly effective in terms of tackling *data integration* challenges in enterprises especially in the life sciences and health care domains. Furthermore, the authors highlighted that consumers were starting to adopt *FOAF* profiles and to embrace *decentralized* social-networking. However, they also point to new *privacy* concerns when linking disparate data sources.

In 2016, discussing present research topics, Bernstein et al. [2] noted a large spectrum between two opposite research lines on expressivity and *reasoning* on the Web on the one hand and ecosystems of *Linked Data* on the other. Particularly notable is the adoption of Semantic Web technologies in several large, more applied systems centered around *knowledge graphs*, which use Semantic Web representations yet ensure the functionality of applied systems which resulted in less formal and precise representations than expected at the earlier stages of Semantic Web research. Based on these considerations, the authors predict moving from logic-based to evidence-based approaches in an effort to build truly *intelligent applications* using vast, *heterogeneous*, *multi-lingual* data.

5. Semantic Web Publications Topic and Trend Analysis

In this section we describe the results of topic and trend analysis by employing data-driven tools. The bottom up analysis was performed with three different tools (i.e., PoolParty, Rexplore, and Saffron) that enable users to gain insights into the various research topics that appear in research papers published at popular Semantic Web publishing venues.

5.1. PoolParty quantitative analysis

The analysis conducted by PoolParty was based on a coarse grained taxonomy of 3,420 unique dictionary topics (that were crowd-sourced from experts in the community in the form of conference and journal metadata), which was generated by assigning each topic to one or more foundational technologies worked on by the community.

The chart presented in *Figure* 3 provides details on the % coverage for each of the eighteen foundations, across the five venues for the 10-year timeframe under examination. As expected, *Knowledge Representation & Data Creation/Publishing/Sharing* is the top foundation, with almost 23% of the total occurrences in all documents. This foundation includes several topics that are fundamental to the Semantic Web community (i.e., the ability to represent semantic data and to publish and share such data). Next in order of importance, the management of such knowledge (*Data Management*) and the construction of feasible systems (*System Engineering*), constitute almost 16% and 11% of the occurrences, respectively. Important functional areas such as *Searching, Browsing & Exploration*, *Data Integration and Ontology/Thesaurus/Taxonomy Man-*

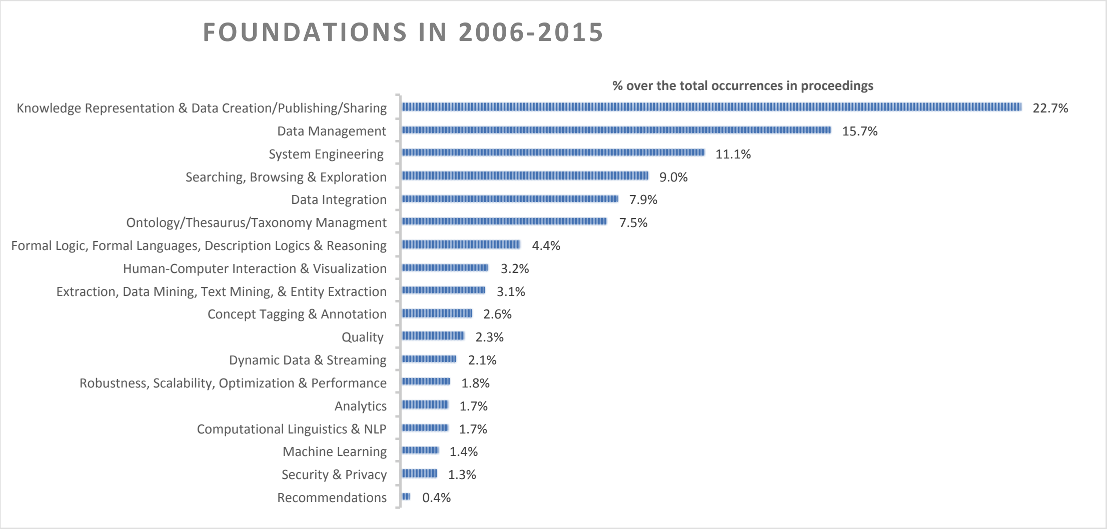


Fig. 3. PoolParty: % coverage per foundational technology across the 5 venues for the 10-year timeframe

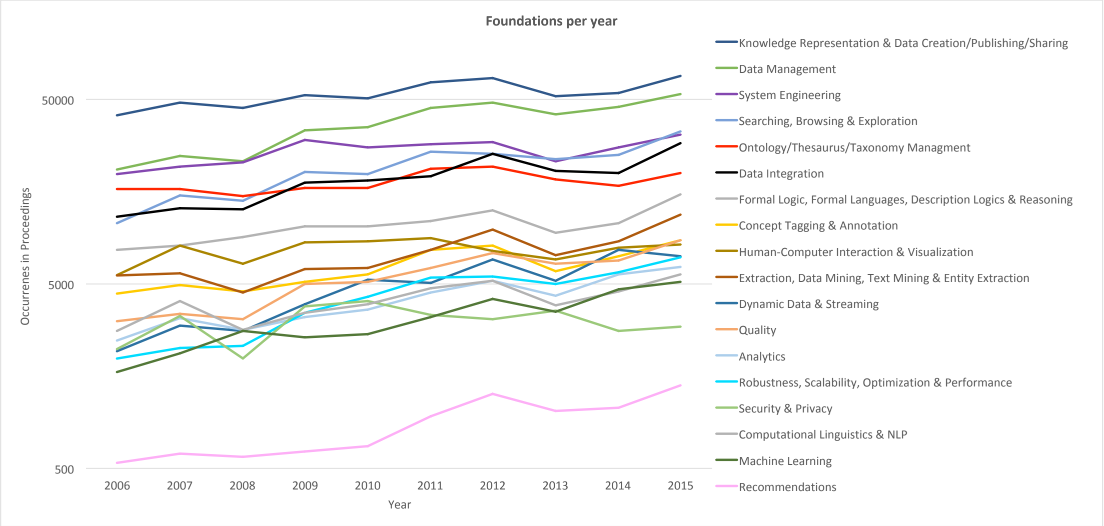
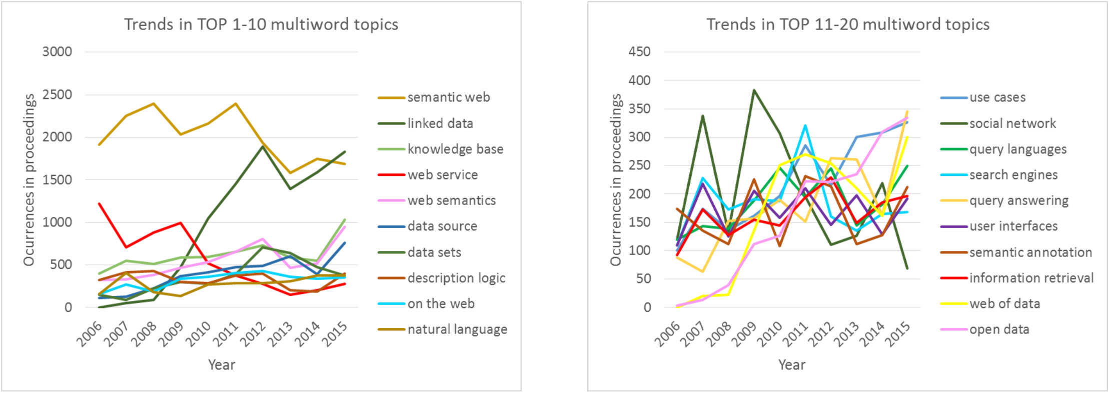


Fig. 4. PoolParty: Growth/Decline of foundational technologies across the 5 venues for the 10 year timeframe

agement also figure strongly in comparison to the other foundations (all of them with more than 7.5% occurrences). In contrast, very specific topics, such as *Formal Logic, Formal Languages, Description Logics & Reasoning*, and *Concept Tagging & Annotation* represent a modest 4.4% and 2.6% respectively, and cross-topics, such as *Human Computer Interaction & Visualization*, *Machine Learning*, *Computational Linguistics & NLP*, *Security & Privacy*, *Recommendations*, and *Analytics* are only marginally represented. Topics that relate to *Quality*, *Dynamic Data & Streaming*, and *Robustness, Scalability, Optimization & Performance* are also under-represented (at around 2%).

In order to gain some insights into the research trends over the last decade, *Figure* 4 depicts the growth/decline of each of the foundations over the 10-year timeframe. Although the general trend for all topics shows year on year increases, we note that *Robustness, Scalability, Optimization & Performance*, *Dynamic Data & Streaming*, *Searching, Browsing & Exploration*, and *Machine Learning* have increased by more than 200% since 2005. In contrast, *Security & Privacy*, and *Ontology/Thesaurus/Taxonomy Manage-*



(a) top 10 multi-word topics

(b) top 11-20 multi-word topics

Fig. 5. PoolParty: Growth/Decline of the (a) top 10 and (b) top 11-20 multi-word topics across the 5 venues for the 10 year timeframe.

ment have had marginal growth of only 30% for the same period.

Figure 5 focuses on the growth/decline of the top 20 multi-word topics. Interestingly, results show a sharp increase of *Linked Data* at the expense of *Semantic Web*. Note also that *Natural Language* is in the top-10 multi-word topics, even though this is a cross topic which may be more represented in a different community. Finally, the decrease in the occurrence of *Web Services* can also be seen here.

5.2. *Rexplore quantitative analysis*

Rexplore characterizes topics according to the Computer Science Ontology (CSO) [8] [35], which is a large-scale automatically generated ontology of research areas. Since it is interesting to compare the trends exhibited by high-tier domain conferences with the ones appearing in the full literature, we analysed both the SWVs corpus (described in *Section* 3) and a more comprehensive dataset (here labelled Full Semantic Web, FSW) containing 32,431 publications associated with the topic Semantic Web or with its 96 associated subtopics in CSO (e.g., Linked Data, RDF, Semantic Web Services) from a dump including all Scopus Computer Science papers in the interval 2006-2015.

The analysis presented here follows the Expert-Driven Automatic Methodology (EDAM) [33] for performing systematic reviews of scholarly articles. EDAM is a methodology that reduces the amount of manual tedious tasks involved in systematic reviews by 1) applying data driven methods for automatically generating an ontology of research areas, 2) revising it with domain experts, and 3) using it to annotate papers and produce relevant analytics. The papers were associated to a topic if they contained in the title, abstract, or keywords: 1) the label of the topic (e.g., “Semantic Web”), 2) a *relevantEquivalent* of the topic (e.g., “Semantic Web Tecnologies”), 3) a *skos:broaderGeneric* of the topic (e.g., “ontology matching”), or 4) a *relevantEquivalent* of any *skos:broaderGeneric* of the topic (e.g., “ontology mapping”)[9]. We chose this straightforward approach instead of other more complex methods based on string similarity [36] or word embeddings [37], since it is simple to reproduce and yields the best precision, as discussed in Salatino et al. [37].

Figure 6 shows the main research fields addressed by the Semantic Web papers in both SWVs and FSW, ranked by the percentage of their publications in the field of Semantic Web. We excluded from this view any super and sub areas of Semantic Web that will be discussed later in detail. Unsurprisingly, the topic *Ontology* appears in about 61.2% of the papers (55.3% for FSW), followed by *Artificial Intelligence* (35.1%, 27.2%), *Information Retrieval* (32.7%, 25.2%), *Query Languages* (26.5%, 17.1%) and *Knowledge Base System* (17.5%, 12.7%). Interestingly, these five core research areas appear more often in the main venues (+7.1% in average), but they are also very important areas for the FSW dataset. Other research areas appear more prominently in one of the datasets. The *Query Language* area is much more frequent in the

[8] https://cso.kmi.open.ac.uk

[9] A detailed description of the relevant semantic relationships is available in Salatino et al. [35].

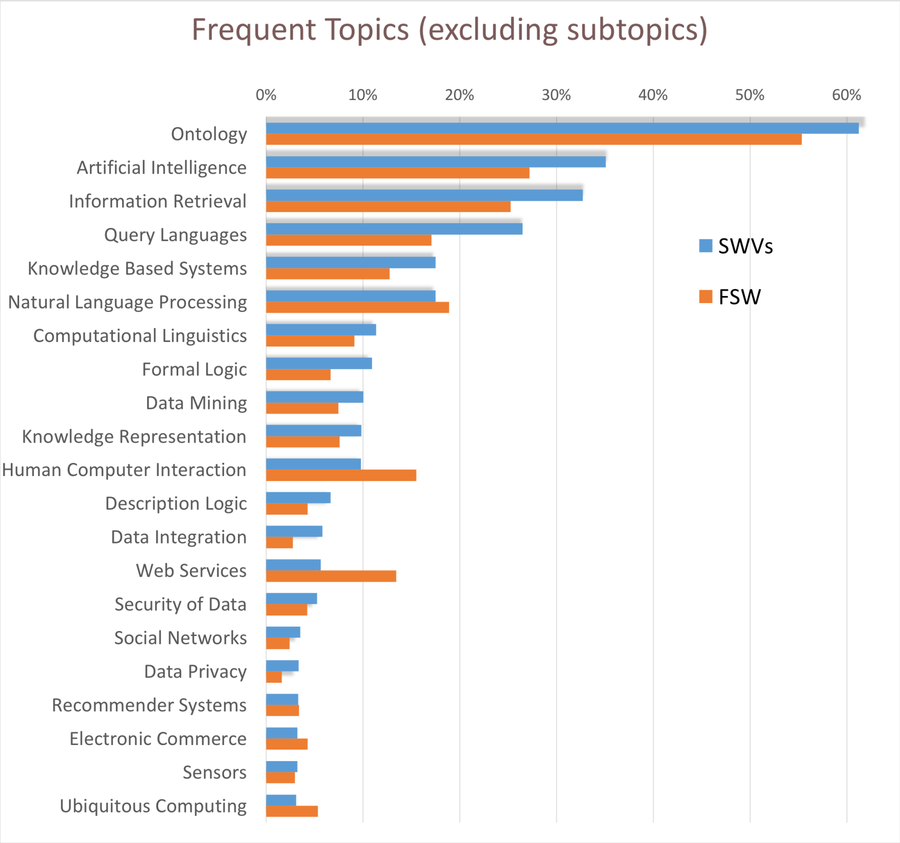


Fig. 6. Rexplore: Frequent topics (excluding subtopics) in SWVs (blue) and FSW (red).

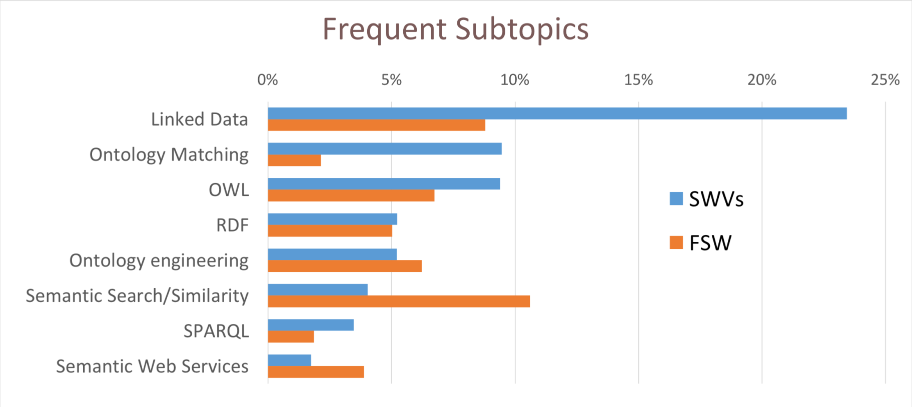


Fig. 7. Rexplore: Frequent Semantic Web subtopics in SWVs (blue) and FSW (red).

SWVs, probably due to the fact that the main venues traditionally are focused on Semantic Web query languages, such as SPARQL. *Formal Logic* has a similar behavior (10.9%, 6.6%), suggesting a stronger focus of the main venues on this topic. Conversely, other research fields appear more often in the FSW dataset. This is the case of *Natural Language Processing* (17.5%, 18.9%), *Human Computer Interaction* (9.8%, 15.5%), *Web Services* (5.6%, 13.4%), *Electronic Commerce* (3.2%, 4.3%) and *Ubiquitous Computing* (3.1%, 5.3%). This seems to suggest that there is a good amount of research in the intersection of

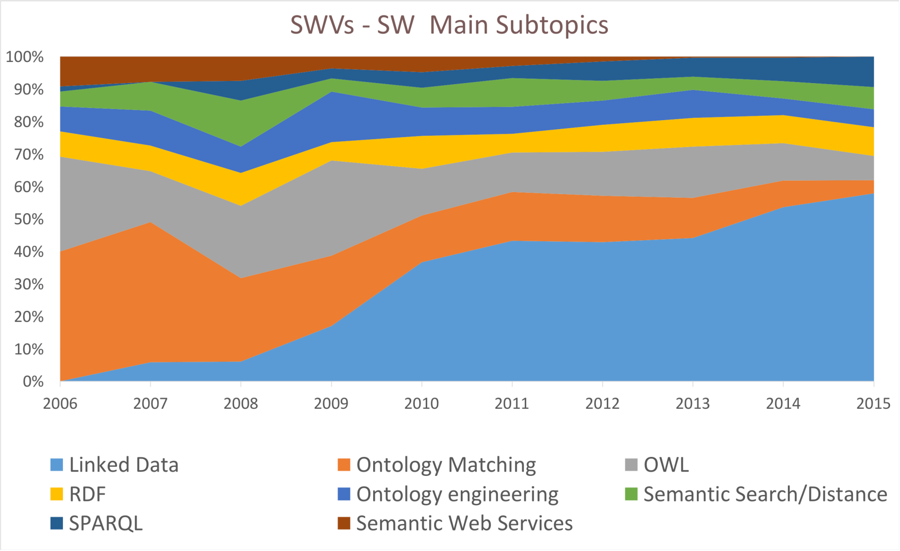


Fig. 8. Rexplore: Number of publications associated with eight Semantic Web subtopics in SWVs.

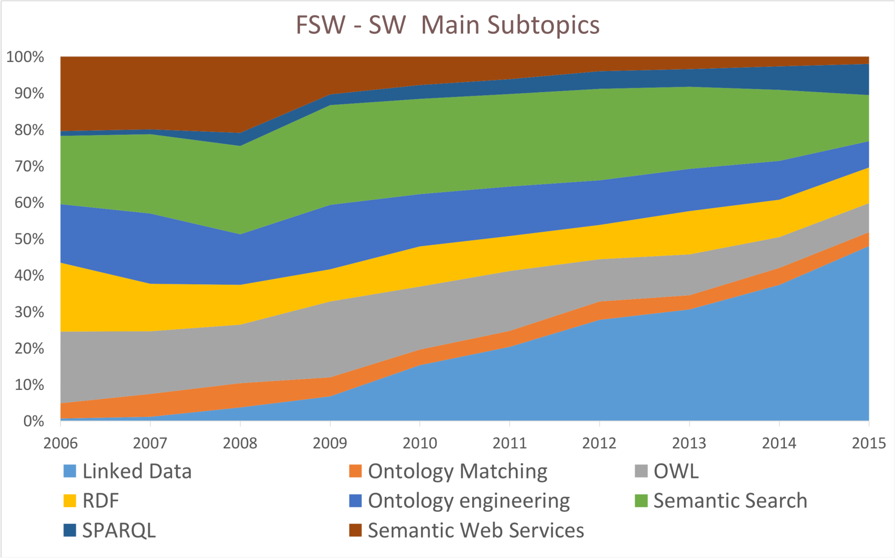


Fig. 9. Rexplore: Number of publications associated with eight Semantic Web subtopics in FSW.

these topics and Semantic Web that is not fully represented in the main venues.

The Semantic Web field subsumes several heterogeneous research areas dealing with different aspects of its vision. *Figure* 7 shows the popularity of the main Semantic Web direct subtopics in the two datasets. We include in this view also the area of *Ontology Engineering*, which is not formally a sub-topic of Semantic Web, since a very large portion of its outcomes are published in the main Semantic Web venues. It is again interesting to consider the difference between the datasets. The topics *Linked Data* (23.4, 8.8%), *Ontology Matching* (9.5%, 2.1%), *OWL* (9.4%,6.7%), and *SPARQL* (3.5%,1.9%) are more frequent in the main venues. Conversely publications addressing *Semantic Search* (4.0%, 10.6%) and *Semantic Web Services* (1.7%, 3.9%) are more popular outside these venues.

Figure 8 and *Figure* 9 show the popularity of the main sub-topics over the years. The two main dynamics, evident in both datasets, are the fading of Semantic Web Services and the rapid growth of Linked Data and to a lesser extent of SPARQL. Indeed, *Semantic Web Services* is one of the main areas in 2004, and an integral part of the initial Semantic Web vision [1]. However, the number of papers about these topics consistently decreases and from 2013 there are almost no publications about them in the SWVs corpus and very few in FSW. The second trend is the steady growth of Linked Data from 2007. In 2015 about half of Semantic Web papers in the main venues refer to this topics. Interestingly, both trends are first anticipated by the main venues, and only later evident also in the FSW dataset. It thus seems that the tendencies of the main venues influence in time all the Semantic Web research.

5.3. Saffron quantitative analysis

Saffron employs a domain-independent approach to topic extraction, which is one of its biggest advantages compared to most systems in the area, in that it does not require external domain-specific classifications. Such information is often not readily available especially in niche domains, and creating a classification is very costly in terms of time, human expertise needs, and maintenance. Saffron bypasses this barrier by automatically building a domain model from the input corpus itself, and by capturing the expertise knowledge of the corpus by isolating its most generic concepts. The constructed hierarchical taxonomy can be visualized as a graph. We use Cytoscapes, an open source software tool for complex networks graph visualization[10]. It allows us to perform a network analysis on the output provided by Saffron, and a customization of the layout. In our case, the size and the color of the nodes are proportional to the number of neighbors each topic is connected to.

Figure 10 shows the general picture of the graph displaying the interconnected topics from the results of the analysis. The size and the colour of the nodes in the graph are related to the number of edges that are connected to them, ie. the bigger nodes with red shades are the most connected topics while the smaller and blue nodes are the leaves of the tree. The first and predominant node (i.e., the root of the taxonomy) is the *Semantic Web* topic itself. Around it, several main clusters with major keyphrases emerge, including: *RDF Data* and *Linked Data*, followed by *Natural Language*, *Data Source* and *Reasoning Task*. A strong focus is also put on *Machine Learning*, *Ontology Engineering*, *Query Execution*, and the mark-up language *OWL-S*. By concentrating on the clusters, we identify the importance of data in terms of its representation (*RDF Data*, *RDF Graph*, *Linked Data*), its accessibility (*Open Data*), and its querying (*Query Execution*, *Query Processing*). Some other main interests in the domain are visible, represented by a cluster made up of *Natural Language* and topics related to the querying of information such as *Semantic and Keyword Search*, *Keyword Query*, *Semantic Similarity* or *Information Retrieval*. *Natural Language* is also connected to another dominating topic that is *Machine Learning*, associated to *Ontology Matching and Mapping*. One of the branches originating from the *Semantic Web* topic brings together concepts related to the structure and representation of the ontology (*Knowledge Base*, *Knowledge Representation*, *Ontology Language*, *OWL Ontology*), while a sub-branch leads to logic and reasoning related topics (*Description Logic*, *Reasoning task*, *Reasoning Algorithm*). The *Ontology Engineering* node is related to topics such as *User Interface*, *Ontology Development* and *Ontology Editing*.

As demonstrated above, the main nodes are at the centre of clusters of topics that are semantically related to them. In the following analysis, we focus thus on the evolution of those major terms, which are the most prominent for a cluster. We selected the top 20 topic terms (i.e., the most connected ones) and observe the distribution of their use in the SW corpus. The two charts in *Figure* 11 show the percentage of documents containing the aforementioned topics (i.e., the number of documents where the term appear at least once), per year. We observe that *Semantic Web* as a topic is on the decline with a decrease of 20% between the beginning and the end of the studied time frame. It is still the most used topic nonetheless, reaching 91% of distribution in the documents in 2006 and lowering down to 70% in 2015. The reasons for this decline could be manifold: the term/field may be so established that it is not named explicitly in the papers anymore or the community is trying to re-brand their research with new terms such as *Linked Data*.

Indeed, the most significant progression is the use of the term *Linked Data*. While it was completely missing in 2006, it experienced a very rapid growth in particular between 2008 and 2010 where its rise was 9-fold, to eventually reach 64% of the distribution in the documents by 2015. Similarly, the *Open Data* topic increased from about 1% in 2006/2007 to 45% in 2015.

[10] Cytoscapes, http://www.cytoscape.org/

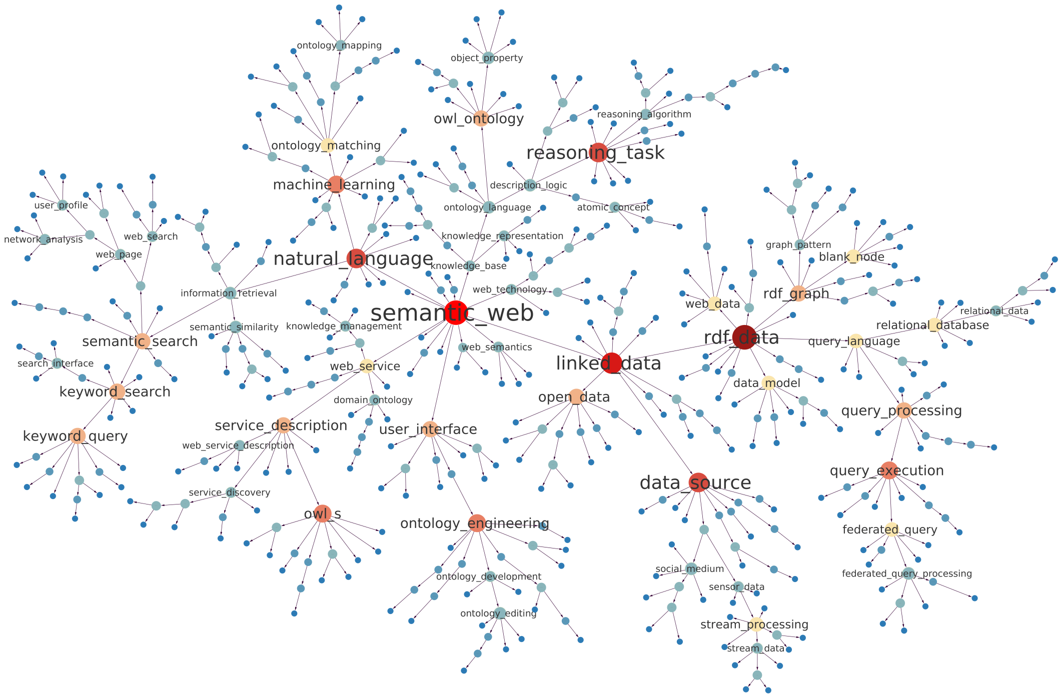
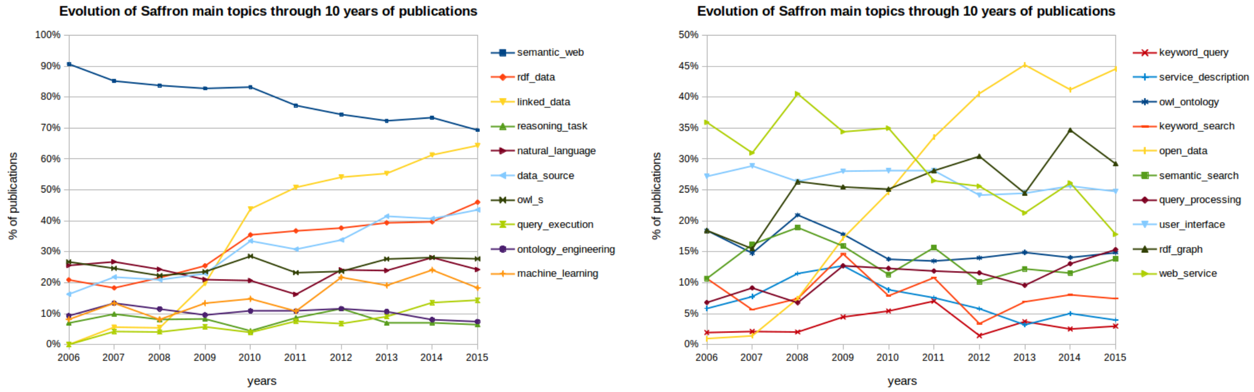


Fig. 10. Saffron: Taxonomy of Semantic Web topics.



(a) top 10 topics

(b) top 11-20 topics

Fig. 11. Saffron: Topic term occurrence evolution over time for (a) top 10 topics and (b) top 11-20 topics.

Other emerging topics include: *Query Execution*, appearing in 2015 in 15% of the publications as well as *RDF Data* and *Data Source* which doubled their presence since 2006. Topics whose popularity increased by at least twice their initial proportion include *RDF Graph* (with two peaks in 2008 and 2014), *Machine Learning* (with a peak in 2012) and *Query Processing* (with a small peak in 2009 then a quite steady line).

Among the topics experiencing strong variations through time, the term *Web Service* is a declining one. After experiencing a peak of use in 2008 with a 40% distribution in the documents, it then dropped to less than 20% in 2015. *Semantic Search* experienced two small peaks in 2008 and 2011, and slight drops in 2010 and 2012 to a more steady curve thereafter. Some topics appear to be consistent over the years, such as

Ontology Engineering, while some others are more volatile. The *Natural Language* topic, despite being equally cited in 2006 and in 2015, gradually dropped in the first half of the period examined, to gain in popularity again after 2011. *Keyword Search* shows quite a varied pattern, with drops in 2007, 2010 and 2012, and peaks in 2009 and 2011. As for *Service Description*, it increased slowly up to 13% by 2009, but gradually declined towards its initial value by 2015.

5.4. Comparison of top forty topics extracted by each tool

Table 8 and *Table* 9 in *Appendix* A, highlight the top 40 multi-word topics that were extracted by at least two data-driven tools and those that were only identified by a single data-driven tool, respectively, based on a simple syntactic matching of the topics. After normalizing the topic names across the sets, we found 86 unique topics. 12 of these were detected by all systems and 23 by at least two systems. We thus computed the Spearman's Rank-Order Correlation on the intersection of the three sets. We found that Rexplore and Poolparty exhibit a moderate correlation ($\rho = 0.61$) and a statistically significant association (p (2-tailed) = 0.035). Conversely, the list produced by Saffron is not correlated with the ones of Rexplore (ρ = 0.01, p (2-tailed) = 0.966.) or Poolparty (ρ = 0.01, p (2-tailed) = 0.681).

The topics uncovered by all three tools could be categorized as reflecting the core focus of the community (*knowledge base, linked data, semantic search, semantic web, web services, ontology matching, query languages*) and several well established sub-communities (*information retrieval, machine learning, natural language processing, ontology engineering*). The topics uncovered by two ore more tools further elaborate on core research topics within the community (*data integration, data source, linked open data (LOD), ontology language, open data, query processing, social networks, user interfaces, web data, web semantics, web ontology language (OWL))*. While, the topics uncovered by only one tool are a mix between supporting technology (e.g., *rdf data, rdf Graph, search engines, logic programming, SPARQL*), very specific topics (e.g., *human computer interaction, stream processing, data privacy, federated query processing*), commonly used data sources (e.g., *DBpedia, wikipedia*), and frequently used terms (e.g., *on the web, use cases, web of data*).

Although in this paper we do not go into details of the specific algorithms employed by PoolPary, Rexplore and Saffron, it is possible to speculate as to why certain topics appear in the top forty list of the various tools. For instance, considering that the PoolParty taxonomy is created from conference and journal metadata, it is not surprising that topics such as *case studies*, *use cases* and references to *on the web* or *web of data* appear, as these terms could frequently occur in calls for papers. In the case of Rexplore we see evidence of broader topics, such as *artificial intelligence* and *human computer interaction* that are reflective of the broader nature of the Rexplore taxonomy, which was generated from a more general computer science corpus. Finally, considering that Saffron not only learns the topics from the corpus, but also tries to identify distinguishing topics for papers, it is not surprising that we see evidence of specfic topics such as *federated query processing* and *stream processing*.

6. Topic Alignment and Findings

In this section, we compare and contrast the topics extracted by the three bottom-up data-driven approaches (Rexplore, Saffron, PoolParty) and the core and marginal topics mentioned in the seminal Semantic Web papers (discussed in *Section* 4), with primary topics identified by the data-driven approaches presented in *Section* 5. Initially we conducted the mapping exercise with the top 20 topics, however after seeing that there were no mappings for several core topics we elected to use the top 40 multi-word topics from PoolParty, Rexplore and Saffron (see *Table* 7 in *Appendix* A).

6.1. Core and marginal topic analysis

The analysis presented in this section is based on a comparison between the core and marginal topics mentioned in the seminal Semantic Web papers and the predominant topics uncovered by PoolParty, Rexplore and Saffron. In contrast to the aforementioned data-driven topic analysis, which was based primarily on the syntactic cross-correlation of topics extracted by PoolParty, Rexplore and Saffron, the analysis presented in this section is based on the clustering of similar topics.

Table 3

Core research topics identified in the seminal papers and their coverage by the data-driven approaches.

	Coverage			Matched topics		
Core topic	PoolParty	Rexplore	Saffron	PoolParty	Rexplore	Saffron
knowledge representation languages and standards	✓	✓	✓	knowledge representation,	knowledge based systems, knowledge representation, Resource Description Framework (RDF), Web Ontology Language (OWL)	rdf data, owl s, blank node, object property
Knowledge structures and modeling	✓	✓	✓	ontology/thesaurus/taxonomy management, web semantics, ontology engineering, ontology language, data models, ontology matching	ontology, ontology engineering	owl ontology, ontology engineering, rdf graph, data model, ontology language, ontology editing, web semantics, ontology development, ontology matching
logic and reasoning	✓	✓	✓	description logic, formal logic/ formal languages/description logics, logic programming	formal logic, description logic, Web Ontology Language (OWL)	reasoning task, description logic
search, retrieval, ranking, question answering	✓	✓	✓	search engines, semantic search, web search, natural language, searching/ browsing/ exploration, computer linguistics & NLP systems, information retrieval	information retrieval, semantic search/similarity, computer linguistics	keyword search, semantic search, natural language, information retrieval
matching and data integration	✓	✓	✓	ontology matching, ontology alignment, similarity measures, data integration	ontology matching, data integration	ontology matching, semantic similarity
privacy, trust, security, provenance	✓	✓	-	security & privacy	security of data, data privacy	-
semantic web databases	✓	-	✓	data sets, knowledge base, data source, knowledge management, data management	knowledge base systems	data source, relational database, knowledge base
distribution, decentralization, federation	-	-	✓	-	-	federated query, federated query processing
query languages and mechanisms	✓	✓	✓	query languages, query answering, query processing	query languages, SPARQL, SPARQL queries	query execution, keyword query, query processing, query language
linked data	✓	✓	✓	linked data, linked open data, semantic web, web of data, data integration, data creation/publishing/sharing	linked data, semantic web, linked open data, data integration	linked data, semantic web
knowledge extraction, discovery and acquisition	✓	✓	✓	information retrieval, machine learning, extraction, data mining, text mining, entity, extraction, analytics, machine learning	information retrieval, natural language processing, data mining, machine learning, natural language processing systems	machine learning, information retrieval

Core topic analysis: As shown in *Table* 3 all three data-driven approaches uncovered eight out of eleven of the Research Landscape topics and all topics were uncovered by at least one data-driven approach. Notable omissions include the *distribution, decentralization and federation* topic, which was not uncovered by PoolParty and Rexplore, the *privacy, trust, security, and provenance* topic, which did not figure in the primary topics uncovered by Saffron, and the *semantic web databases* topic which was not ranked highly by Rexplore.

Marginal topic analysis: Comparing the output from the data-driven approaches to the marginal topics presented in *Table* 4 we observe reduced coverage, with the *multilingual intelligent agents* and *change management and propagation* topics not featuring in any of the top 40 topic lists produced by PoolParty, Rexplore and Saffron. While, the *scalability, efficiency, robust semantic approaches* topic was only identified by PoolParty and not by Rexplore and Saffron.

Additional topics: In order to complete the analysis in *Table* 5 we highlight the topics that were extracted by the data-driven approaches, however were not mentioned in the seminal papers. All three tools identified topics that are very general in nature and as such could not be easily mapped to the primary topics appearing

Table 4
Marginal research topics identified in the seminal papers and their coverage by the data-driven approaches.

	Coverage			Matched topics		
Marginal topic	PoolParty	Rexplore	Saffron	PoolParty	Rexplore	Saffron
multilingual intelligent agents	-	-	-	-	-	-
semantic web services	✓	✓	✓	web service, semantic web service	web services, semantic web services	web service, service description
visualization, user interfaces and annotation	✓	✓	✓	user interfaces, semantic annotation, human computer interaction & visualization, annotation, concept tagging	human computer interaction, visualization	user interface
(scalability, efficiency, robust semantic approaches)	✓	-	-	robustness, scalability, optimization and performance	-	-
change management and propagation	-	-	-	-	-	-
(social semantic web, FOAF)	✓	✓	✓	social network	social networks	social medium

Table 5
Research topics covered by the data-driven approaches that were not identified by the seminal papers.

PoolParty	Rexplore	Saffron
recommendations, use cases, case studies, open data, information systems, web data, semantic technology, structured data	computational linguistics, recommender systems, mobile devices, cloud computing, e-learning system, robotics, electronic commerce systems, decision support systems	open data, web data, web technology

Table 6
Visionary research topics from the seminal papers and their coverage by the data-driven approaches.

	Coverage			Matched topics		
Future topic	PoolParty	Rexplore	Saffron	PoolParty	Rexplore	Saffron
scale changes drastically	✓	-		robustness, scalability, optimization and performance	-	-
intelligent software agents	-	✓	-	-	artificial intelligence	-
(Internet of Things), high volume and velocity of data, e.g., streaming & sensor data	✓	✓	✓	dynamic data / streaming	Internet of Things	stream processing
data quality, e.g, representation, assessment	✓	-	-	quality	-	-

in the seminal papers. For instance, *recommendations, use cases, case studies, open data, information systems, web data, semantic technology*, and *structured data* in the case of PoolParty, *computational linguistics, recommender systems, mobile devices, cloud computing, e-learning system, robotics, electronic commerce systems*, and *decision support systems* in the case of Rexplore, and *open data*, *web data*, *web technology* in the case of Saffron. Several of the topics uncovered by Rexplore stand out from the others as they are not topics per se but rather application or use case oriented keywords that were not extracted from the seminal papers.

6.2. Evidence of future topics

Besides using the data-driven approaches to look for evidence of the topics that the community have been actively working on, we also investigated if the data-driven approaches could also find evidence of future trends predicted in the seminal papers, in particular those mentioned by Bernstein et al. [2]. According to our mapping presented in *Table* 6, evidence with respect to each of the four main lines of future research topics was uncovered by at least one of the data-driven approaches. Interestingly, all approaches found topics relating to the *Internet of Things, streaming and sensor data*, indicating a rise in importance of this topic within the Semantic Web community. However, at the same time, the other three topics that relate to *scale*, *in-*

telligent software agents and *quality* were only weakly identified by the seminal papers.

6.3. Evidence of trends

In the following we summarize the analysis of the trends identified by PoolParty (cf. *Figure* 4- 5), Rexplore (cf. *Figure* 8- 9) and Saffron (cf. *Figure* 11). The foundational topic and trend analysis conducted via PoolParty did not yield any useful results, as generally speaking work on each of the foundational topics appear to be increasing year on year. A cross correlation of the trends highlighted by PoolParty, Rexplore and Saffron provides evidence that topics such as *linked data*, *open data* and *data sources* have an upward trend, whiles topics such as *semantic web*, *web service*, *service description* and *ontology matching* appear to be on a downward trend. When it comes to trend analysis using the data-driven approaches, it is clear that neither foundational topic analysis nor topic specific analysis, provides us with enough evidence to confirm the visions outlined in the seminal papers. For this there is a need for a more focused analysis that maps visions to relevant research topics and uses year on year aggregate counts to depict trends. Although, Fernandez Garcia et al. [16] made some initial attempts at mapping the trends identified by PoolParty to the visions from the seminal paper, unfortunately such a mapping is not very straightforward even for manual mappings and as such is left to future work.

6.4. Mixed methods observations

The comparative analysis of the research topics identified with the qualitative and quantitative methods, discussed in the previous sections, reveals several interesting observations on the benefits and drawbacks of these approaches, as discussed next.

Qualitative vs. Quantitative approaches. Comparing the quality of topic detection using data-driven methods with that of expert-driven methods (cf. *Table* 3), we observe that data-driven approaches had a high recall when it comes to detecting core topics identified by experts in the seminal papers. Data-driven methods failed however to cover multidisciplinary topics, (i.e., topics that cross boundaries between areas), such as *distribution, decentralization, federation*, or *privacy, trust, security, provenance*, or *semantic web databases*. These weakly covered topics are particularly interesting, as they indicate research areas that, although considered important by experts, have not yet attracted a critical mass of research to be reliably identified with quantitative methods.

Analyzing the coverage of *marginal topics* (cf. *Table* 4), we find an opposite phenomenon of research topics for which there is marginal agreement among experts, but strong data-driven evidence of work on those topics. Indeed, data-driven approaches confirm some of the marginal topics such as *social semantic web* and *human computer interaction*. These are topics on which a sufficient volume of work is performed to allow identification by data-driven approaches, but for which a core community has not yet been formed.

As expected, the coverage of visionary topics (*Table* 6) was lower. Although these periphery topics are somehow addressed by the Semantic Web community, the data-driven analysis failed to represent them with the required fine-grained details. It is clear from the results of our analysis that further work on trend detection and analysis is needed in order to better detect emerging topics and to understand the research gaps with respect to the vision.

A major benefit of data-driven methods is that they are capable of providing evidence of the popularity of research areas and topics over time and consequently can be used to derive research trends (although these are somewhat sensitive to the available data and can be less accurate when data is missing, for instance towards the end of the analysis period). When it comes to topics that appear in the Research Landscape but are underrepresented according to our data-driven analysis, such information could be used to encourage publications on these topics via calls for papers of future conferences or via workshops or journal special issues.

Comparison of Quantitative Methods. For the quantitative analysis of our work, we employed data-driven methods that differed, among others, in the way the topic taxonomy was created. In the case of PoolParty a manually built topic taxonomy was employed which closely reflected the topics on which the community are looking for in call for papers or in conference programs. Rexplore made use of the CSO ontology, a large-scale ontology of computer science extracted from a very large corpus and covering key research areas as well as associated research topics. Finally, Saffron extracted its taxonomy of topics entirely from the corpus under analysis and used clustering to identify topics that belong to a research area (without actually deriving research area names). Obviously, these approaches of procuring the topic taxonomy are decreasing in terms of cost as per the time of expert involvement.

In terms of overall performance, (cf. *Tables* 3, 4, 6), PoolParty identified 17/21 core, marginal and future topics (10/11 core topics; 4/6 marginal topics; 3/4 future topics). Together with Saffron, PoolParty identified the most core topics, while achieving the highest recall for the other two topic categories too (i.e., marginal and future topics). Closely after PoolParty, Rexplore identified 14 of the 21 topics of the Research Landscape (9/11 core topics; 3/6 marginal topics; 2/4 future topics), identifying in each category just one topic less than PoolParty. Finally, Saffron is overall very close in its coverage to that of the other two tools by identifying 13 out of 21 topics (10/11 core topics; 2/6 marginal topics; 1/4 future topics). While having a very good coverage of the core topics, Saffron's performance was remarkably inferior to the other tools for the other topic categories, where it primarily identified those topics which were already identified by the other tools. From the above, we conclude that the use of a-priory built taxonomies of research areas, while more expensive, leads to a better coverage of research topics, especially in the analysis of marginal or emerging research topics. Moreover, we attribute the high success of PoolParty to covering research topics to the fact that it relied on a high-quality, manually built topic taxonomy that was well aligned to the domain as the topics were extracted from conference and journal metadata.

While the most cost-effective, Saffron identified a bag of topics that was less straightforward to align to research areas than the output of the other two approaches that relied on taxonomies of research areas (and associated topics). The alignment and interpretation of Saffron topics required expert knowledge and therefore Saffron should ideally also be used in settings where such expert knowledge is available.

While PoolParty had the best performance in confirming research topics from the qualitative analysis, Rexplore provided the most additional topics (cf. *Table* 5), clearly identifying research topics at the intersection of the Semantic Web and other research communities (e.g., *computational linguistics* and *cloud computing*), thus providing invaluable support in positioning the work of our community in a broader research context.

7. Conclusion

The analysis of research topics and trends is an important aspect of scientometrics which is expanding from qualitative expert-driven approaches to also include data-driven methods. The Semantic Web community is no different, with several seminal papers reflecting on and predicting the work of the community and data-driven methods (based on Semantic Web technologies) trying to achieve similar topic and trend detection activities (semi-)automatically.

With this study, we aimed to go beyond the various views on our community's Research Landscape scattered in several papers and obtained with different methods. To that end, we proposed the use of a *mixed methods approach* that can converge, unify but also critically compare conclusions reached with both expert or data-driven approaches. Finally, we conclude this study by revisiting the original research questions:

Is it possible to identify the predominant Semantic Web research topics using both expert based predictions and topic and trend identification tools? A key benefit and novelty of our work is that we identified and aligned core research topics mentioned in the seminal papers and then verified these using data-driven methods. After extracting, grouping and aligning the topics from the seminal papers, we concluded on *eleven core Semantic Web topics* (cf. *Table* 3), out of which eight were confirmed by all the data-driven approaches, while the remaining three indicate topics that are important but not sufficiently represented in papers at the key Semantic Web venues. Besides these core topics, we capture *six marginal topics* (cf. *Table* 4) out of which two are very strongly supported by evidence from data-driven methods.

From a trends perspective, it was clearly visible that topics such as *linked data*, *open data* and *data sources* have increased in importance over the years. While, at the same time, topics such as *semantic web*, *web service*, *service description* and *ontology matching* seem to appear less and less. Although we could speculate as to why this is the case (e.g., a push by the community towards using semantic technology to open up and link data may have caused a decline in work in relation to service based machine-to-machine interaction), however a more in depth analysis, involving sources other than over research papers, would be needed in order to conform our suspicions.

Looking into the future, we identify *four future topics* (cf. *Table* 6), from which the topics on *IoT, sensor and streaming data* has ample evidence in the analyzed research corpus. Finally, the Rexplore data-driven method provided insights into the interactions of our fields with other research areas, highlighting its cross-disciplinary nature. Considering the growing interest in scientomentrics within the Semantic Web community, our findings could be used as a baseline for benchmarking other topic and trend detection methods for the same time period, or extended to cater for more recent work by the community.

What are the strengths and weaknesses of expert-driven and data-driven topic and trend identification methods? Qualitative, expert-driven methods benefit from insights by experts who reflect on past or present research topics and trends and predict future directions. As such, they remain valuable assets in the scientometrics tool-box. Data-driven methods challenge expert-analysis by providing a surprisingly high recall, especially for core research topics, and naturally less for marginal and emerging topics. However, a major benefit of data-driven methods is that their findings are backed-up by quantitative data which can be used to perform a range of other analytics such as research trend detection or identifying connections between research topics.

A key element of the data-driven approaches considered here is the use of a topic taxonomy which can be derived with costly, manual effort, semi-automatically or fully-automatically. Not-entirely surprising, well-curated taxonomies lead to the best performance, but these naturally age very quickly and their maintenance is not sustainable. Therefore, semi-automatic or fully-automatic taxonomy construction methods offer a cheaper and more sustainable alternative with only a slight loss of recall.

In this paper, we proposed and demonstrated the use of a mixed methods approach, which combines both qualitative and quantitative methods in an attempt to overcome their respective weaknesses. This mixed methods approach has several strengths. Firstly, it allowed us to synchronize the results of several qualitative studies and propose a unified Research Landscape of the area. Secondly, by comparing and contrasting the Research Landscape with the results of the data-driven methods, we could: (1) confirm those topics that are both seen as important by experts and for which quantitative evidence can be gathered - these are clearly core topics in the community; (2) identify topics that experts consider important but for which data-driven methods do not (unanimously) find sufficient evidence in the corpus - these are topics that the community should encourage; (3) identify topics on which not all experts agree (which is natural given some bias inadvertently brought in by experts) but which are strongly represented in the research data - these topics could benefit from community building efforts. To summarize, mixed methods allows for drawing interesting conclusions in areas where quantitative and qualitative methods agree or disagree. A weak point of the presented method is the use of manual extraction and alignment of topics which could have introduced bias. We tried to minimize this by performing each of these steps with multiple experts and then reaching agreement where their opinions differed.

In this paper we have focused on approaches to analyse and reflect about the past and to some extent the future development of our research community, using expert opinions, on the one hand, and applying our own data-driven methods, on the other. As such, the comparison and benchmarking of topic detection tools was outside the scope of the paper. Nevertheless, the collected document corpus and the results of our analysis provide the foundations for performing further analysis and benchmarking among topic detection tools in future work.

A first interesting direction would be to apply methods for citation network analysis [9,11] in order to characterize each research field with relevant clusters of papers. We could also apply techniques from the field of spatial scientometrics [18] for analyzing the geographical trends.

Additionally, we could adopt (as mentioned in the end of Section 4.2) emerging methods such as crowdsourcing for a similar reflectional exercise. That is, based on the findings and topics presented here, let the community itself on a larger scale than relying on the insights of a few of its established experts, assess the importance and future of topics for the community. Such an analysis should probably counteract biases in terms of ensuring that researchers do not assess/favor the (future) importance of their own field of research, but we would expect this to be an interesting future direction.

Other avenues for further study include: a more focused analysis that maps visions to relevant research topics and generates the corresponding trends; the deepening of the work to better understand the type of coverage offered in each of the identified research topics; and a broadening of the work to consider not only the research topics but also the application areas and domains where these technologies are routinely applied.

Also, it would be interesting to test this method in other communities (e.g., Software Engineering) and to further improve the topic alignment methods to further reduce bias.

Acknowledgments

This publication has emanated from research supported in part by the PROPEL research project funded by the Austrian Federal Ministry of Transport, Innovation and Technology (BMVIT) and the Austrian Research Promotion Agency (FFG) under the program "ICT of the Future", and a research grant from Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289_P2 co-funded by the European Regional Development Fund. We would like to thank our colleagues from the Semantic Web Company for their support with the PoolParty analysis.

References

[1] Tim Berners-Lee, James Hendler, Ora Lassila, et al. The semantic web. *Scientific American*, 284(5):28–37, 2001.

[2] Abraham Bernstein, James A. Hendler, and Natalya Fridman Noy. A new look at the semantic web. *Commun. ACM*, 59 (9):35–37, 2016. . URL https://doi.org/10.1145/2890489.

[3] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3 (Jan):993–1022, 2003.

[4] Georgeta Bordea. *Domain adaptive extraction of topical hierarchies for Expertise Mining*. Thesis, National University of Ireland, Galway, Ireland, 2013. uri: http://hdl.handle.net/10379/4484.

[5] Georgetas Bordea and Paul Buitelaar. Expertise mining. In *Proceedings of the 21st National Conference on Artificial Intelligence and Cognitive Science, Galway, Ireland*, 2010.

[6] Michel Callon, Jean-Pierre Courtial, William A Turner, and Serge Bauin. From translations to problematic networks: An introduction to co-word analysis. *Information (International Social Science Council)*, 22(2):191–235, 1983. .

[7] Mario Cataldi, Luigi Di Caro, and Claudio Schifanella. Emerging topic detection on twitter based on temporal and social terms evaluation. In *Proceedings of the tenth international workshop on multimedia data mining*, page 4. ACM, 2010.

[8] David Chavalarias and Jean-Philippe Cointet. Phylomemetic patterns in science evolution—the rise and fall of scientific fields. *PloS one*, 8(2), 2013. .

[9] Chaomei Chen. Citespace II: detecting and visualizing emerging trends and transient patterns in scientific literature. *J. Assoc. Inf. Sci. Technol.*, 57(3):359–377, 2006. . URL https://doi.org/10.1002/asi.20317.

[10] Manuel J. Cobo, Antonio Gabriel López-Herrera, Enrique Herrera-Viedma, and Francisco Herrera. Science mapping software tools: Review, analysis, and cooperative study among tools. *J. Assoc. Inf. Sci. Technol.*, 62(7):1382–1402, 2011. . URL https://doi.org/10.1002/asi.21525.

[11] Manuel J. Cobo, Antonio Gabriel López-Herrera, Enrique Herrera-Viedma, and Francisco Herrera. Scimat: A new science mapping analysis software tool. *J. Assoc. Inf. Sci. Technol.*, 63(8):1609–1630, 2012. . URL https://doi.org/10.1002/asi.22688.

[12] Xiang-Ying Dai, Qingcai Chen, Xiaolong Wang, and Jun Xu. Online topic detection and tracking of financial news based on hierarchical clustering. In *International Conference on Machine Learning and Cybernetics, ICMLC 2010, Qingdao, China, July 11-14, 2010, Proceedings*, pages 3341–3346. IEEE, 2010. . URL https://doi.org/10.1109/ICMLC.2010.5580677.

[13] Sheron Levar Decker. *Detection of bursty and emerging trends towards identification of researchers at the early stage of trends*. PhD thesis, uga, 2007.

[14] Jörg Diederich, Wolf-Tilo Balke, and Uwe Thaden. Demonstrating the semantic growbag: automatically creating topic facets for faceteddblp. In Edie M. Rasmussen, Ray R. Larson, Elaine G. Toms, and Shigeo Sugimoto, editors, *ACM/IEEE Joint Conference on Digital Libraries, JCDL 2007, Vancouver, BC, Canada, June 18-23, 2007, Proceedings*, page 505. ACM, 2007. . URL https://doi.org/10.1145/1255175.1255305.

[15] Lee Feigenbaum, Ivan Herman, Tonya Hongsermeier, Eric Neumann, and Susie Stephens. The semantic web in action. *Scientific American*, 297(6):90–97, 2007.

[16] Javier David Fernandez Garcia, Elmar Kiesling, Sabrina Kirrane, Julia Neuschmid, Nika Mizerski, Axel Polleres, Marta Sabou, Thomas Thurner, and Peter Wetz. Propelling the potential of enterprise linked data in austria. roadmap and report., 2016. URL https://www.linked-data.at/wp-content/uploads/2016/12/propel_book_web.pdf.

[17] Volker Frehe, Vilius Rugaitis, and Frank Teuteberg. Scientometrics: How to perform a big data trend analysis with scienceminer. In Erhard Plödereder, Lars Grunske, Eric Schneider, and Dominik Ull, editors, *44. Jahrestagung der Gesellschaft für Informatik, Informatik 2014, Big Data - Komplexität meistern, 22.-26. September 2014 in Stuttgart, Deutschland*, volume P-232 of *LNI*, pages 1699–1710. GI, 2014. URL https://dl.gi.de/20.500.12116/2780.

[18] Koen Frenken, Sjoerd Hardeman, and Jarno Hoekman. Spatial scientometrics: Towards a cumulative research program. *J. Informetrics*, 3(3):222–232, 2009. . URL https://doi.org/10.1016/j.joi.2009.03.005.

[19] Birte Glimm and Heiner Stuckenschmidt. 15 years of semantic web: An incomplete survey. *KI*, 30(2):117–130, 2016. . URL https://doi.org/10.1007/s13218-016-0424-1.

[20] Thomas Hofmann. Probabilistic latent semantic indexing. *SIGIR Forum*, 51(2):211–218, 2017. . URL https://doi.org/10.1145/3130348.3130370.

[21] William W. Hood and Concepción S. Wilson. The literature of bibliometrics, scientometrics, and informetrics. *Scientometrics*, 52(2):291–314, 2001. . URL https://doi.org/10.1023/A:1017919924342.

[22] Yingjie Hu, Krzysztof Janowicz, Grant McKenzie, Kunal Sengupta, and Pascal Hitzler. A linked-data-driven and semantically-enabled journal portal for scientometrics. In Harith Alani, Lalana Kagal, Achille Fokoue, Paul T.

Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha F. Noy, Chris Welty, and Krzysztof Janowicz, editors, *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part II*, volume 8219 of *Lecture Notes in Computer Science*, pages 114–129. Springer, 2013. . URL `https://doi.org/10.1007/978-3-642-41338-4_8`.

[23] Yookyung Jo, Carl Lagoze, and C. Lee Giles. Detecting research topics via the correlation between graphs and texts. In Pavel Berkhin, Rich Caruana, and Xindong Wu, editors, *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, California, USA, August 12-15, 2007*, pages 370–379. ACM, 2007. . URL `https://doi.org/10.1145/1281192.1281234`.

[24] Jon M. Kleinberg. Bursty and hierarchical structure in streams. *Data Min. Knowl. Discov.*, 7(4):373–397, 2003. . URL `https://doi.org/10.1023/A:1024940629314`.

[25] Nancy L Leech and Anthony J Onwuegbuzie. A typology of mixed methods research designs. *Quality & quantity*, 43(2): 265–275, 2009. .

[26] Loet Leydesdorff and Staša Milojević. Scientometrics. *arXiv preprint arXiv:1208.4566*, 2012.

[27] Fergal Monaghan, Georgeta Bordea, Krystian Samp, and Paul Buitelaar. Exploring your research: Sprinkling some saffron on semantic web dog food. In *Semantic Web Challenge at the International Semantic Web Conference*, volume 117, pages 420–435. Citeseer, 2010.

[28] Satoshi Morinaga and Kenji Yamanishi. Tracking dynamics of topic trends using a finite mixture model. In Won Kim, Ron Kohavi, Johannes Gehrke, and William DuMouchel, editors, *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, Washington, USA, August 22-25, 2004*, pages 811–816. ACM, 2004. . URL `https://doi.org/10.1145/1014052.1016919`.

[29] Mizuki Oka, Hirotake Abe, and Kazuhiko Kato. Extracting topics from weblogs through frequency segments. In *Proceedings of WWW 2006 annual workshop on the weblogging ecosystem: aggregation, analysis, and dynamics*, 2006.

[30] Francesco Osborne and Enrico Motta. Klink-2: Integrating multiple web sources to generate semantic topic networks. In Marcelo Arenas, Óscar Corcho, Elena Simperl, Markus Strohmaier, Mathieu d'Aquin, Kavitha Srinivas, Paul T. Groth, Michel Dumontier, Jeff Heflin, Krishnaprasad Thirunarayan, and Steffen Staab, editors, *The Semantic Web - ISWC 2015 - 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part I*, volume 9366 of *Lecture Notes in Computer Science*, pages 408–424. Springer, 2015. . URL `https://doi.org/10.1007/978-3-319-25007-6_24`.

[31] Francesco Osborne, Enrico Motta, and Paul Mulholland. Exploring scholarly data with rexplore. In Harith Alani, Lalana Kagal, Achille Fokoue, Paul T. Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha F. Noy, Chris Welty, and Krzysztof Janowicz, editors, *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part I*, volume 8218 of *Lecture Notes in Computer Science*, pages 460–477. Springer, 2013. . URL `https://doi.org/10.1007/978-3-642-41335-3_29`.

[32] Francesco Osborne, Angelo Antonio Salatino, Aliaksandr Birukou, and Enrico Motta. Automatic classification of springer nature proceedings with smart topic miner. In Paul T. Groth, Elena Simperl, Alasdair J. G. Gray, Marta Sabou, Markus Krötzsch, Freddy Lécué, Fabian Flöck, and Yolanda Gil, editors, *The Semantic Web - ISWC 2016 - 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016, Proceedings, Part II*, volume 9982 of *Lecture Notes in Computer Science*, pages 383–399, 2016. . URL `https://doi.org/10.1007/978-3-319-46547-0_33`.

[33] Francesco Osborne, Henry Muccini, Patricia Lago, and Enrico Motta. Reducing the effort for systematic reviews in software engineering. *CoRR*, abs/1908.06676, 2019. URL `http://arxiv.org/abs/1908.06676`.

[34] Sergey Parinov and Mikhail R. Kogalovsky. Semantic linkages in research information systems as a new data source for scientometric studies. *Scientometrics*, 98(2):927–943, 2014. . URL `https://doi.org/10.1007/s11192-013-1108-3`.

[35] Angelo Antonio Salatino, Thiviyan Thanapalasingam, Andrea Mannocci, Francesco Osborne, and Enrico Motta. The computer science ontology: A large-scale taxonomy of research areas. In Denny Vrandecic, Kalina Bontcheva, Mari Carmen Suárez-Figueroa, Valentina Presutti, Irene Celino, Marta Sabou, Lucie-Aimée Kaffee, and Elena Simperl, editors, *The Semantic Web - ISWC 2018 - 17th International Semantic Web Conference, Monterey, CA, USA, October 8-12, 2018, Proceedings, Part II*, volume 11137 of *Lecture Notes in Computer Science*, pages 187–205. Springer, 2018. . URL `https://doi.org/10.1007/978-3-030-00668-6_12`.

[36] Angelo Antonio Salatino, Thiviyan Thanapalasingam, Andrea Mannocci, Francesco Osborne, and Enrico Motta. Classifying research papers with the computer science ontology. In Marieke van Erp, Medha Atre, Vanessa López, Kavitha Srinivas, and Carolina Fortuna, editors, *Proceedings of the ISWC 2018 Posters & Demonstrations, Industry and Blue Sky Ideas Tracks co-located with 17th International Semantic Web Conference (ISWC 2018), Monterey, USA, October 8th - to - 12th, 2018*, volume 2180 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2018. URL `http://ceur-ws.org/Vol-2180/paper-55.pdf`.

[37] Angelo Antonio Salatino, Francesco Osborne, Thiviyan Thanapalasingam, and Enrico Motta. The CSO classifier: Ontology-driven detection of research topics in scholarly articles. In Antoine Doucet, Antoine Isaac, Koraljka Golub, Trond Aalberg, and Adam Jatowt, editors, *Digital Libraries for Open Knowledge - 23rd International Conference on Theory and Practice of Digital Libraries, TPDL 2019, Oslo, Norway, September 9-12, 2019, Proceedings*, volume 11799 of *Lecture Notes in Computer Science*, pages 296–311. Springer, 2019. . URL `https://doi.org/10.1007/978-3-030-30760-8_26`.

[38] Thomas Schandl and Andreas Blumauer. Poolparty: SKOS thesaurus management utilizing linked data. In Lora Aroyo, Grigoris Antoniou, Eero Hyvönen, Annette ten Teije, Heiner Stuckenschmidt, Liliana Cabral, and Tania Tudorache, editors, *The Semantic Web: Research and Applications, 7th Extended Semantic Web Conference, ESWC 2010, Heraklion, Crete, Greece, May 30 - June 3, 2010, Proceedings, Part II*, volume 6089 of *Lecture Notes in Computer Science*, pages

421–425. Springer, 2010. . URL https://doi.org/10.1007/978-3-642-13489-0_36.

[39] J Michael Schultz and Mark Liberman. Topic detection and tracking using idf-weighted cosine coefficient. In *Proceedings of the DARPA broadcast news workshop*, pages 189–192. San Francisco: Morgan Kaufmann, 1999.

[40] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Arnetminer: extraction and mining of academic social networks. In Ying Li, Bing Liu, and Sunita Sarawagi, editors, *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008*, pages 990–998. ACM, 2008. . URL https://doi.org/10.1145/1401890.1402008.

[41] Nees Jan van Eck and Ludo Waltman. Software survey: Vosviewer, a computer program for bibliometric mapping. *Scientometrics*, 84(2):523–538, 2010. . URL https://doi.org/10.1007/s11192-009-0146-3.

Appendix

A. Additional results

Table 7

Extended topics: Top-40 multiwords in Poolparty and top-40 topics in Rexplore (MV) and Saffron

	Poolparty	Rexplore	Saffron
1	semantic web	semantic web	semantic web
2	linked data	ontology	rdf data
3	knowledge base	artificial intelligence	linked data
4	web service	information retrieval	natural language
5	web semantics	query languages	data source
6	data source	linked data	reasoning task
7	data sets	knowledge based systems	machine learning
8	description logic	natural language processing systems	query execution
9	on the web	Computational Linguistics	owl S
10	natural language	formal logic	ontology engineering
11	use cases	data mining	rdf Graph
12	social network	knowledge representation	User Interface
13	query languages	human computer interaction	service description
14	search engines	ontology matching	open data
15	query answering	web ontology language (OWL)	semantic search
16	user interfaces	description logic	query processing
17	semantic annotation	linked open data (LOD)	keyword search
18	information retrieval	data integration	keyword query
19	web of data	web services	owl ontology
20	open data	resource description framework (RDF)	web service
21	data models	security of data	query language
22	semantic search	ontology engineering	data model
23	ontology matching	semantic search/similarity	ontology matching
24	information systems	social networks	web data
25	query processing	SPARQL	federated query
26	machine learning	data privacy	stream processing
27	ontology language	recommender systems	relational database
28	semantic web service	electronic commerce	blank node
29	linked open data	sensors	information retrieval
30	logic programming	ubiquitous computing	ontology language
31	knowledge management	semantic information	description logic
32	data integration	SPARQL queries	federated query processing
33	ontology engineering	pattern recognition	semantic similarity
34	semantic technology	data visualization	object property
35	ontology alignment	knowledge acquisition	ontology editing
36	web search	information technology	social medium
37	web data	mobile devices	knowledge base
38	structured data	wikipedia	web technology
39	case studies	machine learning	web semantics
40	similarity measures	DBpedia	ontology development

Table 8

Extended topics extracted by two or more tools

Topic	PoolParty	Rexplore	Saffron
description logic	✓	✓	✓
information retrieval	✓	✓	✓
knowledge base	✓	✓	✓
linked data	✓	✓	✓
machine learning	✓	✓	✓
natural language processing	✓	✓	✓
ontology engineering	✓	✓	✓
semantic search	✓	✓	✓
semantic web	✓	✓	✓
web services	✓	✓	✓
ontology matching	✓	✓	✓
query languages	✓	✓	✓
data integration	✓	✓	-
data source	✓	-	✓
linked open data (LOD)	✓	✓	-
ontology language	✓	-	✓
open data	✓	-	✓
query processing	✓	-	✓
social networks	✓	✓	-
user interfaces	✓	-	✓
web data	✓	-	✓
web semantics	✓	-	✓
web ontology language (OWL)	-	✓	✓

Table 9

Extended topics extracted by only one tool

Topic	PoolParty	Rexplore	Saffron
artificial intelligence	-	✓	-
blank node	-	-	✓
case studies	✓	-	-
Computational Linguistics	-	✓	-
data mining	-	✓	-
data models	✓	-	-
data privacy	-	✓	-
data sets	✓	-	-
data visualization	-	✓	-
DBpedia	-	✓	-
electronic commerce	-	✓	-
engineering data model	-	-	✓
federated query	-	-	✓
federated query processing	-	-	✓
formal logic	-	✓	-
human computer interaction	-	✓	-
information systems	✓	-	-
information technology	-	✓	-
keyword query	-	-	✓
keyword search	-	-	✓
knowledge acquisition	-	✓	-
knowledge management	✓	-	-
knowledge representation	-	✓	-
logic programming	✓	-	-
mobile devices	-	✓	-
object property	-	-	✓
on the web	✓	-	-
ontology	-	✓	-
ontology alignment	✓	-	-
ontology development	-	-	✓
ontology editing	-	-	✓
owl ontology	-	-	✓
owl S	-	-	✓
pattern recognition	-	✓	-
query answering	✓	-	-
rdf data	-	-	✓
rdf Graph	-	-	✓
reasoning task	-	-	✓
recommender systems	-	✓	-
relational database	-	-	✓
resource description framework (RDF)	-	✓	-
search engines	✓	-	-
security of data	-	✓	-
semantic annotation	✓	-	-
semantic information	-	✓	-
semantic similarity	-	-	✓
semantic technology	✓	-	-
semantic web service	✓	-	-
sensors	-	✓	-
service description	-	-	✓
similarity measures	✓	-	-
social medium	-	-	✓
SPARQL	-	✓	-
SPARQL queries	-	✓	-
stream processing	-	-	✓
structured data	✓	-	-
systems query execution	-	-	✓
ubiquitous computing	-	✓	-
use cases	✓	-	-
web of data	✓	-	-
web search	✓	-	-
web technology	-	-	✓
wikipedia	-	✓	-

2. Intelligent software web agents: A gap analysis

Bibliographic Information

Kirrane, S., 2021. Intelligent software web agents: A gap analysis. Journal of Web Semantics, 71, p.100659.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization, Project administration, and Funding acquisition.

Copyright Notice

Intelligent Software Web Agents: A Gap Analysis

Sabrina Kirrane

Institute for Information Systems and New Media, Vienna University of Economics and Business, Austria

Abstract

Semantic web technologies have shown their effectiveness, especially when it comes to knowledge representation, reasoning, and data integration. However, the original semantic web vision, whereby machine readable web data could be automatically actioned upon by intelligent software web agents, has yet to be realised. In order to better understand the existing technological opportunities and challenges, in this paper we examine the status quo in terms of intelligent software web agents, guided by research with respect to requirements and architectural components, coming from the agents community. We use the identified requirements to both further elaborate on the semantic web agent motivating use case scenario, and to summarise different perspectives on the requirements from the semantic web agent literature. We subsequently propose a hybrid semantic web agent architecture, and use the various components and subcomponents in order to provide a focused discussion in relation to existing semantic web standards and community activities. Finally, we highlight open research opportunities and challenges and take a broader perspective of the research by discussing the potential for intelligent software web agents as an enabling technology for emerging domains, such as digital assistants, cloud computing, and the internet of things.

Key words: Intelligent Software Agents, Agent Architectures, Intelligent Software Web Agents, Web Standards

1. Introduction

At the turn of the millennium, Berners-Lee et al. [8] coined the term semantic web and set a research agenda for this new research field. The authors used a fictitious scenario to describe their vision for a web of machine-readable data, which would be exploited by intelligent software agents who would carry out data centric tasks on behalf of humans. Hendler [68] further elaborated on the intelligent software agent vision with a particular focus on the key role played by ontologies in terms of service capability advertisements that are necessary in order to facilitate interaction between autonomous intelligent software web agents.

Several years later, in 2007, Hendler [69] highlighted that although the interoperability and intercommunication infrastructure necessary to support intelligent agents was available the intelligent agent vision had not yet been realised. Almost a decade later, in 2016, Bernstein et al. [9] discussed the evolution of the semantic web community and identified several open research questions, which they categorised under the following headings: (i) representation and lightweight semantics; (ii) heterogeneity, quality, and provenance; (iii) latent semantics; and (iv) high volume and velocity data. Interestingly, the authors identified the need to better understand the needs of intelligent software web agents both from a semantics and a deployment perspective. Hinting that the semantic web agent vision had not progressed from a practical perspective. Further evidence that the intelligent software agents vision has received limited attention in recent years was provided by Kirrane et al. [89], who used three data driven approaches in order to extract topics and trends from a corpus of semantic web venue publications from 2006 to 2015 inclusive. The authors highlighted that the intelligent agents topic did not feature in any of the top 40 topic lists produced by the three topic and trend detection tools used for their analysis.

Nevertheless, according to Luck et al. [98], a semantically rich data model, vocabularies, and ontologies, which can be used to describe media and services in a manner that facilitates discovery and composition, are key components of the proposed strategic agent technology roadmap. Indeed, over the years, the semantic web community has produced various standards and best practices that support data integration and reasoning using web technologies [9]. Many of which are discussed in the recent knowledge graphs tutorial article [72], which examines the role of semantic technologies when it comes to publishing and consuming knowledge graphs. Although several application areas (e.g., web search, commerce, social networks, finance) are discussed, there is no mention of agency. Interestingly, agents are briefly mentioned in several places, especially in the context of Findable, Accessible, Interoperable and Reusable (FAIR) principles, however agency from a conceptual perspective is not discussed. More broadly,

Email address: sabrina.kirrane@wu.ac.at (Sabrina Kirrane)

agent technology and semantic web agents in particular could potentially serve as an enabling technology for various emerging domains (e.g., digital assistants, cloud computing, and the internet of things), especially when it comes to integration and governance. However, an important stepping stone to positioning intelligent software web agents as an enabling technology for more complex domains, is to determine what standards, tools, and technologies have been proposed, and to identify open research opportunities and challenges.

Thus, motivated by the desire to better understand the status quo, we perform a focused literature review shepherded by agent requirements and architectural components commonly discussed in the literature. Our work is guided by three primary research questions:

1. Which core requirements and architectural components are routinely used to guide software agent research?
2. What is the status quo in terms of intelligent software web agent research in terms of standards, tools, and technologies?
3. What are the primary opportunities and challenges for intelligent software web agents from both a requirements and an architectural perspective?

In order to answer the aforementioned research questions we adopt an integrative literature review methodology [159, 149]. The goal being to integrate literature, in order to better understand the various proposals and how they relate to one another. The objective of our analysis is not to survey all literature that could be used to realise intelligent software web agents, but rather to use agent requirements and standard components used in agent architectures in order to perform a targeted analysis of the original intelligent software web agents motivating use case scenario and the potential solutions proposed to date.

Towards this end, we start by examining well known literature from the agents community that relates specifically to intelligent agent requirements and architectural components. Next, we use the intelligent agent requirements in order to better understand the functional and non functional aspects of the envisaged intelligent software web agents, and the various perspectives on said requirements coming from the semantic web literature. Following on from this, we use the intelligent agent architectural components in order to examine existing standards, tools, and technologies that could be used to realise the proposed hybrid agent architecture. We subsequently provide pointers, in the form of opportunities and challenges, that could be used to realise the semantic web agent vision. Finally, we discuss the potential for intelligent software web agents as an enabling technology for digital assistants, cloud computing, and the internet of things.

Our primary contributions can be summarised as follows: (i) we provide the necessary background information concerning intelligent agent requirements and architectures; (ii) we introduce an agent task environment requirements assessment framework that can be used to perform a detailed analysis of various agent based use case scenarios; (iii) we propose a web based hybrid agent architecture and use it to perform a gap analysis in terms of existing standards, tools, and technologies; and (iv) we identify existing research opportunities and challenges, and reinforce the need for intelligent software web agents as an enabling technology for several emerging domains.

The remainder of the paper is structured as follows: *Section* 2 presents the necessary background information in relation to intelligent agent requirements and architectures. *Section* 3 outlines our motivating use case scenario and presents the results of our requirements analysis. *Section* 4 examines related work on intelligent software web agents from the perspective of the various agent requirements. *Section* 5 proposes a web based intelligent agent architecture and discusses the standards, tools, and technologies that could be leveraged by the individual components. *Section* 6 performs a gap analysis in terms of existing standards and various research activities, summarises open research challenges and opportunities, and discusses the intelligent software agent potential beyond the original motivating use case scenario. Finally, we present our conclusions in *Section* 7.

2. Intelligent Software Agents

Originally robotics was the primary driver for agent based research, however the concept evolved to include software mimicking or acting on behalf of humans (i.e., software agents) and internet robots (i.e., bots) [107]. In this paper we focus specifically on intelligent software agents that use web resources in order to perform data centric tasks on behalf of humans. In order to provide a theoretical grounding for our assessment of the maturity of intelligent software web agents, we provide the necessary background information on intelligent software agent requirements and provide a high level overview of the most prominent agent architectures.

2.1. Agent Requirements

Wooldridge and Jennings [163] distinguish between weak and strong intelligent software agents. In the case of the former, the agent is capable of acting autonomously, has the ability to interact both with humans and other agents, is capable of reacting to environmental changes, and exhibits proactive goal directed behaviour. In the case of the latter, the agent exhibits each of the aforementioned traits, however these agents are conceptualised based on human like attributes, such as knowledge, belief, intention, or obligation. In the following, we summarise different desiderata for intelligent agent behaviour and group related requirements based on the overarching function:

Basic Functions.

Autonomy: Agents should manage both their state and their actions, and should be able to adapt to changes in their environment without direct intervention by humans [29, 163, 99, 48, 76].

Reactivity: Agents should be able to autonomously respond to environmental changes in a timely manner [163, 99, 48, 76].

Pro-activeness: Agents should be able to pursue proactive goal directed behaviour [163, 99, 48, 76].

Social ability: Agents should be able to interact with humans and other agents [163, 54, 48, 76].

Behavioural Functions.

Benevolence: Assumes that agents do not have goals that conflict with one another and thus are well meaning [129, 163].

Rationality: Assumes that an agent does not act in a manner that would be counter productive when it comes to achieving its goals [163].

Responsibility: Involves acting according to the authority level that is entrusted to the agent either by the person or organisation that the agent represents or another agent [70].

Mobility: Refers to the ability to move around an electronic network, for instance using remote programming in order to execute tasks on other machines [158, 163, 48].

Collaborate Functions.

Interoperability, communication, and brokering services: Agents need to be able to discover services and to interact with other agents [70].

Inter-agent coordination: Agents need to be able to work together with other agents in order to facilitate collective problem solving [70].

Code of Conduct Functions.

Identification: The ability to verify the identity of an agent and the person or organisation that the agent represents [70].

Security: Involves taking measures to secure resources against accidental or intentional misuse [70].

Privacy: Relates to being mindful of the privacy of the person or organisation that an agent represents [70], however more broadly an agent should respect the privacy of anyone with whom it interacts.

Trust: Involves ensuring that the system does not knowingly relay false information [163].

Ethics: Involves leaving the world as it was found, limiting the consumption of scarce resources, and ensuring predictable results [70], however in essence this could be interpreted as ensuring that agents do no harm.

Robustness Functions.

Stability, performance, and scalability: This is a broad category that relates to ensuring that agents and multi-agent systems can handle increasing workload effectively and are highly available [70, 99, 48].

Verification: Relates to governance mechanisms that can be used to verify that everything works as expected [70].

2.2. Agent Architectures

Existing architectures, encapsulating the software components and interfaces that ultimately denote an agents capabilities, can be classified as reactive, deliberative, or hybrid [162, 163, 107, 59]. Reactive architectures are ideally suited for real time decision making (where time is of the essence), whereas deliberative architectures are designed to facilitate complex reasoning. Learning architectures are designed to enable the agent to improve its performance over time. While, hybrid architectures strive to leverage the benefits of both deliberative and reactive architectures. In the following, we provide a high level overview of the external inputs and interfaces that are common to all architectures:

Environment: Agents act in their environment, possibly together with other agents. When it comes to web agents, the internet is a complex space that consists of a variety of different networking technologies, devices, information sources, applications, and both human and artificial agents.

Performance Measure: According to Russel and Norvig [130], when it comes to evaluating the effectiveness of an agent it is necessary to define success in terms of the state of the environment. Here there is a need for desirable qualities, taking into consideration that there may be conflicting goals making it necessary to assess and manage trade-offs.

Sensors: Software agents perceive the world through a variety of sensors, for instance the keyboard, cameras, microphone, network ports, etc., that can be used by agents to sense their environment.

Actuators: Software agents are capable of performing actions via a variety of actuators, for instance the screen, printer, headphones, network ports, etc., that can be used by agents to act on their environment.

2.2.1. Reactive Architectures

Reactive agents are modelled on human based instinctive or reflexive behaviour [59]. When it comes to reactive agents there is a tight coupling between what the agent perceives and how the agent acts in the form of condition action rules [107]. In the following, we briefly introduce the components predominantly found in reactive architectures:

Condition-action rules: The agent simply retrieves the action associated with a particular condition perceived by its sensor(s) and uses the action to give instructions to the actuator(s).

State: More advanced reactive agents maintain state in the form of information about the world, and previous interactions with the environment. Given a new perception, the agent chooses an action based on both the current perception and its history of previous perceptions.

2.2.2. Deliberative Architectures

Deliberative agents are rooted in the physical symbolic system hypothesis proposed by Newell and Simon [108], whereby a symbolic language is used to model the environment and decisions are taken based on logical reasoning [162]. In the following, we highlight the components predominantly found in deliberative architectures:

Knowledge base: The knowledge base is a symbolic encoding of both the agents knowledge of the world and the knowledge that governs its own behaviour.

Reasoning mechanism: Logical reasoning (e.g., deduction, induction, abduction and analogy) relating to conditions perceived by the agents sensor(s), the possible alternative actions, and their impact on the environment are used to enable the agent to give instructions to its actuator(s).

Goal encoding: Goals can be used to guide the agents decision making by describing behaviours that are desirable. The reasoning mechanism is used to select the action or set of actions that will lead to the satisfaction of a given goal.

Utility function: Agents that need to choose between different possible actions or sets of actions, can be guided via a utility function that allows the agent to perform a comparative assessment, based on preferences, such that it maximises its utility.

2.2.3. Learning Architectures

Learning agents strive to become more effective over time, and are deemed especially useful when the agent environment is not known a priori [130]. Although the deliberative component could potentially be enhanced with learning abilities, Bryson [22] argues that modularity from an agent architecture perspective simplifies both design and control, thus in this paper we treat them as separate components. Nonetheless there is a tight coupling between both the deliberative and the learning components. Generally speaking learning agent architectures are composed of four additional components, representing the performance, problem generator, critic, and learning functions:

Performance: The performance component is an all encompassing term used to refer to the core inner functions of the agent.

Problem generator: The goal of the problem generator is to suggest actions that will lead to learning in the form of new knowledge and experiences.

Critic: The critic provides feedback to the agent (in the form of a reward or a penalty) with respect to its performance, which is measured against a fixed performance standard.

Learning element: The learning element performs actions assigned by the problem generator, and uses the feedback mechanism provided by the critic to determine how the core inner functions of the agent should be amended.

2.2.4. Hybrid Architectures

The individual reactive and deliberative architectural components described thus far can be organised into horizontal and/or vertical layers [162, 163]. The proposed layering can be used to combat the shortcomings in terms of both reactive and deliberative architectures, however it increases the complexity of the system from a control perspective [107]. In the following, we briefly introduce the components predominantly found in hybrid architectures:

Layering: Horizontal and/or vertical layers are used to combine different functions, such as reactivity, deliberation, cooperation, and learning.

Controllers: Controller components are necessary for planning the work, executing and monitoring activities, and managing interactions between activities.

3. Intelligent Software Web Agents

The goal of this section is to revisit the original vision for the web, whereby machine-readable data would be exploited by intelligent software agents that carry out data centric tasks on behalf of humans. We start by summarising the use case scenario originally proposed by Berners-Lee et al. [8] in their seminal semantic web paper. Following on from this, we use the task environment framework proposed by Russel and Norvig [130] together with the requirements from the agents literature, presented in *Section* 2, in the form of a task environment requirements assessment, in order to provide additional insights into the functions necessary to realise the proposed intelligent software web agents.

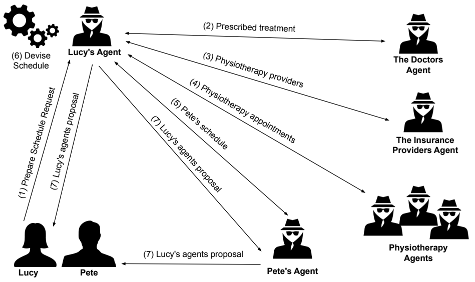

(6) Devise Schedule
Lucy's Agent
(2) Prescribed treatment
The Doctors Agent
(3) Physiotherapy providers
The Insurance Providers Agent
(4) Physiotherapy appointments
(5) Pete's schedule
(7) Lucy's agents proposal
(1) Prepare Schedule Request
(7) Lucy's agents proposal
Physiotherapy Agents
Lucy
Pete
(7) Lucy's agents proposal
Pete's Agent

Figure 1: Physiotherapy appointment planning workflow.

3.1. *Motivating Use Case Scenario*

The original semantic web vision [8] was presented with the help of a motivating use case scenario, revolving around Pete and Lucy and their mother who has just found out that she needs to attend regular physiotherapy sessions. Pete and Lucy ask their personal agents to prepare a physiotherapy appointment schedule such that they are able to share the chauffeuring duties.

Physiotherapy appointment planning workflow. Lucy's agent is tasked with finding a physiotherapist who: (i) is covered by their mothers insurance; (ii) has a trusted service rating of very good or excellent; (iii) is located within a 20 mile radius of their mother's home; and (iv) has appointments that work with Lucy and Pete's busy schedules. The workflow depicted in *Figure* 1 can be summarised as follows:

(1) Lucy requests that her agent devises a plan considering the given constraints.

(2) Lucy's agent consults with the doctor's agent in order to retrieve information relating to the prescribed treatment.

(3) Lucy's agent subsequently consults with the insurance provider agent in order to find physiotherapists considering the given constraints.

(4) Lucy's agent consults the various physiotheraphy agents in order to retrieve available appointment times.

(5) Lucy's agent asks Pete's agent to give her access to his schedule.

(6) Lucy's agent uses the available appointment times provided by the various appointment agents, together with Pete's schedule provided by his agent, and Lucy's own schedule (that her agent already had access to) in order to prepare a physiotherapy appointment schedule considering available appointments and Pete's and Lucy's busy schedules.

(7) Lucy's agent shares the proposed plan with Lucy, and Pete's agent who in turn shares it with Pete.

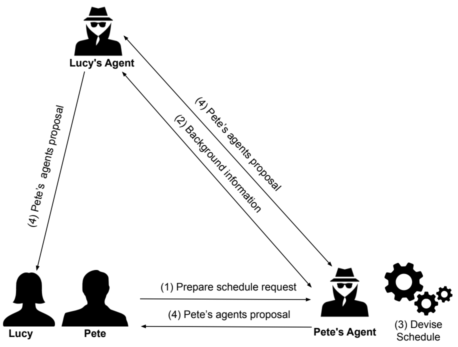

Lucy's Agent
(4) Pete's agents proposal
(4) Pete's agents proposal
(2) Background information
(1) Prepare schedule request
(4) Pete's agents proposal
Lucy
Pete
Pete's Agent
(3) Devise Schedule

Figure 2: Physiotherapy appointment planning workflow with additional constraints.

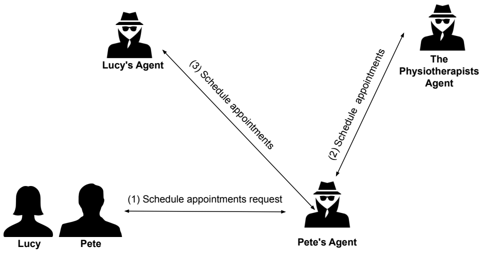

Lucy's Agent
(3) Schedule appointments
(2) Schedule appointments
The Physiotherapists Agent
(1) Schedule appointments request
Lucy
Pete
Pete's Agent

Figure 3: Physiotherapy appointment conformation workflow.

Physiotherapy appointment planning workflow with additional constraints. Given the physiotherapist is quite far from Pete's work and the appointment times coincide with the lunch time rush hour, Pete instructs his agent to redo the task with stricter constraints on location and time. The workflow depicted in *Figure* 2 is described as follows:

(1) Pete requests that his agent propose a new plan that takes into consideration the additional location and time constraints.

(2) Pete's agent obtains all relevant background information relating to the initial proposal from Lucy's agent.

(3) Pete's agent uses the new constraints, together with the available appointment times and information related to Pete's and Lucy's schedules, in order to prepare a new plan, with two compromises: (i) Pete needs to reschedule some conflicting appointments, and (ii) the provider was not on the insurance companies list, thus his agent verified that the service provider was eligible for reimbursement using an alternative mechanism.

(4) Pete's agent shares the proposed plan and the compromises with Pete, and Lucy's agent, who in turn shares it with Lucy.

Physiotherapy appointment conformation workflow. Both Pete and Lucy agree to the plan, and Pete instructs his agent to make the appointments with the physiotherapist and update their schedules accordingly. The workflow depicted in *Figure* 3 is outlined below:

(1) Pete instructs his agent to confirm the appointments and update his schedule accordingly.

(2) Pete's agent makes the booking with the physiotherapists agent.

(3) Pete's agent instructs Lucy's agent to update her schedule.

3.2. A Task Environment Assessment

Next we use the Performance Measure, Environment, Actuators, and Sensors (PEAS) assessment criteria proposed by Russel and Norvig [130] to get a better understanding of high level goals and to examine the external interfaces and inputs.

3.2.1. Information Agents

In the given scenario, the doctor's agent and the insurance company service provider's agent can be classified as *information agents.* Given a request the agent uses the search parameters submitted with the request together with its own knowledge base in order to return an appropriate response. The details of our PEAS assessment can be found below:

Agent: The doctor's agent.

Performance Measure: The treatment information is provided.

Environment: The patient's agent.

Sensors: The doctor's agent `RetrievePrescribedTreatment` interface.

Actuators: The doctor's agent `RetrievePrescribedTreatment` interface.

Agent: The insurance company service provider's agent.

Performance Measure: The physiotherapy service provider information is provided.

Environment: The client's agent.

Sensors: The insurance company service provider's agent `RetrieveServiceProviderInfo` interface.

Actuators: The insurance company service provider's agent `RetrieveServiceProviderInfo` interface.

3.2.2. Booking Agents

The physiotherapy appointment agent has two functions: (i) to provide information about available appointments; and (ii) to accept and confirm appointment requests. Thus this agent can be classified as a *booking agent* that allows for appointments to be scheduled based on availability. The details of our PEAS assessment is as follows:

Agent: The physiotherapy provider agents.

Performance Measure: The available appointments are provided, and/or the requested appointments are confirmed.

Environment: The client's agent.

Sensors: The physiotherapy appointment agent `RetrieveAvailableAppointments` and `BookAvailableAppointments` interfaces.

Actuators: The physiotherapy appointment agent `RetrieveAvailableAppointments` and `BookAvailableAppointments` interfaces.

3.2.3. Personal Planning Agents

Taking the given scenario into consideration, both Pete and Lucy's personal agents can be classified as `planning agents` that are tasked with providing an optimal plan based on the given constraints in terms of available treatments, location, time, etc. In addition, the agents need to work together in order to find a schedule that works for both Pete and Lucy.

Agent: Lucy's personal agent.

Performance Measure: The physiotherapist is covered by insurance, the physiotherapist is located near their mothers house, and the appointments fit with Pete and Lucy's schedules.

Environment: The doctor's agent, the insurance company agent, the physiotherapy provider agents, Pete's agent, and Lucy.

Sensors: Personal agent `PrepareSchedule`, `RequestInfo`, `MakeBooking` interfaces, and a web user interface.

Actuators: Personal agent `PrepareSchedule`, `RequestInfo`, `MakeBooking` interfaces, and a web user interface.

Agent: Pete's personal agent.

Performance Measure: The physiotherapist is covered by insurance, the physiotherapist is located near their mothers house, appointments fit with Pete and Lucy's schedules, the physiotherapist is near Pete's work, and appointments are not during busy traffic periods.

Environment: (Potential) The doctor's agent, the insurance company agent, the physiotherapy provider agents, Lucy's agent, and Pete.

Sensors: Personal agent `PrepareSchedule`, `RequestInfo`, `MakeBooking` interfaces, and a web user interface.

Actuators: Personal agent `PrepareSchedule`, `RequestInfo`, `MakeBooking` interfaces, and a web user interface.

3.3. A Task Environment Requirements Assessment

Unfortunately, the PEAS assessment does not provide any guidance with respect to the inner workings of the various agents, even though such information is necessary in order to determine the various architectural components and how they interact. Thus, in this section we propose a task environment requirements assessment and use it to perform a more detailed analysis of our motivating use case scenario.

We start by examining the *basic functions*, summarised in *Table* 1, that are needed to determine the type of architecture (i.e., reactive, deliberative, learning, and hybrid) required:

Autonomy: All three agent types are able to perform tasks without human interaction.

Reactivity: The information and booking agents are simple request response agents, however scheduling agents need to both interact with other agents and to examine possible solutions to the task that they have been given.

Pro-activeness: Although all three agents exhibit goal directed behaviour, scheduling agents would be classified as more pro-active as they need to explore various alternatives.

Social ability: When it comes to the information and booking agents, although the scenario focuses primarily on agent to agent interaction, all three agent types need to be able to interact with humans and agents.

Next, we examine the various *behavioural functions*, summarised in *Table* 2, that govern how our agents are expected to act:

Benevolence: We assume that information, booking, and scheduling agents are well meaning, however it is conceivable that different personal agents in the broader sense may have conflicting goals.

Rationality: Agents should be designed in order to ensure that agents do not act in a manner that would be counter productive when it comes to achieving their goals.

Responsibility: In the case of all three agent types, there is a tight coupling between responsibility and the overarching goals of the various agents (namely, providing access to the requested information, completing the booking, and finding an optimal schedule given a set of constraints).

Mobility: In the given use case scenario, the agents either request information from other agents or request that other agents perform a specific action, thus we assume that the agents are immobile.

Next we examine the *collaborate functions*, summarised in *Table* 3, that can be used to determine how agents interact with humans and other agents, and how internal communication between agent components should work:

Interoperability: The requester needs to know which services it can call and how it can process the responses, thus there is a need for a schema that is understood by both the requester and the requestee.

Communication: In the given scenario, we assume that agents cater for pull requests via services, and in the case of scheduling agents push notifications on completion of a task.

Brokering services: Information agents are responsible for gathering information from multiple providers, the booking agents need to handle bookings for multiple service providers, and the personal agents are tasked with collecting the information needed in order to complete its task.

Inter-agent coordination: In the given scenario, the personal agents engage in a form of collaborative problem solving. The personal agents do not work collectively but rather support each other via information sharing (e.g., Pete's agent obtained all relevant background information relating to its task from Lucy's agent).

The *code of conduct functions*, summarised in *Table* 4, refer to the security, privacy, and ethical requirements that need to be built into the system:

Identification: In the case of the information agent it may or may not be necessary to authenticate the requester, for instance the personal heath records are available only to patients or their agents, however service provider information is usually public. Similarly, given that some service providers work on an honours system, authentication may or may not be needed in order to make a booking, however in both cases the client would need to be identifiable. When it comes to the scheduling agents, considering the amount of personal data needed by the agents in our

Table 1: Basic functional requirements assessment.

	Information Agent	Booking Agent	Scheduling Agent
Autonomy	handles information requests	handles information and booking requests	consult relevant sources, devise an optimal schedule
Reactivity	immediate response	immediate response	immediate response where possible
Pro-activeness	information goal	booking goal	scheduling goal, explore alternatives
Social ability	humans and agents	humans and agents	humans and agents

Table 2: Behavioural functional requirements assessment.

	Information Agent	Booking Agent	Scheduling Agent
Benevolence	well meaning by design	well meaning by design	well meaning by design, manages conflicting goals
Rationality	rational by design	rational by design	rational by design
Responsibility	provides access to information	provides access to information, complete the booking	manages access to information, finds an optimal schedule given a set of constraints.
Mobility	-	-	interacts with several other agents

Table 3: Collaborative functional requirements assessment.

	Information Agent	Booking Agent	Scheduling Agent
Interoperability	agreed/common schema	agreed/common schema	agreed/common schema
Communication	pull requests	pull requests	push and pull requests
Brokering services	collects information from multiple service providers	handles bookings for multiple service providers and clients	collects information from a variety of sources
Inter-agent coordination	-	-	agents support each other via information sharing

Table 4: Code of conduct functional requirements assessment.

	Information Agent	Booking Agent	Scheduling Agent
Identification	may need to differentiate between public and private information providers	may need to differentiate between public and private information providers, some service providers may work on an honours system	needs to differentiate between public and private information, may need to prove who they represent
Security	protect against unauthorised access, inappropriate use, and denial of service	protect against unauthorised access, inappropriate use, and denial of service	protect against unauthorised access, inappropriate use, and denial of service
Privacy	may need to handle personal information	needs to handle personal information	handles personal information appropriately
Trust	manages information accuracy	manages information accuracy	manages information and scheduling accuracy, reliable and fair, provides transparency and explainability, robust in terms of verification and validation
Ethics	do no harm by design	do no harm by design	do no harm by design

motivating scenario, they would need to be able to authenticate their owners, and also the other personal agents with whom they interact.

Security: The system should be designed to protect against unauthorised access to, and inappropriate use of, data, as well as protecting against denial of service attacks.

Privacy: In the case of the information agents, it is necessary to differentiate between public and private information providers. However, in all other cases, agents will need to manage personal data and thus they need to adhere to the respective data protection legislation.

Trust: All three agent types need to be able to assess if they can trust the providers that they interact with, and ideally should be able to assess if the information they obtain from others is indeed correct. In this context trust is a broad concept linked to reliability, fairness, transparency, explainability, verification and validation.

Ethics: Although there are many things that could be discussed in detail under the ethics umbrella, here we envisage systems that do no harm, thus the agents should avoid behaving in a way that would bring about negative consequence either for the agent itself, or the agents and humans it interacts with.

Finally, the *robustness* requirements assessment, summarised in *Table* 5, defines criteria that should be used to determine the effectiveness of the architecture in terms of both functional and non functional requirements:

Stability: Stability is an all encompassing term used for availability, reliability, and security. Such metrics have an important role to play when it comes to evaluating the effectiveness of any system.

Table 5: Robustness functional requirements assessment.

	Information Agent	Booking Agent	Scheduling Agent
Stability	available reliable & secure	available reliable & secure	available reliable & secure
Performance	provides real time access to information	provides real time access to information	provides real time access to information, timely goal completion
Scalability	handles increasing requests & data	handles increasing requests & data	handles increasing requests, data, & task complexity
Verification	checks information is correct	checks information is correct	checks information is correct, the reasoning is explainable

Performance: While information and booking agents need to be able to respond in real time, the scheduling agents will require time in order to source and analyse the data needed to derive an optimal schedule that satisfies a given set of constraints.

Scalability: All three agent types need to be able to scale with increasing data and increasing requests.

Verification: All three agent types need mechanisms that can be used to verify that everything works as expected. In addition, scheduling agents need to be able to explain what information sources they used and the logic behind their proposal.

4. Intelligent Software Web Agents: Requirements

In this section, we take a closer look at intelligent software agents and how the various requirements are perceived from a semantic web perspective. Considering the broad nature of the topic, the goal is not to summarise all relevant literature, but rather to better understand the different perspectives on the various requirements introduced in *Section* 2 and discussed specifically in the context of the original semantic web agent use case scenario in *Section* 3.

4.1. Basic Functions

The works categorised in *Table* 6 and discussed below describe how intelligent agents could leverage semantic technologies, usually from a theoretical perspective.

Autonomy. Paolucci and Sycara [117] focus on service provision and usage, using the term autonomous semantic web services to refer to services that are capable of reconfiguring their interaction patterns, such that it is possible for them to react to changes with minimal human involvement. Several authors [32, 25, 43, 86, 147, 52, 53] focus specifically on autonomous agents and how they can leverage web services. While, Bryson et al. [23, 24] take a more conservative view referring to agent behaviours as semi-autonomous intelligent modules. Artz and Gil [5] in turn discuss the relationship between autonomy and trust. While, Van Riemsdijk et al. [154] focus on adaptability, discussing how agents can adapt their behaviour in order to comply with norms. Tamma et al. [146] identify autonomous components as a desiderata for searching the semantic web and Sycara et al. [144] highlight the key role played by brokers when it comes to discovery and synchronisation between autonomous agents. More generally,Tamma and Payne [145] argue that the sheer scale and heterogeneity of knowledge and services available on the web calls for autonomy not only on the part of the data and service providers but also the intelligent agents that are best placed to adapt to such dynamic, uncertain, and large scare environments. Payne [120], Leite et al. [95], Leite and Girardi [94], Buoncompagni et al. [26], Kootbally et al. [90], Merkle and Philipp [101] and Ghanadbashi and Golpayegani [56] discuss autonomy from a learning perspective, highlighting the need for agents to be self-aware by building up a knowledge base that allows them to learn alternative strategies and solutions that can be used to fulfil future goals. While, Huhns [74] focuses specifically on the tension between autonomy and co-ordination when it comes to inter-agent co-operation. In particular, the author highlights the need for extending web service standards to cater for federated servers and co-operating clients. Whereas, the autonomous agent architectures proposed by Fornara et al. [47], Fornara and Colombetti [46], Tonti et al. [148] and Van Riemsdijk et al. [154] are designed to cater for constraints in the form of policies or norms.

Reactivity. Bryson et al. [23, 24] discuss how an agent-oriented approach to software engineering, entitled behaviour-oriented design, can be used to define reactive intelligent software web agents that are capable of managing interconnected (possibly conflicting) reactive plans. Boley et al. [13], Papamarkos et al. [118, 118, 119], Poulovassilis et al. [123], Gomes and Alferes [62], Ksystra and Stefaneas [92] and Jochum et al. [77] propose solutions that can be used to encode reactive functionality in the form of event-condition-action rules. Whereas, the architecture proposed by Käfer and Harth [80] makes use of simple condition-action rules. The discussion on web services from an agents perspective by Payne [120] and the framework proposed by Khalili et al. [86] consider reactivity in terms of an agents response to environmental changes. Bonatti et al. [17] in turn propose a formal framework that can be use to express and enforce reactive policies, while at the same time catering for trust negotiation between agents. While, Tamma et al. [146] discuss the role played by both reactive and pro-active components in their searching for semantic web content system, where a reactive approach is used to keep indexes up to date.

Table 6: Intelligent software web agents basic function perspectives.

	Autonomy	Reactivity	Pro-activeness	Social ability
Artz and Gil [5]	autonomy & trust	-	-	social networks & reputation
Boley et al. [13]	-	event condition action rules	-	-
Bonatti et al. [17]	-	trust negotiation between agents	-	social network use case
Bryson et al. [23] Bryson et al. [24]	semi-autonomous modules & autonomous agents	reactive plans	-	-
Buhler and Vidal [25]	autonomous agents & workflows	-	semantic web services & behavioural descriptions	social structures & workflows
Buoncompagni et al. [26]	learning agents	-	learning ability	-
Challenger et al. [30]	-	-	belief-desire intention	-
Chiu and Leung [32]	autonomous agents	-	believe-desire-intention framework & ontologies	-
Demarchi et al. [36]	-	-	belief-desire intention	-
Dong et al. [41]	-	-	belief-desire intention	-
Ermolayev et al. [43]	autonomous agents	-	collaborative goals	social commitments & conventions
Fornara et al. [47] Fornara and Colombetti [46]	policy agents	-	-	-
García-Sánchez et al. [52] García-Sánchez et al. [53]	autonomous agents	-	semantic web services & behavioural descriptions	-
Ghanadbashi and Golpayegani [56]	learning agents	-	learning ability	-
Gomes and Alferes [62]	-	event condition transaction language	-	-
Harth and Käfer [66]	-	condition action rule language	-	-
Huhns [74]	autonomy & cooperation	-	-	-
Jochum et al. [77]	-	event condition action rules	-	-
Käfer and Harth [80]	-	condition action rule language	-	-
Khalili et al. [86]	autonomous agents	environmental changes	goals	communication language
Kootbally et al. [90]	learning agents	-	learning ability	-
Ksystra and Stefaneas [92]	-	event condition action rules	-	-
Leite et al. [95] Leite and Girardi [94]	learning agents	-	learning ability	-
Merkle and Philipp [101]	learning agents	-	learning ability	-
Paolucci and Sycara [117]	autonomous semantic web services	-	-	-
Papamarkos et al. [118] Papamarkos et al. [119]	-	event-condition-action rule language	-	-
Payne [120]	learning agents	environmental changes	goals	social awareness
Pham and Stacey [121]	-	-	goals	-
Poulovassilis et al. [123]	-	event condition action rules	-	-
Rajpathak and Motta [126]	-	-	goals	-
Sycara et al. [144]	discover & synchronisation	-	-	-
Tamma and Payne [145]	service providers & autonomous agents	-	-	-
Tamma et al. [146]	autonomous system components	update indexes	update indexes	collaborative query answering
Terziyan [147]	autonomous resources	-	modelling context, dynamics, & coordination	-
Tonti et al. [148]	policy agents	-	-	social awareness
Van Riemsdijk et al. [154]	normative agents	-	-	socially adaptive agents

Pro-activeness. Buhler and Vidal [25] argue that semantic web services together with semantic behavioural description can be used by agents in order to achieve pro-active behaviour. The proposed approach also serves as a foundation for the ontology based intelligent agent framework proposed by García-Sánchez et al. [53, 52]. Rajpathak and Motta [126], Khalili et al. [86], Payne [120] and Pham and Stacey [121] consider pro-activeness in terms of goal directed agent behaviours, with Ermolayev et al. [43] also considering pro-activeness in terms of collaborative goals in a multi-agent system. The agents proposed by Chiu and Leung [32], Dong et al. [41], Demarchi et al. [36], Challenger et al. [30] all employ the belief–desire–intention software model. While, Payne [120], Leite et al. [95], Leite and Girardi [94], Buoncompagni et al. [26], Kootbally et al. [90], Merkle and Philipp [101] and Ghanadbashi and Golpayegani [56] examine how agents can be enhanced with pro-active learning ability. In turn, the multi-agent information system infrastructure proposed by Chiu and Leung [32] is rooted in the believe-desire-intention framework whereby ontologies are used to encode knowledge that the agent acts upon. While, Terziyan [147] argues that semantic web standards need to be extended in order to cater for context, dynamics, and co-ordination, necessary to facilitate proactivity between agents. In the context of their searching for semantic web content system, according to Tamma et al. [146] a proactive approach should be used by agents to inform other agents of any local changes.

Social ability. Tamma et al. [146] are guided by requirements relating to searching the semantic web whereby agents collaborate in order to answer queries. Artz and Gil [5] highlight the importance of social networks when it comes to trust in and among agents. Buhler and Vidal [25] discuss the role of agent cooperation and coordination from a workflow enactment perspective. While, Ermolayev et al. [43] identify the need for social commitments and conventions to regulate group activities. Both Khalili et al. [86] and Payne [120] highlight the fact that agents are socially aware, however Khalili et al. [86] furthers the notion by highlighting the importance of a common agent communication language. Bonatti et al. [17] demonstrate the effectiveness of their reactive policies and negotiation framework using a social network communication tool. The agents proposed by Van Riemsdijk et al. [154] are socially adaptive agents in the sense that they strive towards norm compliance. Tonti et al. [148] also adopt a social perspective, highlighting the need for policies that can constrain agent behaviour.

4.2. Behavioural Functions

The works presented in *Table* 7 and discussed in more detail below provide different perspectives on the behavioural functions that could potentially be built into intelligent software web agents.

Benevolence. Both Artz and Gil [5] and Jutla et al. [79] briefly mention benevolence in the context of making decisions, however Artz and Gil [5] qualifying its use as a willingness to expend the effort needed to establish trust. Khalili et al. [86] focus on the assumption that benevolent agents do not have conflicting goals. Ermolayev et al. [43] discuss benevolence from a multi-agent group utility perspective, highlighting the need to balance self-interest and benevolence. While, Gandon [50] focus on the societal benefit of the web, arguing that artificial intelligence based applications need to be benevolent by design.

Rationality. According to Payne [120] agents need to act rationally when it comes to decision making, for instance by considering the utility gain in terms of a reward or a perceived advantage. In addition to defining rational behaviour, Khalili et al. [86] also specifically state that it is assumed that agents don't act in a counter productive manner. While, Ermolayev et al. [43] discuss rationality from a multi-agent perspective, focusing on the need to balance individual-rationality from a self-interest perspective and benevolence when it comes to group dynamics. Whereas, Tamma and Payne [145] identify the need for bounded rational deliberation when it comes to partial knowledge and updates to existing knowledge.

Responsibility. Paolucci and Sycara [117] discuss responsibility purely from a web service architecture perspective. While, Bryson et al. [24] focus on responsibility from a data retention perspective. The context broker architecture proposed by Chen et al. [31] focuses specifically on the responsibilities of the context broker agent which is at the core of the proposed meeting system. Demarchi et al. [36] in turn examine responsibility from an architectural perspective., identifying the need for responsible components. While, García-Sánchez et al. [52, 53] take an intelligent agent perspective, identifying several different types of agents that are differentiated from one another via roles and responsibilities.

Mobility. Khalili et al. [86] list mobility (in terms of ability to move around a network) as one of the requirements of an agent based system. Ermolayev et al. [43] highlight the need for mobile agents in order to ensure the robustness of the system from an availability and a performance perspective. Several authors [139, 31, 133] highlight the key role played by semantic web services when it comes to service discovery in mobile and ubiquitous environments. While, Outtagarts [111] performs a broad survey of mobile agent applications, with semantic web services being one of them.

4.3. Collaborate Functions

In the following, we further elaborate on various works that fall under the collaborative functions heading. *Table* 8 presents existing proposals for intelligent software web agents that are particularly relevant for both agent to human and agent to agent interactions, as well as internal interactions between agent components.

Interoperability. Several authors [68, 43, 117, 144, 52, 53, 105] focus on interoperability from a web service perspective, putting a particular emphasis on automatic discovery, execution, selection, and composition. However, only García-Sánchez et al. [52, 53] distinguish between data, process, and functionality interoperability. Both Gladun et al. [60] and Tamma and Payne [145] highlight the need for standardisation when it comes to the interoperability in multi-agent systems. In particular, Tamma and Payne [145] differentiate between syntactic, semantic, and semiotic interoperability. While, Shafiq et al. [138] focus specifically on communication between software agents and semantic web services by proposing an architecture that allows for interoperability via middleware that performs the necessary transformations. More recently, Harth and Käfer [66], Käfer and Harth [80] and Schraudner and Charpenay [132] have proposed agent architectures that are heavily reliant on linked data standards, which are interoperable by design.

Table 7: Intelligent software web agents behavioural function perspectives.

	Benevolence	**Rationality**	**Responsibility**	**Mobility**
Artz and Gil [5]	benevolence & trust	-	-	-
Bryson et al. [24]	-	-	data retention	-
Chen et al. [31]	-	-	context broker agent	service discovery
Demarchi et al. [36]	-	-	agent platform components	-
Ermolayev et al. [43]	self-interest & benevolence.	rationality & group dynamics	-	service mobility & availability
Gandon [50]	benevolence & societal benefit	-	-	-
García-Sánchez et al. [52] García-Sánchez et al. [53]	-	-	agent types, roles, & responsibilities	-
Jutla et al. [79]	benevolence, trust & integrity	-	-	-
Khalili et al. [86]	conflicting goals	goal oriented decision making	-	move around a network
Paolucci and Sycara [117]	-	-	web service architecture	-
Payne [120]	-	goal oriented decision making	-	-
Scioscia et al. [133]	-	-	-	service discovery
Sheshagiri et al. [139]	-	-	-	service discovery
Tamma and Payne [145]	-	partial & updated knowledge	-	-

Table 8: Intelligent software web agents collaborative function perspectives.

	Interoperability	**Communication**	**Brokering services**	**Inter-agent coordination**
Bonatti et al. [17]	-	policies & trust	-	-
Berners-Lee et al. [8]	-	ontologies, co-ordination & collaboration	-	-
Bryson et al. [23]	-	protocols	-	internal co-ordination
Ermolayev et al. [43]	web services	standard languages & vocabularies	-	ontologies
García-Sánchez et al. [52] García-Sánchez et al. [53]	web services	ontologies	data, process & function mediation	ontologies
Gibbins et al. [57]	-	standard languages & vocabularies	system architecture	-
Gladun et al. [60]	standards for interoperability	-	-	-
Harth and Käfer [66]	linked data standards	-	-	-
Hendler [68]	web services	ontologies	logical descriptions	-
Huhns [74]	-	protocols & standard languages & vocabularies	enhanced directory services	autonomy vs coordination
Käfer and Harth [80]	linked data standards	-	-	-
Motta et al. [105]	web services	protocols	interaction framework	-
McIlraith et al. [100]	-	-	interaction framework	-
Paolucci and Sycara [117]	web services	protocols	-	coordinating role
Schraudner and Charpenay [132]	linked data standards	indirect communication	-	-
Shafiq et al. [138]	web services & agents	protocols	-	-
Sycara et al. [144]	web services & agents	ontologies	interaction framework	matchmaking & brokering
Tamma and Payne [145]	standards for interoperability	ontological equivalence & reconciliation	-	-
Tonti et al. [148]	-	communication policy	-	-

Communication. Berners-Lee et al. [8] discusses the difficulties encountered when it comes to co-ordination and communication internationally. Huhns [74], Bryson et al. [23], Paolucci and Sycara [117], Shafiq et al. [138], and Motta et al. [105] highlight the role played by various protocols (e.g., Web Services Description Language (WSDL), Universal Description, Discovery and Integration (UDDI), and Simple Object Access Protocol (SOAP)), when it comes to web service publishing, finding, and binding. Berners-Lee et al. [8], Hendler [68], Sycara et al. [144], and García-Sánchez et al. [52, 53] propose the use of shared vocabularies in the form of ontologies for communication between service providers and consumers. García-Sánchez et al. [52, 53] extend their use in order to cater for communication between architectural components. Both of which raise issues from an interoperability perspective, especially in relation to ontological equivalence and reconciliation, as argued by Tamma and Payne [145]. When it comes to communication between agents, Ermolayev et al. [43], Gibbins et al. [57], and Huhns [74] highlight the need to standardise communication languages and vocabularies, in order to facilitate communication between agents. While, Bonatti et al. [17] discuss the role played by policies and trust with a particular focus on negotiation. From a communication management perspective, the communication architecture proposed by Schraudner and Charpenay [132] ensures that agents can only communicate with each other indirectly via the environment, whereas Tonti et al. [148]

use policies to control communication between agents.

Brokering services. Hendler [68] highlight that adding logical descriptions to web services will facilitate automated match making and brokering. McIlraith et al. [100], Motta et al. [105], and Sycara et al. [144] propose frameworks whereby agent brokers are used to manage the interaction between service providers and consumers. While, Gibbins et al. [57] propose a system architecture and discuss its effectiveness via a proof of concept simulator based application. Huhns [74] in turn discusses how directory services could be enhanced via brokerage services that help to refine the number of potential sources that need to be consulted. García-Sánchez et al. [52, 53] highlight the importance of interoperability when it comes to the brokering process, which is further subdivided into data, process, and functional mediation.

Inter-agent coordination. Huhns [74] highlights the tensions between autonomy and coordination, as it is necessary to relinquish some autonomy in order to honour commitments. The architecture proposed by Paolucci and Sycara [117] distinguishes between peers and super peers, the latter being responsible for coordinating several peers. While, Bryson et al. [23] argue that there is also the need to have co-ordination internally, for instance between software modules, such that it is possible to develop composite services. Both Ermolayev et al. [43] and García-Sánchez et al. [52, 53] propose the use of common vocabularies in the form of ontologies for both inter-agent communication and co-ordination. While, Sycara et al. [144] highlight the key roles played by matchmaking and brokering when it comes to multi-agent co-ordination.

4.4. Code of Conduct Functions

The functions summarised in *Table* 9 and further elaborated on below are particularly relevant for the controller component, however they may also impact the design of several other components, thus they need to be considered when it comes to the architectural design of the system.

Identification. Several authors [8, 68, 43, 110, 52, 53, 145, 66, 80] highlight the role played by Uniform Resource Identifiers (URIs) when it comes to the identification of resources (e.g., web services, ontologies, agents). While, Artz and Gil [5], Gandon and Sadeh [51] and Kirrane and Decker [87] focus on the authentication of actors using credentials in the form of digital signatures together with policies.

Security. Both Gandon and Sadeh [51] and Kirrane and Decker [87] identify the need for access control policy specification and enforcement. Although, Chen et al. [31] argue that together ontologies and declarative policies can be used for both privacy and security, they do not go into specific details on their use from a security perspective. Artz and Gil [5] discuss security in terms of using credentials and policy languages used in order to determine trust in an entity. While, Kagal et al. [82] propose ontologies that can be used to sign and encrypt messages exchanged between service providers and consumers. Sycara et al. [144] in turn argue that matchmakers can be used to address security concerns by offering a choice of providers.

Privacy. Chen et al. [31] discuss how their context broker architecture can be used to control the sharing and use of personal data. Jutla et al. [79] propose an agent based architecture that can be used to allow the specification and enforcement of privacy preferences. Gandon and Sadeh [51] in turn propose an agent based architecture that protects and mediates access to personal resources. Both Jutla and Xu [78] and Palmirani et al. [115] propose high level ontologies that can be used to specify privacy protection mechanisms in the form of laws, standards, societal norms, and guidelines. In addition, the authors describe how the proposed ontologies could be used by privacy agents to identify privacy issues. Whereas, Bao et al. [7] propose a framework that can be used for privacy preserving reasoning when only partial access to data is permitted. Artz and Gil [5] discuss privacy from a trust negotiation perspective and point to several policy languages that can be used to protect privacy. Kravari et al. [91] also focus on enhancing privacy via trust, proposing a policy-based e-Contract workflow management methodology. Sycara et al. [144] identify matchmakers as a means to cater for better privacy by offering a choice of providers.

Trust. Berners-Lee et al. [8] discuss the role played by digital signatures when it comes to verifying that information has been provided by trusted sources. While, Hendler [68] focuses more broadly on using proof exchange to facilitate trust. Additionally, the detailed survey on trust models and mechanisms at the intersection of trust and the semantic web, conducted by Artz and Gil [5], is motivated by the need for agents to make trust judgements based on available data that may vary in terms of quality and truth. The web service composition framework proposed by Ermolayev et al. [43] considers both the credibility and trustworthiness of service providers as a key requirement that needs to be considered. While, Chen et al. [31], Jutla and Xu [78], and Jutla et al. [79] examine trust from a personal data processing perspective, proposing a system that can be used to determine if the users privacy preferences are adhered to. Kirrane and Decker [87] highlight the link between trust, transparency and provenance and point to several potential starting points. Kravari et al. [91] propose a policy-based e-Contract workflow management methodology that can be used to establish trust between agents and service providers. While, Tonti et al. [148] highlight the role between trust and policies from a trust management perspective.

Ethics. When it comes to intelligent software agents, Casanovas [28] argues that there is a need for both normative and institutional regulatory models in order to not

Table 9: Intelligent software web agents code of conduct function perspectives.

	Identity	Security	Privacy	Trust	Ethics
Artz and Gil [5]	credentials & policies	credentials & policies	policies	trust judgements	-
Bao et al. [7]	-	-	privacy preserving reasoning	-	-
Berners-Lee et al. [8]	URIs	-	-	digital signatures	-
Casanovas [28]	-	-	-	-	regulatory models
Chen et al. [31]	-	ontologies & policies	ontologies & policies	trust & privacy	-
Ermolayev et al. [43]	URIs	-	-	credibility & trust	-
García-Sánchez et al. [52] García-Sánchez et al. [53]	URIs	-	-	-	-
Gandon and Sadeh [51]	credentials & policies	access policies	privacy architecture	-	-
Harth and Käfer [66]	URIs	-	-	-	-
Hendler [68]	URIs	-	-	proofs	-
Jutla and Xu [78]	-	-	privacy ontology	trust & privacy	-
Jutla et al. [79]	-	-	privacy architecture	trust & privacy	-
Käfer and Harth [80]	URIs	-	-	-	-
Kagal et al. [82]	-	digital signatures & encryption	-	-	-
Kirrane and Decker [87]	credentials & policies	access policies	-	trust & provenance	usage policies
Kravari et al. [91]	-	-	privacy contracts	trust via contract	-
Oren et al. [110]	URIs	-	-	-	social conventions e.g., robots.txt
Palmirani et al. [115]	-	-	privacy ontology	-	-
Sycara et al. [144]	-	matchmakers & security	matchmakers & privacy	-	-
Tamma and Payne [145]	URIs	-	-	-	-
Tonti et al. [148]	-	-	-	trust & policies	

only reason over legal norms, but also judicial and political decision making, best practices, ethical principles and values. Oren et al. [110] focus specifically on good behaviour when it comes to crawling data, stating it is important to respect the robot.txt access restrictions and to be mindful of the resource limitations of the data provider. Kirrane and Decker [87] identify usage restrictions in the form of access policies, usage constraints, regulatory constrains, and social norms as key requirements needed to realise the intelligent web agent vision.

4.5. Robustness Functions

Finally, the existing work at the intersection of intelligent software web agents and robustness, summarised in *Table* 10, is useful for both assessing the maturity of the exiting proposals, and comparing and contrasting different technological choices.

Stability, Performance & Scalability. Shafiq et al. [138] examine the performance of both their service lookup via UDDI and service invocation via WSDL, and conclude that lookups scale linearly with increasing parameters, while service invocation depends on the complexity of the input and output parameters. Although Jutla et al. [79] do not conduct a performance assessment they identify the need for borrowing/extending metrics from other domains, such as response time, throughput, effectiveness, ease of use, and usefulness. Likewise, García-Sánchez et al. [52, 53] discuss the importance of performance assessment and provide detailed plans that they aim to execute in future work. While, Scioscia et al. [133] used a reference dataset to compare their reasoning engine performance, in terms of both classification and satisfiability, to other well known reasoners, and the memory usage on a mobile device to that of a personal computer. Sycara et al. [144] take a broad view on performance identifying the need to assess different performance characteristics, such as privacy, robustness, adaptability, and load balancing.

Verification. Scioscia et al. [133] used a reference dataset to assess the effectiveness of their reasoner on a classification task, in terms of correctness, parsing errors, memory exceptions, and timeouts. While, Gandon and Sadeh [51] perform an empirical evaluation of their architecture via a campus community agent application, where participants were asked to perform various tasks with a view to obtaining feedback on the effectiveness of the system. Leite et al. [95] and Leite and Girardi [94] demonstrate the effectiveness of their proposals by performing a comparitive analysis to other hybrid agent architectures. Additionally, a number of authors evaluated their proposals using smart home [80], production environment [132], and real time traffic [56] simulations. Other evaluations included ruleset correctness [77], safety [92], and norm compliance checking [154].

5. Intelligent Software Web Agents: Architectural Components

In the following, we propose a hybrid semantic web agent architecture, and discuss how the architecture components introduced in *Section* 2, together with semantic

Table 10: Intelligent software web agents robustness function perspectives.

	Stability, Performance & Scalability	**Verification**
Gandon and Sadeh [51]	-	empirical evaluation
García-Sánchez et al. [52] García-Sánchez et al. [53]	evaluation plans	-
Ghanadbashi and Golpayegani [56]	-	simulation
Jochum et al. [77]	-	ruleset correctness
Jutla et al. [79]	borrowing/extending metrics	-
Käfer and Harth [80]	-	simulation
Ksystra and Stefaneas [92]	-	safety properties
Leite et al. [95], Leite and Girardi [94]	-	comparative analysis
Merkle and Philipp [101]	-	learning tasks
Schraudner and Charpenay [132]	-	simulation
Scioscia et al. [133]	reasoning engine performance & memory usage	reasoning correctness
Shafiq et al. [138]	service lookup & invocation performance	-
Sycara et al. [144]	several different performance characteristics	-
Van Riemsdijk et al. [154]	-	norm compliance checking

web standards and community activities, could potentially be used to realise our information, booking, and planning agents. Rather than proposing three different architectures we propose a single architecture with optional components. The proposed hybrid agent architecture, which is depicted in *Figure* 4, provides support for realtime interaction via its reactive component and sophisticated reasoning via its deliberative component, both of which are necessary in order to realise our scheduling agent.

5.1. Interface Component

In our use case scenario, `Sensors` and `Actuators` take the form of either web interfaces, rendered via networked devices used for agent to human interaction, or web services, residing on networked devices used for agent to agent interactions. *Table* 11 summarises the relevant W3C standardisation efforts and community activities discussed in detail below.

Sensors & Actuators. The Hypertext Transfer Protocol (HTTP)[1] is an application level protocol that forms the basis for data communication via the web. Communication involves a simple request/response protocol that can be used for data exchange. The Linked Data Notifications (LDN)[2] specification in turn describes how HTTP together with the Resource Description Framework (RDF)[3] can be used by senders to push messages to recipients. When it comes to serving web content there are numerous web servers to choose from (c.f., NGINX[4], Apache Tomcat[5]). From a web interface perspective, the Hypertext Markup Language (HTML)[6] and Cascading Style Sheets (CSS)[7] can be used to develop responsive web applications that enable humans to interact with intelligent software agents. From a web services perspective, the Simple Object Access Protocol (SOAP)[8] and the Representational State Transfer (REST)[9] architecture style are the predominant Application Programming Interface (API) approaches used in practice. Web service discovery is supported via registries and indexes, whereby protocols such as the Universal Description, Discovery and Integration (UDDI)[10] can be used to publish and discover web services [18]. There are also several standardisation initiatives relating to semantic web services that use formal ontology-based annotations to describe the service in a manner that can be automatically interpreted by machines (c.f., the Web Ontology Language for Web Services (OWL-S)[11], the Web Service Modeling Language (WSML)[12], the W3C standard Semantic Annotations for Web Services Description Language (WSDL) and XML Schema (SAWSDL)[13]). When it comes to agent specific standardisation efforts, the Foundation for Intelligent Physical Agents (FIPA) propose several standards that support agent to agent communication [122], such as the FIPA Agent Communication Language (ACL)[14] and the FIPA RDF Content Language Specification[15] which describes how RDF can be used to encode the message content.

Additionally, there have been numerous works that focus on using enhancing, and supplementing existing standards, from an intelligent software web agent perspective. In terms of agent specific languages, Wang et al. [157] compare OWL-S, Web Service Modeling Ontology (WSMO)[16] and SAWSDL from the providers, requesters, and brokers perspectives. On the other hand Pai et al. [112] propose a lightweight ontology-based content language based on the FIPA RDF Content Language (CL). While, Challenger et al. [30] introduce a semantic web enabled agent modelling Language (SEA_ML), which they apply in the context of an E-barter system. Gibbins et al. [57] propose a process ontology, inspired by the FIPA agent commu-

[1] `https://tools.ietf.org/html/rfc7231`

[2] `https://www.w3.org/TR/ldn/`

[3] `https://www.w3.org/TR/rdf11-concepts/`

[4] `https://www.nginx.com/`

[5] `http://tomcat.apache.org/`

[6] `https://www.w3.org/TR/html52/`

[7] `https://www.w3.org/TR/CSS2/`

[8] `https://www.w3.org/TR/soap12-part1/`

[9] `https://www.ics.uci.edu/~fielding/pubs/dissertation/rest\_arch\_style.htm`

[10] `http://uddi.xml.org/specification`

[11] `https://www.w3.org/Submission/OWL-S/`

[12] `https://www.w3.org/Submission/WSML/`

[13] `https://www.w3.org/TR/sawsdl/`

[14] `http://www.fipa.org/specs/fipa00061/index.html`

[15] `http://www.fipa.org/specs/fipa00011/XC00011B.html`

[16] `https://www.w3.org/Submission/WSMO/`

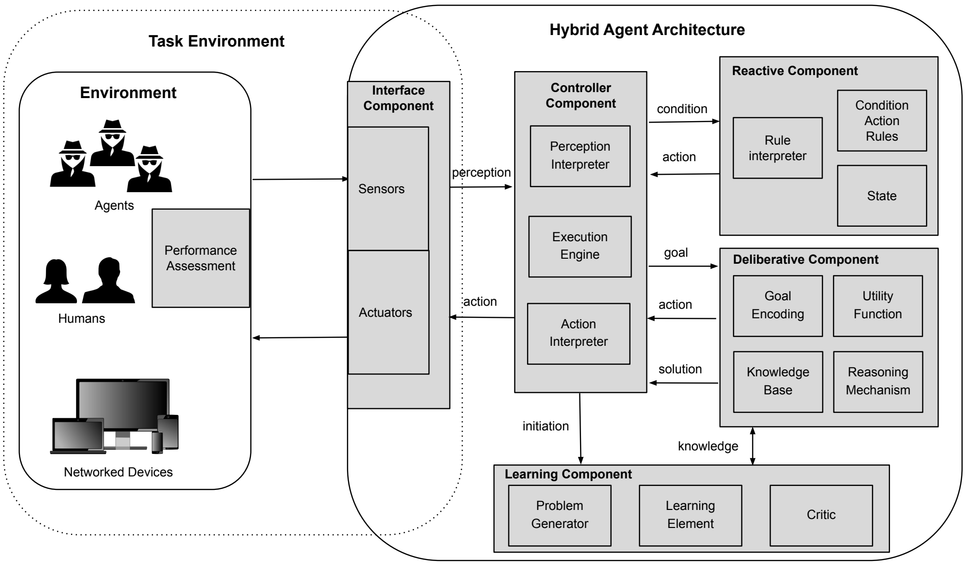


Figure 4: A hybrid agent architecture.

Table 11: Intelligent software web agents interface component.

	Standards	Community Activities
Languages, Ontologies & Vocabularies	HTML, CSS, WSML, WSDL, FIPA ACL, FIPA RDF CL, OWL, OWL-S, SAWSDL, WSMO	FIPA RDF CL extension [112], SEA_ML [30], compare OWL-S, WSMO and SAWSDL [157], FIPA process ontology [57], OWL-S & policies
Models & Frameworks	RDF, UDDI, REST	SWS discovery & composition [136, 155, 137, 135], SWS & JADE [165], SWS design methodology [65], hypermedia controls [156]
Protocols	HTTP, SOAP, LDN	WS publication protocol [137], LDN agent communication protocol [27], ACL/SOAP converter [138]

nication language, that describes various messages types that can be used to describe web services. Whereas, Kagal et al. [82] demonstrate how policies can be embedded into OWL-S descriptions.

When it comes to models and frameworks, Venkatachalam et al. [155] provide a comprehensive survey of existing work on semantic web service (SWS) composition and discovery. More recent works primarily focus on using ontologies to semantically described RESTful web services [34], new approaches for service discovery that leverage user profiles and metadata catalogs [136, 137, 135], and proposing methodologies that support the modelling and design of SWSs [65]. From an implementation perspective, Zapater et al. [165] demonstrate how the JADE Multi-agent System (MAS) development platform can be enhanced with service discovery capabilities. Verborgh et al. [156] in turn argue that there is a need to construct Web APIs out of reusable building blocks and for the use of hypermedia controls to describe both the functional and non-functional aspects of the service.

From a protocol perspective, Seghir et al. [137] propose web service publication and discovery protocols that are represented in the form of sequence diagrams. While, Shafiq et al. [138] propose an abstract architecture, combining web service and FIPA standardisation efforts, which is capable of translating FIPA ACL to SOAP and visa versa. Others have demonstrated how LDNs can be extended to cater for agent communication [27].

5.2. Reactive Component

The `Reactive Component` takes as input a condition and returns an action based on a set of `Condition Action Rules`. More sophisticated reactive components use `State` to further refine the conditions used to determine the action that is required. *Table* 12 summarises existing work both in terms of standardisation and community activities.

Condition Action Rules. When it comes to the specification of condition action rules there are several applicable standardisation efforts. The Production Rule Representation (PRR)[17] specification, developed by the Object Management Group, provides a standard mechanism for encoding rules of the form IF *condition* THEN *action* statements. The Rule Markup Language (RuleML)[18] is a family of languages that provide support for the specification and interchange of both derivation and reaction rules

[17] `https://www.omg.org/spec/PRR/About-PRR/`

[18] `http://wiki.ruleml.org/index.php/Specification_of_RuleML`

Table 12: Intelligent software web agents reactive component.

	Standards	Community Activities
Condition Action Rules		
Languages, Ontologies & Vocabularies	PRR, RuleML, RIF, SWRL, RDF, XML	condition action rules [66, 80], event condition action rules [13, 92, 118, 118, 119, 123, 77], event condition transaction language [62]
Models & Frameworks	RDF	standards based system architectures [66, 132], formal verification framework [92]
State		
Languages, Ontologies & Vocabularies	RDFS, OWL LD	RDF, RDFS, and OWL LD [66], RDF graphs [118, 119, 123]
Models & Frameworks	RDF	memory, commitments, claims, goals, and intentions [13], sequence of states [62], active, inactive & done workflow state [77], check violated state using model checking [92]
Rule Interpreter		
Languages, Ontologies & Vocabularies	RuleML engine, SPARQL	RuleML design rationale [13], SPARQL enabled interpreter [66], parser and translator [118, 119, 123]
Models & Frameworks	RDF	workflow meta model [77], Protune policy engine [17]

[14]. The W3C Rule Interchange Format (RIF)[19] in turn is an interchange format that can be used to exchange rules between different rule systems. The RIF Production Rule Dialect[20] caters specifically for production rules. While, the W3C Semantic Web Rule Language (SWRL)[21] is a language that combines rules and logic, for a subset of RuleML and a subset of the Web Ontology Language (OWL)[22].

Over the years, researchers have proposed a variety of reactive rules languages that allow for the specification of condition action rules [66, 80], event condition action rules [13, 92, 118, 119, 123, 77], and event condition transaction rules that combines condition action rules with transaction logic [62]. When it comes to models and frameworks, Harth and Käfer [66] and Schraudner and Charpenay [132] propose W3C standard based architectures. While, Ksystra and Stefaneas [92] propose a formal framework for analysing reactive rules in order to safeguard against unpredictable behaviour.

State. A reactive agent with state maintains knowledge about the world and the current state of the environment. RDF is a general purpose language that could be used to represent information in a machine interpretable format. The PRR specification describes a metamodel for encoding production rules using the Extensible Markup Language (XML) Metadata Interchange[23], which is abstract in nature. XML is the native encoding for RuleML, however a JavaScript Object Notation (JSON) serialisation is also provided. Although RIF supports several different encodings, XML is the primary medium of exchange between different rule systems. The SWRL specification uses an abstract Extended Backus-Naur Form (EBNF) syntax, which can easily be encoded in XML and/or RDF.

The agents envisaged by Harth and Käfer [66] model state using RDF, RDFS, and OWL LD. While, Papamarkos et al. [118, 119], Poulovassilis et al. [123] use state to refer to RDF graphs. Boley et al. [13] enumerate several categories that encompass the mental state of an agent, from a memory, commitments, claims, goals, and intentions perspective. Considering their transactional focus, Gomes and Alferes [62] work with a sequence of knowledge base states, which they refer to as paths. The system proposed by Jochum et al. [77] proposes three workflow states: active, inactive and done. In their formal verification framework Ksystra and Stefaneas [92] use model checking to find violated states. Whereas, the basic agent architecture proposed by Schraudner and Charpenay [132] does not maintain state.

Rule Interpreter. Although some rule engines provide support for RuleML and SWRL rules (c.f., RDFox[24]), when it comes to condition action rules, a simple interpreter that is able to match conditions would suffice. The rule interpreter is responsible for finding rules whose conditions are satisfied, and for triggering the corresponding actions. In the case of conflicting rules, a conflict resolution mechanism is required.

Boley et al. [13] elaborate on the design rationale underpinning RuleML, which provides support for reactive rules, derivative rules, and integrity constraints. Poulovassilis et al. [123] propose an event condition action rule language that can be applied to RDF data, entitled RDF Triggering Language (RDFTL) in the form of an RDF repository wrapper that leverages the repositories querying capabilities. The interpreters used by Papamarkos et al. [118, 119] and Poulovassilis et al. [123] includes a parser that performs syntactic validation and a translator that translates queries such that they can be executed by the underlying RDF store. In the system proposed by Harth and Käfer [66] conditions are checked against state, by a SPARQL enabled interpreter, with optional support for some simple RDFS and OWL LD based reasoning. Bonatti et al. [17] in turn propose a reactive policy framework, called Protune, that can be used to guide system behaviour.

[19] `https://www.w3.org/TR/rif-overview/`

[20] `https://www.w3.org/TR/rif-prd/`

[21] `https://www.w3.org/Submission/SWRL/`

[22] `https://www.w3.org/TR/2012/REC-owl2-overview-20121211/`

[23] `https://www.omg.org/spec/XMI/2.5.1/PDF`

[24] `https://www.oxfordsemantic.tech/`

Table 13: Intelligent software web agents deliberative component.

	Standards	Community Activities
Knowledge Base		
Languages, Ontologies & Vocabularies	RDFS, OWL, ODRL	belief-augmented OWL [41], normative language ontology [47, 46]
Models & Frameworks	RDF, ODRL	ODRL policy activation & temporal validity [47, 46], belief desire intention principles [30], Jason MAS Platform & ontological knowledge [36], ontology integration & automatic reconciliation
Reasoning Engine		
Languages, Ontologies & Vocabularies	RDFS, OWL, ODRL, SPARQL 1.1 Entailment Regimes	OWL reasoning [2], rule based reasoning[102], reasoning over obligations & permissions [47, 46]
Models & Frameworks	RDF, ODRL	belief desire intention reasoning [30, 36], reasoning over incomplete, subjective & inconsistent data [41]
Goal Encoding		
Languages, Ontologies & Vocabularies	RDFS, OWL	roles, behaviors, plans, beliefs, and goal concepts [30], task, planing & scheduling ontologies [103, 126, 121],
Models & Frameworks	RDF	constraint logic programming goals [41]
Utility Function		
Languages, Ontologies & Vocabularies	RDFS, OWL	degree of inclination [41], utility values assigned to classes using the uDecide protégé plugin [2]
Models & Frameworks	RDF	utility theory based modelling [102], e-bartering economics [30]

5.3. Deliberative Component

The `Deliberative Component` takes as input a goal and either returns a solution or an action (that needs to be carried out before a solution can be determined). The `Knowledge Base` is used to store the knowledge the agent has about the world. The `Goal Encoding` is responsible for intercepting the request and updating the knowledge base accordingly. Together the `Reasoning Engine` and the `Utility Function` are responsible for deriving a solution or further actions that need to be fed back to the execution engine. *Table* 13 summarises relevant standardisation efforts that could be used to realise this component and various community activities that make use of them.

Knowledge Base. A deliberative agent maintains knowledge about the world and the current state of the environment in its knowledge base. In the case of our intelligent software web agents, RDF is used to represent information about resources accessible via the web. The RDF Schema[25] specification defines a set of classes and properties used to describe RDF data. However, using RDFS, it is not possible to represent complex statements that include cardinality constraints, or to model complex relations between classes, such as disjointness or equivalence. The OWL Web Ontology Language[26] standard caters for the encoding of relations between classes, roles and individuals, while at the same time providing support for logical operations and cardinality constraints. Additionally, the Open Digital Rights Language (ODRL)[27] could potentially be used to represent other constraints in the form of policies and norms.

Dong et al. [41] combine OWL with belief augmented frames based logic, which can be used to model evidence for or against a statement. While, Fornara et al. [47] and Fornara and Colombetti [46] describe an OWL based normative language ontology and demonstrate how their ODRL extension caters for policy activation and temporal relevancy can be used by agents to reason about obligations and permissions. The agents envisaged by Challenger et al. [30] are modelled based on the belief-desire intention (BDI) principles [127]. Whereas, Demarchi et al. [36] demonstrates how the Jason multi-agent system development platform [19] can be amended to make use of ontological knowledge available on the web in order to update the agents knowledge base. From an interoperability perspective, Lister et al. [97] propose several alternative strategies that could be used for automatic ontology reconciliation.

Reasoning Engine. OWL2[28] comes in two flavours: OWL2 Full and OWL2 DL. In turn, OWL2 DL is composed of three profiles (OWL2EL, OWL2QL and OWL2RL) that are based on well used DL constructs. The syntactic restrictions imposed on each profile are used to significantly simplify ontological reasoning. OWL 2 EL is designed for applications that require very large ontologies., whereby polynomial time reasoning is achieved at the cost of expressiveness. OWL 2 QL is particularly suitable for applications with lightweight ontologies and a large number of individuals, which need to be accessed via relational queries. Finally, OWL 2 RL provides support for applications with lightweight ontologies, and a large number of individuals that make use of rule based inference and constraints mechanisms. Over the years the community has developed several reasoners that are capable of reasoning over OWL ontologies, albeit often with some restrictions (c.f., Pellet[29], HermiT[30], FACT++[31], Racer[32], and RDFox[24]). Additionally, SPARQL 1.1 Entailment Regimes[33]

[25] `https://www.w3.org/TR/rdf-schema/`

[26] `https://www.w3.org/TR/owl2-overview/`

[27] `https://www.w3.org/TR/odrl-model/`

[28] `https://www.w3.org/TR/owl2-overview/`

[29] `https://www.w3.org/2001/sw/wiki/Pellet`

[30] `http://www.hermit-reasoner.com/`

[31] `http://owl.cs.manchester.ac.uk/tools/fact/`

[32] `http://www.ifis.uni-luebeck.de/~moeller/racer/`

[33] `https://www.w3.org/TR/sparql11-entailment/`

can be used to consider implicit (i.e. inferred) data during query execution based on RDF entailment regimes.

Beyond simple OWL based [2, 41, 47, 46] and rule based [102, 121] reasoning, researchers have demonstrated the potential for reasoning over beliefs, desires and intentions [30, 36], policies and norms [47, 46], and incomplete, subjective, and inconsistent data [41].

Goal Encoding. Given a goal, or set of goals, the agent uses the current state of the world together with the desired state of the world, deduced from its goal(s), in order to infer a solution or further actions that need to be performed. Here RDF, RDFS, and OWL can be used to encode the agents goal(s).

Although there are no standard mechanisms for goal encoding, the domain-specific modelling language proposed by Challenger et al. [30] covers goals and other predominant agent concepts (i.e., roles, behaviors, plans, and beliefs). Additionally, several researchers have proposed task, planing, and scheduling ontologies [103, 126, 121]. While, the belief framework proposed by Dong et al. [41] uses constraint logic programming goals.

Utility Function. The utility function is responsible for assessing possible solutions based on the desired state of the world, and the preferences defined by the person or agent that specifies the goal(s) and associated constraints. According to utility theory [104] informed decisions should be made by examining the goal(s), the actions needed to achieve the goal(s), and the various preferences from a greatest expected satisfaction perspective. Here, a utility theory based modelling, such as that adopted by Brown et al. [21] and Ming et al. [102], could be used to guide the development of the utility function.

Dong et al. [41] define a utility function based on the difference between belief and disbelief values (i.e. the degree of inclination). While, Acar et al. [2] propose a protégé plugin called uDecide that can be used to assign utility values to classes that are subsequently used by the utility function in order to determine the optimal course of action. The template-based ontological method proposed by Ming et al. [102] is rooted in utility theory based modelling [21]. From a domain specific perspective, the semantic web enabled BDI multi-agent system proposed by Challenger et al. [30] builds upon the utility function research specifically focused on the proposed e-bartering system.

5.4. Learning Component

The learning component, which is composed of the `Problem Generator`, `Learning Element`, and `Critic`, could be used to develop more advanced intelligent software web agents that are capable of learning from past experiences and thus becoming more effective over time. This component interacts with both the `Controller Component` and the `Deliberative Component`. The former is responsible for initiating the learning process, while the latter is used to ascertain existing knowledge, perform learning based reasoning tasks, and store the outputs of the learning process. Considering that simple agents (such as the information and booking agents presented in *Section* 3) do not necessarily need learning capabilities, in the proposed architecture, following a typical separation of duties engineering practice, we separate the learning component from the deliberative component. That being said, it is worth noting that there is a high level of interaction between these components. Although the tools and techniques that could be used to support agent learning have not yet been considered from a standardisation perspective, in *Table* 14 we summarise preliminary research that could form a starting point for potential standardisation discussions.

Learning Element. Although the W3C doesn't have any specific groups exploring standardisation potential with respect to learning agents, these agents could benefit from many of the standards discussed under the deliberative component. Additionally there has been several related initiatives that could be considered for the realisation of the learning element. For instance, the W3C Ontology-Lexicon Community Group[34] has developed a lexicon model for ontologies[35] that can be used to enrich ontologies with linguistic information. While, the W3C Web Machine Learning Working Group[36] aims to develop Web APIs that enable machine learning in the browser.

From an ontological perspective, both Wong [160] and Puerto et al. [124] demonstrate how various ontology learning techniques can be used to enhance manually crafted ontologies. While, Merkle and Philipp [101] show how reinforcement learning can be used to enhance the policies or strategies used by agents to complete their tasks. That being said, it's worth noting that according to Albrecht and Stone [3] many learning techniques are computationally complex making them unsuitable for many real world use case scenarios. When it comes to the agent learning semantic web models and frameworks, Leite et al. [95] and Leite and Girardi [94] propose high level ontology-driven hybrid agent architectures that include separate problem generator, critic and learning components that are used by the deliberative component in order to improve both the deliberative knowledge base and the reactive rules. While, Young et al. [164] demonstrate how spatial information about unknown objects together with their semantic web meaning can be used by robots to classify the unknown object. Ghanadbashi and Golpayegani [56] in turn introduce their automatic goal generation model and a corresponding workflow that enables agents to evolve existing goals or create new goals based on emerging requirements. More broadly, Asim et al. [6] highlight that ontology learning has

[34] `https://www.w3.org/community/ontolex/`

[35] `https://www.w3.org/2016/05/ontolex/`

[36] `https://www.w3.org/2021/04/web-machine-learning-charter.html`

Table 14: Intelligent software web agents learning component.

	Standards	Community Activities
Problem Generator & Critic		
Languages, Ontologies & Vocabularies	RDFS, OWL	OWL & PDDL [26, 90]
Models & Frameworks	RDF	critic, learning element & problem generator [95], adaptive behaviour & norms [154, 26]
Learning Element		
Languages, Ontologies & Vocabularies	OWL	ontology learning [160, 124], reinforcement learning & policies/strategies [101], computationally complex [3]
Models & Frameworks Models	RDF	critic, learning element & problem generator [95, 94], spatial information & semantic web mining [164], automatic goal generation model [56], linguistic, statistic and logic based [6]

benefited from a variety of domains, namely natural language processing, machine learning, information retrieval, data mining and knowledge representation. The authors perform a comprehensive survey of existing work, categorising them as linguistic, statistic, and logic based.

Problem Generator & Critic. According to Russel and Norvig [130] the problem generator suggests actions that will lead to learning in the form of new knowledge and experiences. While the critic provides feedback to the agent in the form of a reward or a penalty. Although the problem generator and the critic could vary greatly from an internal implementation perspective, there is a need for standardised vocabularies and APIs that can be used to manage synchronisation and communication between the various internal and external components.

From a vocabularies perspective, there has been some relevant work in terms of robotics, whereby Buoncompagni et al. [26] and Kootbally et al. [90] demonstrate how the Planning Domain Definition Language (PDDL) [37] problem generator can make use of OWL reasoners to check for and solve issues with respect to norm compliance. As for models and frameworks, Leite et al. [95] and Leite and Girardi [94] present architectures whereby the agents perceive the effects that their actions have on the environment, pass this information to the critic, which in turn informs the learning component about poor performance. The learning component also recommends improvements and the problem generator is responsible for proposing new actions based on these recommendations. Van Riemsdijk et al. [154] focus on the weaker notion of norm compliance and propose a semantic framework that demonstrates how agents identify problems and adapt their behaviour in order to avoid violating norms.

5.5. Controller Component

The `Controller Component` is responsible for interpreting perceptions from sensors via the `Perceptions Interpreter`, devising execution plans that leverage the reactive and deliberative components, and executing the plans via the `Execution Plan Management`. In addition, this component is responsible for advising the actuators what action(s) need to be taken via the `Actions and Solutions Interpreter`. Although the tools and techniques that are needed to support agent control have not yet matured in terms of W3C standardisation efforts, in *Table* 15 we summarise preliminary research that could form the basis of initial standardisation discussions.

Perception & Action Interpreters. Irrespective of whether we are dealing with a web service or a web application, there is a need to define interfaces that can be used to interact with the agent. The perception interpreter is responsible for forwarding perceptions to the `Execution Engine`, while the action interpreter in turn is responsible for initiating actions forwarded by the `Execution Engine`. When it comes to simple read and write operations, the Linked Data Platform (LDP)[38] specification provides a set of best practices for an architecture that supports accessing, updating, creating and deleting Linked Data resources. However, said architecture would need to be amended to provide support for additional functions required in order to cater for interaction between intelligent agents.

Challenger et al. [30] discuss the role played by messages and message sequences when it comes to agent interaction and highlight that they could be based on some standard such as FIPA_Contract_Net. More broadly, there are a range of FIPA standards[39] that could potentially be leveraged by intelligent software web agents. From a goal encoding perspective, Pham and Stacey [121] propose an ontology than can be used for modelling planning problems that could be worked on by goal driven agents. In the frameworks proposed by Leite et al. [95] and Leite and Girardi [94] perceptions and actions are represented as ontologies, however the authors focus on the general framework as opposed to the languages, vocabularies and ontologies that could be used for modelling perceptions and actions. Young et al. [164] in turn focus on leveraging external knowledge bases in order to obtain semantic descriptions for unknown objects.

[37] `https://helios.hud.ac.uk/scommv/IPC-14/repository/kovacs-pddl-3.1-2011.pdf`

[38] `https://www.w3.org/TR/ldp/`

[39] `http://www.fipa.org/specs/fipa00025/XC00025E.html`, `http://www.fipa.org/repository/ips.php3`

Table 15: Intelligent software web agents controller component.

	Standards	Community Activities
Perception & Action Interpreters		
Languages, Ontologies & Vocabularies	FIPA_Contract_Net	messages & message sequences [30], generic planning ontology [121],
Models & Frameworks		ontological representation [95, 94], semantic descriptions for unknown objects [164]
Execution Engine		
Languages, Ontologies & Vocabularies	OWL, OWL-S, ODRL	control via policies [148], Protune policy engine [17], normative language ontology [47, 46], agent modelling language [30], legal ontology Palmirani et al. [115]
Models & Frameworks	LDP	KAoS, Rei, Ponder comparison [148], context broker architecture [31], agent platform comparison [113], JACK & OWL-S [30], abstract state machines, Linked Data-Fu & LDP [80], RDF/RDFS & RuleML [13], Jason interpreter [36],

Execution Engine. The `Execution Engine` is responsible for routing conditions to the `Reactive Component` and goals to the `Deliberative Component`, thus the component needs to be able to handle both real-time and delayed responses. Although there is a lack of specific W3C standardisation activity concerning the execution engine, the semantic web community have proposed several tools and technologies that make use of web standards.

Boley et al. [13] discuss how RDF/RDFS and RuleML can together be used to develop simple reactive software agents. Tonti et al. [148] investigate how policies can be used to control agent behaviour by separating a systems functional and governance aspects. The authors compare and contrast the policy management approaches of the KAoS [151, 152], Rei [83, 82] and Ponder [33] policy languages and frameworks when it comes to controlling communication. Chen et al. [31] also focus on policy enforcement, proposing a context broker architecture that can be used to control the sharing and use of personal data. When it comes to general policy languages, the Protune policy engined proposed by Bonatti et al. [17] has also been used to control reactive behaviour. Fornara et al. [47] and Fornara and Colombetti [46] in turn propose a normative language ontology, derived from the ODRL standard, that could be used to control agent behaviour. While, Palmirani et al. [115] introduce their legal ontology that could be used to design privacy preserving intelligent agents. Poulovassilis et al. [123] propose an abstract architecture that could guide the development of an event-condition-action reactive agent. Whereas, Käfer and Harth [80] use abstract state machines in order to model the internals of their reflexive agents and demonstrate the effectiveness of their proposal using Linked Data-Fu[40] together with their Linked Data Platform implementation[41]. Challenger et al. [30] propose a platform independent multi-agent system development methodology and demonstrate its effectiveness using the JACK multi-agent system development framework together with OWL-S models. Whereas, Demarchi et al. [36] demonstrate how the Jason interpreter can be adapted to benefit from ontological knowledge. More broadly, Pal et al. [113] provide a comprehensive review of platforms that can be used to develop agent based systems, however their suitability for developing intelligent web agents is still and open area of research.

6. Intelligent Software Web Agents: The Future

The goal of this section is to use insights gained from the analysis of the intelligent software web agent requirements and the hybrid agent architecture components, in order to highlight existing research opportunities and challenges. In addition, we take a broader perspective of the research by discussing the potential for intelligent software web agent as an enabling technology for emerging domains, such as digital assistants, cloud computing, and the internet of things.

6.1. Opportunities and Challenges

A condensed overview of the intelligent software web agents requirements analysis (focusing on the scheduling agent, which is the most complicated out of the three agents we examined), their impact from an architectural perspective, and the corresponding opportunities and challenges discussed below is presented in *Table* 16.

Core aspects of the hybrid agent architecture. The reactivity, pro-activeness, interoperability, and communication requirements are classified as core modules that are inherent to the hybrid agent architecture presented in the previous section. When it comes to instantiating the architecture there are several standards and technologies that could be leveraged in order to realise the `Interface`, `Reactive`, and `Deliberative Components`. Additionally, over the years the research community have proposed various approaches for semantic web service discovery and composition methods; event condition action rule languages and frameworks; and approaches for representing and reasoning over roles, behaviours, norms, beliefs, goals and plans, that could serve as a basis for developing a simple scheduling agent prototype. However, the suitability of the various proposals from both a practical perspective and a performance and a scalability perspective has yet to be determined.

[40] `https://linked-data-fu.github.io/`

[41] `https://github.com/kaefer3000/ldbbc/`

Table 16: Intelligent software web agents requirements assessment.

Functions	Use Case Requirements	Architectural Impact	Opportunities	Challenges
Basic Functions				
Autonomy	consult relevant sources, devise an optimal schedule	cross cutting	ontology learning and reinforcement learning techniques	adopt a learning theory perspective
Reactivity	immediate response where possible	core	event condition action rule languages and frameworks	reference architecture
Pro-activeness	scheduling goal, explore alternatives	core	well established standards, techniques for representing & reasoning over roles, behaviours, norms, beliefs, goals & plans	reference architecture
Social ability	humans and agents	cross cutting	policy & norm languages	virtual organisations management techniques, policy & norm standards
Behavioural Functions				
Benevolence	well meaning by design, manage conflicting goals	cross cutting	-	benevolent by design
Rationality	rational by design	cross cutting	-	rational by design
Responsibility	mange access to information, finds optimal schedule given a set of constraints	task specific	task environment requirements assessment	requirements elicitation techniques
Mobility	interacts with several other agents	cross cutting	-	discovery of services in mobile and ubiquitous environments, technological advances supporting agent mobility
Collaborative Functions				
Interoperability	agreed/common schema	core	established standards	reference architecture
Communication	push and pull requests	core	established standards, norms & policies	reference architecture
Brokering services	collects information from a variety of sources	cross cutting	established standards, semantic web service discovery & composition techniques	reference architecture
Inter-agent co-ordination	agents support each other via information sharing motivating	cross cutting	policies & norms	virtual organisations management techniques, policy & norm standards
Code of Conduct Functions				
Identification	handle public and private information, may need to prove who they represent	cross cutting	unique identifiers & authentication mechanisms	agent architectures adaptation
Security	protect against unauthorised access, inappropriate use, and denial of service	cross cutting	access control, encryption	agent architectures adaptation
Privacy	handle personal information appropriately	cross cutting	privacy policies, anonymisation	agent architectures adaptation
Trust	manages information and scheduling accuracy, consults reliable sources	cross cutting	trust frameworks and architectures	agent architectures adaptation
Ethics	do no harm by design	cross cutting	legal policies, norms, general guidelines	agent architectures adaptation
Robustness Functions				
Stability	available, reliable & secure	robustness	-	attacker models
Performance	real time access to information, timely goal completion	robustness	-	agent benchmarking tools
Scalability	handles increasing requests, data, & task complexity	robustness	-	agent benchmarking tools
Verification	checks information is correct, the reasoning is explainable	task specific & robustness	task environment requirements assessment, formal method, simulation	requirements elicitation techniques, test driven development

Other challenges relate to the development of the `Controller Component`, which is responsible for internal co-ordination. Existing proposals have focused on defining messages and message protocols; proposing ontologies for specifying norms and legal requirements; and controlling agent behaviour via policies. Unfortunately, much of the work has focused on basic technology research, and many of the proposals have not been validated via prototyping. Here a reference architecture [106] could serve to bridge the gap between theory and practice and to identify potential open challenges that still need to be addressed from an architecture perspective.

Additionally, the `Learning Component`, which is often discussed in the context of learning agents or normative/policy agents, has received little attention to date. Broadly speaking, existing proposals focus on using ontology or reinforcement learning techniques to enhance the agents knowledge base, or demonstrating how agents can adapt their behaviour based on changes in the environment. Here again, there is the need to determine the effectiveness of existing proposal in the form of a prototype. When it comes to multi-agent learning in general, there are several survey articles (cf., [116, 141, 12]) that could serve as the basis for the development of this component. While, from a practical implementation perspective, further research is needed to better understand its role in the overall architecture and what are the concrete standardisation needs.

Task specific considerations. Both the responsibility and verification requirements has been classified as task specific. In our motivating use case scenario, we identified three different types of agents, namely information agents, booking agents, and scheduling agents. Clearly this is not an exhaustive list of agent types, however considering the original semantic web vision has not yet been realised it is beneficial to revisit this simple use case, before moving on to more complex scenarios that can leverage intelligent web agents. The key take home from an architectural perspective, is that we should strive to develop optional task modules that could serve a variety of use case scenarios. Here again there is an need to consider how such task specific modules could be integrated into a reference architecture and described in detail from a system design perspective. The agent task environment requirements assessment framework proposed herein can be used not only to perform a detailed analysis of various agent based use case scenarios, but also to better understand the potential solutions and the technological and standardisations gaps that still exist. Interesting directions for future work include adopting software engineering requirements elicitation techniques [166, 143, 61] and lessons learned [35] in order to better understand the various use case requirements.

Cross cutting generic considerations. The behavioural functions (i.e., benevolence, rationality, and mobility), code of conduct functions (i.e., identification, security, privacy, trust, and ethics) and two of the basic functions (i.e., autonomy and social ability) have been classified as cross cutting, as they need to be considered when it comes to the architecture as a whole and also the individual components. Ideally they should be integrated into a reference architecture in the form of optional generic modules that can be used by the agent depending on the task that needs to be carried out. Following common engineering practices, these modules would need to be described from a system design perspective.

In the early days of semantic web research, the behavioural functions (i.e., benevolence, rationality, and mobility) received some interest from intelligent software web agent researchers. In particular, researchers highlighted the need for agents to be benevolent and rational by design, to be capable of balancing self interest and group interests, to take on various roles and responsibilities, and to the need to cater for mobility from a robustness perspective. However, when it comes to the proposed tools, technologies, and standards benevolence, responsibility, and mobility requirements were not even mentioned in the corresponding papers. The lack of recent research in terms of intelligent software web agents behavioural functions is indicative of the communities diversification of interests and the need to better understand the needs of intelligent software web agents both from semantics and a deployment perspective, as argued by Bernstein et al. [9]. The analysis of the intelligent software web agent requirements, and the standards, tools and technologies presented herein is a first step towards better understanding the status quo and the requirements that should guide agents that leverage semantic web technologies.

When it comes to the code of conduct functions, there is a body of work from the semantic web community that has not been applied directly to the semantic web agent use case, that could potentially be leveraged in order to realise the proposed architecture. In the following, we identify several interesting works that could potentially inform the design of our intelligent software web agents. Broadly speaking, existing work in terms of identification focuses on access control for RDF [128, 75, 1, 45, 37, 49, 88] or demonstrating how policy languages can be used for the specification and enforcement of access restrictions [151, 81, 16]. Besides access control, security based research has primarily focused on applying encryption algorithms [58, 55, 84, 44] and digital signatures [85] to RDF data. Work on privacy primarily focuses on applying and extending existing anonymisation techniques such that they work with graph data [125, 67, 96, 142] or catering for the specification and enforcement of privacy preferences [15, 131]. When it comes to trust, Artz and Gil [5] conducted a survey of existing trust mechanisms in computer science in general, and the Semantic Web in particular. In addition, several authors have proposed trust frameworks and architectures [39, 40, 93, 10]. When it comes to ethics, Gordon et al. [63] focus on requirements that are necessary for modelling and reasoning over legal rules and regulations, whereas Palmirani et al. [114] extend RuleML in the form of LegalRuleML such that it can be used to model and reason over both legal norms and business rules. More generally, existing work at the intersection of intelligent agents and ethics [42, 38] or behavioural aspects of intelligent agents [163, 161] could provide insights into the detailed design of these cross cutting generic modules. Interestingly, the *Ethics Guidelines for Trustworthy AI* [71], recently released by the European Commission only briefly mentions agent technologies, instead focusing on artificial intelligence in general. Thus, while this document serves as a useful starting point with respect to codes of conduct for agents, further work is needed to make these guidelines actionable from an intelligent agents perspective.

Basic agent functional requirements relating to autonomy and social ability serve as motivation for research concerning the role of policies and norms when it comes to controlling intelligent software web agent behaviour. However, there has been limited research by the semantic web community in terms of developing truly autonomous agents that are capable of interacting with other agents and the tools, technologies, and standards needed to enable agents to form virtual organisations in order to collaboratively solve problems. Beyond the semantic web community Van Der Vecht et al. [153] propose gradual levels of autonomy that can be catered for via commitments and contracts. More generally, the technical opportunities and challenges relating to the field of agent based computing, identified by Luck et al. [98], are also relevant from an intelligent software web agents perspective. Primary considerations include: viewing autonomy from a learning theory perspective and examining social ability in terms of virtual organisations. The authors also highlight the need for the advancement of tools and technologies to support scalable service discovery and composition, and semantic integration and additional research in terms of transparency, trust, reputation, and negotiation.

Robustness considerations. All four robustness functions (i.e., stability, performance, scalability, and verification) have simply been classified as robustness from an architectural perspective. These requirements need to be considered both when it comes to the detailed design of the system and the choice of technologies. In the proposed architecture the `Performance Assessment` entity which is part of the `Task Environment` is responsible for evaluating the effectiveness of the system from both a functional and a non-functional perspective.

Several of the requirements papers identified the need to evaluate existing proposals in terms of stability, performance and scalability. However, when it comes to the development of tools, technologies, and standards that could be used to evaluate the effectiveness of existing proposals, researchers have primarily focused on developing proof of concepts in the form of basic simulations or assessing formal aspects such as correctness, safety, and compliance. Here again, we see evidence that intelligent software web agent research is more foundational than applied. Considering the crucial role played by both functional and non functional testing from an engineering perspective, there is a need to develop testing strategies and benchmarks in order to advance the research further. More broadly, when it comes to measuring robustness, besides an array of individual performance evaluations for various query and reasoning engines, there is a body of work in relation to benchmarking that could be used/extended in order to benchmark the proposed architecture. For instance, there are well established benchmarks, such as the Lehigh University Benchmark (LUBM)[64] or the Berlin SPARQL Benchmark (BSBM) [11] and promising newcomers, such as the Linked Data Benchmark Council (LDBC) Social Network Benchmark [4]. In addition, it may be possible to leverage existing benchmarking frameworks, such as the general entity annotator benchmarking framework (GERBIL) [150] or the holistic benchmarking for big linked data framework (HOBBIT) [109].

6.2. Semantic Web Agents as an Enabling Technology

Moving beyond the original intelligent software web agent motivating scenario, the tools, technologies, and standards discussed herein could potentially have a much broader impact. For instance, according to Luck et al. [98], the semantic web community provides a semantically rich data model, vocabularies, and ontologies that can be used to describe media and services in a manner that facilitates discovery and composition; and allows for agent to agent information exchange. In the following, we move beyond the original motivating scenario by highlighting the potential impact of intelligent software web agents on emerging domains, such as digital assistants, cloud computing, and the internet of things.

Digital Assistants. Although well known voice assistants, such as Siri, Alexa, and Cortana, are not as sophisticated as the Knowledge Navigator concept proposed by Sculley [134] (when he was the chief executive officer at Apple) the technology has been embedded in various smart home and smart phone products. Common features include sending and receiving text messages and emails, making calls, setting timers and reminders, and control of hardware (e.g., thermostats, lights, audio, video) [73]. However, in order to realise Sculley's Knowledge Navigator these voice assistants need to be enhanced with data discovery and reasoning capabilities, which are at the core of envisaged intelligent software web agents. Considering the sensitive personal nature of the data often captured by such agents, intelligent software web agents could also be employed in order to provide users with more control and transparency with respect to personal data processing.

Cloud Computing. Cloud computing has been around for quite some time, however as technology rapidly evolves so too does the service offering, for instance edge computing is a paradigm whereby computation is performed closer to where the data is consumed [140]. New data infrastructure initiatives, such as GAIA-X [20], envisage virtual data spaces developed on top of federated infrastructure (including high performance computing and edge systems), where data sovereignty and secure exchange are built-in by design. Recently, the term *private 5G networks* is used to refer to industrial networks that require increased reliability, low latency, and strong security [73]. In this context, intelligent software web agents could potentially play a major role both from a resource allocation and a governance perspective. In the case of the former, agents could take on a coordinating role when it comes to virtual organisation / private network formation and monitoring. In the case of the latter, both data and service providers could encode usage constraints and provenance trails using policy languages and ontologies in a manner that supports agent based negotiation and automated compliance checking.

The Internet of Things. The W3C Web of Things initiative[42] focuses on building on existing Web standards in order to facilitate data integration across various IoT platforms. Here, semantic technologies have already been used in order to describe things[43] and facilitate thing discovery[44]. When it comes to the intersection of intelligent

[42] `https://www.w3.org/WoT/`

[43] `https://www.w3.org/TR/wot-thing-description/`

[44] `https://www.w3.org/TR/wot-discovery/`

software web agents and the internet of things, semantic web agents could also play a crucial role in terms of coordinating the usage, management, and governance of things. Additionally, the standards, tools, and technologies discussed herein could provide support for analytics needed in order to optimise supply and value chains that make use of IoT technologies.

7. Conclusions

Motivated by the desire to further advance existing research into intelligent software web agents, in this paper we revisited the original use case scenario proposed in the seminal semantic web paper from a gap analysis perspective. We started by collating and summarising requirements and core architectural components relating to intelligent software agents in general. Following on from this, we used the intelligent software agent requirements to both further elaborate on the semantic web agent motivating use case scenario, and to summarise and classify existing semantic web agent literature. We subsequently used the insights gained in order to propose a hybrid semantic web agent architecture that guided our discussion with respect to relevant standards, tools, and technologies. Following on from this, we used the functional and non-functional agent requirements together with the scheduling agent use case requirements to better understand the opportunities and challenges concerning the realisation of intelligent software web agents. Finally, we broadened the discussion and highlighted the potential of intelligent software web agent as an enabling technology for digital assistants, cloud computing, and the internet of things.

Key outputs include: (i) a task environment requirements assessment framework, based on agent requirements gleaned from the literature, that could be used to perform an in-depth assessment of various agent use case scenarios; and (ii) a hybrid architecture and the corresponding assessment of existing standards, tools, and technologies, which serves as the basis for developing a reference architecture that can be used to realise the original intelligent software web agent vision and to build the foundations needed in order to support more complex use case scenarios.

Based on our analysis, there are a number of gaps that still need to be addressed in order to move the intelligent software web agent vision forward. Firstly, from an architectural perspective, there is a need to develop a reference architecture that could serve to bridge the gap between theory and practice, and to identify potential open research challenges that still need to be addressed. Secondly, from an implementation perspective there is a need to better understand the specific requirements relating to the cross cutting behavioural functions (i.e., benevolence, rationality, and mobility), code of conduct functions (i.e., identification, security, privacy, trust, and ethics), and basic functions (i.e., autonomy, and social ability), the adaptations/extensions needed to existing tools and technologies, and insights into how these tools and technologies fit together with core intelligent software agent technologies and with each other. Finally, from a robustness perspective, there is a need to develop/extend existing benchmarks such that they can be used to both validate and assess the performance and scalability of various instantiations of our hybrid agent architecture.

Acknowledgments

This research is funded by the FWF Austrian Science Fund and the Internet Foundation Austria under the FWF Elise Richter and netidee SCIENCE programmes as project number V 759-N.

References

[1] F. Abel, J. De Coi, N. Henze, A. Koesling, D. Krause, and D. Olmedilla. Enabling advanced and context-dependent access control in rdf stores. In *The Semantic Web*, volume 4825, pages 1–14. Springer Berlin Heidelberg, 2007.

[2] E. Acar, M. Fink, C. Meilicke, and H. Stuckenschmidt. udecide: A protégé plugin for multiattribute decision making. In *Proceedings of the 8th International Conference on Knowledge Capture*, pages 1–4, 2015.

[3] S. V. Albrecht and P. Stone. Autonomous agents modelling other agents: A comprehensive survey and open problems. *Artificial Intelligence*, 258:66–95, 2018.

[4] R. Angles, J. B. Antal, A. Averbuch, P. Boncz, O. Erling, A. Gubichev, V. Haprian, M. Kaufmann, J. L. L. Pey, N. Martínez, et al. The ldbc social network benchmark. *arXiv preprint arXiv:2001.02299*, 2020.

[5] D. Artz and Y. Gil. A survey of trust in computer science and the semantic web. *Journal of Web Semantics*, 5(2):58–71, 2007.

[6] M. N. Asim, M. Wasim, M. U. G. Khan, W. Mahmood, and H. M. Abbasi. A survey of ontology learning techniques and applications. *Database*, 2018, 2018.

[7] J. Bao, G. Slutzki, and V. Honavar. Privacy-preserving reasoning on the semanticweb. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI'07)*, pages 791–797. IEEE, 2007.

[8] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific american*, 284(5):34–43, 2001.

[9] A. Bernstein, J. Hendler, and N. Noy. A new look at the semantic web. *Communications of the ACM*, 2016.

[10] C. Bizer and R. Oldakowski. Using context-and content-based trust policies on the semantic web. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pages 228–229. ACM, 2004.

[11] C. Bizer and A. Schultz. The berlin sparql benchmark. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 5(2):1–24, 2009.

[12] D. Bloembergen, K. Tuyls, D. Hennes, and M. Kaisers. Evolutionary dynamics of multi-agent learning: A survey. *Journal of Artificial Intelligence Research*, 53:659–697, 2015.

[13] H. Boley, S. Tabet, and G. Wagner. Design rationale for ruleml: A markup language for semantic web rules. In *SWWS*, volume 1, pages 381–401, 2001.

[14] H. Boley, A. Paschke, and O. Shafiq. Ruleml 1.0: the overarching specification of web rules. In *International Workshop on Rules and Rule Markup Languages for the Semantic Web*, pages 162–178. Springer, 2010.

[15] P. A. Bonatti and S. Kirrane. Big data and analytics in the age of the gdpr. In *2019 IEEE International Congress on Big Data (BigDataCongress)*, pages 7–16. IEEE, 2019.

[16] P. A. Bonatti and D. Olmedilla. Rule-based policy representation and reasoning for the semantic web. In *Proceedings of the Third International Summer School Conference on Reasoning Web*, pages 240–268. Springer-Verlag, 2007.

[17] P. A. Bonatti, P. Kärger, and D. Olmedilla. Reactive policies for the semantic web. In *Extended Semantic Web Conference*, pages 76–90. Springer, 2010.

[18] D. Booth, H. Haas, F. McCabe, E. Newcomer, M. Champion, C. Ferris, and D. Orchard. Web services architecture. Technical report, W3C Working Group, 2004. URL `https://www.w3.org/TR/ws-arch/#wsdisc`.

[19] R. H. Bordini and J. F. Hübner. Bdi agent programming in agentspeak using jason. In *International workshop on computational logic in multi-agent systems*. Springer, 2005.

[20] A. Braud, G. Fromentoux, B. Radier, and O. Le Grand. The road to european digital sovereignty with gaia-x and idsa. *IEEE Network*, 35(2):4–5, 2021.

[21] S. M. Brown, E. Santos, and S. B. Banks. Utility theory-based user models for intelligent interface agents. In *Conference of the Canadian Society for Computational Studies of Intelligence*, pages 378–392. Springer, 1998.

[22] J. Bryson. Cross-paradigm analysis of autonomous agent architecture. *Journal of Experimental & Theoretical Artificial Intelligence*, 12(2):165–189, 2000.

[23] J. J. Bryson, D. L. Martin, S. A. McIlraith, and L. A. Stein. Toward behavioral intelligence in the semantic web. *Computer*, 35(11):48–54, 2002.

[24] J. J. Bryson, D. Martin, S. A. McIlraith, and L. A. Stein. Agent-based composite services in daml-s: The behavior-oriented design of an intelligent semantic web. In *Web Intelligence*, pages 37–58. Springer, 2003.

[25] P. A. Buhler and J. M. Vidal. Towards adaptive workflow enactment using multiagent systems. *Information technology and management*, 6(1):61–87, 2005.

[26] L. Buoncompagni, A. Capitanelli, and F. Mastrogiovanni. A ros multi-ontology references services: Owl reasoners and application prototyping issues. *arXiv preprint arXiv:1706.10151*, 2017.

[27] J.-P. Calbimonte, D. Calvaresi, and M. Schumacher. Multi-agent interactions on the web through linked data notifications. In *Multi-Agent Systems and Agreement Technologies*, pages 44–53. Springer, 2017.

[28] P. Casanovas. Semantic web regulatory models: Why ethics matter. *Philosophy & technology*, 28(1):33–55, 2015.

[29] C. Castelfranchi. Guarantees for autonomy in cognitive agent architecture. In *International Workshop on Agent Theories, Architectures, and Languages*, pages 56–70. Springer, 1994.

[30] M. Challenger, B. T. Tezel, O. F. Alaca, B. Tekinerdogan, and G. Kardas. Development of semantic web-enabled bdi multi-agent systems using sea_ml: An electronic bartering case study. *Applied Sciences*, 8(5):688, 2018.

[31] H. Chen, T. Finin, A. Joshi, L. Kagal, F. Perich, and D. Chakraborty. Intelligent agents meet the semantic web in smart spaces. *IEEE Internet computing*, 8(6):69–79, 2004.

[32] D. K. Chiu and H.-f. Leung. Towards ubiquitous tourist service coordination and integration: a multi-agent and semantic web approach. In *Proceedings of the 7th international conference on Electronic commerce*, pages 574–581, 2005.

[33] N. Damianou, N. Dulay, E. C. Lupu, and M. Sloman. Ponder: A language for specifying security and management policies for distributed systems. Technical report, Imperial College, Department of Computing, 2000.

[34] J. R. V. Dantas, H. A. Lira, B. de Azevedo Muniz, T. M. Nunes, and P. P. M. Farias. Semantic web services discovery adopting serin. In *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, pages 387–394. IEEE, 2015.

[35] A. Davis, O. Dieste, A. Hickey, N. Juristo, and A. M. Moreno. Effectiveness of requirements elicitation techniques: Empirical results derived from a systematic review. In *14th IEEE International Requirements Engineering Conference (RE'06)*, pages 179–188. IEEE, 2006.

[36] F. Demarchi, E. R. Santos, and R. A. Silveira. Integration between agents and remote ontologies for the use of content on the semantic web. In *ICAART (1)*, pages 125–132, 2018.

[37] S. Dietzold and S. Auer. Access control on rdf triple stores from a semantic wiki perspective. In *Proceedings of the ESWC'06 Workshop on Scripting for the Semantic Web*, 2006.

[38] V. Dignum. Ethics in artificial intelligence: introduction to the special issue, 2018.

[39] L. Ding, L. Zhou, and T. W. Finin. Trust based knowledge outsourcing for semantic web agents. In *Web Intelligence*, pages 379–387, 2003.

[40] L. Ding, P. Kolari, T. Finin, A. Joshi, Y. Peng, and Y. Yesha. On homeland security and the semantic web: A provenance and trust aware inference framework. In *AAAI Spring Symposium: AI Technologies for Homeland Security*, 2005.

[41] J. S. Dong, Y. Feng, Y.-F. Li, C. K.-Y. Tan, B. Wadhwa, and H. H. Wang. Bowl: augmenting the semantic web with beliefs. *Innovations in Systems and Software Engineering*, 11 (3), 2015.

[42] C. Dowling. Intelligent agents: some ethical issues and dilemmas. *Proc. AIC 2000*, pages 28–32, 2000.

[43] V. Ermolayev, N. Keberle, S. Plaksin, O. Kononenko, and V. Terziyan. Towards a framework for agent-enabled semantic web service composition. *International Journal of Web Services Research (IJWSR)*, 1(3):63–87, 2004.

[44] J. D. Fernández, S. Kirrane, A. Polleres, and S. Steyskal. Self-enforcing access control for encrypted rdf. In *European Semantic Web Conference*, pages 607–622. Springer, 2017.

[45] G. Flouris, I. Fundulaki, M. Michou, and G. Antoniou. Controlling access to rdf graphs. In *Proceedings of the Third Future Internet Conference on Future Internet*, pages 107–117. Springer-Verlag, 2010.

[46] N. Fornara and M. Colombetti. Using semantic web technologies and production rules for reasoning on obligations, permissions, and prohibitions. *AI Communications*, 32(4):319–334, 2019.

[47] N. Fornara, A. Chiappa, and M. Colombetti. Using semantic web technologies and production rules for reasoning on obligations and permissions. In *International Conference on Agreement Technologies*, pages 49–63. Springer, 2018.

[48] S. Franklin and A. Graesser. Is it an agent, or just a program?: A taxonomy for autonomous agents. In *International workshop on agent theories, architectures, and languages*, pages 21–35. Springer, 1996.

[49] A. Gabillon and L. Letouzey. A view based access control model for sparql. In *Network and System Security (NSS), 2010 4th International Conference on*, pages 105–112, Sept 2010.

[50] F. Gandon. The web we mix: Benevolent ais for a resilient web. In *Proceedings of the 10th ACM Conference on Web Science*, pages 115–116, 2019.

[51] F. L. Gandon and N. M. Sadeh. Semantic web technologies to reconcile privacy and context awareness. *Journal of Web Semantics*, 1(3):241–260, 2004.

[52] F. García-Sánchez, J. T. Fernández-Breis, R. Valencia-García, J. M. Gómez, and R. Martínez-Béjar. Combining semantic web technologies with multi-agent systems for integrated access to biological resources. *Journal of Biomedical Informatics*, 41(5): 848–859, 2008.

[53] F. García-Sánchez, R. Valencia-García, R. Martínez-Béjar, and J. T. Fernández-Breis. An ontology, intelligent agent-based framework for the provision of semantic web services. *Expert Systems with Applications*, 36(2):3167–3187, 2009.

[54] M. R. Genesereth and N. J. Nilsson. *Logical foundations of artificial intelligence*. Morgan Kaufmann, 2012.

[55] S. Gerbracht. Possibilities to Encrypt an RDF-Graph. In *Proc. of Information and Communication Technologies: From Theory to Applications*, pages 1–6, 2008.

[56] S. Ghanadbashi and F. Golpayegani. Using ontology to guide reinforcement learning agents in unseen situations. *Applied Intelligence*, pages 1–17, 2021.

[57] N. Gibbins, S. Harris, and N. Shadbolt. Agent-based semantic web services. In *Proceedings of the 12th international conference on World Wide Web*, pages 710–717, 2003.

[58] M. Giereth. On Partial Encryption of RDF-Graphs. In *Proc. of International Semantic Web Conference*, volume 3729, pages 308–322, 2005.

[59] R. Girardi and A. Leite. A survey on software agent architectures. *IEEE Intell. Informatics Bull.*, 14(1):8–20, 2013.

[60] A. Gladun, J. Rogushina, F. Garcı, R. Martínez-Béjar, J. T. Fernández-Breis, et al. An application of intelligent techniques and semantic web technologies in e-learning environments. *Expert Systems with Applications*, 36(2):1922–1931, 2009.

[61] J. A. Goguen and C. Linde. Techniques for requirements elicitation. In *[1993] Proceedings of the IEEE International Symposium on Requirements Engineering*, pages 152–164. IEEE, 1993.

[62] A. S. Gomes and J. J. Alferes. A procedure for an event-condition-transaction language. In *International Conference on Web Reasoning and Rule Systems*, pages 113–129. Springer, 2015.

[63] T. F. Gordon, G. Governatori, and A. Rotolo. Rules and norms: Requirements for rule interchange languages in the legal domain. In *International Workshop on Rules and Rule Markup Languages for the Semantic Web*, pages 282–296. Springer, 2009.

[64] Y. Guo, Z. Pan, and J. Heflin. Lubm: A benchmark for owl knowledge base systems. *Journal of Web Semantics*, 3(2-3): 158–182, 2005.

[65] M. S. Hajji and A. Al Maghrabi. Semantic web services methodology and tool extensions. *International Journal of Applied Engineering Research*, 12(2):256–262, 2017.

[66] A. Harth and T. Käfer. Specifying and executing user agent behaviour with condition-action rules. In *Proceedings of the 1st Workshop on Decentralizing the Semantic Web co-located with the 16th International Semantic Web Conference*, 2017.

[67] B. Heitmann, F. Hermsen, and S. Decker. k-rdf-neighbourhood anonymity: Combining structural and attribute-based anonymisation for linked data. In *Proceedings of the 5th Workshop on Society, Privacy and the Semantic Web - Policy and Technology (PrivOn2017) (PrivOn)*, 2017.

[68] J. Hendler. Agents and the semantic web. *IEEE Intelligent systems*, 16(2):30–37, 2001.

[69] J. Hendler. Where are all the intelligent agents? *IEEE Annals of the History of Computing*, 22(03):2–3, 2007.

[70] B. Hermans. Intelligent software agents on the internet: Chapters 6-7. *First Monday*, 1997. URL `https://journals.uic.edu/ojs/index.php/fm/article/download/516/437`.

[71] High-Level Expert Group on Artificial Intelligence (AI HLEG). Ethics Guidelines for Trustworthy AI, 2019. URL `https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai`.

[72] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutierrez, J. E. L. Gayo, S. Kirrane, S. Neumaier, A. Polleres, R. Navigli, A.-C. N. Ngomo, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, and A. Zimmermann. Knowledge graphs, 2021.

[73] M. B. Hoy. Alexa, siri, cortana, and more: an introduction to voice assistants. *Medical reference services quarterly*, 37(1): 81–88, 2018.

[74] M. N. Huhns. Agents as web services. *IEEE Internet computing*, 6(4):93–95, 2002.

[75] A. Jain and C. Farkas. Secure resource description framework: An access control model. In *Proceedings of the Eleventh ACM Symposium on Access Control Models and Technologies*, pages 121–129. ACM, 2006.

[76] N. Jennings and M. Wooldridge. Software agents. *IEE review*, 42(1):17–20, 1996.

[77] B. Jochum, L. Nürnberg, N. Aßfalg, and T. Käfer. Data-driven workflows for specifying and executing agents in an environment of reasoning and restful systems. In *International Conference on Business Process Management*, pages 93–105. Springer, 2019.

[78] D. Jutla and L. Xu. Privacy agents and ontology for the semantic web. *AMCIS 2004 Proceedings*, page 210, 2004.

[79] D. N. Jutla, P. Bodorik, and Y. Zhang. Pecan: An architecture for users' privacy-aware electronic commerce contexts on the semantic web. *Information Systems*, 31(4-5):295–320, 2006.

[80] T. Käfer and A. Harth. Rule-based programming of user agents for linked data. In *LDOW@ WWW*, 2018.

[81] L. Kagal and T. Finin. A policy language for a pervasive computing environment. In *Proceedings POLICY 2003. IEEE 4th International Workshop on Policies for Distributed Systems and Networks*, pages 63–74. IEEE Comput. Soc, 2003.

[82] L. Kagal, T. Finin, M. Paolucci, N. Srinivasan, K. Sycara, and G. Denker. Authorization and privacy for semantic web services. *IEEE Intelligent Systems*, 19(4):50–56, 2004.

[83] L. Kagal et al. Rei: A policy language for the me-centric project. Technical report, HP Laboratories Palo Alto, 2002.

[84] A. Kasten, A. Scherp, F. Armknecht, and M. Krause. Towards search on encrypted graph data. In *Proc. of the International Conference on Society, Privacy and the Semantic Web-Policy and Technology*, pages 46–57, 2013.

[85] A. Kasten, A. Scherp, and P. Schauß. *A Framework for Iterative Signing of Graph Data on the Web*, pages 146–160. Springer International Publishing, 2014. doi: 10.1007/978-3-319-07443-6_11. URL `https://doi.org/10.1007/978-3-319-07443-6_11`.

[86] A. Khalili, A. H. Badrabadi, and F. Khoshalhan. A framework for distributed market place based on intelligent software agents and semantic web services. In *2008 IEEE Congress on Services Part II (services-2 2008)*, pages 141–148. IEEE, 2008.

[87] S. Kirrane and S. Decker. Intelligent agents: The vision revisited. In *Proceedings of the 2nd Workshop on Decentralizing the Semantic Web co-located with the 17th International Semantic Web Conference*, 2018.

[88] S. Kirrane, A. Abdelrahman, A. Mileo, and S. Decker. Secure manipulation of linked data. In *The Semantic Web - ISWC 2013*, volume 8218. Springer Berlin Heidelberg, 2013.

[89] S. Kirrane, M. Sabou, J. D. Fernández, F. Osborne, C. Robin, P. Buitelaar, E. Motta, and A. Polleres. A decade of semantic web research through the lenses of a mixed methods approach. *Semantic Web*, 11(6):979–1005, 2020.

[90] Z. Kootbally, T. R. Kramer, C. Schlenoff, and S. K. Gupta. Overview of an ontology-based approach for kit building applications. In *2017 ieee 11th international conference on semantic computing (icsc)*, pages 520–525. IEEE, 2017.

[91] K. Kravari, N. Bassiliades, and G. Governatori. A policy-based b2c e-contract management workflow methodology using semantic web agents. *Artificial Intelligence and Law*, 24(2): 93–131, 2016.

[92] K. Ksystra and P. Stefaneas. Formal analysis and verification support for reactive rule-based web agents. *International Journal of Web Information Systems*, 2016.

[93] C. Laufer and D. Schwabe. On modeling political systems to support the trust process. In *Proceedings of the 5th Workshop on Society, Privacy and the Semantic Web - Policy and Technology (PrivOn2017) (PrivOn)*, 2017.

[94] A. Leite and R. Girardi. A case-based reasoning architecture of a hybrid software agent. In *2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, volume 3, pages 79–86. IEEE, 2014.

[95] A. Leite, R. Girardi, and P. Novais. Using ontologies in hybrid software agent architectures. In *2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, volume 3, pages 155–158. IEEE, 2013.

[96] Z. Lin. From isomorphism-based security for graphs to semantics-preserving security for the resource description

framework (rdf). Master's thesis, University of Waterloo, 2016.

[97] K. Lister, L. Sterling, et al. Reconciling ontological differences for intelligent agents. *Meaning Negotiation, AAAI Press (Menlo Park, America, 2002)*, 2002.

[98] M. Luck, P. McBurney, O. Shehory, and S. Willmott. Agent technology: computing as interaction (a roadmap for agent based computing). Technical report, University of Southampton, 2005.

[99] P. Maes. Artificial life meets entertainment: lifelike autonomous agents. *Communications of the ACM*, 38(11):108–114, 1995.

[100] S. A. McIlraith, T. C. Son, and H. Zeng. Semantic web services. *IEEE intelligent systems*, 16(2):46–53, 2001.

[101] N. Merkle and P. Philipp. Cooperative web agents by combining semantic technologies with reinforcement learning. In *Proceedings of the 10th International Conference on Knowledge Capture*, pages 205–212, 2019.

[102] Z. Ming, G. Wang, Y. Yan, J. Dal Santo, J. K. Allen, and F. Mistree. An ontology for reusable and executable decision templates. *Journal of Computing and Information Science in Engineering*, 17(3), 2017.

[103] R. Mizoguchi, M. Ikeda, K. Seta, and J. Vanwelkenhuysen. Ontology for modeling the world from problem solving perspectives. In *Proc. of IJCAI-95 Workshop on Basic Ontological Issues in Knowledge Sharing*, pages 1–12, 1995.

[104] O. Morgenstern and J. Von Neumann. *Theory of games and economic behavior.* Princeton university press, 1953.

[105] E. Motta, J. Domingue, L. Cabral, and M. Gaspari. Irs–ii: A framework and infrastructure for semantic web services. In *International Semantic Web Conference*, pages 306–318. Springer, 2003.

[106] G. Muller. A reference architecture primer. *Eindhoven Univ. of Techn., Eindhoven, White paper*, 2008.

[107] J. P. Müller. Architectures and applications of intelligent agents: A survey. *Knowledge Engineering Review*, 13(4):353–380, 1998.

[108] A. Newell and H. A. Simon. Computer science as empirical inquiry: Symbols and search. In *ACM Turing award lectures*, page 1975, 2007.

[109] A.-C. N. Ngomo and M. Röder. Hobbit: holistic benchmarking for big linked data. *ERCIM News*, 2016(105), 2016.

[110] E. Oren, R. Delbru, M. Catasta, R. Cyganiak, H. Stenzhorn, and G. Tummarello. Sindice. com: a document-oriented lookup index for open linked data. *International Journal of Metadata, Semantics and Ontologies*, 3(1):37–52, 2008.

[111] A. Outtagarts. Mobile agent-based applications: A survey. *International Journal of Computer Science and Network Security*, 9(11):331–339, 2009.

[112] F.-P. Pai, I.-C. Hsu, and Y.-C. Chung. Semantic web technology for agent interoperability: a proposed infrastructure. *Applied Intelligence*, 44(1):1–16, 2016.

[113] C.-V. Pal, F. Leon, M. Paprzycki, and M. Ganzha. A review of platforms for the development of agent systems. *arXiv preprint arXiv:2007.08961*, 2020.

[114] M. Palmirani, G. Governatori, A. Rotolo, S. Tabet, H. Boley, and A. Paschke. Legalruleml: Xml-based rules and norms. In *International Workshop on Rules and Rule Markup Languages for the Semantic Web*. Springer, 2011.

[115] M. Palmirani, M. Martoni, A. Rossi, C. Bartolini, and L. Robaldo. Pronto: Privacy ontology for legal reasoning. In *International Conference on Electronic Government and the Information Systems Perspective*. Springer, 2018.

[116] L. Panait and S. Luke. Cooperative multi-agent learning: The state of the art. *Autonomous agents and multi-agent systems*, 11(3):387–434, 2005.

[117] M. Paolucci and K. Sycara. Autonomous semantic web services. *IEEE Internet computing*, 7(5):34–41, 2003.

[118] G. Papamarkos, A. Poulovassilis, and P. T. Wood. Event-condition-action rule languages for the semantic web. In *Proceedings of the First International Conference on Semantic Web and Databases*, pages 294–312. Citeseer, 2003.

[119] G. Papamarkos, A. Poulovassilis, and P. T. Wood. Rdftl: An event-condition-action language for rdf. In *Proc. of the 3rd International Workshop on Web Dynamics*, 2004.

[120] T. R. Payne. Web services from an agent perspective. *IEEE Intelligent Systems*, 23(2):12–14, 2008.

[121] H. Pham and D. Stacey. Practical goal-based reasoning in ontology-driven applications. In *KEOD*, pages 99–109, 2011.

[122] S. Poslad. Specifying protocols for multi-agent systems interaction. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 2(4):15–es, 2007.

[123] A. Poulovassilis, G. Papamarkos, and P. T. Wood. Event-condition-action rule languages for the semantic web. In *International conference on extending database technology*, pages 855–864. Springer, 2006.

[124] E. Puerto, J. Aguilar, et al. Automatic learning of ontologies for the semantic web: Experiment lexical learning. *Respuestas*, 17(2):5–12, 2012.

[125] F. Radulovic, R. García Castro, and A. Gómez-Pérez. Towards the anonymisation of rdf data, 2015. URL `https://doi.org/10.18293/SEKE2015-167`.

[126] D. Rajpathak and E. Motta. An ontological formalization of the planning task. In *International Conference on Formal Ontology in Information Systems (FOIS 2004)*, pages 305–316, 2004.

[127] A. S. Rao, M. P. Georgeff, et al. Bdi agents: From theory to practice. In *Icmas*, volume 95, pages 312–319, 1995.

[128] P. Reddivari, T. Finin, and A. Joshi. Policy-based access control for an rdf store. In *Proceedings of the Policy Management for the Web workshop*, pages 78–83, 2005.

[129] J. S. Rosenschein and M. R. Genesereth. Deals among rational agents. In *Readings in Distributed Artificial Intelligence*, pages 227–234. Elsevier, 1988.

[130] S. Russel and P. Norvig. *Artificial intelligence: a modern approach.* Pearson Education Limited, 2013.

[131] O. Sacco and A. Passant. A privacy preference ontology (ppo) for linked data. In *Linked Data on the Web.* CEUR-WS, 2011.

[132] D. Schraudner and V. Charpenay. An http/rdf-based agent infrastructure for manufacturing using stigmergy. In *European Semantic Web Conference*, pages 197–202. Springer, 2020.

[133] F. Scioscia, M. Ruta, G. Loseto, F. Gramegna, S. Ieva, A. Pinto, and E. Di Sciascio. A mobile matchmaker for the ubiquitous semantic web. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 10(4):77–100, 2014.

[134] J. Sculley and J. A. Byrne. *Odyssey.* Harper & Row Publishers, Inc., 1987.

[135] N. B. Seghir and O. Kazar. A new framework for web service discovery in distributed environments. In *2017 First International Conference on Embedded & Distributed Systems (EDiS)*, pages 1–6. IEEE, 2017.

[136] N. B. Seghir, O. Kazar, and K. Rezeg. A decentralized framework for semantic web services discovery using mobile agent. *International Journal of Information Technology and Web Engineering (IJITWE)*, 10(4):20–43, 2015.

[137] N. B. Seghir, O. Kazar, K. Rezeg, and S. Bourekkache. A semantic web services discovery approach based on a mobile agent using metadata. *International Journal of Intelligent Computing and Cybernetics*, 2017.

[138] M. O. Shafiq, Y. Ding, and D. Fensel. Bridging multi agent systems and web services: towards interoperability between software agents and semantic web services. In *2006 10th IEEE International Enterprise Distributed Object Computing Conference (EDOC'06)*, pages 85–96. IEEE, 2006.

[139] M. Sheshagiri, N. Sadeh, and F. Gandon. Using semantic web services for context-aware mobile applications. In *MobiSys 2004 Workshop on Context Awareness*. Citeseer, 2004.

[140] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu. Edge computing: Vision and challenges. *IEEE internet of things journal*, 3(5):637–646, 2016.

[141] Y. Shoham, R. Powers, and T. Grenager. Multi-agent reinforcement learning: a critical survey. Technical report, Technical report, Stanford University, 2003.

[142] R. R. C. Silva, B. C. Leal, F. T. Brito, V. M. P. Vidal, and J. C. Machado. A differentially private approach for querying rdf data of social networks. In *Proceedings of the 21st International Database Engineering & Applications Symposium*, pages 74–81. ACM, 2017. ISBN 978-1-4503-5220-8.

[143] I. Sommerville, P. Sawyer, and S. Viller. Viewpoints for requirements elicitation: a practical approach. In *Proceedings of IEEE International Symposium on Requirements Engineering: RE'98*, pages 74–81. IEEE, 1998.

[144] K. Sycara, M. Paolucci, J. Soudry, and N. Srinivasan. Dynamic discovery and coordination of agent-based semantic web services. *IEEE Internet computing*, 8(3):66–73, 2004.

[145] V. Tamma and T. R. Payne. Is a semantic web agent a knowledge-savvy agent? *IEEE Intelligent Systems*, 23(4):82–85, 2008.

[146] V. Tamma, I. Blacoe, B. L. Smith, and M. Wooldridge. Serse: searching for semantic web content. In *ECAI*, volume 16, page 63, 2004.

[147] V. Terziyan. Smartresource–proactive self-maintained resources in semantic web: Lessons learned. *International Journal of Smart Home*, 2(2):33–57, 2008.

[148] G. Tonti, J. M. Bradshaw, R. Jeffers, R. Montanari, N. Suri, and A. Uszok. Semantic web languages for policy representation and reasoning: A comparison of kaos, rei, and ponder. In *International Semantic Web Conference*, pages 419–437. Springer, 2003.

[149] R. J. Torraco. Writing integrative literature reviews: Guidelines and examples. *Human resource development review*, 4 (3):356–367, 2005.

[150] R. Usbeck, M. Röder, A.-C. Ngonga Ngomo, C. Baron, A. Both, M. Brümmer, D. Ceccarelli, M. Cornolti, D. Cherix, B. Eickmann, et al. Gerbil: general entity annotator benchmarking framework. In *Proceedings of the 24th international conference on World Wide Web*, pages 1133–1143, 2015.

[151] A. Uszok, J. Bradshaw, R. Jeffers, N. Suri, P. Hayes, M. Breedy, L. Bunch, M. Johnson, S. Kulkarni, and J. Lott. Kaos policy and domain services: Toward a description-logic approach to policy representation, deconfliction, and enforcement. In *Proceedings of the 4th IEEE International Workshop on Policies for Distributed Systems and Networks*, pages 93–. IEEE Computer Society, 2003.

[152] A. Uszok, J. M. Bradshaw, M. Johnson, R. Jeffers, A. Tate, J. Dalton, and S. Aitken. Kaos policy management for semantic web services. *IEEE Intelligent Systems*, 19(4):32–41, 2004.

[153] B. Van Der Vecht, A. P. Meyer, M. Neef, F. Dignum, and J.-J. C. Meyer. Influence-based autonomy levels in agent decision-making. In *International Workshop on Coordination, Organizations, Institutions, and Norms in Agent Systems*, pages 322–337. Springer, 2006.

[154] M. B. Van Riemsdijk, L. Dennis, M. Fisher, and K. V. Hindriks. A semantic framework for socially adaptive agents: Towards strong norm compliance. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 423–432. Citeseer, 2015.

[155] K. Venkatachalam, N. Karthikeyan, and S. Kannimuthu. Comprehensive survey on semantic web service discovery and composition. *Adv Nat Appl Sci AENSI Publ*, 10(5):32–40, 2016.

[156] R. Verborgh, E. Mannnens, and R. Van de Walle. Bottom-up web apis with self-descriptive responses. In *Proceedings of the First Karlsruhe Service Summit Workshop-Advances in Service Research*, page 143, 2015.

[157] H. H. Wang, N. Gibbins, T. Payne, A. Patelli, and Y. Wang. A survey of semantic web services formalisms. *Concurrency and Computation: Practice and Experience*, 27(15):4053–4072, 2015.

[158] J. White. Telescript technology: the foundation for the electronic marketplace. Technical report, General Magic, 1994. URL `http://www.datarover.com/Telescript/Whitepapers/wp1/whitepaper-1.html`.

[159] R. Whittemore and K. Knafl. The integrative review: updated methodology. *Journal of advanced nursing*, 52(5):546–553, 2005.

[160] W. Y. Wong. *Learning lightweight ontologies from text across different domains using the web as background knowledge.* University of Western Australia, 2009.

[161] M. Wooldridge. Intelligent agents: The key concepts. In *ECCAI Advanced Course on Artificial Intelligence*, pages 3–43. Springer, 2001.

[162] M. Wooldridge and N. R. Jennings. Agent theories, architectures, and languages: a survey. In *International Workshop on Agent Theories, Architectures, and Languages*, pages 1–39. Springer, 1994.

[163] M. J. Wooldridge and N. R. Jennings. Intelligent agents: Theory and practice. *The knowledge engineering review*, 10(2): 115–152, 1995.

[164] J. Young, V. Basile, L. Kunze, E. Cabrio, and N. Hawes. Towards lifelong object learning by integrating situated robot perception and semantic web mining. In *22nd European Conference on Artificial Intelligence, ECAI 2016*, volume 285, pages 1458–1466. IOS Press, 2016.

[165] J. J. S. Zapater, D. M. L. Escrivá, F. R. S. García, and J. J. M. Durá. Semantic web service discovery system for road traffic information services. *Expert Systems with Applications*, 42(8): 3833–3842, 2015.

[166] D. Zowghi and C. Coulin. Requirements elicitation: A survey of techniques, approaches, and tools. In *Engineering and managing software requirements*, pages 19–46. Springer, 2005.

3. Governance of autonomous agents on the web: Challenges and opportunities

Bibliographic Information

Kampik, T., Mansour, A., Boissier, O., **Kirrane, S.**, Padget, J., Payne, T.R., Singh, M.P., Tamma, V. and Zimmermann, A., 2022. Governance of Autonomous Agents on the Web: Challenges and Opportunities. ACM Transactions on Internet Technology, 22(4), pp.1-31.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Formal analysis, Investigation, Data Curation, Writing - Original Draft, and Writing - Review & Editing.

Copyright Notice

Governance of Autonomous Agents on the Web: Challenges and Opportunities

TIMOTHEUS KAMPIK, Umeå University, Sweden
ADNANE MANSOUR, Mines Saint-Étienne, France
OLIVIER BOISSIER, Mines Saint-Étienne, France
SABRINA KIRRANE, Wirtschaftsuniversität Wien, Austria
JULIAN PADGET, University of Bath, UK
TERRY R. PAYNE, University of Liverpool, UK
MUNINDAR P. SINGH, North Carolina State University, USA
VALENTINA TAMMA, University of Liverpool, UK
ANTOINE ZIMMERMANN, Mines Saint-Étienne, France

The study of autonomous agents has a long tradition in the Multiagent Systems and the Semantic Web communities, with applications ranging from automating business processes to personal assistants. More recently, the Web of Things (WoT), which is an extension of the Internet of Things (IoT) with metadata expressed in Web standards, and its community provide further motivation for pushing the autonomous agents research agenda forward. Although representing and reasoning about norms, policies and preferences is crucial to ensuring that autonomous agents act in a manner that satisfies stakeholder requirements, normative concepts, policies and preferences have yet to be considered as first-class abstractions in Web-based multiagent systems. Towards this end, this paper motivates the need for alignment and joint research across the Multiagent Systems, Semantic Web, and WoT communities, introduces a conceptual framework for governance of autonomous agents on the Web, and identifies several research challenges and opportunities.

CCS Concepts: • **Computing methodologies → Multiagent Systems**; **Intelligent agents**; • **Information systems → web services**.

Additional Key Words and Phrases: autonomous agents, norms, policies, preferences, governance

ACM Reference Format:
Timotheus Kampik, Adnane Mansour, Olivier Boissier, Sabrina Kirrane, Julian Padget, Terry R. Payne, Munindar P. Singh, Valentina Tamma, and Antoine Zimmermann. 2021. Governance of Autonomous Agents on the Web: Challenges and Opportunities. *ACM Trans. Internet Technol.* 20, 1, Article 123 (December 2021), 30 pages. https://doi.org/10.1145/1122445.1122456

Authors' addresses: Timotheus Kampik, Umeå University, Umeå, Sweden, tkampik@cs.umu.se; Adnane Mansour, Mines Saint-Étienne, France, adnane.mansour@emse.fr; Olivier Boissier, Mines Saint-Étienne, France, olivier.boissier@emse.fr; Sabrina Kirrane, Wirtschaftsuniversität Wien, Vienna, Austria, sabrina.kirrane@wu.ac.at; Julian Padget, University of Bath, Bath, UK, j.a.padget@bath.ac.uk; Terry R. Payne, University of Liverpool, Liverpool, UK, t.r.payne@liverpool.ac.uk; Munindar P. Singh, North Carolina State University, Raleigh, North Carolina, USA, 27695, mpsingh@ncsu.edu; Valentina Tamma, University of Liverpool, Liverpool, UK, v.tamma@liverpool.ac.uk; Antoine Zimmermann, Mines Saint-Étienne, France, antoine.zimmermann@emse.fr.

1533-5399/2021/12-ART123 $15.00
https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Over the last two decades, the Web has evolved extensively in response to a variety of different requirements. From originally providing a distributed information dissemination architecture, it has encompassed support for publication, discovery, consumption and aggregation of information, knowledge, and services, thereby interconnecting the digital, social and physical worlds. The Web's ubiquity, as well as the simplicity of its underlying communication protocols has resulted in it becoming the de facto standard for communication between services, and more recently, connected things. With the rise of the Internet of Things (IoT), the combination of the Web of Things (WoT) (as an extension of the IoT with metadata expressed in Web standards), traditional web services, and a knowledge dissemination infrastructure that is both machine navigable and machine understandable, has facilitated a new generation of applications that utilise the Web.

Berners-Lee et al. [14] outline how autonomous agents could comprehend and exploit this machine-readable knowledge to achieve a variety of tasks. Thus, the notion of *autonomy* provides a framework whereby individual agents (e.g., those representing or controlling services, things, or applications) may plan, collaborate, and cooperate to achieve complex but disparate goals. Such multiagent systems avoid centralised control, which is the bane of business process management [129]. By seeking mutually beneficial interactions, agents of heterogeneous construction (potentially originating from different developers) can evolve a mutually supportive economy across the Web, performing a multitude of tasks for Web users. However, to achieve this notion of collaborative agents that use the Web infrastructure, it is crucial to consider a *governance* perspective, which defines how agents should act in a given situation (also considering the consequences of their potential actions) and defines how frameworks that govern communities of agents should be designed, interoperate, and evolve. This perspective is of particular importance for the Web, where usage may cross social contexts and jurisdictions, and where no centralised control over the different agents is possible. Indeed, the need for intelligent system governance is, at the time of writing, a focus point of legislative and regulatory efforts; e.g., by the European Commission [74].

Therefore, what is needed is a new governance framework supported by a review of the related literature on the use of norms, policies, and preferences for autonomous normative agents, as well as contextualising these with respect to the notion of the Web (of Things). Towards this end, this paper makes three main contributions. Firstly, it motivates the need for norms, policies and preferences for autonomous agents on the Web by means of a simple motivating scenario. Secondly, it proposes a new governance conceptual framework and gives an overview of the state of the art on norms, policies, and preferences for autonomous normative agents (restricted to those efforts that provide the theoretical background to our proposed framework). Finally, it identifies several challenges and opportunities, for the MAS, Semantic Web and WoT communities, underlining the need for better integration and joint research across the different communities. Each challenge is motivated by a concise review of the state of the art, followed by several opportunities for future investigation. For each of the identified challenges, we discuss its maturity in terms of research and technological approaches, ranging from nascent solutions to those that have received community adoption, whereas the opportunities take the form of open research questions that need to be explored.

The remainder of the paper is structured as follows: we start by outlining our motivating use case scenario (Section 2) and presenting the relevant background (Section 3). A conceptual framework is then proposed (Section 4), accompanied by an instantiation based on our use case scenario (Section 5), after which we subsequently identify several challenges and opportunities (Section 6). We then conclude by proposing a research roadmap for the governance of autonomous agents on the web (Section 7).

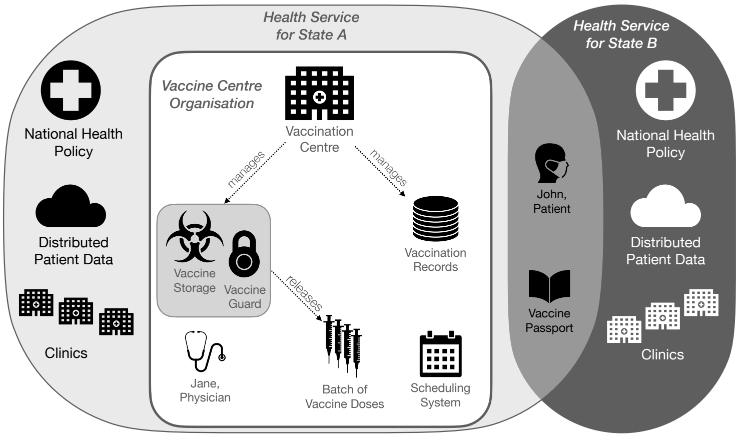


Fig. 1. Organisations, agents, things, and services in the scenario.

2 MOTIVATING SCENARIO

The synergy between autonomous agents that leverage Web technologies, and the resources (i.e., things, services, information) that they can exploit to achieve their goals can be illustrated through a motivating scenario that demonstrates the need for governance through the use of norms, policies, and preferences. Consider a scenario whereby there is a vaccination roll-out (for example, for the COVID-19 pandemic), where patients who request vaccinations may have differing personal circumstances. For example, John, the patient in Figure 1, may ask to be vaccinated early as he is the care giver for a vulnerable member of his family. As the demand for vaccines outstrips supply, policies exist that determine vaccination eligibility. Furthermore, as vaccines are available from different manufacturers (e.g., AstraZeneca and Pfizer-BioNTech) and can be of different types (e.g., mRNA or inactivated vaccines), these vaccination policies may vary depending on the recipient's personal health record and/or their preferences, as well as vaccine availability.

Patients may be registered to different clinics or health centres that follow local or national policies or guidance on health care. In this case, John is registered at a clinic in his country (labelled State B in Figure 1), but has a preference for vaccination near his current residential address in State A. Each country or state can be seen as having an *organisation* of different health centres (clinics, hospitals, and vaccination centres), following their own national health policy that prescribe a specific specification/format for patient medical records, which may be held under disparate data models and access policies. Patient medical records are available (subject to appropriate authorisation) via web services using secure protocols across the web infrastructure [135], and are encoded using established medical ontologies and vocabularies to facilitate record exchange within and across different national health organisations.

Vaccination centres store batches of vaccines within one or more temperature-controlled vaccine storage systems, where each storage system is responsible for both inventory management and the dispensation of the different COVID-19 vaccine batches from a specialised cold store via a robotic arm. The release and retrieval of vaccine batches is guarded by policies that must be satisfied to ensure appropriate use by authorised personnel (i.e., the vaccine guard in Figure 1). Once a batch of vaccines has been released, the vaccine doses should be used within a given time-frame to avoid spoilage and wastage, as they have a short shelf-life once thawed. Furthermore, a scheduling system determines which patients can be vaccinated in a given time-slot, based on vaccination demand and patient requirement (determined by the current vaccination policy that may change frequently). This scheduling system should ensure that no vaccines are wasted, whilst ensuring that the policies determining which patients can receive which vaccines is adhered to. Thus, the vaccination centre could be considered as an organisation that coordinates and exploits a variety of disparate information technology (IT) systems integrated through a Web infrastructure, including data management, scheduling, patient-facing services, and IoT-based physical assets such as the robot arm and the automated vaccine stores. Typically, however, the task of orchestrating and using these different systems requires costly and time-consuming human intervention. Finally, once a vaccine has been administered, the patient's medical records should be updated, and the patient should be able to prove their vaccination status if required (e.g., using a vaccine passport [55]). The vaccine records should ideally be resilient to forgery whilst being privacy preserving and easy to administer [55]; thus they may utilise a passport mechanism that itself exploits web-based resources such as *verifiable credentials*[1], decentralised data platforms [142], blockchains [135], etc.

This scenario raises challenges due to the decentralised and dynamic characteristics of the involved organisations, policies, services, and stakeholders. Patients can request vaccination based on their interpretation of eligibility, which should then be validated by the vaccination centre. The handling of requests may require the collection of patient data from multiple sources and the mapping to a shared data model. The vaccination eligibility policy can change frequently due to, for example, the emergence of a new *variant of concern*, that may accelerate the need for vaccinating a specific population cohort or demographic. Changes to vaccination administration guidance may prioritise the use of certain types of vaccine over others for specific sub-groups (e.g., prioritising Pfizer-BioNTech over AstraZeneca, where possible, for certain patients based on medical risk assessments, or prohibiting certain vaccines for users where safety data is not available). Thus, the verification of vaccination eligibility for patients may rely on the aggregation of multiple policies, and on resolving inconsistencies between them. A further challenge involves ensuring that the process for adhering to the national prioritisation criteria is fair and transparent.

Additional legal and ethical challenges arise when considering the complete socio-technical system, including electronic health record access [81] and supply chains [127]. Finally, vaccination scheduling needs to take into account patient availability (to avoid no-show cases and thus avoid vaccine wastage), as well as stock availability. Scheduling is therefore a collaborative process involving factors such as the vaccination centre capacity, vaccine availability, and patient availability. However, availability data may be distributed across multiple sources and, for privacy reasons, cannot be held centrally.

This scenario underlines the need for systematic and scalable approaches for the governance of the different IT systems and IoT-based physical assets, taking into account the need to operate under different governance institutions, as well as interact across organisational boundaries (e.g., between countries). Such interactions must comply with applicable norms and policies encountered at different stages of the vaccination roll-out. For example, the European Commission recently proposed a Digital Green Certificate, recognised by all EU member states, that facilitates the safe free movement of citizens within the EU during the COVID-19 pandemic.[2]

Given the intrinsic openness of the Web, coupled with the fact that autonomous agents can act on behalf of both patients and medical practitioners that need access to critical medical applications, the need for regulation, security, and privacy are of utmost importance. Additionally, there is a need to facilitate coordination between stakeholders and ensure that relevant regulatory requirements are adhered to throughout.

[1] https://www.w3.org/TR/vc-data-model/

[2] https://ec.europa.eu/info/live-work-travel-eu/coronavirus-response/safe-covid-19-vaccines-europeans/covid-19-digital-green-certificates_en

3 BACKGROUND

The vaccine administration scenario detailed in Section 2 relies on the availability of a uniform access layer that sits on top of several different systems (e.g., data management, services, and IoT platforms). The Web provides the necessary infrastructure to integrate and make accessible all of these systems, effectively becoming an application architecture for the proposed MAS [39], on top of which autonomous agents may interact and cooperate to achieve common goals. In the following subsections, we present the relevant background in multiagent systems, the Semantic Web, and the WoT, followed by a discussion of the related work in norms, policies and preferences, with a focus on the governance of autonomous agents, both within and spanning those communities.

3.1 Multiagent Systems

A multiagent system is composed of a (dynamic) set of *agents* interacting inside a shared, possibly distributed, *environment* which itself comprises a dynamic set of *artefacts*. Agents are goal-oriented autonomous entities, encapsulating a logical thread of control, that pursue their tasks by communicating with other agents and by perceiving and acting upon artefacts within the environment. In essence, a MAS addresses the challenges of how agents may coordinate their efforts and cooperate in light of their autonomy [145]. Artefacts model any kind of (non-autonomous) resource or tool that agents can use and possibly share to achieve their goals. An agent perceives the observable state of an artefact, reacts to events related to state changes, and performs actions that correspond to operations provided by the artefact's interface. The coordinated and organised activities taking place in the system result from the concurrent and complex tasks handled by *groups of agents* interacting with each other, or acting within an environment. Such activities may lead to recurrent patterns of cooperation captured by agent *organisations*. Changes in the state of the environment may also lead agents to react and possibly affect the state of the organisation.

Research into multiagent systems has led to a number of concrete programming models.[3] These models[4]are concerned with agent-oriented programming [20], interaction and protocol languages [123], environment infrastructures [146], and agent organisation model and management systems [58]. The results produced so far have clearly demonstrated the importance of these concepts and abstractions for the development of multiagent applications. Additionally, a variety of languages, tools, and platforms for agent-oriented programming (MAOP) have been developed and application success stories exists (e.g., [49]). This type of research is often referred to under the umbrella of *Engineering Multiagent Systems (EMAS)*. An overview and a comparative analysis of several prominent MAOPs can be found in [91]. One of the most prominent underlying architectures used by many agent-oriented programming systems is the *Belief-Desire-Intention (BDI)* architecture, which models: knowledge (i.e., *beliefs*) that the agent knows about, either through observation of the environment or interaction with other agents; goals (i.e., *desires*) that the agent would like to bring about; and goals and plans of action (i.e., *intentions*) that the agent is currently focused on.

From an agent development environment perspective, the Jade platform [12] provides a variety of behaviours (one-shot, cyclic, contract net) and is still available, although the last release dates back to 2017. Although Jade does not directly provide support for BDI-based agents, they can be added through extensions such as Jadex [22]. Jack [27] is an example of a closed source BDI architecture, whereas the practical Agent Programming Language (2APL) is another open source language that retains BDI semantics [47]. GOAL [75] offers a further BDI architecture which is actively maintained, whereas SPADE[5] is a recently introduced Python-based BDI platform. The

[3] Refer to the proceedings of the EMAS or PAAMS series for broad overviews.

[4] The models presented here reflect the relevant state of the art with respect to different MAS and are by no means exhaustive.

[5] https://spade-mas.readthedocs.io/en/latest/index.html

JaCaMo MAOP framework, based on the JaCaMo conceptual meta-model [17], offers first-class abstractions to program the agents working environment and their organisation, in addition to offering the Jason interpreter for the BDI-based *AgentSpeak* language [20].

Whilst MAOP is thriving within the academic community, industrial adoption of MAOP technologies is in its infancy, and standardisation efforts such as FIPA [63] (that superseded KQML) have received little attention in recent years [99].

3.2 Agents and the Semantic Web

Attempts to tightly integrate autonomous agents and Web technologies date back to the vision of the *Semantic Web* of the early 2000s. Berners-Lee et al. [14] originally envisioned "a web of data that can be processed directly and indirectly by machines", in which intelligent agents act on behalf of humans, by searching for and understanding relevant information published on the web or acquired via services. Such information could potentially be made available by multiple sources, using alternative ontologies, often with different provenance. Autonomous agents rely on communication languages and protocols to exchange data and coordinate their behaviour and thus collaborate. Early approaches based on speech acts [8], focused on message types or *performatives* (e.g., *request*, *inform*, and *promise*) based on a folk categorisation of the intended meaning of the communication. This evolved through the DARPA funded Knowledge Sharing Effort (KSE) resulting in a communication language, the *Knowledge Query Manipulation Language (KQML)*, defining the mechanism by which agents communicated; and an ontology language, the *Knowledge Interchange Format (KIF)*, describing the knowledge that the performative referred to [62]. Although agents could perform services on behalf of their peers, discovered through capability registries [51], service invocation occurred as a by-product of requesting information. This contrasts with the notion of *web services* and *things*, which use web-based communication protocols, whereby the invocation of services could be requested explicitly (in a similar manner to calling methods or functions within a programming language) by providing the relevant input parameters, as data or knowledge fragments.

The prominent view from a Semantic Web perspective is that multiagent systems operate on the Web through the provision of services, using HTTP as the de facto standard transport protocol. Additionally, the Semantic Web community have developed standards, protocols, vocabularies, ontologies, and knowledge representation formalisms to facilitate the integration of machine-processible data from diverse sources at scale, using the existing web infrastructure. As such, the two communities diverged due to different priorities, though there is increasing recognition [39] that the Web is a natural application architecture for MAS and can support different types of interactions between agents and resources.

From a knowledge representation perspective, standards such as RDFS [24] and OWL [69] facilitate the representation of complex knowledge about agents, services, things and their relationship in an explicit and processable way. An example is the Provenance ontology (PROV-O), a data model for workflows expressed using agents, their actions, and other assets.[6] Additionally, reasoning engines have been developed that are capable of reasoning over OWL ontologies, albeit often with some restrictions (cf., Pellet[7], HermiT[8], FACT++[9], Racer[10], and RDFox[11]). However, the use of ontologically grounded annotations for services within agent communication pre-dates the

[6] https://www.w3.org/TR/prov-o/

[7] https://www.w3.org/2001/sw/wiki/Pellet

[8] http://www.hermit-reasoner.com/

[9] http://owl.cs.manchester.ac.uk/tools/fact/

[10] http://www.ifis.uni-luebeck.de/ moeller/racer/

[11] https://www.oxfordsemantic.tech/

Semantic Web [57, 77], and in some cases the Web itself [76]. Semantic Web service research exploited both F-Logic [84] as used by WSMO [119], and DAML-S [6] (based on the DARPA Agent Markup Language) which evolved into OWL-S [98]. Other approaches to support service utilisation were developed using OWL, e.g., the OWL ontology for protocols, OWL-P [52], or using federated service discovery mechanisms such as the semantically annotated version of UDDI [109]. These frameworks and ontologies were key in facilitating the discovery and use of services by autonomous agents, and provided an alternative communication paradigm built on web-based infrastructure. In addition, from the knowledge perspective, bespoke protocols were developed to support the decentralised management and exchange of knowledge and information amongst networks of agents or peers [131].

Other efforts include the provision of infrastructures for supporting the cleaning and validation of the data published on Linked Open Data Platforms; e.g., LOD Laundromat [11][12] and OOPS [113].[13] Such techniques help detect errors in the data exchanged between agents and things. The SPARQL [72] query language facilitates federated querying over distributed data sources accessible via the web, whereas the Linked Data Platform [130] can be used to manipulate RDF data via HTTP operations. Approaches have also been proposed to enrich SPARQL with qualitative and quantitative preferences [70, 111] to select query results that satisfy user-defined criteria.

In recent years the Semantic Web community has broadened its focus beyond knowledge representation, reasoning, and querying to include knowledge extraction, discovery, search, and retrieval. However, many of the proposed tools and techniques have yet to be used extensively within MAS or by the MAS community. A recent survey [85] identified several open research challenges and opportunities in relation to the suitability of existing proposals for autonomous agent use cases, the combination of symbolic and sub-symbolic AI techniques for enhancing agent learning, and the development of tools and techniques for validation and verification.

3.3 Agents and the Web of Things

The *Web of Things (WoT)* [90] refers to the Internet of Things (IoT) with an application of Web standards and technologies for improving interoperability of IoT devices and infrastructure. Things are resources that can be acted upon or queried via APIs (e.g., WoT scripting API [88]); *autonomous goal-driven agents*[14] thus can make use of a WoT environment via WoT technologies and become part of the WoT ecosystem. Indeed, bringing agents to the Web requires more than simply exploiting Web protocols (such as HTTP [61]) and data formats (e.g., XML [23], RDF [44]). The communication infrastructure used by agents should comply with an architectural style based on well-defined principles, such as *Representational State Transfer (REST)* [60] as instantiated in the Architecture of the World Wide Web [79].[15] Furthermore, for things to be used without human intervention, they must be formally described. To this end, the W3C published the *Thing Description* [80] standard, which specifies how a JSON-LD representation of thing affordances (i.e., properties or actions) via Web APIs can be provided. In addition, the *WoT Discovery* [35] standard provides a mechanism for the automatic discovery of thing descriptions (thus obviating the need to hard-code the location of such descriptions beforehand). These standards support improved heterogeneity by decoupling agents from thing implementation details.

The WoT activity highlights the importance of metadata with clear semantics, and made their standards, especially thing descriptions, compatible with RDF and Semantic Web technologies. In

[12] https://github.com/LOD-Laundromat/LOD-Laundromat

[13] http://oops.linkeddata.es

[14] Here, we refer to agents in the sense of multiagent systems as discussed in Section 3.1.

[15] Note that the specification of the Web architecture defines the concept of *Web agents* as "a person or a piece of software acting on the information space on behalf of a person, entity, or process".

fact, even before a standardisation effort for the WoT started, multiple initiatives suggested the use of the Semantic Web to improve IoT systems [118]. More precisely, in REST style hypermedia systems such as the WoT, things and agents are resources that interact by producing and consuming hypermedia about their state and the artefacts surrounding them [38]. All resources are identified through IRIs[16] to support global referencing, irrespective of contextual information. Therefore, resources can be represented through semantic descriptions that are expressed in a uniform data exchange format such as RDF using terms from some standardised and interlinked vocabulary expressed in OWL [69]. This standardised knowledge model hides the specifics of the implementation and facilitates interconnected resources that can be queried by exposing SPARQL endpoints. Of particular interest to WoT environments are the vocabularies that describe sensors and actuators (SOSA/SSN [71]), provenance (PROV [92]), and temporal entities (OWL-Time [40]).

The WoT provides a natural substrate for multiagent systems based on the vision that systems of interconnected things should be open and easily reconfigurable, and therefore such systems should comprise autonomous and collaborative components. This notion was supported by Singh and Chopra [125] who argue that IoT systems need the kind of decentralised intelligence that MAS provides. Likewise, Ciortea et al. [39] recommend integrating the Web and MAS to leverage the proven benefits of hypermedia systems for MAS. Importantly, these papers emphasise governance as a major challenge.

The technologies that emerge from the WoT community are often industry-oriented and paralleled by standardisation efforts. A recent example is the abstract *WoT architecture* design document [90], supported by the *Thing Description* [80] and the *WoT Scripting API* [88] specifications, for which a reference implementation is provided.[17] Although these technologies are more mature than MAOP technologies from an engineering perspective, and have a clear path to industry adoption, they lack the rich abstractions related to agents and autonomy that MAOP technologies provide. For example, the notion of a *servient*, as introduced in the WoT architecture design document can be considered an evolutionary step from a stricter server-client separation; a notion that is considered simplistic within the MAS community. Recent approaches have sought to form a bridge between the MAOP and WoT technology ecosystems [36, 37]; however, this line of research is young and the corresponding technologies are nascent.

3.4 Norms, Policies and Preferences

Norms, policies, and preferences can help govern autonomous agent behaviour. The term *norm* has several meanings in natural language and is used widely in economics and social science. In MAS, the term "norm" typically expresses a deontic concept (e.g., a prohibition, permission, obligation, or dispensation). A coherent set of norms, i.e., created and evaluated as a unit, is referred to as an institution [103]. The same understanding of norms is found in the Semantic Web literature, where there is also a body of work focusing on policy specification and enforcement. Here, *policy* is an overarching term used to refer to a variety of system constraints, whereas the term preferences is primarily used in connection with privacy and personal data protection.

The study of norms is a long-running and active line of research within the MAS community, as evidenced by numerous Dagstuhl seminars [5, 48], and a handbook on the topic [32]. Normative MAS [16] are realised and characterised in multiple ways, including those based on: (1) the agents reasoning capabilities; (2) whether norms are implicit or explicit; and (3) whether or not the architecture includes monitoring and enforcement mechanisms.

[16] Internationalised Resource Identifiers [54]

[17] https://github.com/eclipse/thingweb.node-wot

Agent capabilities vis-à-vis norms typically fall into three categories: (i) *norm unaware*, whereby agents may be regimented by external agencies to enforce norm compliance [7]; (ii) *norm-aware*, where agents may choose whether or not to comply with norms, depending on the alignment of their goals with those norms, the penalties for non-compliance, and the likelihood of enforcement [122]; and (iii) *value aware*, whereby agents, in addition to being norm-aware, are able to participate in norm creation and norm revision, by reasoning about the values supported (or not) by particular norms [41]. Thus, compliance in normative systems depends on how individual agents reason and adapt to norms at both design and run time [93, 136, 141]. Implicit norms that reside within the agents themselves are expressed through agent behaviour, but are not otherwise externally discernible, whereas explicit or referenceable norms may have an abstract representation involving variables and a grounded (detached) representation in an entity such as a contract [124], institution [53, 56, 103], or organisation [17, 139]. Agents that are norm or value aware should be able to: (i) recognise norms; (ii) decide whether they want to follow them; and (iii) adapt their behaviour according to the norms, if they decide to do so. Such agents may additionally be able to engage in norm revision processes. Norms, and more broadly *conventions* or *social norms* [94], are established in an agent society in one of two ways, namely top-down and bottom-up [101, 149].

In *top-down* systems, norms are identified as part of the MAS design process and are either: hard-coded into the agents' behaviour (implicit representation), eschewing any form of normative reasoning and narrowing the scope for behavioural adaptation; or are prescriptive and explicitly represented, and thus external to the agents, typically represented in the form of abstract regulations (for example, ungrounded terms over variables) that, as a result of agent actions, become detached (for example, grounded terms over literals). The n-BDI variant [43] is a BDI-based agent architecture that allows for the internalisation of norms where the design suggests an agent-internal process that synthesise norm-style rules based on observed behaviour, whereas N-Jason [93] agents perceive institutional facts, which they internalise as beliefs and hence incorporate in their reasoning. Norms designed offline, however thoughtfully crafted for the long-term, are at risk of losing relevance in open, always-on, environments such as the Web, because it is not possible to anticipate all eventualities at design time. Furthermore, drift in the agent demographic or in systems goals, are likely to make norm revision essential over any sufficiently long system lifetime. With explicit norms, any norm change will affect the entire population. Such changes can be effected through a human-in-the-loop approach, where human designers revise the norms and then switch the system over at some suitable point; such as through a shutdown/reboot sequence, or the use of norm-aware planning [122]. In the latter case, an agent must manage a plan sequence that although initially compliant, may cease to be part of the way through the plan due to the change in norms. Such an agent must also be able to check that its learned way of achieving a goal is compliant with the new norms, perhaps by means of some oracle [107], or by being able to acquire a fresh plan that is compliant.

In *bottom-up* systems, an individual agent decides whether or not to adopt a norm: with implicit norms, it may seek advice from others or apply indirect reinforcement learning over its observations, as a basis for prediction, possibly in combination with a *strategy update function* [149]. In such systems, norms are deemed to have emerged once they have been adopted by a sufficiently large fraction of the population; this is typically 90% in most of the literature, and 100% in some cases (which is hard to achieve), or assumes a simple majority, which can risk oscillatory outcomes. However, convergence (this term appears to be used interchangeably with emergence in the literature [102]) is a function of the *capabilities* of the agents. Emergence with explicit norms depends on agent reasoning capabilities. An agent might inform the regulator that it wants to take a particular action in a particular state (without sanction) – the agent knows what it wants but not how to get it—as a request to change the norms without having to reason about norm representation. A more difficult approach is that an agent might propose a new (abstract) norm – the agent knows how to define a new norm to get what it wants [73, 102]. As above, changes have to be actioned, which could be as outlined previously, although pluralist approaches are possible, as put forward by Ostrom [106], or by using one of the many voting mechanisms. The challenge for an agent then becomes how to decide which way to vote, which depends on their reasoning capabilities: are they able to evaluate the consequences of the norm change; and are they selfish (i.e., vote "yes" if the change is individually beneficial, e.g., increases their utility) or altruistic? (i.e., vote "yes" if the change is collectively beneficial). More sophisticated still would be the use of argumentation to determine if the revision is consistent with the population's values [122, 134].

In the early days, Semantic Web researchers proposed general policy languages, such as KAoS [21], Rei [82] and Protune [19], which cater for a variety of different constraints (access control, privacy preferences, regulatory requirements, etc.). A prominent early attempt to provide a semantic model of polices as soft constraints for agents was OWL Polar [121], an OWL DL explicit policy representation language. OWL Polar aims to fulfil the essential requirements of policy representation, reasoning, and analysis, where policies are system-level principles of ideal activity that are binding upon the components of that system, and thus are used to regulate the behaviour of agents [121]. Over the years the Semantic Web community have also proposed policy languages that are tailored to better cater for access control, privacy preferences, licensing, and regulatory governance requirements, including detailed surveys, for example, of the various policy languages, and the different access control enforcement strategies for RDF [87]. From a privacy perspective, the Platform for Privacy Preferences Project (P3P) [42] specification, deemed obsolete in 2018, aimed to allow websites to express their privacy preferences in a machine readable format that could be interpreted by agents that could automate decision making on behalf of humans. The P3P initiative, despite having failed, inspired subsequent work on representing and reasoning over privacy preferences, such as using OWL [65], catering to more expressive privacy preferences [89], and representing consent for personal data processing [18].

Many existing proposals rely on WebID [120], a community-driven specification that offers an identification mechanism making use of Semantic Web technologies to provide password-less authentication. An extension of WebID (specifically WebID-OIDC that relies on OpenID Connect[18]) is used in the *Solid* project. Solid[19] is an ongoing initiative, lead by Tim Berners-Lee, aimed at deploying a distributed Linked Data infrastructure for governing one's personal data, which is built on top of Linked Data Platforms. Additionally, there has been work on *usage* control in the form of licensing [28, 66–68, 143], and more recently, policy languages have been used as a means to represent regulatory constraints [50, 108]. The Open Digital Rights Language [64, 78], although primarily designed for licensing, has been extended to cater for: access policies [133]; requests, data offers and agreements [132]; and regulatory policies [50]. Usage control, however, often proves challenging for organisations and users, and any constraints imposed on the use of data need to ensure that policies are applied consistently across organisations and that there are robust propagation mechanisms preventing policies from becoming invalid [45, 46]. The notion of FAIR ICT Agents [86] is based on FAIR (Findable, Accessible, Interoperable and Reusable) principles [147], where ICT denotes *interactive intelligent agents that are constrained via goals, preferences, norms and usage restrictions.* Thus far, the WoT standards offer only limited support for norms, policies and preferences, which are currently described in guidelines targeted at human developers rather than as declarative, machine-readable statements usable by agents [117].

[18] https://openid.net/connect/

[19] https://solidproject.org/

Although research on norm-aware agents has made reasonable progress to date, much remains to be done to elevate human oversight to align with the three categories [74]: *human-in-the-loop*, where there may be human intervention in each decision cycle; *human-on-the-loop*, where there is human intervention in the design cycle and operation monitoring; and *human-in-command*, where there is human oversight of the overall system, including the means to decide when and how to engage the AI system. The motivated scenario presented herein draws on human-on-the-loop and human-in-command, and indeed it is these levels of abstraction that inspire the governance framework introduced in Section 4, since those are the system characteristics we aim to facilitate.

4 CONCEPTUAL FRAMEWORK

The overarching goal of this section is to identify governance entities, their relations and their purpose, with no aim to be prescriptive in their instantiation. In doing so, we propose a blueprint for the governance of socio-technical systems that can be instantiated in a variety of ways, using a variety of concrete software components. Thus, this section aims to provide guidance for developers on how different parts of an agent governance system fit together and the functions that they contribute. Our objective here is to enable a range of solutions, fit for different purposes, realisable through available (rather than prescribed) software, but still coherent through the framework set out in the three layers shown in Figure 2.

4.1 Global View

In order to provide something actionable for designers and implementors, we ground our framework for the governance of autonomous agents on the Web through three layers that structure the various entities and abstractions needed for the development of socio-technical systems on the Web. Each layer is assigned concepts that are necessary for governance: norms, policies, and preferences (as illustrated in Figure 2). The way in which these different parts are realised, and how they interact is dependent on various design decisions. In setting out this framework, we draw on and organise existing work on norms, policies, and preferences (as described in Section 3) to cater for abstract requirements for the governance of socio-technical systems. This gives rise to the following three layers:

Reactive Things & Services Layer. This layer comprises non-autonomous entities in the environment. As seen in Section 3, such entities are key notions of the WoT architecture [79] for which

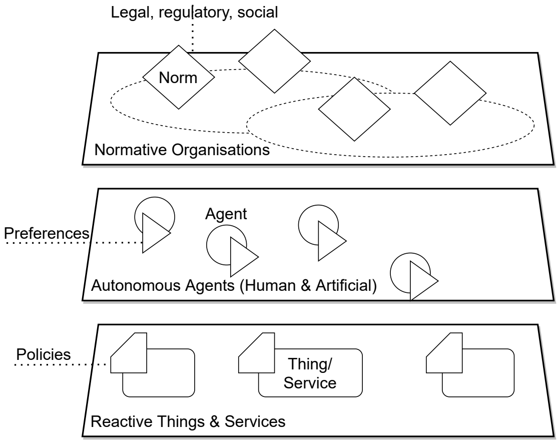

Legal, regulatory, social
Norm
Normative Organisations
Agent
Preferences
Autonomous Agents (Human & Artificial)
Policies
Thing/
Service
Reactive Things & Services

Fig. 2. Conceptual Framework for Governing Agents on the Web structured along three layers. Interactions (not represented here in this Figure) take place within each of the layer and between the layers.

first-class abstractions are proposed for specifying and discovering them and other entities within the MAOP approach (e.g., artefacts in the JaCaMo meta-model [17]). Such entities are perceived and acted upon by agents. We propose the use of *policies* for dealing with the governance of such non autonomous entities, following the same approach adopted by Semantic Web community. These policies state who can access them, and constraints on their usage (if any). Semantic Web technologies such as OWL-POLAR [121] can provide a means to implement, manage, and enforce policies that constrain access to things and services, and the affordances they provide.

Autonomous Agents (Human & Artificial) Layer. The agents layer is composed of entities that autonomously perceive and act upon their environment (i.e., things and services) and interact with the other entities. Agents are the main abstractions for specifying and managing autonomous behaviours. In contrast to the conventional model of programs on the Web as *servers* or *clients*, the WoT architecture introduces a *servient* that can both pro-actively access other things and services and reactively respond to requests from other things and services. In addition, servients can host one or several things. Whilst the Web architecture does not provide first-class abstractions for autonomy, it is possible to distinguish between "agentified" things that exhibit pro-active behaviours and reactive things by introducing custom properties into the W3C WoT *Thing Description* [83]. Agents have *preferences* that inform and constrain their actions with respect to things, web services, and other agents. Preferences control the local reasoning and decision-making undertaken by the agents, and can thus support governance. In traditional deliberation architectures for autonomous agents, preferences are specified (or emerge) as part of the often complex reasoning cycles. Hence, the management of these preferences given the presence of norms and policies can be challenging. Semantic Web approaches that consider preferences (e.g., SPARQL with preferences [112]) can enable declarative preference management, especially when an agent's preferences are to be considered.

Normative Organisations Layer. In MAOP, organisations are first-class abstractions [17] that group agents and their governance (i.e., norms). Although the WoT architecture does not provide such abstractions, its security and privacy guidelines reflect similar notions to organisational norms. Whilst the previous two layers (discussed above) included governance concepts dedicated to the local governance of each entity (e.g., policies for thing, preferences for agent), this layer addresses the governance of autonomous entities *participating in the system.* This layer manages abstractions for the logical grouping of agents with a particular purpose, and the provision of legal, regulatory, and social norms that may possibly span multiple organisations. However, organisations are entirely virtual and passive (i.e., shaped by their members), thus it is up to these member agents to stipulate, comply with (or violate), enforce, and evolve organisational norms. Semantic Web technologies such at ODRL [64] allow for the formalisation of norms for specific domains and purposes; hence, they can be integrated seamlessly with the more abstract MAOP abstractions for organisations and norms that are agnostic to these details.

From a MAS perspective, this framework is coherent with the JaCaMo meta-model [17]; from a WoT perspective, it is coherent with the WoT architecture [79]; and furthermore, it is coherent with the Semantic Web perspective, although with enhancements with respect to policies, preferences, and norms. It is worth noting that our conceptual framework provides *software engineering abstractions.* Analogously to the Web architecture, we do not recommend a one-to-one mapping of software abstractions to physical entities (devices). Considering Web architecture standards, the WoT Scripting API supports, for example, the instantiation of multiple things as part of one *servient*, which may represent a single physical machine.

4.2 The Layers

In this section, we return to each of the layers introduced in Section 4.1 and detail their composition and their governance.

4.2.1 Reactive Things & Services. As defined in Section 3.3, things are physical objects that are endowed with network capacities that allow one to make use of their functions in a digital environment. For example, they can be sensors that provide measurement data through the Internet, or actuators that can be triggered from a Web API. Therefore, in the WoT context, things resemble and are sometimes assimilated within web services. These web services are normally purely digital entities that simply exchange data via their input parameters and output results. In the WoT architecture, things may be autonomous, whereas, in contrast, our conceptual framework distinguishes between autonomous agents (which may be things in the WoT architecture) and *reactive* things.

When it comes to things and services, policies serve many purposes. Access control policies ensure that only authorised agents use specific things and services. Here, there is a need to provision both authentication and authorisation mechanisms, and policies may help resist security threats. Additionally, policies may govern the use of data that is produced by things and services; e.g., to ensure personal data protection or intellectual property rights.

From a policy governance perspective, it is useful to distinguish between *enforcement* and *compliance* of the autonomous agents acting on these things and services given their respective policies. Enforcement means that any violations are prevented, whereas compliance means there is a need for retrospective conformance checking.

4.2.2 Autonomous Agents (Human & Artificial). In contrast to (reactive) things and services, agents are entities that pursue their own goals autonomously. They determine the necessary actions that should be executed on the things and services situated in the environment. In the MAS literature, several agent architectures that are based on the different properties exhibited by the agents have been proposed [148]. They range from purely reactive (i.e., those that respond to stimuli without complex symbolic reasoning to reason about future actions) to deliberative ones (those that maintain a symbolic world model for reasoning about plans and decision making) [29]. A notable example of deliberative architectures is the BDI architecture [115], where agents are programmed using their mental attitudes such as beliefs, desires, wishes, etc. [20] and that is one of the mainstream architectures for cognitive agents in MAS.[20] This contrasts with reactive architectures (such as the subsumption architecture [25, 137]) typically used by robotic systems, whereby *behaviours* define the actions a robot should perform as a consequence of some stimuli (e.g., from sensor data or direct communication). Many hybrid agent architectures [26, 148] combine elements of both reactive and deliberative ones, where prominence is often given to the reactive aspect over the deliberative aspect (such as obstacle avoidance versus goal deliberation). Our conceptual model focuses on governance and is agnostic with respect to any particular architecture, and thus cater for the heterogeneity of agents.

In addition to taking decisions on their own, agents may also coordinate with humans or with other agents to adjust and align their goals with the other agents' goals and identify joint goals, and as such, they may communicate with other agents or human users by exchanging messages. We address the various means of interaction among agents in Section 4.3 below.

Each agent maintains a representation of its internal state that is built from the agent's internal reasoning, from its perceptions of the environment, i.e., the observable state of the things and services deployed in the system, and from its interactions with other agents. Acting on behalf of human users (e.g., assistant agents) or abstract entities (e.g., service agents), agents manage

[20] A number of different MAOP frameworks that adopt a BDI architecture were discussed previously in Section 3.1.

preferences that guide their decision process. It is important to differentiate between agents developed by the application designer and those that enter the system at run-time. This differentiation emphasises the level of control the application designer has over the agent with respect to its internal state. It also justifies the proposition of two levels of governance within our conceptual model: preferences for local and individual control; and organisations for global and collective control. In our conceptual framework, preferences cover many dimensions, ranging from privacy preferences to moral values or ethical principles. Additionally, there can be either agreement or conflict between preferences and access control as defined in the previous layer, due to the fact that an agent may need to verify someone's identity, and based on this determine what information to disclose. Being part of a MAS, the reasoning and decision mechanisms of the agents are enriched with mechanisms to reason over several factors including: norms; regulatory requirements coming from the organisation in which the agent participates; and over policies or access control rules attached to the resources, things, and services with which the agent interacts.

4.2.3 Normative Organisations. Organisations act as coordination mechanisms by which agents work together to achieve their joint goals. The design of agents within an organisation focuses mainly on the agents' capabilities and constraints, as well as on organisational concepts such as *roles* (or functions, or positions), *groups* (or communities), *tasks* (or activities) and *interaction protocols* (or dialogue structures); therefore on what relates the structure of an organisation to the externally observable behaviour of its agents [58]. Organisations usually have a structure defined by: (i) groups, whereby agents are classed together and possibly organised hierarchically; and (ii) roles, whereby agents assume various duties. For example, agents can belong to multiple organisations, be part of various groups, assume different roles (possibly at the same time), and join or leave organisations at will. Organisations can be formed at design time or emerge due to interactions between agents at run-time.

The dynamics of the organisational structure, for example an agent changing its role or joining a group, is governed by rules that are formalised as the norms of the organisation. Norms define what communication is possible, allowed, or forbidden between agents. An organisation is a means to regulate agent behaviour, and such organisations may be governed by norms, including laws and regulations adopted from the social setting or jurisdiction and those legislated within the organisation. The organisation structure and its normative part are described in such a way that agents can autonomously take part in the organisation and regulate themselves automatically with the aim of achieving their (individual or collective) goals. However, a formal, explicit encoding of norms is necessary to facilitate automated compliance and conformance checking.

4.3 Interactions Within and Among the Layers

Within a MAS that is fully aligned with our layered conceptual framework, several interactions may take place within each of the layers and across them (i.e., both inter and intra-layer interactions). Similarly to the Web architecture, the conceptual framework is protocol-agnostic. Some protocols may be chosen based on the underlying things and services to be used; whereas other protocols would be custom to the desired agent-agent interactions; and some of these latter protocols may be designed whereas others evolve. We identify the following types of interactions in the conceptual framework:

Agent-to-agent interactions. Agents can interact with other agents *directly*, by exchanging messages or acting upon each other, or *indirectly*, by observing each other's actions on the reactive things and services of their environment. Because agents are autonomous, the requests that one agent sends to another are handled at the discretion of the receiving agent. In comparison to the interaction with Web services, interaction with an agent may imply a higher likelihood that the response deviates in a complex and nuanced manner from the requested resource.

Agent-to-thing/Thing-to-agent interactions. Agents *proactively* interact with things and services by *acting* upon them, *accessing* their properties, and by listening to (perceiving) events that things and services emit.

Thing-to-thing interactions. While things and services are purely *reactive*, they may interact with other things or services as part of a reaction chain. In this context, existing standards that are part of the Web architecture can be applied for basic communication, but more expressive approaches may be required to manage norms, policies, and preferences, for example when a thing communicates on behalf of an agent across organisational boundaries.

Agent-to-organisation/Organisation-to-agent interactions. An agent's preference depends on the norms of the organisations that the agent is a part of. However, because the agent is an autonomous entity, it may choose to not adopt an organisational norm. At the same time, the agent may attempt to change an organisation's norm, for example by proposing a norm update that then requires approval by a majority of the organisation's agents.

Thing-to-organisation/Organisation-to-thing interactions. In contrast to agents, things and services cannot directly affect organisations. Things and services can be implemented to dynamically adopt policies that reflect organisational norms, and the state of a thing or service can be considered by an organisation, but in both directions, the organisation is the *leading system.*

Organisation-to-organisation interactions. Several organisations may have (unidirectional or bidirectional) dependencies. For example, in a hierarchy of organisations, the norms of lower ranking organisations may depend on norms that are specified on a higher level in the hierarchy; still, a higher ranking organisation may have some norms that depend on the norms of multiple lower ranking organisations (consider dependencies between a federated state and its federal entities).

In the MAS community, interaction protocols are typically designed from a global perspective and aim to facilitate interaction and coordination between agents. A protocol specifies the permitted enactments; i.e., the possible sequences of message exchanges. Proposals for languages for interaction protocols include process algebra [59], Petri Nets [116], and information protocols [123]. Petri nets may then be mapped to models that are more accessible to human users, such as Business Process Model and Notation (BPMN) diagrams. In practice, protocol design and protocol discovery can go hand-in-hand: in particular, Petri Net-based protocols (*processes*) can be *mined* from IT system event logs [140], which, for example, can be used for organisational compliance checking [30]. Recently, *agent system mining* has been proposed as a novel process mining variant that focuses on the agents that participate in one or several (organisational) processes, i.e., on the micro-level instead of on the macro-level process view that an organisation imposes [138]. In the service-oriented community, the notion of a *choreography* is similar to an interaction protocol [9], in that a choreography describes interactions between services from a global perspective.

Human-Agent interactions are typically modelled as conversations between the different agents, i.e., *dialogues* [144]. A dialogue has a normative aspect: it is regulated by norms, and can establish new norms. In normative systems, dialogue protocols are specific notations for norms that specify the violation contexts. Utterances in a dialogue can be seen as moves in the underlying protocol that create obligations and permissions for the participating agents. Of particular interest are persuasive dialogues, where an agent can *convince*, *suggest*, or *command.* Agents can use persuasive dialogues to convince other agents to add new beliefs; to enter into some form of negotiation; or, in the case of the command, a new violation rule is introduced thus creating a new obligation [15].

Besides the protocols themselves and their logical organisation in process choreography, the Agent Communication Language (ACL) and the agreed vocabularies are crucial when it comes to the interaction and co-ordination of agents in a MAS (Section 3.2). The Web architecture specification lists *properties*, *actions*, and *events* as the central abstraction of its interaction model.

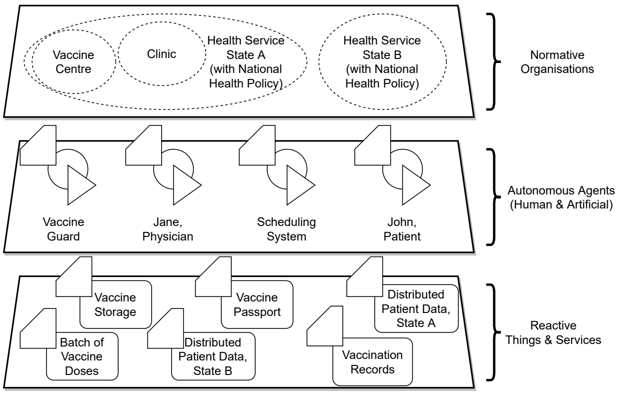

Vaccine Centre
Clinic
Health Service State A (with National Health Policy)
Health Service State B (with National Health Policy)
Normative Organisations
Vaccine Guard
Jane, Physician
Scheduling System
John, Patient
Autonomous Agents (Human & Artificial)
Vaccine Storage
Vaccine Passport
Distributed Patient Data, State A
Batch of Vaccine Doses
Distributed Patient Data, State B
Vaccination Records
Reactive Things & Services

Fig. 3. Normative organisations, autonomous agents, and reactive things and services in the scenario.

In our conceptual framework, organisations, agents, things and services may expose properties and generate events, but only agents may execute actions. From a governance perspective, there is a need for policy, preferences, and norm-aware interaction protocols. For example, agents may need to authenticate themselves to other agents as well as to things and services, whereas collaborating agents may need to engage in preference elicitation and negotiation, and norms may need to be communicated and possibly agreed upon by agents that form part of an organisational structure.

5 USE CASE REVISITED

In this section, we demonstrate how a *normative* MAS that leverages web services, things, and Semantic Web technologies could be used to realise our motivating scenario (Section 2). We show how several example situations can be modelled using the proposed conceptual framework (Section 4) and highlight technologies that could be used to instantiate our governance framework.

5.1 The Global Setting

Agents encapsulate knowledge, goals, and preferences corresponding to the autonomous entities involved in the vaccination process. The resulting conceptual model is illustrated in Figure 3.

An assistant agent is in charge of managing personal data on behalf of a patient (e.g., the patient John). A physician agent is in charge of managing administrative tasks to act on behalf of the physician (e.g., Jane). Other types of agents access the things and services (i.e., a *vaccine guard* agent controls access to the freezer), and to manage the vaccination process by collecting patients' data and checking their eligibility (i.e., *scheduling system* agents). It is worth noting that, contrary to the other agents, the first two kinds of agents (i.e., assistant agents and physician agents) may not be under complete control of the stakeholders who develop and own the application.

We introduce a *vaccination centre organisation* to delimit the *vaccination* application and to provide scope for the adherence to regulations and behaviours for both artificial and human agents that are part of this structure. To this end, the organisation specifies roles and norms, whereby the roles are used to structure agent responsibilities, and the norms (i.e., duties, rights, and interdictions) regulate the vaccination application. Agents with a given role are expected to fulfil the corresponding norms. The vaccination centre is, in turn, part of the *health service organisation* of a particular state, in which *clinic organisations* complement the normative framework provided by state and vaccine centres. In addition to norms, the definition of the organisation may impose hard constraints on its composition that should be enforced by service policies. For example, by stating an upper limit on the number of agents that can adopt specific roles, the vaccination application may consequently limit the number of patients or physicians that may enter the organisation.

5.2 Illustrative Use of the Conceptual Framework with our Motivating Scenario

The following paragraphs describe the use of the conceptual model in situations derived from the motivating scenario. For narrative convenience, we use the terms *obligation*, *permission*, and *authority* in an informal sense.

5.2.1 Initialising the Vaccination Application. At the launch of the application, the *vaccination centre organisation* is created, by endowing the agents that support the business processes within the *vaccination centre organisation* with the roles necessary to fulfil their goals. The definition of the organisation (e.g., the roles and distribution of norms on the roles) is published as a web resource in a machine readable and understandable format, accessible to any agent wishing to become a member of that organisation. The current state of the organisation (i.e., which agent is assigned the various roles) is published and updated as necessary, over the entire lifetime of the organisation. The *freezer agent* adopts the *guard role*, which results in it being assigned the duty of managing access to the inventory of COVID-19 vaccine doses stored in the freezer. It obtains the permission to use the robotic arm to retrieve a vaccine dose when asked, and to deliver it to the staff. The *manager agent* is assigned the *organiser role*, and consequently inherits the obligation to compile lists of eligible patients based on the patient data and the vaccination eligibility policy. The *data agent* adopts the *collector role* and obtains the authority to collect personal information about each patient requesting a vaccination appointment; it also has the obligation to verify the patient's eligibility for receiving the vaccine as well as the obligation to solicit patients through dissemination channels when vaccine doses and scheduling slots are available.

5.2.2 Joining the Vaccination Centre Organisation. When a patient obtains the credentials to access and use the vaccination application, the *assistant agent* acting on behalf of the patient is provided access to the web resource describing the organisation. After reasoning over its obligations and authorities, as imposed by the *vaccination centre organisation*, the agent decides to adopt the role. The *assistant agent* subsequently acquires the obligation to provide access to the patient medical data. This role may create internal conflicts between preferences provided by the patient and the obligations assumed when the agent took on the *patient assistant role*. After accessing and reasoning about the description of the *vaccination centre organisation*, the *physician agent* discovers that it has the obligation to coordinate with agents that are assigned to other roles. To assume the *medical practitioner role*, the *physician agent* must authenticate itself; upon adopting the role, it captures the associated permissions, obligations, and authorisation for further decision making. The same process of role adoption applies to other agents.

5.2.3 Assessing Patient Eligibility. While assigned to the *organiser role*, the *manager agent* takes into account its preferences in defining the patient information collection policy, and sends it to the agent with the *collector role* (as stated by the organisation definition). Fulfilling its obligation, the agent checks the eligibility of all arriving patients so that each dose is only administered to an eligible patient and that doses are administered before their expiry date.[21] To fulfil its goals, the agent therefore requests that agents adopting the new *patient assistant role* share the necessary patient personal data. It is worth noting that agents with the *collector role* need to consider the obligations stated by the organisation as soft constraints, and identify contexts in which these constraints may be relaxed. Further complications may arise if any of the agents attempt to negotiate relaxations of these obligations, either in anticipation of, or after a (perceived or factual) violation. For example, an *assistant agent* can negotiate an exception for a potential obligation violation by using computational

[21] In a practical scenario, we would not expect 100% compliance with this constraint, but rather that the number of excess or wasted doses (relative to administered doses) does not exceed a specified threshold.

models and algorithms of formal argumentation where the *assistant agent* believes that the data it has for its patient satisfies the eligibility criteria. On behalf of the organisation, the agent that has adopted the *organiser role* is in charge of the definition of the eligibility policy, and consequently may interact with the agent in charge of the data collection by granting or denying the request for an exception, or even by updating the organisation's norms in order to accept the request.

5.2.4 Administering the Vaccine. When administering the vaccine, the agent with the *physician role* must respect the priority order for vaccine administration as defined by the agent in charge of the collection; for example, the elderly and vulnerable population must be vaccinated first, unless respecting the priority order implies wasting the dose. Importantly, the physician must not violate this priority order by, for example, preferentially vaccinating friends or relatives. In some cases, the *physician agent* must choose between either administering a vaccine dose despite the eligibility status being uncertain, or allowing the dose to expire. For example, existing regulations indicate that administering a vaccine dose to close relatives is impermissible, but the agent may conclude that in accordance with practitioner norms (such as the Hippocratic Oath), it is preferable if vaccines are not wasted. In such cases, the agent may need to prepare a defence strategy to avoid sanctioning, for example, via argumentation approaches [10, 13]. These issues merely relate to the permissions needed for the obligations implied by opening the fridge. Additional challenges arise when considering the complete socio-technical system, including electronic health record access [81] and supply chain integration [127].

6 CHALLENGES AND OPPORTUNITIES

We now present research challenges (i.e., technical limitations of existing proposals) and opportunities (i.e., open research questions) related to normative agents on the Web. The four horizontal challenges that characterise the contributions of norm-based multiagent systems for the Web are described below, in addition to two orthogonal challenges that need to be tackled in order to address the horizontal issues (and illustrated in Figure 4). For each area, we label the challenges in the context of limitations of the state of the art and subsequently identify future research opportunities. Figure 4 also indicates the practical maturity of each challenge, from *nascent* (blue sky challenges with basic research potential, using a white background) via *developing* (basic research with immediate practical potential, using a white-grey gradient background) to *practical* (challenges that can be addressed primarily from an engineering perspective using a grey background).

6.1 Relating Norms and Interaction Protocols

Challenges. A broad challenge in engineering normative MAS is that we need a way to operationalise norms in the sense of giving them a computational interpretation. Interaction protocols characterise interactions based on message order and occurrence – that is, in operational terms. However, it is nontrivial to produce protocols that are as flexible as necessary, yet enactable in a decentralised manner, while at the same time being verifiably correct. Although the W3C provides Web-based standards for retrieving and querying machine-readable data, these standards do not cater for usage constraints, such as access policies, intellectual property rights, and privacy preferences. In our scenario, an agent may, for example, want to decide with whom and in which context it shares its vaccination status. Existing work on interaction protocols [63] largely focuses on request-response interactions and imposes restrictions on computation for scenarios involving the interaction of three or more parties [33, 59]. In particular, traditional approaches entwine control flow details into the protocol, thereby making it difficult to separate them from the content, for which a declarative meaning can be specified. Prior work on specifying protocols based on norms (commitments) [31, 97] was hindered by the lack of declarative specification of the constraints on messages. More

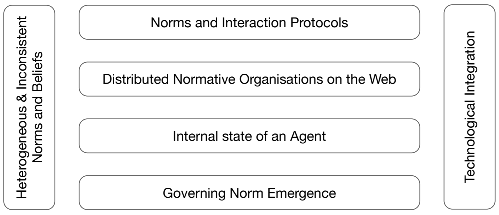

Heterogeneous & Inconsistent Norms and Beliefs
Norms and Interaction Protocols
Distributed Normative Organisations on the Web
Internal state of an Agent
Governing Norm Emergence
Technological Integration

Fig. 4. Overview: Research challenges for normative MAS and the Web

recent approaches describe causality and integrity constraints on messages declaratively [123]; such protocols (whilst sufficiently flexible to support all enactments of the stated norms) can grow quite large [126], but emerging verification approaches aim to tackle this challenge [128].

Opportunities. When it comes to operationalising norms, from a service-provisioning perspective, there is a need to develop policy-aware querying and data retrieval protocols; whereas from an agent interaction perspective, norms should be mapped both to the agent platform and the environment. This raises several important questions, including: (i) How can we design norm-aware dynamic interaction protocols? (ii) How can existing querying and data retrieval protocols be extended, such that they are policy aware? (iii) What new languages are needed to facilitate norm governance? (iv) How do we model and reason with respect to norm changes and temporal validity?

When operating in deployed systems modelled according to the conceptual framework described in Section 4, these interaction protocols also need to take into account that the autonomous agent layer includes human agents which might have conflicting requirements, and therefore strategies might need to be employed in order to resolve conflicts. This raises a number of additional questions that need to be addressed, including: (i) How do we ensure protocol compliance by human agents? (ii) How do we model protocols that implement persuasion? (iii) What mechanisms do we use to resolve conflicting requirements?

6.2 Distributed Normative MAS and Open Organisations on the Web

Challenges. Organisations, institutions, and contracts are useful abstractions to structure norms and make them accessible to agents. Although agents have the choice of joining such structures, they may be subject to conditions that regulate their admission (and exit), as well as there being an expectation to comply with the organisation's norms. Due to the scale of the Web, numerous permanent, ephemeral, or evolving structures may exist. Consequently, an agent needs to be able to discover and reason about such organisations and the corresponding norms. In the vaccination scenario, for example, an agent may need to be able to discover organisations that model the healthcare systems of other jurisdictions that potential patients may need to refer to, when claiming that they are eligible to receive a specific type of vaccine dose. Ontologies facilitate the discovery of services [109], and their use as a means to represent organisations is promising. A major challenge related to the distributed management of such structures [17] is to monitor and enforce norm compliance, and to instantiate organisations, agents, or complete multiagent (sub-)systems at run-time on the Web, which is an emerging line of research in the MAS community [2–4]. Another challenge is that agents require abstractions and mechanisms to build and adapt organisations on the fly [34]. Additionally, an agent may participate in multiple structures that operate at different timescales and scopes, and hence accommodating their diversity is nontrivial.

Opportunities. Addressing the above challenges requires answers to the following research questions, in the context of Web-based and WoT-based technology ecosystems: (i) How can agents discover organisational norms on the Web? (ii) How can norm compliance be monitored and enforced in dynamic scenarios, in which agents, organisations, or entire (sub-)MAS are instantiated at run-time? (iii) How can autonomous agents create and change organisations *on-the-fly*? (iv) How can normative organisations accommodate agents that participate in multiple organisations, with potentially inconsistent norms, and partial semantic interoperability? (v) How can compliance checking and enforcement approaches that are prevalent in the information systems literature be adapted and applied to normative MAS on the Web?

6.3 Internal State of an Agent and Norms

Challenges. An agent should be able reason about norms, taking into consideration its internal state (e.g., its beliefs, goals, and intentions), and explain its normative reasoning to others. This is, for example, important for the *administering the vaccine* scenario (Section 5.2.4), when a decision is to be made about whether to administer a vaccine to a patient whose eligibility status is uncertain, using for example qualitative methods [110], argumentation [10] or sub-symbolic techniques such as classification or Bayesian networks. Several languages express and support automated reasoning about agent internals, such as beliefs, desires or goals, and intentions. However, challenges exist when it comes to reasoning not only with respect to goals, but privacy preferences, regulatory constraints, and norms. The problems here not only relate to knowledge representation – how to represent all these aspects so they are accessible to an agent – but also to the impact such enrichment may have on deliberation performance. The most popular (symbolic) agent architecture is practical reasoning by means of beliefs, desires and intention (BDI), but this offers no guarantees about how long the deliberation cycle may take. Indeed this falls to the developer, in as much as they can endeavour to keep rules relatively simple whilst limiting the number of overlapping conditions, but the actual response time is out of their hands. Additionally, there is the more significant problem that arises due to the fact that agent architecture research has primarily focused on agent internals. Irrespective of whether they are comprised of symbolic or sub-symbolic aspects (such as reasoning, reinforcement or probabilistic learning), such architectures are not normally conceived or designed for interfacing with non-agent technological frameworks and their underlying abstractions (for example as provided by the W3C Web architecture and related standards).

Opportunities. From a norms perspective, important open questions in specification and enforcement include: (i) How can we ensure consistent representation of and adherence to norms? (ii) How should a governance architecture be designed in which rational agents are incentivised to comply with norms? (iii) Can a norm violation be excused, based on explanation or argumentation? (iv) Could transparency facilitate the persuasiveness of the explanation or argument? (v) How do we ensure that an agent is aware of the implications of violating a norm? (vi) How do we cater for agents that are not rational (as such agents may not be designed for criteria such as rationality)? (vii) How should performance limitations affect normative decision-making in practical reasoning architectures? (viii) How can normative reasoning be implemented in real-world agent architectures (which may only be agent-oriented in the broader sense)? (ix) How can agent architectures be better integrated with web technologies and standards?

6.4 Governing Norm Emergence

Challenges. Approaches for the governance of norm emergence are dependent on the capabilities of the agents in a MAS, bearing in mind that population properties may not be homogeneous. In our example scenario, the governance of norm emergence is, for example, important to facilitate vaccinations (i.e., the belief that getting vaccinated is, while typically not mandatory, good for one's health and more broadly for the public health at large), and to balance "hard" rules and "soft" recommendations to decrease the spread of COVID-19. The challenges here include modelling and managing the spread of beliefs and counter-beliefs, the potential resolution of contrary positions through argumentation, and how to make hard and soft policies accessible to different agent architectures with different reasoning capabilities. We differentiate between a decentralised approach to norm emergence with implicit norms, where the norms emerge through the interactions of agents – [1] is one example of such a scheme – and various centralised approaches to the governance of norm emergence [101], which latter we classify by adapting the oversight terminology put forward by the High-Level Expert Group on Artificial Intelligence (AI HLEG) [74, §B.II.1.1]: (i) an external agency observes the behaviour of the population to identify patterns of behaviour and revise norms imposed by that agency to optimise for system goals (external agent/human on-the-loop) – for example Morales et al. [100] look at individual norms in isolation – while the general case of revising a consistent body of norms remains open; (ii) agents propose norm revisions to an external agency, which then implements them subject to an assessment of how those revisions contribute towards system goals (external agent/human in-command), which also remains open; and (iii) agents propose norm revisions and system participants, which may include humans, use an internal decision-making mechanism to establish which changes will be implemented (internal agents/humans in-the-loop). The uHelp system illustrates some preliminary steps in this direction [104], but relegates software agents to a supporting role. The human-in-the loop, human-in-command, human-on-the-loop approaches are closely related to the different strategies that regulate persuasive dialogues [15], where commands introduce new obligations, whilst suggestions introduce new beliefs. However, open challenges relate to devising persuasion strategies and the corresponding obligations.

Opportunities. In order to govern norm emergence in Web-based MAS and their socio-technical contexts, one needs to answer – for example – the following questions:

(i) How can the emergence of norms in conjunction with governance decisions be monitored and managed in Web-based MAS? (ii) What roles do human-on-the-loop, human-in-the-loop, and human-in-command approaches play in the context of the preceding question, and what are the engineering implications of these different human interaction approaches? (iii) What is necessary to maintain alignment of the (evolving) value preferences of participants with the norms that govern them: when does one norm change become many changes? (iv) Which collective decision-making mechanisms are best suited for all agent and for mixed human-agent systems? (v) What is the appropriate capacity for agent reasoning about (self-)governance? Is wanting to do something that is prohibited, or not wanting to do something that is obliged a sufficient statement of intention? (vi) How can an agent and/or a human evaluate the consequences of norm revisions? Will they create a fresh problem while resolving the current one? (vii) How can oscillatory norm change be prevented? An agent-dominated system could potentially change faster than a human-in-command can evaluate the changes.

6.5 Heterogeneous and Inconsistent Norms and Beliefs

Challenges. In heterogeneous information systems, we cannot reasonably assume that norms and policies are globally accepted and thus agents may hold inconsistent beliefs about them. For example, in our vaccination scenario, vaccine administration policies, eligibility requirements, and IT system landscapes may differ between two federated states A and B. However, a patient who moves permanently from A to B should ideally be able to receive the first vaccine dose in state A, and a dose of a matching or complementary type in state B after a reasonable time interval. For aligning norms and policies of sub-entities, reaching (partial) agreements in the face of conflicting beliefs regarding norms and policies is an important challenge that needs to be tackled to enable normative distributed MAS on the Web; using, for example, long-running lines of research on agreement technologies [105] and formal argumentation [10]. Currently, the body of research on belief revision and argumentation-based reasoning is, however, poorly integrated with practical engineering perspectives; standardisation efforts like the specification of an *argument interchange format* [114] exist, but have not found substantial adoption.

Opportunities. There are several open questions when it comes to enabling reasoning and decision-making in the face of inconsistent norms and beliefs of agents on the Web, including: (i) To what extent is there a practical need for engineering abstractions that treat conflict and inconsistency in the context of a normative Web (of Things)? (ii) What systematic approaches to drawing inferences and making decisions in a Web (of Things) governance context can be designed, implemented, and standardised as software engineering abstractions? (iii) How can existing research on belief revision and argumentation-based reasoning be made more accessible both from an engineering and a standardisation perspective?

6.6 Technological Integration

Challenges. In order to facilitate the practical applicability of research on norms and policies for autonomous agents on the Web, it is crucial to build bridges across the technology ecosystems of the different communities. Section 3 provides an overview of the technology ecosystems that have emerged from the MAS, Semantic Web, and WoT communities. To summarise, in the WoT community, engineering-oriented work has been conducted in a highly practice-oriented manner, in close alignment with industry practitioners as well as standardisation bodies such as the W3C. An example of practice-oriented work can be observed though W3C IoT standards that feature an abstract architecture [90] and an interface specification (W3C WoT Scripting API) [88], supported by a JavaScript reference implementation.[22] Research on engineering autonomous agents and MAS has primarily gained traction within the academic community [99], and standardisation attempts such as FIPA[23] have lacked significant adoption. Adjusting agent-oriented programming and software engineering approaches to better serve the Semantic Web and WoT communities is a way for the MAS community to move their engineering research closer to practice. This lets us conclude that while each of the communities has its own thriving technology ecosystem, a key challenge lies in integrating these ecosystems, which exhibit different degrees of practical maturity.

Opportunities. The above observations raise two questions: (i) How can the technology ecosystems of (normative) MAS and the Semantic Web be integrated with the WoT, and in particular with the W3C Web architecture? (ii) How can issues of practical maturity be mitigated (by the integration strategy)? With respect to *(i)*, we argue that an integration strategy can employ a combination of two approaches across two dimensions that requires pragmatic trade-offs, considering the discrepancies between the technology ecosystems and their underlying conceptual abstractions.

Approach 1: Full-Fledged Framework Adoption. In order to facilitate implementations that build on research in the different communities, interfaces that integrate Semantic Web, MAOP, and WoT technologies can be devised that either re-implement their abstractions or integrate technology frameworks and specification languages [36–38]. A benefit of this approach is that it facilitates the adoption of powerful abstractions and technology ecosystems developed in these communities. A disadvantage, however, is that this approach can cause a high technological overhead.

Approach 2: Modular Abstraction Adoption. In order to facilitate implementations that build on technology stacks established in industry, *minimally viable abstractions* on norms and autonomous

[22] https://github.com/eclipse/thingweb.node-wot

[23] http://www.fipa.org/.

agents can be implemented as reusable modules in mainstream programming languages; or alternatively, specific features of complex technology platforms can be exposed as service-oriented interfaces. This strategy resembles a call to action as made in Logan's *Agent Programming Manifesto* [96] and allows for deliberate trade-offs between conceptual richness and practical feasibility by avoiding the overhead (on conventional developers) of having to learn unfamiliar programming paradigms. For example, one may adopt JaCaMo's capabilities for modelling organisations and artefacts via a Java-based technology stack, and defer adopting Jason [20] since it involves a custom language for agent-oriented programming.

Broadly, the two approaches can be considered analogous to the *integrated system* (Approach 1) and *best of breed* (Approach 2) strategies for implementing large scale enterprise systems [95]. In an actual implementation scenario, these approaches represent the extremes of a scale with valuable trade-offs in between. We suggest that this trade-off is initially made from a conceptual perspective (*Which programming abstractions are useful in a given scenario?*) and followed using a technology perspective (*Which technologies do I want to use to implement these abstractions?*).

With respect to the second question, we argue that the strategy should prioritise mature technologies, and if necessary re-implement the requisite abstractions in technology stacks that are established in industry practice. Specifically, we might consider the WoT standards and technologies as a mature foundation on which to place Semantic Web and MAS technologies. An example of a synergy is in extending WoT servients to autonomous agents without necessarily committing to a BDI architecture for those agents.

7 CONCLUSION

This paper discusses the relevance of norms, policies, and preferences for governing complex socio-technical multiagent systems on the Web. The key challenge – the conceptual and technological integration of normative concepts with WoT abstractions and systematic evaluation of the practical usefulness of the integration results – is aligned with the general challenge for autonomous agents on the Web to transfer the rich theoretical achievements of the broader MAS community to the practical and engineering-oriented WoT community, and to facilitate real-world applications at scale. While the challenge of transferring research on normative agents and multiagent systems into engineering practice is well-known and generally acknowledged, this paper has taken the emergence of new Web standards, as well as the increased research interest in Web-based MAS, as a starting point to provide a new and broad perspective on it, with a focus on the Web and Web of Things Architecture standards.

In this context, the paper proposes a conceptual framework that serves to define the role played by various norms, policies and preferences when it comes to complex socio-technical MAS on the Web, and demonstrated it via a simple but realistic scenario.

In addition, the paper provides a research roadmap outlining the technical and theoretical research challenges and opportunities to support complex socio-technical MAS governance on the Web. In particular, this roadmap calls for: (i) relating norms and interaction protocols; (ii) incorporating normative organisations and norm governance approaches into WoT architectures and standards; (iii) combining agent reasoning to relate policies, preferences, and norms; (iv) tackling the emergence of norms for flexible governance; (v) designing reasoning methods about norms in the face of inconsistency; and (vi) cautiously advancing Semantic Web and (normative) MAS tools and frameworks into practice via the WoT.

Acknowledgments. We thank the anonymous reviewers for the useful and insightful comments on the paper. Timotheus Kampik is supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. Adnane Mansour, Olivier Boissier, and Antoine Zimmermann are supported by the French National Research Agency under grant ANR-19-CE23-00030 (project HyperAgents). Sabrina Kirrane is funded by the FWF Austrian Science Fund and the Internet Foundation Austria under the FWF Elise Richter and netidee SCIENCE programmes as project number V 759-N. Munindar P. Singh acknowledges support from the US National Science Foundation under grant IIS-1908374.

REFERENCES

[1] Nirav Ajmeri, Hui Guo, Pradeep K. Murukannaiah, and Munindar P. Singh. 2018. Robust Norm Emergence by Revealing and Reasoning about Context: Socially Intelligent Agents for Enhancing Privacy. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*. IJCAI, Stockholm, 28–34. https://doi.org/10.24963/ijcai.2018/4

[2] Huib Aldewereld, Julian Padget, Wamberto Vasconcelos, Javier Vazquez-Salceda, Paul Sergeant, and Athanasios Staikopoulos. 2010. Adaptable, Organization-Aware, Service-Oriented Computing. *IEEE Intelligent Systems* 25, 4 (2010), 26–35. https://doi.org/10.1109/MIS.2010.93

[3] Cleber Jorge Amaral and Jomi Fred Hübner. 2020. Jacamo-Web is on the Fly: An Interactive Multi-Agent System IDE. In *Engineering Multi-Agent Systems*, Louise A. Dennis, Rafael H. Bordini, and Yves Lespérance (Eds.). Springer International Publishing, Cham, 246–255.

[4] Cleber Jorge Amaral, Jomi Fred Hübner, and Timotheus Kampik. 2020. Towards Jacamo-rest: A Resource-Oriented Abstraction for Managing Multi-Agent Systems. In *Proceedings of the 14th Workshop-School on Agents, Environments, and Applications*, Gleifer Vaz Alves, Gustavo Guiménez Lug, André Pinz Borgeso, and Carlos Eduardo Pantoja (Eds.). UTFPR, Ponta Grossa, Parana, Brazil, 140–151.

[5] Giulia Andrighetto, Guido Governatori, Pablo Noriega, and Leon van der Torre. 2012. *Normative Multi-Agent Systems (Dagstuhl Seminar 12111).* Technical Report 3. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[6] Anupriya Ankolekar, Mark Burstein, Jerry R. Hobbs, Ora Lassila, David Martin, Drew McDermott, Sheila A. McIlraith, Srini Narayanan, Massimo Paolucci, Terry Payne, and Katia Sycara. 2002. DAML-S: Web Service Description for the Semantic Web. In *The Semantic Web — ISWC 2002*, Ian Horrocks and James Hendler (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 348–363.

[7] Josep Lluís Arcos, Marc Esteva, Pablo Noriega, Juan A. Rodríguez-Aguilar, and Carles Sierra. 2005. Engineering open environments with electronic institutions. *Engineering Applications of Artificial Intelligence* 18, 2 (2005), 191–204. https://doi.org/10.1016/j.engappai.2004.11.019

[8] John Langshaw Austin. 1962. *How to do things with words.* Oxford University Press, Oxford, England.

[9] Adam Barker, Christopher D. Walton, and David Robertson. 2009. Choreographing Web Services. *IEEE Transactions on Services Computing* 2, 2 (2009), 152–166. https://doi.org/10.1109/TSC.2009.8

[10] Pietro Baroni, Dov M. Gabbay, Massimiliano Giacomin, and Leendert van der Torre. 2018. *Handbook of formal argumentation.* College Publications, Rickmansworth.

[11] Wouter Beek, Laurens Rietveld, Hamid R. Bazoobandi, Jan Wielemaker, and Stefan Schlobach. 2014. LOD Laundromat: A Uniform Way of Publishing Other People's Dirty Data. In *The Semantic Web – ISWC 2014*, Peter Mika, Tania Tudorache, Abraham Bernstein, Chris Welty, Craig Knoblock, Denny Vrandečić, Paul Groth, Natasha Noy, Krzysztof Janowicz, and Carole Goble (Eds.). Springer International Publishing, Cham, 213–228.

[12] Fabio Luigi Bellifemine, Giovanni Caire, and Dominic Greenwood. 2007. *Developing Multi-Agent Systems with JADE (Wiley Series in Agent Technology).* John Wiley and Sons, Inc., Hoboken, NJ, USA.

[13] Trevor Bench-Capon, Henry Prakken, and Giovanni Sartor. 2009. *Argumentation in Legal Reasoning.* Springer US, Boston, MA, 363–382. https://doi.org/10.1007/978-0-387-98197-0_18

[14] Tim Berners-Lee, James Hendler, and Ora Lassila. 2001. The Semantic Web. *Scientific American* 284, 5 (2001), 34–43.

[15] G. Boella, J. Hulstijn, and L. van der Torre. 2004. Persuasion Strategies in Dialogue. In *Procs. of CMNA Workshop at ECAI'04.* Online Proceedings, University of Liverpool, Valencia, 29–32.

[16] Guido Boella, Leendert Van Der Torre, and Harko Verhagen. 2006. Introduction to normative multiagent systems. *Computational & Mathematical Organization Theory* 12, 2-3 (2006), 71–79.

[17] Olivier Boissier, Rafael H. Bordini, Jomi Hübner, and Alessandro Ricci. 2020. *Multi-agent oriented programming: programming multi-agent systems using JaCaMo.* MIT Press, Cambridge.

[18] Piero A Bonatti, Sabrina Kirrane, Iliana M. Petrova, and Luigi Sauro. 2020. Machine Understandable Policies and GDPR Compliance Checking. *KI-Künstliche Intelligenz* 34, 3 (2020), 303–315.

[19] Piero A. Bonatti and Daniel Olmedilla. 2007. *Rule-Based Policy Representation and Reasoning for the Semantic Web.* Springer Berlin Heidelberg, Berlin, Heidelberg, 240–268. https://doi.org/10.1007/978-3-540-74615-7_4

[20] Rafael H. Bordini, Jomi Fred Hübner, and Michael Wooldridge. 2007. *Programming Multi-Agent Systems in AgentSpeak Using Jason (Wiley Series in Agent Technology).* John Wiley & Sons, Inc., Hoboken, NJ, USA.

[21] Jeffrey M. Bradshaw (Ed.). 1997. *Software agents.* MIT press, Cambridge, MA.

[22] Lars Braubach, Alexander Pokahr, and Winfried Lamersdorf. 2005. Jadex: A BDI-Agent System Combining Middleware and Reasoning. In *Software Agent-Based Applications, Platforms and Development Kits*, Rainer Unland, Monique Calisti, and Matthias Klusch (Eds.). Birkhäuser Basel, Basel, 143–168.

[23] Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, and François Yergeau. 2008. *Extensible Markup Language (XML) 1.0 (Fifth Edition), W3C Recommendation 26 November 2008.* W3C Recommendation. World Wide Web Consortium (W3C). https://www.w3.org/TR/2008/REC-xml-20081126/

[24] Dan Brickley and Ramanathan V. Guha. 2014. *RDF Schema 1.1, W3C Recommendation 25 February 2014.* W3C Recommendation. World Wide Web Consortium (W3C). http://www.w3.org/TR/2014/REC-rdf-schema-20140225/

[25] Rodney A. Brooks. 1991. Intelligence without representation. *Artificial Intelligence* 47, 1 (1991), 139–159.

[26] Joanna Bryson and Lynn Andrea Stein. 2001. Modularity and Design in Reactive Intelligence. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI 2001, Seattle, Washington, USA, August 4-10, 2001*, Bernhard Nebel (Ed.). Morgan Kaufmann, Palo Alto, California, 1115–1120.

[27] Paolo Busetta, Ralph Rönnquist, Andrew Hodgson, and Andrew Lucas. 2019. JACK intelligent agents - Components for intelligent agents in JAVA. Available from https://aosgrp.com/media/research/jack/busetta99jack.pdf. AgentLink Newsletter Issue 2.

[28] Elena Cabrio, Alessio Palmero Aprosio, and Serena Villata. 2014. These Are Your Rights - A Natural Language Processing Approach to Automated RDF Licenses Generation. In *The Semantic Web: Trends and Challenges - 11th International Conference, ESWC 2014, Anissaras, Crete, Greece, May 25-29, 2014. Proceedings (Lecture Notes in Computer Science)*, Vol. 8465. Springer, Cham, 255–269.

[29] Roberta Calegari, Giovanni Ciatto, Viviana Mascardi, and Andrea Omicini. 2021. Logic-based technologies for multi-agent systems: a systematic literature review. *Autonomous Agents and Multi Agent Systems* 35, 1 (2021), 1. https://doi.org/10.1007/s10458-020-09478-3

[30] Filip Caron, Jan Vanthienen, and Bart Baesens. 2013. Comprehensive rule-based compliance checking and risk management with process mining. *Decision Support Systems* 54, 3 (2013), 1357–1369. https://doi.org/10.1016/j.dss.2012.12.012

[31] Amit Chopra and Munindar P. Singh. 2004. Nonmonotonic Commitment Machines. In *Advances in Agent Communication: Proceedings of the International Workshop on Agent Communication Languages (ACL 2003) (Lecture Notes in Artificial Intelligence)*, Frank Dignum (Ed.). Springer, Melbourne, 183–200. https://doi.org/10.1007/978-3-540-24608-4_11

[32] Amit Chopra, Leendert van der Torre, Harko Verhagen, and Serena Villata. 2018. *Handbook of Normative Multiagent Systems.* College Publications, Rickmansworth.

[33] Amit K. Chopra, Samuel H. Christie V, and Munindar P. Singh. 2020. An Evaluation of Communication Protocol Languages for Engineering Multiagent Systems. *Journal of Artificial Intelligence Research (JAIR)* 69 (Dec. 2020), 1351–1393. https://doi.org/10.1613/jair.1.12212

[34] Amit K. Chopra and Munindar P. Singh. 2016. From Social Machines to Social Protocols: Software Engineering Foundations for Sociotechnical Systems. In *Proceedings of the 25th International World Wide Web Conference.* ACM, Montréal, 903–914. https://doi.org/10.1145/2872427.2883018

[35] Andrea Cimmino, Michael McCool, Farshid Tavakolizadeh, and Kunihiko Toumura. 2020. *Web of Things (WoT) Discovery, W3C First Public Working Draft 24 November 2020.* W3C First Public Working Draft. World Wide Web Consortium (W3C). http://www.w3.org/TR/2020/WD-wot-discovery-20201124/

[36] Andrei Ciortea, Olivier Boissier, and Alessandro Ricci. 2018. Engineering World-Wide Multi-Agent Systems with Hypermedia. In *Engineering Multi-Agent Systems - 6th International Workshop, EMAS 2018, Stockholm, Sweden, July 14-15, 2018, Revised Selected Papers (Lecture Notes in Computer Science)*, Danny Weyns, Viviana Mascardi, and Alessandro Ricci (Eds.), Vol. 11375. Springer, Cham, 285–301. https://doi.org/10.1007/978-3-030-25693-7_15

[37] Andrei Ciortea, Olivier Boissier, Antoine Zimmermann, and Adina Magda Florea. 2018. Give Agents Some REST: Hypermedia-driven Agent Environments. In *Engineering Multi-Agent Systems*, Amal El Fallah-Seghrouchni, Alessandro Ricci, and Tran Cao Son (Eds.). Springer International Publishing, Cham, 125–141.

[38] Andrei Ciortea, Simon Mayer, Olivier Boissier, and Fabien Gandon. 2019. Exploiting Interaction Affordances: On Engineering Autonomous Systems for the Web of Things. In *Second W3C Workshop on the Web of Things The Open Web to Challenge IoT Fragmentation.* W3C, Munich, Germany, 0–4. https://hal.archives-ouvertes.fr/hal-02196903

[39] Andrei Ciortea, Simon Mayer, Fabien Gandon, Olivier Boissier, Alessandro Ricci, and Antoine Zimmermann. 2019. A Decade in Hindsight: The Missing Bridge Between Multi-Agent Systems and the World Wide Web. In *AAMAS 2019 - 18th International Conference on Autonomous Agents and Multiagent Systems.* International Foundation for Autonomous Agents and Multiagent Systems., Montréal, Canada, 5. https://hal-emse.ccsd.cnrs.fr/emse-02070625

[40] Simon Cox and Chris Little. 2017. *The Time Ontology in OWL, W3C Recommendation 19 October 2017.* W3C Recommendation. World Wide Web Consortium (W3C). http://www.w3.org/TR/2017/REC-prov-o-20171019/

[41] Stephen Cranefield, Michael Winikoff, Virginia Dignum, and Frank Dignum. 2017. No Pizza for You: Value-based Plan Selection in BDI Agents. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence,*

(IJCAI), Carles Sierra (Ed.). ijcai.org, Melbourne, 178–184. https://doi.org/10.24963/ijcai.2017/26

[42] Lorrie Cranor, Marc Langheinrich, Massimo Marchiori, Martin Presler-Marshall, and Joseph Reagle. 2002. *The Platform for Privacy Preferences 1.0 (P3P1.0) Specification.* W3C Recommendation. World Wide Web Consortium. Obsolete recommendation since 30 August 2018.

[43] N. Criado, E. Argente, P. Noriega, and V. Botti. 2010. Towards a Normative BDI Architecture for Norm Compliance. In *COIN@*. Citeseer, Informal Proceedings, Lyon, France, 65.

[44] Richard Cyganiak, David Wood, and Markus Lanthaler. 2014. *RDF 1.1 Concepts and Abstract Syntax, W3C Recommendation 25 February 2014.* W3C Recommendation. World Wide Web Consortium (W3C). http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/

[45] Enrico Daga, Mathieu d'Aquin, Alessandro Adamou, and Enrico Motta. 2016. Addressing exploitability of Smart City data. In *2016 IEEE International Smart Cities Conference (ISC2).* IEEE, Piscataway, New Jersey, United States, 1–6. https://doi.org/10.1109/ISC2.2016.7580764

[46] Enrico Daga, Mathieu d'Aquin, Aldo Gangemi, and Enrico Motta. 2015. Propagation of Policies in Rich Data Flows. In *Proceedings of the 8th International Conference on Knowledge Capture* (Palisades, NY, USA) *(K-CAP 2015).* Association for Computing Machinery, New York, NY, USA, Article 5, 8 pages. https://doi.org/10.1145/2815833.2815839

[47] Mehdi Dastani. 2008. 2APL: A Practical Agent Programming Language. *Autonomous Agents and Multi-Agent Systems* 16, 3 (2008), 214–248.

[48] Mehdi Dastani, Jürgen Dix, Harko Verhagen, and Serena Villata. 2018. Normative Multi-Agent Systems (Dagstuhl Seminar 18171). *Dagstuhl Reports* 8, 4 (2018), 72–103.

[49] Fernando De la Prieta, Sara Rodríguez-González, Pablo Chamoso, Juan Manuel Corchado, and Javier Bajo. 2019. Survey of agent-based cloud computing applications. *Future Generation Computer Systems* 100 (2019), 223–236. https://doi.org/10.1016/j.future.2019.04.037

[50] Marina De Vos, Sabrina Kirrane, Julian Padget, and Ken Satoh. 2019. ODRL Policy Modelling and Compliance Checking. In *Rules and Reasoning*, Paul Fodor, Marco Montali, Diego Calvanese, and Dumitru Roman (Eds.). Springer International Publishing, Cham, 36–51.

[51] Keith Decker, Katia Sycara, and Mike Williamson. 1997. Middle-agents for the internet. In *IJCAI-97: Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence* (Nagoya, Japan). Morgan Kaufmann, San Francisco, California, 578–583.

[52] Nirmit Desai, Ashok U. Mallya, Amit K. Chopra, and Munindar P. Singh. 2006. OWL-P: A Methodology for Business Process Development. In *Agent-Oriented Information Systems III, 7th International Bi-Conference Workshop, AOIS2005, Utrecht, Netherlands, July 26, 2005, and Klagenfurt, Austria, October 27, 2005, Revised Selected Papers (Lecture Notes in Computer Science).* Springer, Berlin, 79–94. https://doi.org/10.1007/11916291_6

[53] Mark d'Inverno, Michael Luck, Pablo Noriega, Juan A. Rodríguez-Aguilar, and Carles Sierra. 2012. Communicating open systems. *Artificial Intelligence* 186 (2012), 38–94. https://doi.org/10.1016/j.artint.2012.03.004

[54] Martin J. Dürst and Michel Suignard. 2005. *Internationalized Resource Identifiers (IRIs).* Technical Report. Internet Engineering Task Force. http://tools.ietf.org/html/rfc3987

[55] Marc Eisenstadt, Manoharan Ramachandran, Niaz Chowdhury, Allan Third, and John Domingue. 2020. COVID-19 Antibody Test/Vaccination Certification: There's an App for That. *IEEE Open Journal of Engineering in Medicine and Biology* 1 (2020), 148–155. https://doi.org/10.1109/OJEMB.2020.2999214

[56] Marc Esteva, Julian Padget, and Carles Sierra. 2002. Formalizing a Language for Institutions and Norms. In *Intelligent Agents VIII*, John-Jules Ch. Meyer and Milind Tambe (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 348–366.

[57] Dieter Fensel and Enrico Motta. 2001. Structured Development of Problem Solving Methods. *IEEE Trans. Knowl. Data Eng.* 13, 6 (2001), 913–932. https://doi.org/10.1109/69.971187

[58] Jacques Ferber, Olivier Gutknecht, and Fabien Michel. 2004. From Agents to Organizations: An Organizational View of Multi-agent Systems. In *Agent-Oriented Software Engineering IV*, Paolo Giorgini, Jörg P. Müller, and James Odell (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 214–230.

[59] Angelo Ferrando, Michael Winikoff, Stephen Cranefield, Frank Dignum, and Viviana Mascardi. 2019. On Enactability of Agent Interaction Protocols: Towards a Unified Approach. In *Proceedings of the 7th International Workshop on Engineering Multi-Agent Systems (EMAS) (Lecture Notes in Computer Science)*, Vol. 12058. Springer, Montréal, 43–64. https://doi.org/10.1007/978-3-030-51417-4_3

[60] Roy Thomas Fielding. 2000. *Representational State Transfer (REST).* Ph.D. Dissertation. University of California, Irvine. Chapter 5 of Roy Fielding's thesis.

[61] Roy Thomas Fielding and Julian Reschke. 2014. *Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content.* Technical Report. IETF. http://tools.ietf.org/html/rfc7231

[62] Tim Finin, Richard Fritzson, Don McKay, and Robin McEntire. 1994. KQML as an Agent Communication Language. In *Proceedings of the Third International Conference on Information and Knowledge Management* (Gaithersburg, Maryland, USA) *(CIKM '94).* Association for Computing Machinery, New York, NY, USA, 456–463. https://doi.org/10.1145/

191246.191322

[63] FIPA. 2003. FIPA Interaction Protocol Specifications. http://www.fipa.org/repository/ips.html FIPA: The Foundation for Intelligent Physical Agents.

[64] Nicoletta Fornara and Marco Colombetti. 2019. Using Semantic Web technologies and production rules for reasoning on obligations, permissions, and prohibitions. *AI Communications* 32 (2019), 319–334. https://doi.org/10.3233/AIC-190617

[65] Diego Zuquim Guimarães Garcia and Maria Beatriz Felgar de Toledo. 2008. A web service privacy framework based on a policy approach enhanced with ontologies. In *2008 11th IEEE International Conference on Computational Science and Engineering-Workshops*. IEEE, IEEE, Piscataway, New Jersey, United States, 209–214.

[66] Guido Governatori, Lam Ho-Pun, Antonino Rotolo, Serena Villata, and Fabien Gandon. 2013. Heuristics for Licenses Composition. In *Frontiers in Artificial Intelligence and Applications*, Vol. 259. IOS press, Amsterdam, The Netherlands, 77–86.

[67] Guido Governatori, Ho-Pun Lam, Antonino Rotolo, Serena Villata, Ghislain Auguste Atemezing, and Fabien Gandon. 2014. LIVE: A Tool for Checking Licenses Compatibility between Vocabularies and Data. In *Proceedings of the 2014 International Conference on Posters & Demonstrations Track - Volume 1272* (Riva del Garda, Italy) *(ISWC-PD'14)*. CEUR-WS.org, Aachen, DEU, 77–80.

[68] Guido Governatori, Antonino Rotolo, Serena Villata, and Fabien Gandon. 2013. One License to Compose Them All. In *The Semantic Web – ISWC 2013*, Harith Alani, Lalana Kagal, Achille Fokoue, Paul Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha Noy, Chris Welty, and Krzysztof Janowicz (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 151–166.

[69] W3C OWL Working Group. 2012. *OWL 2 Web Ontology Language Document Overview (Second Edition), W3C Recommendation 11 December 2012*. Technical Report. World Wide Web Consortium (W3C). http://www.w3.org/TR/2012/REC-owl2-overview-20121211/

[70] Marina Gueroussova, Axel Polleres, and Sheila McIlraith. 2013. SPARQL with Qualitative and Quantitative Preferences. In *Proceedings of the 2nd International Conference on Ordering and Reasoning - Volume 1059* (Sydney, Australia) *(OrdRing'13)*. CEUR-WS.org, Aachen, DEU, 2–8.

[71] Armin Haller, Krzysztof Janowicz, Simon Cox, Danh Le Phuoc, Jamie Taylor, and Maxime Lefrançois. 2017. *Semantic Sensor Network Ontology, W3C Recommendation 19 October 2017*. W3C Recommendation. World Wide Web Consortium (W3C). https://www.w3.org/TR/2017/REC-vocab-ssn-20171019/

[72] Steve Harris and Andy Seaborne. 2013. *SPARQL 1.1 Query Language, W3C Recommendation 21 March 2013*. W3C Recommendation. World Wide Web Consortium (W3C). http://www.w3.org/TR/2013/REC-sparql11-query-20130321/

[73] Chris Haynes, Michael Luck, Peter McBurney, Samhar Mahmoud, Tomas Vitek, and Simon Miles. 2017. Engineering the emergence of norms: A review. *Knowledge Eng. Review* 32 (2017), e18.

[74] High-Level Expert Group on Artificial Intelligence (AI HLEG). 2019. Ethics Guidelines for Trustworthy AI. https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai

[75] Koen V Hindriks and Jürgen Dix. 2014. GOAL: A Multi-Agent Programming Language Applied to an Exploration Game. In *Agent-Oriented Software Engineering*, O Shehory and A Sturm (Eds.). Springer International Publishing, Cham, 235–258. https://doi.org/10.1007/978-3-642-54432-3_12

[76] Michael N. Huhns, Nigel Jacobs, Tomasz Ksiezyk, Wei-Min Shen, Munindar P. Singh, and Philip E. Cannata. 1993. Integrating Enterprise Information Models in Carnot. In *Proceedings of the International Conference on Intelligent and Cooperative Information Systems (ICICIS)*. IEEE, Rotterdam, 32–42. https://doi.org/10.1109/ICICIS.1993.291772

[77] Michael N. Huhns and Munindar P. Singh. 1997. Ontologies for Agents. *IEEE Internet Computing (IC)* 1, 6 (Nov. 1997), 81–83. https://doi.org/10.1109/4236.643942 Instance of the column *Agents on the Web*.

[78] Renato Ianella and Serena Villata. 2018. *ODRL Information Model 2.2, W3C Recommendation 15 February 2018*. W3C Recommendation. World Wide Web Consortium (W3C). http://www.w3.org/TR/2018/REC-odrl-model-20180215/

[79] Ian Jacobs and Norman Walsh. 2004. *Architecture of the World Wide Web, Volume One, W3C Recommendation 15 December 2004*. W3C Recommendation. World Wide Web Consortium (W3C). http://www.w3.org/TR/2004/REC-webarch-20041215/

[80] Sebastian Kaebisch, Takuki Kamiya, Michael McCool, Victor Charpenay, and Matthias Kovatsch. 2020. *Web of Things (WoT) Thing Description, W3C Recommendation 9 April 2020*. W3C Recommendation. World Wide Web Consortium (W3C). https://www.w3.org/TR/2020/REC-wot-thing-description-20200409/

[81] Özgür Kafali, Nirav Ajmeri, and Munindar P. Singh. 2020. DESEN: Specification of Sociotechnical Systems via Patterns of Regulation and Control. *ACM Transactions on Software Engineering and Methodology (TOSEM)* 29, 1, Article 7 (Feb. 2020), 50 pages. https://doi.org/10.1145/3365664

[82] Lalana Kagal, Tim Finin, and Anupam Joshi. 2003. A Policy Language for a Pervasive Computing Environment. In *Proceedings of the 4th IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY '03)*. IEEE Computer Society, USA, 63.

[83] Timotheus Kampik, Andres Gomez, Andrei Ciortea, and Simon Mayer. 2021. *Autonomous Agents on the Edge of Things.* International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1767–1769.

[84] Michael Kifer, Georg Lausen, and James Wu. 1995. Logical Foundations of Object-Oriented and Frame-Based Languages. *J. ACM* 42, 4 (July 1995), 741–843. https://doi.org/10.1145/210332.210335

[85] Sabrina Kirrane. 2021. Intelligent Software Web Agents: A Gap Analysis. arXiv:2102.06607 [cs.AI]

[86] Sabrina Kirrane and Stefan Decker. 2018. Intelligent Agents: The Vision Revisited. In *Proceedings of the 2nd Workshop on Decentralizing the Semantic Web co-located with the 17th International Semantic Web Conference.* RWTH Aachen University (CEUR-WS Proceedings), Aachen, Germany, 9–17.

[87] Sabrina Kirrane, Alessandra Mileo, and Stefan Decker. 2017. Access control and the resource description framework: A survey. *Semantic Web* 8, 2 (2017), 311–352.

[88] Zoltan Kis, Daniel Peinter, Cristiano Aguzzi, Johannes Hund, and Kazuaki Nimura. 2020. *Web of Things (WoT) Scripting API, W3C Working Group Note 24 November 2020.* W3C Working Group Note. World Wide Web Consortium (W3C). https://www.w3.org/TR/2020/NOTE-wot-scripting-api-20201124/

[89] Pranam Kolari, Li Ding, G Shashidhara, Anupam Joshi, Tim Finin, and Lalana Kagal. 2005. Enhancing web privacy protection through declarative policies. In *Sixth IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY'05).* IEEE, IEEE, Piscataway, New Jersey, United States, 57–66.

[90] Matthias Kovatsch, Ryuichi Matsukura, Michael Lagally, Toru Kawaguchi, Kunihiko Toumura, and Kazuo Kajimoto. 2020. *Web of Things (WoT) Architecture, W3C Recommendation 9 April 2020.* W3C Recommendation. World Wide Web Consortium (W3C). https://www.w3.org/TR/2020/REC-wot-architecture-20200409/

[91] Kalliopi Kravari and Nick Bassiliades. 2015. A survey of agent platforms. *Journal of Artificial Societies and Social Simulation* 18, 1 (2015), 11.

[92] Timothy Lebo, Satya Sanket Sahoo, and Deborah L. McGuinness. 2013. *PROV-O: The PROV Ontology, W3C Recommendation 30 April 2013.* W3C Recommendation. World Wide Web Consortium (W3C). http://www.w3.org/TR/2013/REC-prov-o-20130430/

[93] Jeehang Lee, Julian Padget, Brian Logan, Daniela Dybalova, and Natasha Alechina. 2014. N-Jason: Run-Time Norm Compliance in AgentSpeak(L). In *Engineering Multi-Agent Systems - Second International Workshop, EMAS 2014, Paris, France, May 5-6, 2014, Revised Selected Papers (Lecture Notes in Computer Science)*, Fabiano Dalpiaz, Jürgen Dix, and M. Birna van Riemsdijk (Eds.), Vol. 8758. Springer, Cham, 367–387. https://doi.org/10.1007/978-3-319-14484-9_19

[94] David Lewis. 1969. *Convention — A Philosophical Study.* Harvard University Press, Cambridge, MA.

[95] Ben Light, Christopher P. Holland, and Karl Wills. 2001. ERP and best of breed: a comparative analysis. *Business Process Management Journal* 7, 3 (2021/05/25 2001), 216–224. https://doi.org/10.1108/14637150110392683

[96] Brian Logan. 2018. An agent programming manifesto. *International Journal of Agent-Oriented Software Engineering* 6, 2 (2018), 187–210.

[97] Ashok U. Mallya and Munindar P. Singh. 2007. An Algebra for Commitment Protocols. *Journal of Autonomous Agents and Multi-Agent Systems (JAAMAS)* 14, 2 (April 2007), 143–163. https://doi.org/10.1007/s10458-006-7232-1

[98] David Martin, Massimo Paolucci, Sheila McIlraith, Mark Burstein, Drew McDermott, Deborah McGuinness, Bijan Parsia, Terry Payne, Marta Sabou, Monika Solanki, Naveen Srinivasan, and Katia Sycara. 2005. Bringing Semantics to Web Services: The OWL-S Approach. In *Semantic Web Services and Web Process Composition*, Jorge Cardoso and Amit Sheth (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 26–42.

[99] Viviana Mascardi, Danny Weyns, Alessandro Ricci, Clara Benac Earle, Arthur Casals, Moharram Challenger, Amit Chopra, Andrei Ciortea, Louise A. Dennis, Álvaro Fernández Díaz, Amal El Fallah-Seghrouchni, Angelo Ferrando, Lars-Åke Fredlund, Eleonora Giunchiglia, Zahia Guessoum, Akin Günay, Koen Hindriks, Carlos A. Iglesias, Brian Logan, Timotheus Kampik, Geylani Kardas, Vincent J. Koeman, John Bruntse Larsen, Simon Mayer, Tasio Méndez, Juan Carlos Nieves, Valeria Seidita, Baris Tekin Teze, László Z. Varga, and Michael Winikoff. 2019. Engineering Multi-Agent Systems: State of Affairs and the Road Ahead. *SIGSOFT Softw. Eng. Notes* 44, 1 (March 2019), 18–28. https://doi.org/10.1145/3310013.3322175

[100] Javier Morales, Maite López-Sánchez, Juan Antonio Rodríguez-Aguilar, Wamberto Weber Vasconcelos, and Michael J. Wooldridge. 2015. Online Automated Synthesis of Compact Normative Systems. *ACM Trans. Auton. Adapt. Syst.* 10, 1 (2015), 2:1–2:33. https://doi.org/10.1145/2720024

[101] Andreasa Morris-Martin, Marina De Vos, and Julian Padget. 2019. Norm emergence in multiagent systems: A viewpoint paper. *Autonomous Agents and Multi-Agent Systems* 33, 6 (2019), 706–749. https://doi.org/10.1007/s10458-019-09422-0

[102] Andreasa Morris-Martin, Marina De Vos, and Julian Padget. 2021. A Norm Emergence Framework for Normative MAS – Position Paper. In *Coordination, Organizations, Institutions, Norms, and Ethics for Governance of Multi-Agent Systems XIII*, Andrea Aler Tubella, Stephen Cranefield, Christopher Frantz, Felipe Meneguzzi, and Wamberto Vasconcelos (Eds.). Springer International Publishing, Cham, 156–174.

[103] Pablo Noriega. 1997. *Agent Mediated Auctions: The Fishmarket Metaphor.* Ph.D. Dissertation. Universitat Autònoma de Barcelona. https://www.iiia.csic.es/research/thesis-details?pastphd_id=85

[104] Nardine Osman, Ronald Chenu-Abente, Qiang Shen, Carles Sierra, and Fausto Giunchiglia. 2021. Empowering Users in Online Open Communities. *SN Comput. Sci.* 2, 4 (2021), 338. https://doi.org/10.1007/s42979-021-00714-5

[105] Sascha Ossowski. 2012. *Agreement Technologies.* Springer Publishing Company, Incorporated, Berlin.

[106] Elinor Ostrom. 1990. *Governing the Commons. The Evolutions of Institutions for Collective Action.* Cambridge University Press, Cambridge.

[107] Julian Padget, Marina De Vos, and Charlie Ann Page. 2018. Deontic Sensors. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence* (Stockholm, Sweden) *(IJCAI'18).* AAAI Press, Palo Alto, 475–481.

[108] Monica Palmirani, Guido Governatori, Antonino Rotolo, Said Tabet, Harold Boley, and Adrian Paschke. 2011. LegalRuleML: XML-Based Rules and Norms. In *Proceedings of the 5th International Conference on Rule-Based Modeling and Computing on the Semantic Web* (Ft. Lauderdale, FL, USA) *(RuleML'11).* Springer-Verlag, Berlin, Heidelberg, 298–312.

[109] Massimo Paolucci, Takahiro Kawamura, Terry R. Payne, and Katia Sycara. 2002. Semantic Matching of Web Services Capabilities. In *The Semantic Web — ISWC 2002*, Ian Horrocks and James Hendler (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 333–347.

[110] Simon Parsons. 2001. *Qualitative methods for reasoning under uncertainty.* MIT Press, Cambridge, Massachusetts, United States.

[111] Peter F. Patel-Schneider, Axel Polleres, and David Martin. 2018. Comparative Preferences in SPARQL. In *Knowledge Engineering and Knowledge Management - 21st International Conference, EKAW 2018, Nancy, France, November 12-16, 2018, Proceedings (Lecture Notes in Computer Science)*, Catherine Faron-Zucker, Chiara Ghidini, Amedeo Napoli, and Yannick Toussaint (Eds.), Vol. 11313. Springer, Cham, 289–305. https://doi.org/10.1007/978-3-030-03667-6_19

[112] Olivier Pivert, Olfa Slama, and Virginie Thion. 2016. SPARQL Extensions with Preferences: A Survey. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing* (Pisa, Italy) *(SAC '16).* Association for Computing Machinery, New York, NY, USA, 1015–1020. https://doi.org/10.1145/2851613.2851690

[113] María Poveda-Villalón, Asunción Gómez-Pérez, and Mari Carmen Suárez-Figueroa. 2014. OOPS! (OntOlogy Pitfall Scanner!): An On-line Tool for Ontology Evaluation. *International Journal on Semantic Web and Information Systems (IJSWIS)* 10, 2 (2014), 7–34.

[114] Iyad Rahwan and Chris Reed. 2009. *The Argument Interchange Format.* Springer US, Boston, MA, 383–402. https://doi.org/10.1007/978-0-387-98197-0_19

[115] Anand S. Rao and Michael P. Georgeff. 1995. BDI Agents: From Theory to Practice. In *Proceedings of the First International Conference on Multiagent Systems, June 12-14, 1995, San Francisco, California, USA*, Victor R. Lesser and Les Gasser (Eds.). The MIT Press, Cambridge, 312–319.

[116] Wolfgang Reisig. 1985. *Petri Nets: An Introduction.* Springer-Verlag, Berlin, Heidelberg.

[117] Elena Reshetova and Michael McCool. 2019. *Web of Things (WoT) Security and Privacy Guidelines, W3C Working Group Note 6 November 2019.* W3C Working Group Note. World Wide Web Consortium (W3C). https://www.w3.org/TR/2019/NOTE-wot-security-20191106/

[118] Ahlem Rhayem, Mohamed Ben Ahmed Mhiri, and Faïez Gargouri. 2020. Semantic Web Technologies for the Internet of Things: Systematic Literature Review. *Internet Things* 11 (2020), 100206. https://doi.org/10.1016/j.iot.2020.100206

[119] Dumitru Roman, Uwe Keller, Holger Lausen, Jos de Bruijn, Rubén Lara, Michael Stollberg, Axel Polleres, Cristina Feier, Christoph Bussler, and Dieter Fensel. 2005. Web Service Modeling Ontology. *Appl. Ontology* 1, 1 (2005), 77–106. http://content.iospress.com/articles/applied-ontology/ao000008

[120] Andrei Sambra and Stéphane Corlosquet. 2015. *WebID 1.0 - Web Identity and Discovery, W3C Editor's Draft 28 May 2014.* W3C Editor's draft. World Wide Web Consortium (W3C). https://dvcs.w3.org/hg/WebID/raw-file/tip/spec/identity-respec.html

[121] Murat Sensoy, Timothy J. Norman, Wamberto W. Vasconcelos, and Katia Sycara. 2012. OWL-POLAR: A framework for semantic policy representation and reasoning. *Journal of Web Semantics* 12-13 (2012), 148–160. https://doi.org/10.1016/j.websem.2011.11.005 Reasoning with context in the Semantic Web.

[122] Zohreh Shams, Marina De Vos, Nir Oren, and Julian A. Padget. 2020. Argumentation-Based Reasoning about Plans, Maintenance Goals, and Norms. *ACM Trans. Auton. Adapt. Syst.* 14, 3 (2020), 9:1–9:39. https://doi.org/10.1145/3364220

[123] Munindar P. Singh. 2011. Information-Driven Interaction-Oriented Programming: BSPL, the Blindingly Simple Protocol Language. In *Proceedings of the 10th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS).* IFAAMAS, Taipei, 491–498. https://doi.org/10.5555/2031678.2031687

[124] Munindar P. Singh. 2013. Norms as a Basis for Governing Sociotechnical Systems. *ACM Transactions on Intelligent Systems and Technology (TIST)* 5, 1, Article 21 (Dec. 2013), 23 pages. https://doi.org/10.1145/2542182.2542203

[125] Munindar P. Singh and Amit K. Chopra. 2017. The Internet of Things and Multiagent Systems: Decentralized Intelligence in Distributed Computing. In *37th IEEE International Conference on Distributed Computing Systems, ICDCS 2017, Atlanta, GA, USA, June 5-8, 2017*, Kisung Lee and Ling Liu (Eds.). IEEE Computer Society, Washington, D.C., United States, 1738–1747. https://doi.org/10.1109/ICDCS.2017.304

[126] Munindar P. Singh and Amit K. Chopra. 2020. Clouseau: Generating Communication Protocols from Commitments. In *Proceedings of the 34th Conference on Artificial Intelligence (AAAI)*. AAAI Press, New York, 7244–7252. https://doi.org/10.1609/aaai.v34i05.6215

[127] Munindar P. Singh, Amit K. Chopra, and Nirmit Desai. 2009. Commitment-Based Service-Oriented Architecture. *Computer* 42, 11 (2009), 72–79. https://doi.org/10.1109/MC.2009.347

[128] Munindar P. Singh and Samuel H. Christie V. 2021. Tango: Declarative Semantics for Multiagent Communication Protocols. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*. IJCAI, Online, 391–397. https://doi.org/10.24963/ijcai.2021/55

[129] Munindar P. Singh and Michael N. Huhns. 2005. *Service-Oriented Computing: Semantics, Processes, Agents*. John Wiley & Sons, Chichester, United Kingdom. https://doi.org/10.1002/0470091509

[130] Steve Speicher, John Arwe, and Ashok Malhotra. 2015. *Linked Data Platform 1.0, W3C Recommendation 26 February 2015*. W3C Recommendation. World Wide Web Consortium (W3C). http://www.w3.org/TR/2015/REC-ldp-20150226/

[131] Steffen Staab and Heiner Stuckenschmidt (Eds.). 2006. *Semantic Web and Peer-to-Peer - Decentralized Management and Exchange of Knowledge and Information*. Springer, Berlin. https://doi.org/10.1007/3-540-28347-1

[132] Simon Steyskal and Sabrina Kirrane. 2015. If you can't enforce it, contract it: Enforceability in Policy-Driven (Linked) Data Markets. In *Semantics (posters & demos)*. CEUR, Aachen, 63–66.

[133] Simon Steyskal and Axel Polleres. 2014. Defining Expressive Access Policies for Linked Data Using the ODRL Ontology 2.0. In *Proceedings of the 10th International Conference on Semantic Systems* (Leipzig, Germany) *(SEM '14)*. Association for Computing Machinery, New York, NY, USA, 20–23. https://doi.org/10.1145/2660517.2660530

[134] Christian Straßer and Ofer Arieli. 2014. Sequent-Based Argumentation for Normative Reasoning. In *Deontic Logic and Normative Systems*, Fabrizio Cariani, Davide Grossi, Joke Meheus, and Xavier Parent (Eds.). Springer International Publishing, Cham, 224–240.

[135] Sudeep Tanwar, Karan Parekh, and Richard Evans. 2020. Blockchain-based electronic healthcare record system for healthcare 4.0 applications. *Journal of Information Security and Applications* 50 (2020), 102407. https://doi.org/10.1016/j.jisa.2019.102407

[136] Pankaj R. Telang, Munindar P. Singh, and Neil Yorke-Smith. 2019. A Coupled Operational Semantics for Goals and Commitments. *Journal of Artificial Intelligence Research (JAIR)* 65 (May 2019), 31–85. https://doi.org/10.1613/jair.1.11494

[137] Andreas Theodorou, Robert H. Wortham, and Joanna J. Bryson. 2017. Designing and implementing transparency for real time inspection of autonomous robots. *Connect. Sci.* 29, 3 (2017), 230–241. https://doi.org/10.1080/09540091.2017.1310182

[138] Andrei Tour, Artem Polyvyanyy, and Anna Kalenkova. 2021. Agent System Mining: Vision, Benefits, and Challenges. *IEEE Access* 9 (2021), 99480–99494.

[139] Yathiraj B. Udupi and Munindar P. Singh. 2006. Contract Enactment in Virtual Organizations: A Commitment-Based Approach. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI)*. AAAI Press, Boston, 722–727.

[140] Wil Van Der Aalst. 2012. Process mining. *Commun. ACM* 55, 8 (2012), 76–83.

[141] M. Birna van Riemsdijk, Louise A. Dennis, Michael Fisher, and Koen V. Hindriks. 2013. Agent Reasoning for Norm Compliance: A Semantic Approach. In *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-Agent Systems* (St. Paul, MN, USA) *(AAMAS '13)*. IFAAMAS, Richland, SC, 499–506.

[142] Ruben Verborgh. 2018. Decentralizing the Semantic Web through incentivized collaboration. In *Proceedings of the 17th International Semantic Web Conference: Blue Sky Track (CEUR Workshop Proceedings)*, Marieke van Erp, Medha Atre, Vanessa Lopez, Kavitha Srinivas, and Carolina Fortuna (Eds.), Vol. 2189. CEUR-WS.org, Aachen, DEU, 1–5.

[143] Serena Villata and Fabien Gandon. 2012. Licenses Compatibility and Composition in the Web of Data. In *Proceedings of the Third International Conference on Consuming Linked Data - Volume 905* (Boston, MA) *(COLD'12)*. CEUR-WS.org, Aachen, DEU, 124–135.

[144] D. Walton and E.C.W. Krabbe. 1995. *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning*. State University of New York Press, New York, NY, USA.

[145] Gerhard Weiss. 1999. *Multiagent systems: a modern approach to distributed artificial intelligence*. MIT press, Cambridge, Massachusetts, United States.

[146] Danny Weyns, Andrea Omicini, and James Odell. 2007. Environment as a First Class Abstraction in Multiagent Systems. *Journal of Autonomous Agents and Multi-Agent Systems (JAAMAS)* 14, 1 (Feb. 2007), 5–30.

[147] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. 2016. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific data* 3, 1 (2016), 1–9.

[148] Michael Wooldridge and Nicholas R. Jennings. 1995. Intelligent Agents: Theory and Practice. *Knowledge engineering review* 10, 2 (1995), 115–152.

[149] Michael J. Wooldridge. 2001. *Introduction to Multiagent Systems*. John Wiley & Sons, Inc., USA.

4. Machine understandable policies and GDPR compliance checking

Bibliographic Information

Bonatti, P.A., **Kirrane, S.**, Petrova, I.M. and Sauro, L., 2020. Machine understandable policies and GDPR compliance checking. KI-Künstliche Intelligenz, 34, pp.303-315.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Formal analysis, Investigation, Writing - Original Draft, Writing - Review & Editing, Project administration, and Funding acquisition.

Copyright Notice

Noname manuscript No.
(will be inserted by the editor)

Machine Understandable Policies and GDPR Compliance Checking

Piero A. Bonatti · Sabrina Kirrane · Iliana M. Petrova · Luigi Sauro

Received: 31st October 2019 / Accepted: tba

Abstract The European General Data Protection Regulation (GDPR) calls for technical and organizational measures to support its implementation. Towards this end, the SPECIAL H2020 project aims to provide a set of tools that can be used by data controllers and processors to automatically check if personal data processing and sharing complies with the obligations set forth in the GDPR. The primary contributions of the project include: (i) a policy language that can be used to express consent, business policies, and regulatory obligations; and (ii) two different approaches to automated compliance checking that can be used to demonstrate that data processing performed by data controllers / processors complies with consent provided by data subjects, and business processes comply with regulatory obligations set forth in the GDPR.

1 Introduction

The European General Data Protection Regulation (GDPR), which came into force on the 25th of May 2018, defines legal requirements concerning the processing and sharing of personally identifiable data. In addition, the legislation calls for technical and organizational measures to support its implementation.

When it comes to legal informatics there is a large body of work on legal knowledge representation and reasoning (cf., [2, 4, 11, 16, 19, 21]), however said approaches are usually foundational in nature and as such are not readily accessible for companies looking for technical means to demonstrate GDPR compliance.

Recently we have seen the emergence of GDPR compliance tools (cf., [1, 12, 17, 18]) in the form of predefined questionnaires that enable data controllers and processors to assess the compliance of services and products that process personal data. The primary limitation of said tools is their lack of support for automated compliance checking.

In order to fill this gap, SPECIAL builds upon a rich history of policy language research from the Semantic Web community (cf., [7, 13, 14, 25, 26]), and shows how together machine understandable policies and automated compliance checking can be used to demonstrate compliance with legal requirements set forth in the GDPR.

In particular, we introduce the SPECIAL policy language and discuss how it can be used to express consent, business policies, and regulatory obligations. In addition, we describe two different approaches to automated compliance checking used to demonstrate that: (i) data processing performed by data controllers / processors complies with consent provided by data subjects; and (ii) business processes comply with regulatory obligations set forth in the GDPR. In addition, we provide a highlevel overview of our compliance checking algorithm and present the results of our initial performance evaluation.

The remainder of the paper is structured as follows: *Section* 2 describes our analysis of the text of the GDPR. *Section* 4 introduces the SPECIAL policy language, which provides a machine understandable encoding of consent. *Section* 5 discusses how the SPECIAL policy language can be used to encode business policies and regulatory obligations. *Section* 6 presents our compliance checking algorithm and the results of our initial performance evaluation. *Section* 7 points to related work on GDPR compliance. Finally, we present our conclusions and interesting directions for future work in *Section* 8.

Piero Bonatti · Iliana M. Petrova · Luigi Sauro
Università di Napoli Federico II, Naples, Italy
E-mail: pab@unina.it

Sabrina Kirrane
Vienna University of Economics and Business, Vienna, Austria
E-mail: sabrina.kirrane@wu.ac.at

2 Requirements Analysis

The GDPR, which came into effect on the 25th of May 2018, superceeds the Data Protection Directive 95/46/EC [8]. Given that primary goal of the SPECIAL H2020 project is to provide a set of tools that can be used by data controllers and processors to automatically check if personal data processing and sharing complies with the obligations set forth in the GDPR, we specifically focus on personal data processing that is performed after the GDPR came into effect. A necessary first step in this regard is to better understand the text of the GDPR, its interpretation by legal professionals, and the role of machine understandable representations, and automated compliance checking.

2.1 GDPR Analysis

Legal rules are composed of several constructs, prohibitions (used to describe what is not permitted), permissions (used to describe what is permitted), obligations (used to describe requirements that must be fulfilled), and dispensations (used to describe exemptions), commonly referred to as deontic concepts. In addition to these common constructs, the legal language contains constraints (used to limit the scope of permissions, prohibitions, obligations and dispensations), definitions (used to establish meaning), dispositions (used to highlight best practices/ suggestions), and opening clauses (used to indicate the need to consult National or European legislation).

In this paper, we propose a policy language that can be used to represent regulative norms in the form of permissions, obligations, and prohibitions. The analysis presented in this section serves to better understand the structure of the GDPR such that it is possible to identify fragments of the legislation that can be modeled in a manner that supports automated compliance checking of personal data processing and sharing with and between companies. Although constitutive norms could be used to provide requirements to organizations with respect to the processing of personal data within and between organizations, instead, we provide companies with a policy language that can be used to model data processing and sharing requirements and to simply attest to the existence or required controls, procedures, and documentation. The development of a fully fledged legal reasoning system, based on the modelling of normative requirements prescribed in legal text (c.f., [9, 19, 20]) is outside of the scope of the SPECIAL project.

When it comes to encoding legislative requirements using machine understandable representations, such that it is possible to perform automated compliance, major considerations include:

Connectedness of the various articles, paragraphs, and points, which can either explicitly refer to another piece of legislation (e.g., *"scientific or historical research purposes or statistical purposes in accordance with Article 89(1)"*) or implicitly to knowledge about the law (e.g., *"Personal data shall be: (a) processed lawfully, fairly and in a transparent manner in relation to the data subject ('lawfulness, fairness and transparency')"*). In either case, from an automated compliance checking perspective, it is clear that legal requirements are not separate and distinct rules but rather rules need to be linked, clustered, and/or generalized in a manner that enables the validation of a combination rules.

In SPECIAL we do not try to encode the entire GDPR, but rather focus on encoding legislative obligations (relating to several articles, paragraphs, and points) such that: (i) data processing performed by data controllers / processors complies with consent provided by data subjects; and (ii) business processes comply with regulatory obligations set forth in the GDPR.

Temporal expressions provide contextual information that is relevant for the interpretation of actions that need to be taken. Several different types of temporal expressions can be found in the text of the GDPR, for instance:

- ... *"the right to withdraw his or her consent at any time"* (Article 7 paragraph 3);
- ... *"processing based on consent before its withdrawal"* (Article 7 paragraph 3, Article 13 paragraph 2, Article 14 paragraph 2);
- ... *"prior to giving consent"* (Article 7 paragraph 3);
- ... *"at the time when personal data are obtained"* (Article 13 paragraphs 1 and 2);
- ... *"the personal data shall no longer be processed"* (Article 21 paragraph 3);

In SPECIAL we provide support for such temporal requirements by recording in a suitable *transparency ledger* when consent was obtained or when the data processing/sharing happened. This information is used for both ex-ante and ex-post compliance checking (as well as other purposes, discussed later).

2.2 Legal Interpretations

The GDPR defines several potential legal bases (consent, contract, legal obligation, vital interest, public interest, exercise of official authority, and legitimate interest) under which companies can legally process personal data. In order to determine if personal data processing is legally valid, the legal inquiry process usually involves gathering specific information such as: (i) the *personal data* collected from the data subject; (ii) the *processing* that are performed on the personal data; (iii) the *purpose* of such processing;

(iv) where data are *stored and for how long*; and (v) with whom data is *shared*. The answers provided to said questions enable legal professionals to determine which articles need to be consulted in order both to assess the lawfulness of processing and to identify relevant legal obligations.

Although, the open textured nature of legal texts is a highly desirable feature, as it leaves room for interpretation on a case by case basis, such ambiguity poses challenges for automatic compliance checking. In terms of legal interpretations, legal professionals also need to interpret the facts of the case with respect to relevant National or European legislation (e.g., opening clauses) and subjective terms (e.g., single words or parts of a sentence that can be interpreted in various ways). Here legal knowledge graphs could potentially play a crucial role as they allow for the modeling of both legislation and cases in a machine readable format, based on standardization activities such as European Law Identifier (ELI) and the European Case Law Identifier (ECLI), which provide technical specifications for web identifiers and vocabularies that can be used to describe metadata pertaining to legal documents. Such a legal knowledge graph could be used not only to identify case specific legislation, but also to uncover if there have been any prior cases that could be used to reduce ambiguity.

The SPECIAL poly language has been developed together with legal professionals who well versed in the interpretation of legal texts. Going forward we envisage that legal knowledge graphs could be used to reduce subjectivity thus allowing us to perform automated compliance checking for a broader set of legislative requirements.

2.3 Machine Understandable Representations

The GDPR poses at least two requirements that call for a machine-understandable representation of data usage modalities. Article 30 states that each controller shall maintain a record of the personal data processing activities under its responsibility. The first paragraph specifies that such a ledger should describe (among other information) the following aspects of *data usage*:

P1. the *purpose* of processing;
P2. a description of the *categories of data subjects* and of the *categories of personal data*;
P3. the categories of *recipients* to whom the personal data have been or will be disclosed;
P4. *transfers* of personal data to a third country or an international organization (since cross-border data transfer are subject to limitations);
P5. the envisaged *time limits for erasure* of the different categories of data;
P6. information about the *processing*, such as the security measures mentioned in Article 32.

Recital 42 stresses that, where processing is based on the data subject's consent, the controller should be able to demonstrate that the data subject has given consent to the processing operation.

SPECIAL addresses this issue by recording consent in the transparency ledger (cf. Sec. 2.1). The description of consent is similar to the description of processing activities as per Article 30. While Article 6.1.(a) – that introduces consent as a legal basis for personal data processing – and Recital 42 explicitly mention only the purpose of processing, Articles 13 and 14 add the other elements P2–P6 listed above. Concerning P6 (processing), it should be specified whether any automated decision making is involved, including profiling.

2.4 Automated Compliance Checking

Once such data usage descriptions are encoded in a machine-understandable way, several tasks, related to GDPR compliance, can be automated, including:

T1. Checking whether the processing complies with several restrictions imposed by the GDPR, such as additional requirements on the processing of sensitive data, restrictions on cross-border transfers, and compatibility of data usage with the chosen legal basis. This kind of validation requires a machine-understandable formalization of the relevant parts of the GDPR.
T2. Checking whether a specific operation is permitted by the available consent.
T3. Running ex-post auditing on the controller's activities. In SPECIAL this task is supported by logging data processing events in the transparency ledger, and comparing such events with consent.
T4. Finding the consent that justifies a specific processing (for auditing or responding to a data subject's inquiry).

The transparency ledger is also used in SPECIAL to provide dashboards to data subjects, that support them in monitoring the use of their data and *explaining* why their consent allowed specific operations. Such dashboards can also be used as a uniform interface to let data subjects exercise their rights (access to data, right to erasure, etc.) as specified by Articles 15–18 and 21–22.

3 Background

This section provides the necessary background information on RDF, RDFS and OWL. We start by describing the RDF data model concepts. We subsequently discuss the role played by RDF Schema and OWL when it comes to data modelling and reasoning.

The RDF data model was designed to facilitate data sharing and reuse. RDF *vocabularies* (otherwise known as *ontologies*) are collections of RDF triples that can be used to describe both schema and instance data. The `RDF` vocabulary is used to encode basic information pertaining to RDF, such as RDF type. Whereas, `FOAF`[1] is a well known vocabulary that is used to describe people and social relationship on the Web. Vocabularies are often placed in a common *namespaces*. For convenience *prefixes* are used as a shorthand notation for namespaces.

In RDF there is a tight coupling between the schema and instance data. *RDFSchema* (RDFS) [**?**] is a set of classes and properties used to describe RDF data. RDFS does not describe the structure of an RDF graph, but rather provides a framework used by vocabularies to denote classes, properties and relations. RDFS is composed of a set of classes that are used to define types of resources and properties that are used to describe these resources. Using the RDFS vocabulary it is possible to define class and property relations, similar to object oriented programming.

Generally speaking, RDF is used to express binary predicate relations. Whereas RDFS is used to define the domain and the range of these properties, hierarchies of classes and hierarchies of properties. However, using RDFS, it is not possible to represent complex sentences that include cardinality constraints or to model complex relations between classes, such as disjointness or equivalence.

OWL [**?**] is related to a family of logics known as Description Logics (DL). Like RDFS, OWL uses classes and properties (commonly called roles) and instances (known as individuals). However, OWL provides for:

- A rich set of relations between classes, roles and individuals (for example, `owl:sameAs`, `owl:equivalentClass`, `owl:differentFrom`).
- A number of logical operations (for example, `owl:unionOf`, `owl:intersectionOf` and `owl:complementOf`.
- Several cardinality constraints (for example, `owl:someValuesFrom`, `owl:allValuesFrom`, `owl:cardinality`, `owl:minCardinality` and `owl:maxCardinality`).

When it comes to reasoning, in an attempt to balance expressivity and efficiency, the specification presents three sublanguages (OWL Full, OWL DL and OWL Lite), otherwise known as *species* of OWL.

- OWL Full, which includes both OWL DL and OWL Lite, is the most expressive. It is the only OWL language, which supports RDFS. However, it is in general undecidable and it is not supported by the majority of software vendors.
- OWL DL, which includes OWL Lite, is less expressive than OWL Full. A number of restrictions are imposed:
  - it is not permissible to use `rdfs:Class` or `rdfs:Property`;
  - there must be a clear separation between individuals, classes, roles and datatypes; and
  - limitations are applied to several roles.

  As a result OWL DL is decidable and has a worst case computational complexity of NExpTime. In addition, it is fully supported by most software tools.
- OWL Lite, which has a worst case computational complexity of ExpTime, is the least expressive. Several restrictions, in addition to those specified for OWL DL, are imposed:
  - it is not possible to use `rdfs:Class` or `rdfs:Property`;
  - cardinality constraints are limited to 0 and 1;
  - in certain situations unnamed classes cannot be used; and
  - a number of additional limitations are applied to roles.

The second version of the language OWL2 was released as a W3C recommendation in 2012 [**?**]. OWL2 comes in two flavours (OWL2 Full and OWL2 DL). OWL2 DL is composed of three profiles (OWL2EL, OWL2QL and OWL2RL) that are based on well used DL constructs. OWL2 extends OWL with

- additional datatypes,
- additional annotations and
- relaxes the strict separation between classes, properties and annotations.

For specific details on OWL, the reader is referred to the OWL [**?**] and OWL2 [**?**] specification documentation.

4 Consent Compliance Checking

Although there are several potential legal bases that could be used to lawfully process personal data, in SPECIAL we have a particular focus on consent. Thus in this section we present the SPECIAL policy language and demonstrate how it can be used to encode consent in a manner than enables automated compliance checking.

4.1 Encoding Usage Descriptions and Consent

The common structure of the activity records and of the consent forms, consisting of properties P1–P6, is called *simple (usage) policy* in SPECIAL. In general, both the controller's activities and the consent of data subjects can be described by a *set* of simple usage policies (covering different data categories and purposes), called *full (usage) policies*. Each

[1] FOAF Vocabulary Specification, http://xmlns.com/foaf/spec/.

simple policy can be specified simply by attaching to each property P_i (such as purpose, data category, recipients, etc.) a term selected from a suitable *vocabulary* (ontology).

Example 1 A company – call it BeFit – sells a wearable fitness appliance and wants (i) to process biometric data (stored in the EU) for sending health-related advice to its customers, and (ii) share the customer's location data with their friends. Location data are kept for a minimum of one year but no longer than 5; biometric data are kept for an unspecified amount of time. In order to do all this legally, BeFit needs consent from its customers. Consent can be represented with two simple policies, specified using SPECIAL's vocabularies:

```
{
  has_purpose: FitnessRecommendation,
  has_data: BiometricData,
  has_processing: Analytics,
  has_recipient: BeFit,
  has_storage: { has_location: EU }
}

{
  has_purpose: SocialNetworking,
  has_data: LocationData,
  has_processing: Transfer,
  has_recipient: DataSubjFriends,
  has_storage: {
      has_location: EU,
      has_duration: [1year,5year]
      }
}
```

If `HeartRate` is a subclass of `BiometricData` and `ComputeAvg` is a subclass of `Analytics`, then the above consent allows BeFit to compute the average heart rate of the data subject in order to send her fitness recommendations. BeFit customers may restrict their consent, e.g. by picking a specific recommendation modality, like "recommendation via SMS only". Then the first line should be replaced with something like:

```
has_purpose:{
  FitnessRecommendation,
  contact: SMS}
```

Moreover, a customer of BeFit may consent to the first or the second argument of the union, or both. Their consent would be encoded, respectively, with the first simple policy, the second simple policy, or both. Similarly, each single process in the controller's business application may use only biometric data, only location data, or both. Accordingly, it may be associated to the first simple policy, the second simple policy, or both. ∎

The temporary exemplifying policy language vocabularies reported in SPECIAL's deliverables have been obtained by adapting previous standardized terms introduced by initiatives related to privacy and digital rights management, such as P3P[2] and ODRL,[3]. More refined vocabularies have been recently proposed by W3C's *Data Privacy Vocabularies and Controls Community Group*, (DPVCG) [22], promoted by SPECIAL and spanning a range of stakeholders wider than the project's consortium. The current vocabularies can be found on DPVCG's website[4].

As shown in Example 1, usage policies can be formatted with a minor extension of JSON (in particular, compound terms and policy sets require additional operators), while vocabularies can be encoded in RDFS or lightweight profiles of OWL2 such as OWL2-EL and OWL2-QL.

A grammar for SPECIAL policy expressions in Backus–Naur form (BNF) format is presented in Figure 1. The categories DataVocabExpression, PurposeVocabExpression, ProcessingVocabExpression, RecipientVocabExpression, LocationVocabExpression, DurationVocabExpression are specified by DPVCG's vocabularies.

4.2 Compliance Checking

Internally, SPECIAL's components encode also policies and the entries of the transparency ledger with a fragment (profile) of OWL2 called $\mathcal{PL}$ (policy logic) [6]. The adoption of a logic-based description language has manifold reasons. First, it has a clean, unambiguous semantics, that is a must for policy languages. A formal approach brings the following advantages:

- strong correctness and completeness guarantees on the algorithms for permission checking and compliance checking;
- the mutual coherence of the different reasoning tasks related to policies, such as policy validation, permission checking, compliance checking, and explanations (cf. tasks T1–T4 and the subsequent paragraph);
- correct usage after data is transferred to other controllers (i.e. interoperability). When it comes to so-called *sticky policies* [23], that constitute a sort of a license that applies to the data released to third parties, it is essential that all parties understand the sticky policy in the same way.

Policies are modeled as OWL2 *classes*. If the policy describes a controller's activity, then its instances represent all the operations that the controller may possibly execute. If the policy describes consent, then its instances represent all the operations permitted by the data subject. A description of (part of) the controller's activity – called *business policy* in SPECIAL (possibly represented as a *transparency log entry*) – *complies* with a consent policy if the former is a subclass of the latter, that is, all the possible operations described by the business policies are also permitted by the given consent.

[2] http://www.w3.org/TR/P3P11

[3] https://www.w3.org/TR/odrl/

[4] www.w3.org/community/dpvcg/

Fig. 1: SPECIAL's Usage Policy Language Grammar

UsagePolicy :=’ObjectUnionOf’ ’(’ **BasicUsagePolicy**
{ **BasicUsagePolicy** }* ’)’
| **BasicUsagePolicy**

BasicUsagePolicy :=’ObjectIntersectionOf’ ’(’ **Data Purpose Processing Recipients Storage** ’)’

Data :=’ObjectSomeValueFrom’ ’(’ ’spl:hasData’ **DataExpression** ’)’

Purpose :=’ObjectSomeValueFrom’ ’(’ ’spl:hasPurpose’ **PurposeExpression** ’)’

Processing :=’ObjectSomeValueFrom’ ’(’ ’spl:hasProcessing’ **ProcessingExpression** ’)’

Recipients :=’ObjectSomeValueFrom’ ’(’ ’spl:hasRecipient’ **RecipientExpression** ’)’

Storage :=’ObjectSomeValueFrom’ ’(’ ’spl:hasStorage’ **StorageExpression** ’)’

DataExpression :=’spl:AnyData’ | **DataVocabExpression**

PurposeExpression :=’spl:AnyPurpose’ | **PurposeVocabExpression**

ProcessingExpression :=’spl:AnyProcessing’ | **ProcessingVocabExpression**

RecipientsExpression :=’spl:AnyRecipient’ | ’spl:Null’ | **RecipientVocabExpression**

StorageExpression :=’spl:AnyStorage’ | ’spl:Null’ |
’ObjectIntersectionOf’ ’(’ **Location Duration** ’)’

Location :=’ObjectSomeValueFrom’ ’(’ ’spl:hasLocation’ **LocationExpression** ’)’

Duration :=’ObjectSomeValueFrom’ ’(’ ’spl:hasDuration’ **DurationExpression** ’)’
| ’DataSomeValueFrom’ ’(’ ’spl:durationInDays’ **IntervalExpression** ’)’

LocationExpression :=’spl:AnyLocation’ | **LocationVocabExpression**

DurationExpression :=’spl:AnyDuration’ | **DurationVocabExpression**

IntervalExpression :=’DatatypeRestriction’ ’(’ ’xsd:integer’ **LowerBound UpperBound** ’)’

LowerBound :=’xsd:minInclusive’ **IntegerLiteral**

UpperBound :=’xsd:maxInclusive’ **IntegerLiteral**

IntegerLiteral := **stringOfDigits** ’^^’ ’xsd:integer’

stringOfDigits := *a sequence of digits enclosed in a pair of ” (U+22)*

Fig. 2: SPECIAL's Business Policy Language Grammar

BusinessPolicy := **BasicBP** |
’ObjectUnionOf’ ’(’ **BasicBP** { **BasicBP** }* ’)’

BasicBP :=’ObjectIntersectionOf’ ’(’ **Data Purpose Processing Recipients Storage {Duty}* {LegalBasis}** ’)’

Data := *see* Section *4*

Purpose := *see* Section *4*

Processing := *see* Section *4*

Recipients := *see* Section *4*

Storage := *see* Section *4*

Duty :=’ObjectSomeValuesFrom’ ’(’ ’sbpl:hasDuty’ **DutyExpression** ’)’

DutyExpression :=’sbpl:AnyDuty’ | **DutyVocabExpression**

LegalBasis :=’ObjectSomeValuesFrom’ ’(’ ’sbpl:hasLegalBasis **LegalBasisVocabExpression** ’)’

Example 2 Consider again Example 1. The JSON-like representation used there can be directly mapped onto an OWL2 class ObjectUnionOf(P_1 P_2), where P_2 is[5]:

```
ObjectIntersectionOf(
  ObjectSomeValueFrom(
    has_purpose SocialNetworking )
  ObjectSomeValueFrom(
    has_data LocationData)
  ObjectSomeValueFrom(
    has_processing Transfer)
  ObjectSomeValueFrom(
    has_recipient DataSubjFriends)
  ObjectSomeValueFrom(
    has_storage ObjectIntersectionOf(
      ObjectSomeValueFrom(has_location: EU)
      DataSomeValueFrom(has_duration
        DatatypeRestriction(xsd:integer
          xsd:minInclusive "365"^^xsd:integer
          xsd:maxInclusive "1825"^^xsd:integer
)))
```

In order to check whether a business policy BP (encoded as an OWL2 class) complies with the above policy one should check whether the former is a subclass of the latter, that is, whether:

SubClassOf(BP ObjectUnionOf(P_1 P_2))

is a logical consequence of the ontology that defines SPECIAL's vocabularies. ∎

5 Business Processes Compliance Checking

Beyond consent, the GDPR defines obligations that apply to the data controllers / processors internal systems and processes. Here are two examples:

[5] We omit P_1 due to space limitations; the reader may easily derive it by analogy with the above example.

- whenever the data controller operates on personal data, it must *acquire explicit consent* from the involved data subjects, unless the purpose of data processing falls within a set of exceptional cases (e.g. the processing is required by law); cf. Article 6.1, (b)–(f);
- whenever data are transferred to a third country whose data protection regulations do not match the EU requirements, alternative guarantees must be provided, e.g. in the form of company regulations called *binding corporate rules*, cf. Article 47 and, more generally, GDPR Chapter V (Transfer Of Personal Data To Third Countries Or International Organisations).

Moreover, and differently from the above examples, the GDPR sets obligations that are not directly related to the controller's business processes, such as the requirement that data subjects have the right to *access, rectify, and delete* their personal data. In order to fulfill such obligations, data controllers have to set up suitable processes. Last but not least, it is useful to label the controller/processors processes with the legal basis for the processing; this helps in assessing and demonstrating the lawfulness of data processing activities. For automated compliance checking descriptions of internal systems and processes should be adequately formalized in a machine-understandable way; moreover, the formalization should represent accurately the real processes, in order to make the automated compliance verification reliable.

5.1 Encoding Business Processes as Policies

In SPECIAL, we address a concrete setting in which a partial and abstract description of processes is available. Each process description is shaped like a *formalized business policy* consisting of the following set of features:

- the file(s) to be processed;
- the software that carries out the processing;
- the purpose of the processing;
- the entities that can access the results of the processing;
- the details of where the results are stored and for how long;
- *the obligations that are fulfilled while (or before) carrying out the processing;*
- *the legal basis of the processing.*

It is not hard to see that the first five elements in the above list match SPECIAL's usage policy language (UPL) introduced in *Section* 4. As far as the above elements are concerned, the only difference between UPL expressions and a business policy is the granularity of attribute values. For example, the involved data (specified in the first element of the above list) are not expressed as a general, content-oriented category, but rather as a concrete set of data sources or data items. Such objects can be modeled as instances or subclasses of the general data categories illustrated in *Section* 4, thereby creating a link between digital artifacts and usage policies. Similar considerations hold for the other attributes:

- processing is not necessarily described in the abstract terms adopted by the processing vocabulary introduced in *Section* 4; in a business policy, this can be specified by naming concrete software procedures;
- the purpose of data processing may be directly related to the data controller's mission and products;
- recipients may consist of a concrete list of legal and/or physical persons, as opposed to general categories such as `Ours` or `ThirdParty`;
- storage may be specified by a list of specific data repositories, at the level of files and hosts.

With this level of granularity, specific authorizations can be derived from the business policy, for example:

> *The indicated software procedure can read the indicated data sources. The results can be written in the specified repositories. The specified recipients can read the repositories...*

This methodology for generating authorizations fosters a close correspondence between the business policy and the actual behavior of the data controller's systems and processes.

The attribute encoding obligations is not part of usage policies. It plays a dual role, representing:

- preconditions authorizations specified by the business policy, e.g. if the obligation is something like `getValidConsent` then the derived authorizations is a *rule* like *the specified software can read the data sources if consent has been given*;
- obligation assertions (under human responsibility) that the data controller has set up *processes for fulfilling the indicated obligations* – e.g. a process to obtain consent from the data subjects – which is relevant to checking compliance with the GDPR.

5.2 Business Policies in OWL2

A basic business policy is simply a usage policy (as in *Section* 4) extended with zero or more obligations, and a legal basis, encoded with attributes `hasDuty` and `hasLegalBasis`, for example the following policy associates the collection of personal demographic information to the obligations to get consent and let the data subject exercise her rights:

```
ObjectIntersectionOf(
   ObjectSomeValuesFrom
      (spl:hasData svd:Demographic)
   ObjectSomeValuesFrom
      (spl:hasProcessing svpr:Collect)
   ObjectSomeValuesFrom
      (spl:hasPurpose svpu:Account)
   ObjectSomeValuesFrom
      (spl:hasRecipient svr:Ours)
   ObjectSomeValuesFrom
      (spl:hasStorage
      ObjectIntersectionOf(
         spl:hasLocation svl:OurServers
         spl:hasDuration svdu:Indefinitely
      )
   )
   ObjectSomeValuesFrom
      (sbpl:hasDuty getValidConsent)
   ObjectSomeValuesFrom
      (sbpl:hasDuty getAccessReqs)
   ObjectSomeValuesFrom
      (sbpl:hasDuty getRectifyReqs)
   ObjectSomeValuesFrom
      (sbpl:hasDuty getDeleteReqs)
   ObjectSomeValuesFrom
      (sbpl:hasLegalBasis A6-1-a-explicit-consent)
)
```

Similarly to usage policies, *general* business policies can be composed by enclosing several basic business policies inside the `ObjectUnionOf` operator of OWL2. The syntax and the logical semantics of SPECIAL's Business Policy Language are specified in Figure 2. The values for attributes DutyVocabExpression and LegalBasisVocabExpression are specified in DPVCG's vocabularies.

5.3 Partial Encoding of the GDPR in OWL2

The GDPR cannot be fully axomatized due to the usual difficulties that arise in axiomatizing legal text (especially the frequent use of subjective terms as highlighted in *Section* 2). However it is possible to encode some constraints that should hold over the different attributes of a business policy. At the top level, the formalization is organized as follows:

```
ObjectUnionOf(
  ObjectIntersectionOf(
    Chap2_LawfulProcessing
    Chap3_RightsOfDataSubjects
    Chap4_ControllerAndProcessorObligations
    Chap5_DataTransfer
  )
  Chap9_Derogations
)
```

Informally, the above expression says that either the requirements of GDPR Chapters 1–5 are satisfied, or some of the derogations provided by GDPR Chapter 9 should apply. In turn, each of the above terms is equivalent to a compound OWL2 class that captures more details from the regulation. Here we illustrate part of the formalization of GDPR Chapter 2 for an example. `Chap2_LawfulProcessing` is equivalent to the following expression:

```
ObjectUnionOf(
   Art6_LawfulProcessing
   Art9_SensitiveData
   Art10_CriminalData
)
```

The above three conditions apply, respectively, to non-sensitive personal data, sensitive data, and criminal data. At least one of the three conditions should be satisfied. In turn, `Art6_LawfulProcessing` is defined as:

```
ObjectUnionOf(
   ObjectSomeValuesFrom(spl:hasData
     SensitiveData_as_per_Art9
   )
   ObjectSomeValuesFrom(spl:hasData
     CriminalConvictionData_as_per_Art10
   )
   Art6_1_LegalBasis
   Art6_4_CompatiblePurpose
   )
)
```

Roughly speaking, the above union represents an implication in disjunctive normal form, and should be read like this: if the data involved in the processing is neither sensitive nor criminal conviction data, then either the fundamental legal bases of Art. 6(1) apply, or the processing is compatible with the original purpose for collecting the data as per Art. 6(4). In order to capture this meaning, class `Art6_1` is defined as:

```
ObjectSomeValuesFrom(hasLegalBasis
   ObjectUnionOf(
     Art6_1_a_Consent
     Art6_1_b_Contract
     Art6_1_c_LegalObligation
     Art6_1_d_VitalInterest
     Art6_1_e_PublicInterest
     Art6_1_f_LegitimateInterest
   )
)
```

Roughly speaking, this definition means that a business policy satisfies the requirements of Art. 6(1) if it contains a clause

`ObjectSomeValueFrom( hasLegalBasis` X `)`

where X is some of the above classes corresponding to points *a–f* of Art. 6(1). In practice, this means that a human expert has to pick an appropriate legal basis for each business policy. Similarly, the formalization of Article 9 applies to sensitive data categories only, and requires a legal basis from a different list. So the term `SensitiveData_as_per_Art9` is equivalent to:

```
ObjectUnionOf(
   ObjectSomeValuesFrom(spl:hasData
     ObjectComplementOf(SensitiveData_as_per_Art9)
   )
   ObjectSomeValuesFrom(hasLegalBasis
     ObjectUnionOf(
       Art9_2_a_Consent
       Art9_2_b_EmploymentAndSocialSecurity
       Art9_2_c_VitalInterest
       Art9_2_d_LegitimateActivitiesOfAssociations
       Art9_2_e_PublicData
       Art9_2_f_Judicial
       Art9_2_g_PublicInteres
       Art9_2_h_PreventiveOrOccupationalMedicine
       Art9_2_i_PublicHealth
       Art9_2_j_ArchivingResearchStatistics
     )
   )
)
```

The rest of the regulation is formalized with a similar approach.

5.4 Compliance Checking

Let us now make an example of compliance checking of a business policy w.r.t. the above axiomatization. Consider the following business policy:

```
ObjectIntersectionOf(
   ObjectSomeValuesFrom( hasData Religion )
   ObjectSomeValuesFrom( hasProcessing Collect )
   ObjectSomeValuesFrom( hasPurpose
     PersonalisedBenefits )
   ObjectSomeValuesFrom( hasStorage
     ObjectSomeValuesFrom( hasLocation EU ))
   ObjectSomeValuesFrom( hasRecipient
     DataProcessor )
   ObjectSomeValuesFrom( hasDuty
     Art12-22_SubjectRights )
   ObjectSomeValuesFrom( hasDuty
     Art32-37_Obligations )
   ObjectSomeValuesFrom( hasLegalBasis
     Art6_1_a_Consent )
)
```

This policy is not a subclass of the formalized GDPR (hence it does not pass the compliance check) because Religion is classified as sensitive data (it is a subclass of SensitiveData_as_per_Art9). Then the business policy is not a subclass of Art9_SensitiveData, because the legal basis is not among the required list. Moreover, the business policy is not covered by the derogations provided by GDPR Chapter 9 (details are omitted here). As a consequence, the business policy does not satisfy the conditions specified by Chap2_LawfulProcessing. Note that this kind of compliance checking is able to verify the coherency of the different parts of a business policy.

If Religion was replaced by any non-sensitive data category such as Location, then the policy would be compliant because it would be a subclass of Art6_LawfulProcessing. This satisfies the condition called Chap2_LawfulProcessing. The hasDuty attributes of the business policy suffice to satisfy Chap3_RightsOfDataSubjects and Chap4 ControllerAndProcessorObligations. Chap5_DataTransfer would also be satisfied since the processing does not involve any transfers outside the EU.

6 Our Automated Compliance Checking Algorithm

Business policies (that describe the processing of each of the controller's processes) are not only needed to fulfill the requirements of Article 30. They can also be used to check whether a running process complies with the available consent, as a sort of access control system. Several implementation strategies are possible, depending on the controller's system architecture; to fix ideas, the reader may consider the following generic approach: Each of the controller's processes is labeled with a corresponding business policy that describes it, and before processing a piece of data, the business policy is compared with the data subject's consent to check whether the operation is permitted.

In general, such compliance checks occur frequently enough to call for a scalable implementation. Consider, for example, a telecom provider that collects location information to offer location-based services. Locations cannot be stored without a legal basis, such as law requirements or consent – not even temporarily, while a batch process selects the parts that can be legally kept. So compliance checking needs to be executed on the fly. In order to estimate the amount of compliance checks involved, consider that the events produced by the provider's base stations are approximately 15000 per second; the probing records of wi-fi networks are about 850 millions per day.

In order to meet such performance requirements, SPECIAL has developed ad-hoc reasoning algorithms for $\mathcal{PL}$ [6], that leverage $\mathcal{PL}$'s simplicity to achieve unprecedented reasoning speed. Compliance checking is split into two phases: first, business policies are normalized and closed under the axioms contained in the vocabularies; in the second phase, business policies are compared with consent policies with a *structural subsumption* algorithm. We have just completed the evaluation of a sequential Java implementation of those algorithms, called PLR. We chose Java to facilitate the comparison with other engines, by exploiting the standard OWL APIs, and we refrained to apply parallelization techniques in order to assess the properties of the basic algorithms. Before discussing more performant implementation options, we report the performance of PLR over the pilots

	Pilot 1	Pilot 2
Ontology		
inclusions	186	186
disjoint class axioms	11	11
property range axioms	10	10
functional properties	8	8
classification hierarchy height	4	4
Business policies		
# generated policies	120	100
avg. simple pol. per full pol.	2.71	2.39
Consent policies		
# generated policies	12,000	10,000
avg. simple pol. per full pol.	3.77	3.42
Test cases		
# compliance checks	12,000	10,000

Table 1: Test cases derived from SPECIAL's pilots

of SPECIAL.[6] The pilots share a common ontology that defines personal data categories, purposes, and the other features that are needed in usage policies, cf. SPECIAL's deliverable D2.5.[7] In the experiments, business policies are those developed for the pilots, while consent policies are generated randomly by simulating the opt-in and opt-out choices of data subjects, picked from the options provided by the data controllers.

PLR can pre-compute the first phase, since the business policies are known in advance and are typically persistent. So the runtime cost is reduced to structural subsumption. In this way, on the test cases derived from SPECIAL's use cases (cf. Table 1), the performance we achieve, respectively, is 150μsec and 190μsec per compliance check, using the following system:

processor:	Intel Xeon Silver 4110
cores:	8
cache:	11M
RAM:	198 GB
OS:	Ubuntu 18.4
JVM:	1.8.0_181
heap:	32 GB (actually used: less than 700 MB).

This means that PLR alone can execute about 6000 compliance checks per second and more than 518 million checks per day, that is, 60% of wi-fi probing events and 40% of base station events.

In order to raise performance up to the required levels, one can re-engineer PLR using a language more performant than Java, and/or parallelize processing by means of big data architectures. Compliance checking is particularly well suited to parallelization, since each test is independent from the others and no synchronization is required. Additionally, the investigation of parallelization within PLR's algorithms is under investigation.

7 Related Work

From a GDPR compliance perspective, there exist several compliance tools (cf. [1, 12, 17, 18]) that enable companies to assess the compliance of applications and business processes via predefined questionnaires.

There is also a large body of work on legal knowledge representation (cf.[4, 21]) and reasoning (cf. [2, 11, 16, 19]). From a representation perspective, Bartolini et al. [4] and Pandit et al. [21] propose ontologies that can be used to model data protection requirements. While, Palmirani et al. [19] and Athan et al. [2] demonstrate how LegalRuleML can be used to specify legal norms. The work by Lam and Hashmi [16] and Governatori et al. [11] also builds upon LegalRuleML, however the focus is more on ensuring business process compliance.

Both rule languages and OWL2 have already been used as policy languages; a non-exhaustive list is [7, 13, 14, 25, 26]. As noted in [5], the advantage of OWL2 – hence description logics – is that all the main policy-reasoning tasks are decidable (and tractable if policies can be expressed with OWL2 profiles), while compliance checking is undecidable in rule languages, or at least intractable – in the absence of recursion – because it can be reduced to datalog query containment. So an OWL2-based policy language is a natural choice in a project like SPECIAL, where policy comparison is the predominant task. Among the aforementioned languages, both Rei and Protune [7, 14] support logic program rules, which make them unsuitable to SPECIAL's purposes. KAoS [25] is based on a description logic that, in general, is not tractable, and supports role-value maps – a construct that easily makes reasoning undecidable (see [3], Chap. 5). The papers on KAoS do not discuss how to address this issue.

P3P's privacy policies – that are encoded in XML – and simple $\mathcal{PL}$ policies have a similar structure: the tag `STATEMENT` contains tags `PURPOSE`, `RECIPIENT`, `RETENTION`, and `DATA-GROUP`, that correspond to the analogous properties of SPECIAL's usage policies. Only the information on the location of data is missing. The tag `STATEMENT` is included in a larger context that adds information about the controller (tag `ENTITY`) and about the space of web resources covered by the policy (through so-called *policy reference files*). Such additional pieces of information can be directly encoded with simple $\mathcal{PL}$ concepts.

There exist several well-engineered reasoners for OWL2 and its profiles. Hermit [10] is a general reasoner for OWL2. Over the test cases inspired by SPECIAL's use cases, it takes 3.67ms and 3.96ms per compliance check, respectively, that is, over 20 times longer than PLR. ELK [15] is a specialized polynomial-time reasoner for the OWL2-EL profile. It

[6] We have also run sets of synthetic experiments with increasing size to assess the scalability of PLR. They are omitted here due to space limitation and will be published in a forthcoming paper. We anticipate that these experiments confirm that PLR is faster than its competitors.

[7] https://zenodo.org/record/2545177

does not support functional roles, nor the interval constraints used to model storage duration, therefore it cannot be used to reason on the $\mathcal{PL}$ profile. Konclude [24] is a highly optimized reasoner with "pay-as-you-go" strategies (i.e. it becomes more efficient on less complex profiles of OWL2). Konclude is designed for classification, and is currently not optimized for subsumption tests (i.e. the reasoning task underlying compliance checks). Consequently, it turns out to be slower than Hermit on our test cases.

8 Conclusion and Future Work

The overarching goal of the SPECIAL project is to develop tools and technologies that enable data controllers and processors to comply with personal data processing obligations specified in the GDPR. In this paper, we presented the SPECIAL policy language and discussed how it can be used to encode consent, business policies, and regulatory obligations. In addition we described the SPECIAL approaches to GDPR compliance checking and presented the results of our initial performance evaluation.

Ongoing/future work includes: the optimisation of the existing compliance checking algorithm to cater for automated compliance checking for a broader set of legislative requirements; and the development of an algebra that can be used to combine multiple policies, for instance where there is a need to aggregate data from multiple data sources.

Acknowledgment

This research is funded by the European Union's Horizon 2020 research and innovation programme under grant agreement N. 731601. The authors are grateful to all of SPECIAL's partners; without their contribution this project and its results would not have been possible.

References

1. S. Agarwal, S. Steyskal, F. Antunovic, and S. Kirrane. Legislative compliance assessment: Framework, model and gdpr instantiation. In *Annual Privacy Forum*, pages 131–149. Springer, 2018.
2. T. Athan, H. Boley, G. Governatori, M. Palmirani, A. Paschke, and A. Z. Wyner. Oasis LegalRuleML. In *ICAIL*, volume 13, pages 3–12, 2013.
3. F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*, 2003. Cambridge University Press. ISBN 0-521-78176-0.
4. C. Bartolini, R. Muthuri, and C. Santos. Using ontologies to model data protection requirements in workflows. In *JSAI International Symposium on Artificial Intelligence*, 2015.
5. P. A. Bonatti. Datalog for security, privacy and trust. In *Datalog Reloaded - First International Workshop, Datalog 2010, Oxford, UK, March 16-19, 2010. Revised Selected Papers*, pages 21–36, 2010. doi: 10.1007/978-3-642-24206-9_2.
6. P. A. Bonatti. Fast compliance checking in an OWL2 fragment. In J. Lang, editor, *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, pages 1746–1752. ijcai.org, 2018. ISBN 978-0-9992411-2-7. doi: 10.24963/ijcai.2018/241.
7. P. A. Bonatti, J. L. D. Coi, D. Olmedilla, and L. Sauro. A rule-based trust negotiation system. *IEEE Trans. Knowl. Data Eng.*, 22(11):1507–1520, 2010. doi: 10.1109/TKDE.2010.83.
8. E. Directive. 95/46/ec of the european parliament and of the council of 24 october 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data. *Official Journal of the EC*, 23(6), 1995.
9. F. Gandon, G. Governatori, and S. Villata. Normative requirements as linked data. In *The 30th international conference on Legal Knowledge and Information Systems (JURIX)*, 2017.
10. B. Glimm, I. Horrocks, B. Motik, G. Stoilos, and Z. Wang. Hermit: An OWL 2 reasoner. *J. Autom. Reasoning*, 53(3):245–269, 2014. doi: 10.1007/s10817-014-9305-1.
11. G. Governatori, M. Hashmi, H.-P. Lam, S. Villata, and M. Palmirani. Semantic business process regulatory compliance checking using LegalRuleML. In *European Knowledge Acquisition Workshop*, 2016.
12. Information Commissioner's Office (ICO) UK. Getting ready for the GDPR, 2017. URL `https://ico.org.uk/for-organisations/resources-and-support/data-protection-self-assessment/getting-ready-for-the-gdpr/`.
13. S. Jajodia, P. Samarati, M. L. Sapino, and V. S. Subrahmanian. Flexible support for multiple access control policies. *ACM Trans. Database Syst.*, 26(2):214–260, 2001.
14. L. Kagal, T. W. Finin, and A. Joshi. A policy language for a pervasive computing environment. In *4th IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY)*, pages 63–. IEEE Computer Society, June 2003. ISBN 0-7695-1933-4.
15. Y. Kazakov, M. Krötzsch, and F. Simancik. The incredible ELK - from polynomial procedures to efficient rea-

soning with EL ontologies. *J. Autom. Reasoning*, 53(1): 1–61, 2014. doi: 10.1007/s10817-013-9296-3.

16. H.-P. Lam and M. Hashmi. Enabling reasoning with LegalRuleML. *Theory and Practice of Logic Programming*, 19(1):1–26, 2019.
17. Microsoft Trust Center. Detailed GDPR Assessment, 2017. URL `http://aka.ms/gdprdetailedassessment`.
18. Nymity. GDPR Compliance Toolkit. URL `https://www.nymity.com/gdpr-toolkit.aspx`.
19. M. Palmirani, G. Governatori, A. Rotolo, S. Tabet, H. Boley, and A. Paschke. LegalRuleML: XML-based rules and norms. In *International Workshop on Rules and Rule Markup Languages for the Semantic Web*, pages 298–312, 2011.
20. M. Palmirani, M. Martoni, A. Rossi, C. Bartolini, and L. Robaldo. Pronto: Privacy ontology for legal reasoning. In *International Conference on Electronic Government and the Information Systems Perspective*, pages 139–152, 2018.
21. H. J. Pandit, K. Fatema, D. O'Sullivan, and D. Lewis. Gdprtext-gdpr as a linked data resource. In *European Semantic Web Conference*, pages 481–495, 2018.
22. H. J. Pandit, A. Polleres, B. Bos, R. Brennan, B. P. Bruegger, F. J. Ekaputra, J. D. Fernández, R. G. Hamed, E. Kiesling, M. Lizar, E. Schlehahn, S. Steyskal, and R. Wenning. Creating a vocabulary for data privacy - the first-year report of data privacy vocabularies and controls community group (DPVCG). In *On the Move to Meaningful Internet Systems: OTM 2019 Conferences - Confederated International Conferences: CoopIS, ODBASE, C&TC 2019, Rhodes, Greece, October 21-25, 2019, Proceedings*, pages 714–730, 2019.
23. S. Pearson and M. C. Mont. Sticky policies: An approach for managing privacy across multiple parties. *IEEE Computer*, 44(9):60–68, 2011. doi: 10.1109/MC.2011.225.
24. A. Steigmiller, T. Liebig, and B. Glimm. Konclude: System description. *J. Web Semant.*, 27-28:78–85, 2014. doi: 10.1016/j.websem.2014.06.003.
25. A. Uszok, J. M. Bradshaw, R. Jeffers, N. Suri, P. J. Hayes, M. R. Breedy, L. Bunch, M. Johnson, S. Kulkarni, and J. Lott. KAoS policy and domain services: Towards a description-logic approach to policy representation, deconfliction, and enforcement. In *4th IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY)*, pages 93–96. IEEE Computer Society, June 2003. ISBN 0-7695-1933-4.
26. T. Y. C. Woo and S. S. Lam. Authorizations in distributed systems: A new approach. *Journal of Computer Security*, 2(2-3):107–136, 1993. doi: 10.3233/JCS-1993-22-304.

5. User consent modeling for ensuring transparency and compliance in smart cities

Bibliographic Information

Fernández, J.D., Sabou, M., **Kirrane, S.**, Kiesling, E., Ekaputra, F.J., Azzam, A. and Wenning, R., 2020. User consent modeling for ensuring transparency and compliance in smart cities. Personal and Ubiquitous Computing, 24, pp.465-486.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Formal analysis, Investigation, Writing - Original Draft, and Writing - Review & Editing.

Copyright Notice

Personal and Ubiquitous Computing Journal manuscript No.
(will be inserted by the editor)

User Consent Modeling for Ensuring Transparency and Compliance in Smart Cities

Javier D. Fernández · Marta Sabou · Sabrina Kirrane · Elmar Kiesling · Fajar J. Ekaputra · Amr Azzam · Rigo Wenning

Received: date / Accepted: date

Abstract Smart city infrastructures such as transportation and energy networks are evolving into so-called *Cyber-Physical Social Systems* (CPSSs), which collect and leverage citizens' data in order to adapt services to citizens' needs. The privacy implications of such systems are, however, significant and need to be addressed. Current systems either try to escape the privacy challenge via anonymization or use very rigid, hard coded work flows that has been agreed with a data protection authority. In the case of the latter, there is a severe impact on data quality and richness, whereas in the former, only these hard coded flows are permitted resulting in diminished functionality and potential. We address these limitations via *user modeling* in terms of investigating how to model and semantically represent user consent, preferences and data usage policies that will guide the processing of said data in the data lake. Data protection is a horizontal field and consequently very wide. Therefore we focus on a concrete setting where we extend the domain-agnostic SPECIAL

Funded by Austrian Research Promotion Agency (FFG): grant no. 861213 (CitySPIN), the European Union's Horizon 2020 research and innovation programme grant 731601 (SPECIAL), the Christian Doppler Research Association, and the Austrian Federal Ministry for Digital & Economic Affairs and the National Foundation for Research, Technology and Development.

Javier D. Fernández, Sabrina Kirrane and Amr Azzam
Vienna University of Economics and Business, Vienna, Austria
E-mail: firstName.secondName@wu.ac.at

Marta Sabou and Elmar Kiesling
Technical University of Vienna, Vienna, Austria
E-mail: firstName.secondName@ifs.tuwien.ac.at

Fajar J. Ekaputra
Christian Doppler Laboratory for Security and Quality Improvement in the Production System Lifecycle (CDL-SQI), Technical University of Vienna, Vienna, Austria
E-mail: fajar.ekaputra@tuwien.ac.at

Rigo Wenning
ERCIM/W3C, Sophia-Antipolis, France
E-mail: rigo@w3.org

policy language for a smart mobility use case supplied by Vienna's largest utility provider. To that end (1) we create an extension of SPECIAL in terms of a core CPSS vocabulary that lowers the semantic gap between the domain agnostic terms of SPECIAL and the vocabulary of the use case; (2) we propose a workflow that supports defining domain specific vocabularies for complex CPSS systems; and (3) show that these two contributions allow successfully achieving the goals of our setting.

Keywords Cyber Physical (Social) Systems · Smart Mobility · User Consent Modeling · Privacy · GDPR · Linked Data

1 Introduction

Large-scale Smart City infrastructures such as smart transportation or smart energy networks typically span the boundaries of the physical, cyber and social spheres. Sensors in the physical world are used to collect (real-time) data, which is then processed pragmatically by a cyber-component to determine appropriate actuation/adaptation strategies. Increasingly, participants and users of these infrastructures provide data to the system (e.g., through social sensing) and can even act as actuators to optimize the system. Such complex systems, are referred to as *Cyber-Physical Social Systems* (CPSSs) [43].

Considering that CPSSs often make use of and integrate personal information from various sources, privacy protection needs to be at the core of such systems. The reality is often different with systems adopting a take-it-or-leave-it approach [16]. At a first glance, giving users the choice to accept or reject a request to participation in a CPSS seems to be an efficient and simple solution. But such an approach regularly attracts harsh criticism. Once trust is sufficiently eroded via this simple approach, people will reject all those systems by default. It is therefore important to earn and sustain the trust into CPSSs. This philosophy is also underlying the framework that the EU has laid down with the creation of the GDPR.

The GDPR defines a set of obligations for controllers and processors of personal data, including, but not limited to, having a lawful reason for processing personal data and providing full transparency to data subjects with respect to the processing of their personal data. Several tools [22,28,31] that assist companies in assessing their compliance with the GDPR have recently been developed. They are, however, targeted at self-assessment (i.e., companies complete standard questionnaires in the form of privacy impact assessments). The self assessment is used to check a pre-set and fixed work flow against the legal rules. The challenge for CPSSs is to allow for a maximum amount of data to be collected with a maximum of use and re-use permissions granted by data subjects. Confronted with the resulting large amount of legalese that comes with such an approach, privacy scholars start to talk about the end of data self determination[19]. Despite the nominal transparency, data subjects and regulators alike are just overwhelmed by the complexity of CPSSs.

The EU H2020 SPECIAL[1] project, strives to enable companies to work with the data subjects to sustain trust in complex systems, such as CPSSs. With the SPECIAL system, preferences, consent, and legal grounds for processing can be integrated at run time into a CPSS. Because of the semantics involved, the CPSS becomes privacy-aware. Algorithms in the CPSS can react on privacy concerns within the system with a high degree of flexibility. This serves the data controllers and the data subjects alike. Data self-determination allows data subjects to participate in value creation via those CPSSs. The legal or technical term for such participation is consent. If other grounds for processing are used, the quest for trust imposes a high level of digestible transparency, here again semantics plays a critical role.

In this context, the data controller is challenged to make sure that personal data processing actually conforms to the promises made to the data subject. This is of increased importance since the fines for misbehaviour have become significant with the advent of the GDPR. If the data subject sets a preference in the CPSS, e.g. via his mobile device, the CPSS needs to make sure said preference are followed in the subsequent complex work flows that sometimes even transcend organizational borders. The idea here is to enable the system to automatically ingest and interpret the preferences without having a programmer setting switches. The challenge is then to manage the data flows. The SPECIAL approach addresses this challenge by attaching semantics to personal data that specifies possible usages, in the form of a *usage policy*. The SPECIAL engine is capable of using those semantics in order to perform both ex-ante and ex-post compliance checking, and to provide digestible transparency to data subjects concerning what happened to their personal data, why and when.

In this paper we analyze the suitability of the SPECIAL policy language for CPSS user consent modeling and showcase its extension to cover the needs of the *Smart Mobility* use cases provided by Vienna's largest utility provider, Wiener Stadtwerke (WStW). To that end, we adopt a three stage approach. First, we create an extension of SPECIAL with a vocabulary that is generically applicable to CPSS use cases (the SPECIAL-CPSS core vocabulary) and introduces a set of CPSS-specific terms, thus lowering the effort of extending the domain-agnostic policy language to the need of concrete use cases. The core vocabulary is grounded on an overview of CPSS systems obtained with a literature review, and as such aims to be reusable across various CPSS domains, beyond smart mobility. Second, we propose a practical work flow to support CPSS owners in analyzing their complex systems and deriving user consent modeling vocabularies needed for their use cases. Third, we check the usefulness of these two contributions by using them in the context of the WStW's smart mobility use case and successfully deriving use case specific usage policies. In a nutshell, the novelty of this paper lies in a non-trivial extension of the semantics used within Cyber-Physical-Systems in order to

[1] `https://www.specialprivacy.eu/`

prepare a much more sophisticated approach to data protection and GDPR compliance for CPSS. Concretely, we make the following contributions:

- the SPECIAL-CPSS core vocabulary, which serves as a means for describing usage constraints across a variety of CPSS domains and are usable within a SPECIAL – like system;
- the practical work flow, which enables CPSS preference, constraint and consent modeling in general;
- a practical use case to show how these techniques can be applied in a CPSS setting.

The remainder of this paper is organized as follows. In Section 2 we present the state of the art in policy languages and GDPR transparency and compliance. Then, in Section 3 we describe the main components of the SPECIAL consent, transparency and compliance framework, paying particular attention to the SPECIAL usage policy language and vocabularies, and the methodology used to extend SPECIAL in order to cater for CPSSs. The proposed extension is motivated and guided by our CitySPIN use cases, presented in Section 4. Our core CPSS vocabulary is introduced in Section 5. Section 6 presents a workflow to establish data subjects' consent and data usage policies for specific use cases. This workflow is validated using our CitySPIN use cases in Section 7. Finally, we conclude and present future work in Section 8.

2 Related Work

The European General Data Protection Regulation (GDPR) requires data controllers to obtain explicit consent for the processing of personal data from data subjects. Traditionally, this consent is obtained via a human-readable description (i.e., a *contract*, or *terms and conditions*), which does not allow for any automatic processing. Thus, formal policy languages are designed to unambiguously represent usage policies, which makes it possible to automatically verify whether data processing is covered by data subjects' given consent. In the following, we first review current alternative policy languages (Section 2.1) and GDPR transparency and compliance tools (Section 2.2).

2.1 Usage Policies

There are several potential candidates for the formal representation of usage policies, including semantic policy languages [42,23,7,25] and standard based policy languages [13,21]. KAoS [42] is a general policy language which adopts a pure ontological approach, whereas Rei [23] and Protune [7] use ontologies to represent concepts, the relationships between these concepts and the evidence needed to prove their truth, and rules to represent policies. Kolovski et al. [25] demonstrate how together description logic and defeasible logic rules can be used to understand the effect and the consequence of sets of access control policies. They share with our view the set of reasoning tasks over policies, and use description logics. On the other hand, they don't address complexity issues. The Platform for Privacy Preferences (P3P)[2] is a W3C recommendation that enables websites to express their privacy preferences in a machine readable format. A more recent W3C recommendation known as the Open Digital Rights Language (ODRL)[3], released in early 2018, is a general rights language that can be used to define rights or to limit access to digital resources. In principle, any of these languages could be used to encode usage policies in a CPSS scenario. Still, there are other relevant considerations that suggest to define a usage policy language around the recent standard OWL2, and select language constructs carefully in order to adequately trade off expressiveness and computational complexity. This is the main objective of the SPECIAL policy language [5][8], developed within the EU H2020 SPECIAL project. In the next section, we provide an analysis of the policy language and its adaptation to CPSS needs.

2.2 Transparency and Compliance

Since the GDPR has come into effect, data controllers must provide transparency to data subjects with respect to the processing of personal data and *compliance*, i.e. the CPSS data controller must demonstrate that the usage of personal data complies with data subjects' consent (it respects/does not violate any requests).

Transparency. As for transparency about data processing, relevant work primarily focuses on the re-purposing of existing logging mechanisms as the basis for personal data processing transparency and compliance [6]. Many of the existing approaches use a secret key signing scheme based on Message Authentication Codes (MACs) together with a hashing algorithm to generate chains of log records that are in turn used to ensure log confidentiality and integrity [2] (cf. [6] for a summary of existing approaches). MACs use symmetric keys that are generated and verified using collision-resistant secure cryptographic hash functions. Only a few contributions [34,37], however, focused on personal data processing. An alternative distributed architecture to manage access to personal data based on blockchain technology has been proposed by Zyskind et al. [46]. The authors discuss how the blockchain data model and Application Programming Interfaces (APIs) can be extended to keep track of both data and access transactions. More recently, Sutton and Samavi [41] propose an extension of blockchain technology with *Linked Data* to create tamper-proof audit logs and non-repudiation. Very little research has been conducted, however, into transparency requirements and performance/scalability issues of such blockchain-based solutions.

[2] P3P,`http://www.w3.org/TR/P3P/`

[3] ODRL,`https://www.w3.org/TR/odrl-model/`

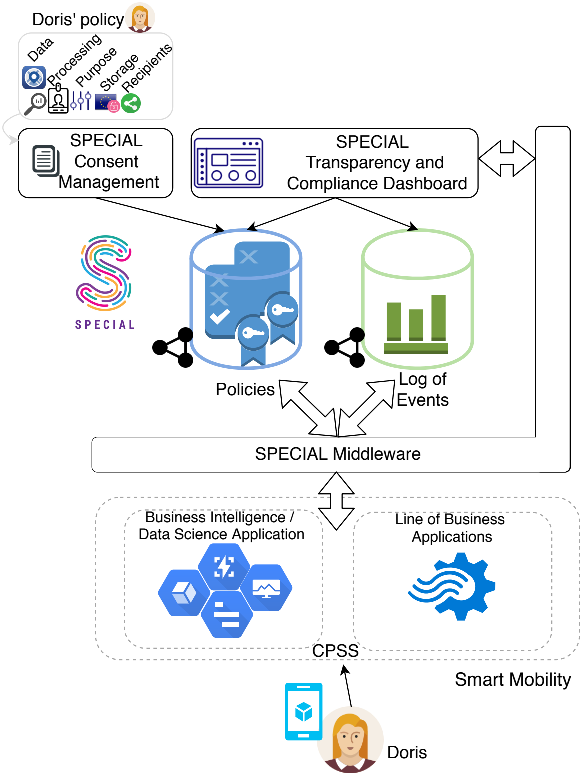

Doris' policy
Data
Processing
Purpose
Storage
Recipients
SPECIAL Consent Management
SPECIAL Transparency and Compliance Dashboard
SPECIAL
Policies
Log of Events
SPECIAL Middleware
Business Intelligence / Data Science Application
Line of Business Applications
CPSS
Smart Mobility
Doris

Fig. 1 The SPECIAL Consent, Transparency and Compliance framework

Compliance. As for GDPR compliance, recently the Information Commissioner's Office (ICO) in the UK [22], Microsoft [28], and Nymity [31] have developed compliance tools that enable companies to assess the compliance of their applications and business processes by completing a predefined questionnaire. Recent works also look at the challenges of representing GDPR concepts and obligations [32] as well as informed consent [17]. The management of events for business process compliance monitoring and process mining [27] can be seen as orthogonal work.

In contrast to existing approaches, in this paper we focus on vocabularies that can be used to record both usage policies and data processing and sharing events in a manner that supports automatic compliance checking.

3 Background and Methodology

In this section, we first present a high level overview of the SPECIAL consent, transparency and compliance framework (Section 3.1). Following on from this, we describe the SPECIAL policy language in detail (Section 3.2). Finally, we propose a methodology that can be used to adapt and extend the SPECIAL policy language and vocabularies to cope with CPSSs requirements in general, and, our smart mobility use case in particular (Section 3.3).

3.1 SPECIAL Consent, Transparency and Compliance Framework

In order to enable companies to comply with consent and transparency requirements stipulated by the GDPR, SPECIAL provides a policy language, vocabularies and a consent, transparency and compliance framework, which can be adapted and extended specifically for CPSS needs. The SPECIAL framework (shown in Figure 1) consists of the following components:

(i) the *SPECIAL Consent Management* component, which is responsible for obtaining consent from the data subject and representing it in the form of a machine readable usage policy;
(ii) the *SPECIAL Transparency and Compliance Component*, which is responsible for presenting data processing and sharing events in an easily digestible manner and demonstrating that existing data processing and sharing complies with the respective usage control policies; and
(iii) the *SPECIAL Middleware*, which includes sub-components that connect the SPECIAL primary components with the access control mechanisms and business logic of existing Line of Business applications, and middleware that enables companies to perform policy aware business intelligence and data science.

Underpinning the framework are a variety of existing data sources that support business operations (i.e., *Line of Business Applications*), and strategic decision making (i.e. *Business Intelligence / Data Science Applications*), and two additional SPECIAL data sources that are needed to support SPECIAL's consent, transparency and compliance framework: a *Policies* store, which is used to record the consent, regulatory and business policies; and an *Events* store, which is used to record (i.e. log) data processing or sharing events.

3.2 SPECIAL's Usage Policy Language

In this section, we provide a detailed overview of the SPECIAL usage policy language vocabularies, which we will analyze and extend in a practical CPSS scenario in subsequent sections.

SPECIAL usage policies are encoded in OWL 2 [30]. In the examples[4] that follow, the `spl` prefix represents `http://www.specialprivacy.eu/langs/usage-policy#`. Additional details, including the full policy expression grammar in Backus normal form (BNF), can be found in the SPECIAL documentation [5].

[4] For the policy language examples we use the OWL functional syntax which is less verbose.

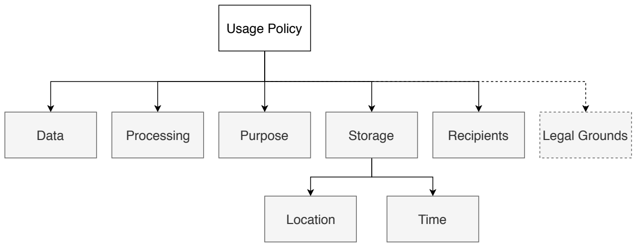


Fig. 2 The SPECIAL policy minimum core model (MCM), extended with optional legal grounds.

3.2.1 Data Usage Policy Model

Conceptually, a *usage policy* is meant to specify a *set of authorized operations*. According to the GDPR, these policies shall specify clearly which data are collected, what is the purpose of the collection, what processing will be performed, where the data is stored, and whether or not the data will be shared with others. The SPECIAL policy language, which was developed in close collaboration with legal experts, consists of five core elements, collectively known as the *minimum core model* (MCM), which is depicted in Figure 2:

- *Data* describes the personal data collected from the data subject (e.g. contact information, financial data, etc).
- *Processing* describes the operations that are performed on the personal data (e.g. collection, analysis, etc).
- *Purpose* specifies the objective that is associated with data processing (e.g. health, marketing, etc).
- *Storage* specifies where data are stored and for how long.
- *Recipients* specifies who is going to receive the results of data processing and, as a special case, whom data are shared with.

Optionally, policies can be extended with zero or more legal ground(s) for processing. In this paper, we focus on consent, but other alternatives (such as *legitimate interest*) can be represented [4].

3.2.2 Encoding SPECIAL Usage Policies

A *basic usage policy* is composed of one or more policies, each of which is an OWL 2 expression of the form presented in Listing 1.

Listing 1 A basic usage policy

```
ObjectIntersectionOf(
   ObjectSomeValueFrom( spl:hasData SomeDataCategory )
   ObjectSomeValueFrom( spl:hasProcessing SomeProcessing )
   ObjectSomeValueFrom( spl:hasPurpose SomePurpose )
   ObjectSomeValueFrom( spl:hasStorage SomeStorage )
   ObjectSomeValueFrom( spl:hasRecipient SomeRecipient )
)
```

Table 1 SPECIAL auxiliary vocabularies for usage policies.

Category	Namespace	#Classes	Examples	Superclass
Data	svd:=(S)/data	27	svd:Activity, svd:Anonymized, svd:Financial, svd:Health, svd:Location, svd:Navigation, svd:Preference, svd:Profile, etc.	spl:AnyData
Processing	svpr:=(S)/processing	9	svpr:Aggregate, svpr:Analyze, svpr:Anonymize, svpr:Collect, svpr:Copy, svpr:Derive, svpr:Move, svpr:Query, svpr:Transfer	spl:AnyProcessing
Purpose	svpu:=(S)/purposes	31	svpu:Account, svpu:Arts, svpu:Delivery, svpu:Education, svpu:Feedback,svpu:Gaming, svpu:Health, svpu:Marketing, svpu:Payment, svpu:Search, etc.	spl:AnyPurpose
Recipient	svr:=(S)/recipients	6	svr:Delivery, svr:OtherRecipient, svr:Ours, svr:Public, svr:Same, svr:Unrelated	spl:AnyRecipient
Storage location	svl:=(S)/locations	7	svl:ControllerServer, svl:EU, svl:EULike, svl:ThirdCountries, svl:OurServers, svl:ProcessorServers, svl:ThirdParty	spl:AnyLocation
Storage duration	svdu:=(S)/duration	4	svdu:BusinessPractices, svdu:Indefinitely, svdu:LegalRequirement, svdu:StatedPurpose	spl:AnyDuration

The policy presented in Listing 1, which follows the minimum core model (MCM), authorizes all operations on data that: (i) belong to *SomeDataCategory*, (ii) fall within the specified *SomeProcessing* category, (iii) have any purpose covered by the *SomePurpose* category, (iv) store the results of the processing in any place belonging to the *SomeStorage* category, and (v) disclose the results to any member(s) of the *SomeRecipient* category.

Additionally, SPECIAL provides several auxiliary vocabularies that provide a set of classes for *SomeDataCategory*, *SomeProcessing*, *SomePurpose*, *SomeRecipient*. Table 1 provides a high-level overview of the proposed vocabularies[5] that were defined in the context of the SPECIAL use cases. For instance, the policy in Listing 2 presents an example of a union of *basic usage policies*, in the context of an online fundraising website. The policy states that financial data can only be used for payment purposes and shall neither be stored nor disclosed to third parties, while the nickname can be used freely.

[5] All namespaces share the `S` which represents `http://www.specialprivacy.eu/vocabs`.

Finally, note that the hasStorage policy attribute is a structured object itself, with two attributes, and is specified in Listing 3, where *SomeLocation* and *SomeDuration* are expressed in terms of the corresponding storage location and duration auxiliary vocabularies.

Considering that it is clearly not possible to enumerate over all possible classes the policy language and by extension the vocabularies were designed to be extensible. This paper builds upon this extensibility to provide support for CPSS scenarios.

Listing 2 A policy composed of a union of basic usage policies

```
ObjectUnionOf(
   ObjectIntersectionOf(
      ObjectSomeValueFrom( spl:hasData svd:Financial )
      ObjectSomeValueFrom( spl:hasProcessing spl:AnyProcessing )
      ObjectSomeValueFrom( spl:hasPurpose svpu:Payment )
      ObjectSomeValueFrom( spl:hasStorage spl:Null)
      ObjectSomeValueFrom( spl:hasRecipient spl:Null ) )
   ObjectIntersectionOf(
      ObjectSomeValueFrom( spl:hasData svd:nickname )
      ObjectSomeValueFrom( spl:hasProcessing spl:AnyProcessing )
      ObjectSomeValueFrom( spl:hasPurpose spl:AnyPurpose)
      ObjectSomeValueFrom( spl:hasStorage spl:AnyStorage)
      ObjectSomeValueFrom( spl:hasRecipient spl:AnyRecipient ) )
)
```

Listing 3 Typical structure of the hasStorage policy attribute

```
ObjectIntersectionOf(
   ObjectSomeValueFrom( spl:hasLocation SomeLocation )
   ObjectSomeValueFrom( spl:hasDuration SomeDuration )
   DataSomeValuesFrom( spl:durationInDays Interval )
)
```

3.3 Methodology

Figure 3 depicts the inputs, main steps and outputs of the methodology we adopted when extending the SPECIAL usage policy language vocabularies, in order to cater for a smart mobility use case, and more generally, making the first steps towards its use for the broader family of Cyber-Physical Social Systems.

The **inputs** to our work are the SPECIAL usage policy language presented in the previous section and the smart mobility use case, which will be discussed in detail in Section 4. The primary **output** is a vocabulary (i.e., a set of terms) that can be used in the current use case to specify usage policies.

SPECIAL's minimum core model (MCM, see Fig. 2) is highly generic (i.e., domain agnostic) and therefore offers little support in deriving vocabularies that are necessary to support specific use cases. To fill this gap, we propose a domain-agnostic approach that can be used to facilitate the creation of use case

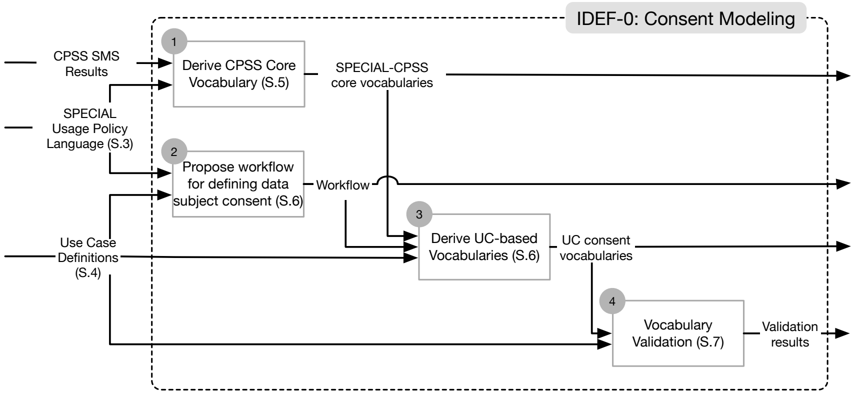

IDEF-0: Consent Modeling
CPSS SMS Results
SPECIAL Usage Policy Language (S.3)
Use Case Definitions (S.4)
1
Derive CPSS Core Vocabulary (S.5)
SPECIAL-CPSS core vocabularies
2
Propose workflow for defining data subject consent (S.6)
Workflow
3
Derive UC-based Vocabularies (S.6)
UC consent vocabularies
4
Vocabulary Validation (S.7)
Validation results

Fig. 3 Methodology for extending the SPECIAL usage policy language vocabularies for a smart mobility use case.

specific vocabularies, by first deriving a specialization of the SPECIAL MCM that captures terminology classes generically valid across a given domain. The objective being to derive a *core* ontology [38] for CPSS usage policies that are applicable and reusable across multiple CPSS subdomains. The proposed approach conforms with ontology engineering best practices, which suggest the development of layered ontology extensions from highly domain independent ontologies (e.g., generic ontologies), to core ontologies (e.g., domain ontologies) and then increasingly specific subdomain and task ontologies. The proposed approach is composed of four steps, which can be summarized as follows:

1. **Step 1: Derive the CPSS core ontology.** In order to bridge the semantic gap between the SPECIAL vocabularies and domain-specific terms required to support our CPSS use cases, we first need to identify domain-specific terms. This core ontology serves as a starting point for extending SPECIAL not just to support smart mobility systems, but also to support other CPSS systems, such as smart manufacturing, smart grids or smart homes. In order to derive this generic CPSS ontology, we rely on a principled approach grounded in a *Systematic Mapping Study*. Specifically, we followed the methodology of Kitchenham et al [24] as we detail in Section 5. The output of this step is a `SPECIAL-CPSS core ontology`.
2. **Step 2: Propose workflow to define CPSS data subjects' consent and data usage policies.** Before deriving specific usage policy vocabularies, it is first necessary to identify components, relationships and data sources based on use case descriptions. The concrete steps taken to that end are captured in a practical workflow to define CPSS data subjects' consent and data usage policies, as described in Section 6.
3. **Step 3: Derive use case specific vocabularies** by applying the workflow for deriving domain specific vocabularies (at Step 2) based on a detailed use case description. This step results in a set of domain-specific vocab-

ularies for usage policy specification (simply referred to as `UC consent vocabularies`). In this step, we rely on the output of Step 1, i.e., the `SPECIAL-CPSS core ontology`, and exemplify its usage to derive a domain ontology.

4. **Step 4: Vocabulary validation.** We validate both the `SPECIAL-CPSS core ontology` and the `UC consent vocabularies` by exemplifying their use to create usage policies required to support our smart mobility use cases. This step results in a set of `validation results` w.r.t. concrete usage policies. We show a practical example in Section 7.

To sum up, the major, reusable outputs of our methodology are:

- The *SPECIAL-CPSS core ontology*, which can also serve as a starting point for describing usage policies vocabularies in other CPSS domains (output of Step1).
- The *workflow to define CPSS data subjects' consent and data usage policies* (output of Step2);
- The *overall methodology* itself, can be followed whenever adapting SPECIAL to new domains. It provides guidance with respect to creating both core extensions of SPECIAL, as well as domain specific vocabularies.

4 CitySPIN Smart Mobility Use Case

In order to exemplify CPSS transparency and compliance requirements in the context of the GDPR, we present a general overview of a *Smart Mobility* use case that emerged in the CitySPIN project (Section 4.1) and subsequently describe four specific use case scenarios (Section 4.2).

4.1 General overview

As Vienna's largest utility provider, Wiener Stadtwerke (WStW) manages a broad and diverse public transportation network. In their long-term planning activities, WStW aims to extend and optimize this network. In the shorter/medium term, the network needs to be adjusted temporarily, e.g., to cater for the transportation needs of large-scale events or to accommodate special situations such as refurbishing and temporary closure of transportation network stations. In this context, information about passenger flows, i.e., movement patterns generally or during (recurring) large-scale events, are a key factor in decision-making processes. Passenger access to the transportation network is currently not monitored digitally (e.g., through access gates). Such information could in principle be obtained from individual citizens, but this requires solid transparency mechanisms and means to ensure compliance, as described next. Note that the following descriptions are based to the CitySPIN project context and that they are not currently put in practice in the company's production environment.

4.2 Scenarios

In the following, we present four scenarios of the CitySPIN use case, exemplified with a generic WienMobil APP user, Doris, and a WStW transportation network planner, Eva.

Doris installs the WienMobil APP, provided by a subsidiary of WStW, which allows her to obtain real-time public transport routing information in Vienna. For a desired destination, the APP provides the best route from the current (or a specified) location by combining several modes of transportation (metro, bus, train, rent-bicycles etc) within Vienna.

During installation, Doris is guided through a number of privacy choices that determine the later behavior of the App. Those choices can later be changed in the settings. Depending on those choices, different policies will apply.

Scenario 1: A personalized mobility planning. The APP states that, in order to provide a more personalized routing service, the APP can record the history of her routing requests, including the GPS location at the moment of the search. This can be integrated with external non-personal data sources (e.g. traffic congestion and city events) and will be used to analyse her mobility patterns, including the attendance to events in the city (e.g. concerts, sport events), in order to recommend her best routes and notify about delays in the future. Additionally, the APP informs her that the data will be stored on the company servers in Austria for a period of 2 years after each collection point, in order to detect yearly recurrent events. The collected data can always be retrieved, amended or deleted via the privacy dashboard.

Doris accepts this option and starts using the APP. As she is a fan of one local soccer club, she makes intensive use of the APP to go to the soccer stadium. As soon as the end of the match is approaching, the WienMobil APP notifies Doris and shows the fastest route (avoiding any congestion) to her house or a restaurant she frequents regularly after matches.

As many people are using the APP, the service can alert Doris to wait and have a coffee in the surrounding until the congestion after the match is over. The APP issues an alarm sound once the conditions are good again.

Scenario 2: Event Partnership. At a certain point in time, WStW establishes a partnership with governmental organizations to promote non-profit cultural and heritage events. Thus, the APP asks for a potential policy update. The APP states that she can also receive partnership recommendations related to her mobility patterns for non-profit cultural and heritage events. In this case, she needs to consent to use the same collected information (history of routes, mobility patterns and GPS location) and demographic data (from her annual pass) for the new purpose. Doris consents to this update, and continues using the application.

Some time later, Doris is recommended to plan her visit to the “Long Night of Museums” (spanning activities around the full city). The day of the event, the APP suggests to keep her current GPS active in order to receive live updates of museum attendance, routes and sub-event recommendations. Doris enables this for 1 day, and the APP provides regular updates on her potential destinations, taking into account her profile (including already visited museums) and crowded locations.

Scenario 3: A Fully-fledged privacy dashboard. As there is a huge festival to promote local/regional products, the APP asks whether she would attend it, inviting her to an appetizer. She declines, and takes the opportunity to use the APP's privacy dashboard to check and modify some of the permissions given. She can also find the data gathered from her, how they were processed, where they were stored and for what purposes they were used.

Scenario 4: Decision support for WStW planners. Eva is a transportation network planner at WStW. She and her team are responsible for planning extensions of the public transport network infrastructure in order to respond to evolving mobility patterns in the city (e.g., creating new lines, increasing/decreasing the capacity and frequency of vehicles) as well as to offer advice on adjusting the transportation schedule during large-scale events (increasing/decreasing the capacity and frequency of vehicles on the transportation lines affected by and/or relevant for the event).

Thus, Eva integrates relevant information from multiple sources in a semantic data lake, together with the associated usage policies. At some point, she wants to check the validity of the existing personalized recommendations. Thus, she uses the WienMobile APP to collect feedback on alternative routes. Doris can now select between 2 suggested routes and add comments on why she has selected a given option. This is integrated into Doris' privacy dashboard, for a period of one year after the data has been collected.

When executing pattern detection algorithms, the system automatically checks to ensure that no usage policy is violated and keeps records about the processing of the data. Thanks to this logging facility, WStW can easily and transparently demonstrate (e.g., through user dashboards) that all data storage and processing complies with previously collected consents.

5 Deriving the SPECIAL-CPSS Core Vocabulary

A key goal of this paper is to illustrate how the SPECIAL vocabularies (presented in Section 3.2) can be extended to cope with practical CPSS scenarios, such as the CitySPIN smart mobility use case (Section 4). To this end, we first derive a CPSS-specific *core ontology* that reduces the semantic gap between the SPECIAL MCM and domain-specific vocabularies by providing a set of concepts that are semantically closer to the needs of the application domain than the SPECIAL MCM. Other benefits of this core vocabulary include that it can provide better guidance on deriving domain-specific vocabularies than just the very abstract SPECIAL MCM concepts. Indeed, the intention is to create

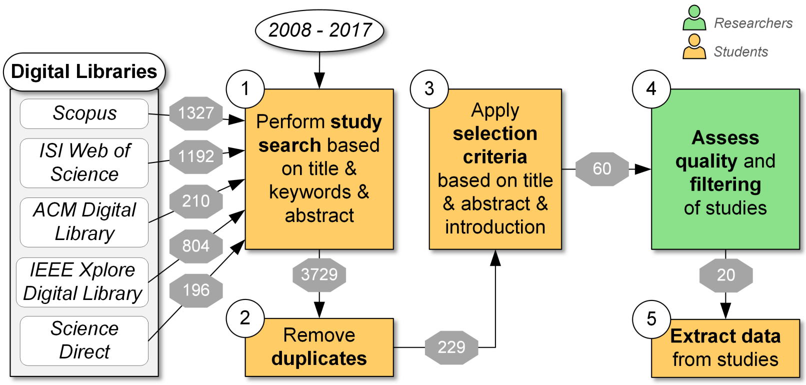

Researchers
Students
2008 - 2017
Digital Libraries
Scopus
ISI Web of Science
ACM Digital Library
IEEE Xplore Digital Library
Science Direct
1327
1192
210
804
196
1
Perform study search based on title & keywords & abstract
3729
2
Remove duplicates
229
3
Apply selection criteria based on title & abstract & introduction
60
4
Assess quality and filtering of studies
20
5
Extract data from studies

Fig. 4 Key stages of the Systematic Mapping Study that allowed extracting the information for creating the SPECIAL-CPSS core ontology.

a core vocabulary that can be reused for deriving usage policy vocabularies for CPSS in other domains as well (such as smart grids, smart home, smart manufacturing).

Methodologically, we ground the CPSS core ontology in information collected from literature describing a broad range of CPSSs. We collected this information by means of a Systematic Mapping Study (SMS) as proposed by Kitchenham et al [24]. The goal of a SMS is to review a specific software engineering topic area and to classify primary research papers (i.e., papers describing concrete systems, but not papers that survey those systems or an aspect thereof) in that domain in order to provide an overview of a certain topic [24].

Before collecting and analyzing the literature, we detailed all envisioned study stages and their parameters in a study protocol [36] which can be consulted for further details. The study aimed to answer the following research questions in order to provide an in-depth understanding of CPSS as described in the literature: *RQ1: What is an overarching definition of CPSS? RQ2: What are application domains, goals and stages specific toCPSS? RQ3:What are main characteristics of CPSS that could be used for their classification? RQ4: What is the role of human and social elements in CPSS? RQ5: What data sources are typically used in CPSS? RQ6: How is data processed and distributed in CPSS? RQ7: What architectural approaches are applied to design and describe CPSS? RQ8: What are currently main research areas and topics and what are key challenges and emerging future work trends in CPSS?* The details of the results obtained with that protocol are available in [35] which we briefly sum up here.

Papers to be included in the study were found through a manually performed *(1) keyword-based search* in five of the largest scientific digital libraries (see Fig. 4). The search spanned the period 2008-2017 and focused on the paper title, keywords and abstract. For the selection of the query terms, the research team collected candidate terms that: were aligned with the focus of the CitySPIN research project on cyber-physical social systems as well as related areas of research such as ”internet of things”, ”sensor networks”, ”participatory sensing”. A number of search queries were formed from these terms and run on digital libraries, in order to determine the number of resulting papers that they would return, as retrieving an overly large number of papers would have made the study unfeasible. For each of the candidate queries, we also took a look at a sample of the returned papers to estimate the quality of these papers, i.e., the level to which they fulfilled our selection criteria, especially IC1 (see details next). Finally, we settled for the following search query which lead to 3729 papers:

(cyber AND physical AND soci) OR (cyber AND physical AND human) OR (cyber AND physical AND soci* AND distributed) OR (cyber AND physical AND participatory)*

The 3729 papers were assessed for relevance based on their titles and collected into a spreadsheet which allowed *(2) duplicate detection and removal* and lead to a total of 229 papers. From these papers, 60 papers were identified as relevant for the study by *(3) applying a set of selection/exclusion criteria* on the information provided in their titles, abstracts and introductions. We tested three inclusion criteria:

- **IC1**: Studies focusing, proposing, leveraging, or analyzing a CPSS in detail. We were looking for papers that provide at least a minimal description of a concrete system in an application scenario or use case. At least one section of the paper should describe a system.
- **IC2**: Studies subject to peer review (e.g., journal papers, papers published as part of conference proceedings).
- **IC3** :Studies published since 2007.

We also checked the following exclusion criteria:

- **EC1**: Studies that are written in a language other than English, or that are not available in full-text.
- **EC2**: Secondary studies (e.g., systematic literature reviews, systematic map-ping studies, and surveys), which do not provide novel research results by their own and instead summarize work done by other researchers.
- **EC3**: Studies where a CPSS is only mentioned as a side-topic, e.g., this term appears only in the title or a reference or as an example.
- **EC4**: Studies focusing only on CPS in general, not on CPSS specifically.

Researchers involved in the study *(4) assessed the quality* of the candidate papers and selected 22 of them to include into the study. *(5) Data extraction* was guided by pre-defined extraction forms (see the study protocol [36]) which allowed to survey each paper in the same way (objectively) and reduced the room for bias. Besides bibliographic information, we collected data-items relevant to our research questions, e.g., *CPSS definition, application domain, CPSS purposes, CPSS process steps/activities, involvement of human actors, data sources, collected data*. The process of *analyzing and synthesizing* the collected

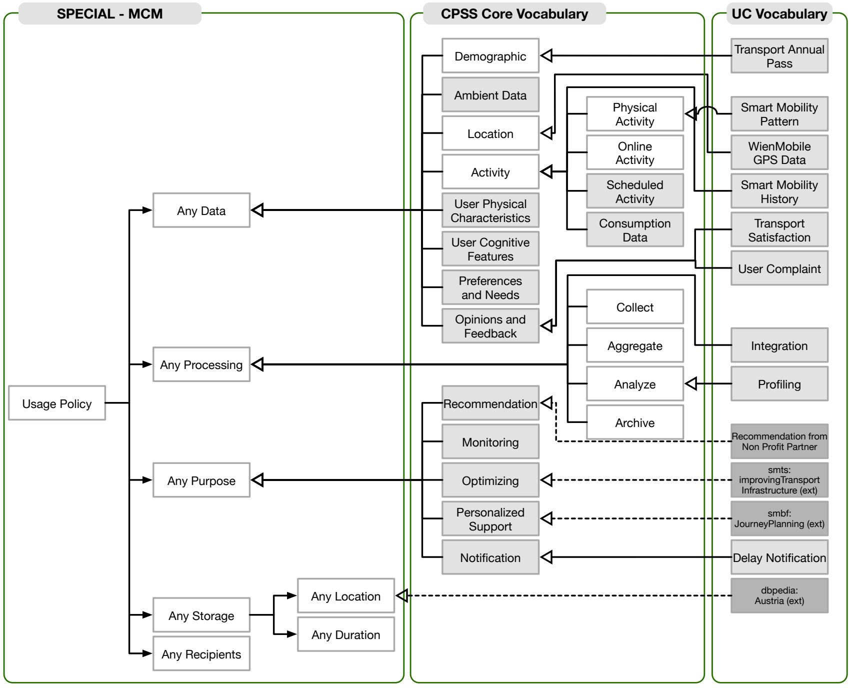


Fig. 5 Layered vocabulary: SPECIAL-MCM, SPECIAL-CPSS core ontology and the use case specific extensions.

data included the application of descriptive statistics and interpretation of the results with respect to the research questions.

The SPECIAL-CPSS core vocabulary is an extension of SPECIAL MCM and provides a point for further extension with use case specific vocabularies (see Figure 5). Specifically, extensions were made to the *Data* and *Purpose* concepts of SPECIAL, as described next and summed up in Tables 2 and 3. Note that we do not consider the extension of other MCM categories (*Processing*, *Storage* and *Recipients*) as (i) those can be seen as more general or domain-agnostic categories and (ii) they are already well-covered in SPECIAL. In any case, we provide specific examples of use-case based extensions in Section 6.

Extensions to SPECIAL Data. CPSS span the physical, the cyber and the social spaces; the data sets most often being used in CPSS describe either the physical or the social space of the system.

In terms of the physical space, *AmbientData* provides information about the surrounding environment such as weather conditions, air quality or temperature. Increasingly, such data is collected with smart sensors installed in the participants' personal sphere, and therefore can be subject to user consent. User *Location* is another frequently collected data category, for example, through smart phones' GPS sensors.

Table 2 CPSS Upper Level Ontology to describe Data Sources

Category	Description	SubClasses	Sources
AmbientData	Characteristics of the physical environment	AirQuality, Temperature	[14], [20]
Location			
ActivityData	General activity data.		
PhysicalActivity	Activities performed in the physical space.	DrivingActivity	
OnlineActivity	Activities performed in the online.	SearchLogs	
ScheduledActivity	Scheduled activities (past and future).	PlannedEventData	[40]
ConsumptionData	Measurement of resource consumption.	EnergyConsumption	[11]
UserPhysicalCharacteristics		HearthRate, BloodPreassure	
UserCognitiveFeatures		MemoryProblems	[33]
PreferencesAndNeeds	Needs and preferences to be taken account during recommendations or personalized support.	WalkingPreferences	[40]
OpinionsAndFeedback	User ratings or complaints.	ServiceSatisfaction, UserComplaint	[20]

In terms of data sets that describe different aspects of the (human) participants in the CPSS, *Activity* data is collected in various ways:

1. *PhysicalActivity* details user actions in the physical space, for example, a user's *DrivingData*, *HomeActivity*, or *MobilityData*.
2. *OnlineActivity* captures activities in the online sphere, for example, various digital traces left by the user, such as *SearchLogs*.
3. *ScheduledActivity* refers to past or future activities that were scheduled, for example, by means of the user's calendar entries. For example, the concept *PlannedEventData* could be introduced to captures events collected from a user's calendar such as done in [40].
4. *ConsumptionData* captures consumption of some resources, for example, energy consumption as recorded by smart meters [11].

Several systems, especially those with applications in the health care domain, actively collect *UserPhysicalCharacteristics* including for example, their *HearthRate* or *BloodPreassure*. Similarly, *UserCognitiveFeatures* (e.g., their attention span) are needed in those CPSSs that aim to adjust a process to these user characteristics. For example, in the smart manufacturing domain, adaptive manufacturing systems aim to improve the working conditions for aging workers by improving the human-machine interaction [33]. To that end, both physical conditions (e.g., colour blindness, short-sightedness, hearing loss) as well as cognitive features (anxiety disorders, memory problems) are collected and used within the smart manufacturing CPSS.

PreferencesAndNeeds, such as the users *WalkingPreferences*, are often used in CPSSs that offer recommendations or personal support. For example, an intelligent parking assistant suggests parking places closer/further to a meeting's place depending on whether the driver prefers to have a shorter/longer walk from the parking place to the meeting's location [40].

Table 3 CPSS Upper Level Ontology to describe Purpose.

Category	Description	SubClasses	Sources
(Personalized) Support	User is guided during a process to achieve goals in the best possible ways while taking into account real-time conditions.	DrivingSupport, NavigationSupport, ParkingSupport, JourneyPlanning	[44], [40]
Monitoring		HealthMonitoring, MonitoringTraffic	
Optimizing	Optimizing a process or a service by adjusting it in order to achieve efficiency or effectiveness.	OptimizingEnergyConsumption, OptimizingManufact.Process, ReducingCommutingTime, SharingResources	[11]
Recommendation	Suggest an object, event or other entity based on the user's constraints/profile and ambient conditions.	EventRecommendation	[44]
Notification	Feedback provided to the system users in diverse situations, ranging from messages to alerts.	EmergencyResponse, HealthWarnings, AnomalyDetection	[14]

Finally, *OpinionsAndFeedback* provided by users are important sources of information for CPSSs that aim to adjust recommendations according to the users' perception of some service (e.g., a restaurant). *ServiceSatisfaction* records as well as *UserComplaints* are typical types of data collected. For example, user ratings are used to recommend suitable airport services in [20].

Extensions to SPECIAL Purpose. We extend the SPECIAL *purpose* category with five broad purposes (see Table 3), which emerged from our overview of CPSS systems in various domains.

(Personalized) Support is the purpose of those CPSSs in which a user is guided during a process (e.g., driving, parking etc.) to achieve goals in the best possible way while taking into account real-time conditions (e.g., traffic congestion). *DrivingSupport*, *ParkingSupport* [40] , *NavigationSupport* (e.g., for visually impaired), *PersonalizedManufacturing*, *JourneyPlanning* are a few examples of more specific purposes in this category.

The purpose of *Monitoring* a process or the state of the environment is common among CPSSs, mostly as a pre-requisite to enable other purposes such as optimization or recommendation. Examples are *ManufacturingProcessTracking* and *HealthMonitoring*.

The *Optimizing* purpose is common among CPSSs. Indeed, many CPSSs have a feedback loop into their environment that allows the systems to modify the environment in ways that lead to optimization. Optimization can focus on a process or a service and it can aim at adjusting it in order to achieve efficiency or effectiveness. These adjustments often respond to changing conditions in the system's environment.

CPSSs that provide *Recommendation* services suggest an object, event or other entity based on the user's constraints/profile and ambient conditions. *EventRecommendation* is, for example, the purpose of the system presented in [44], which supports visually impaired students to find and attend suitable events on the university campus.

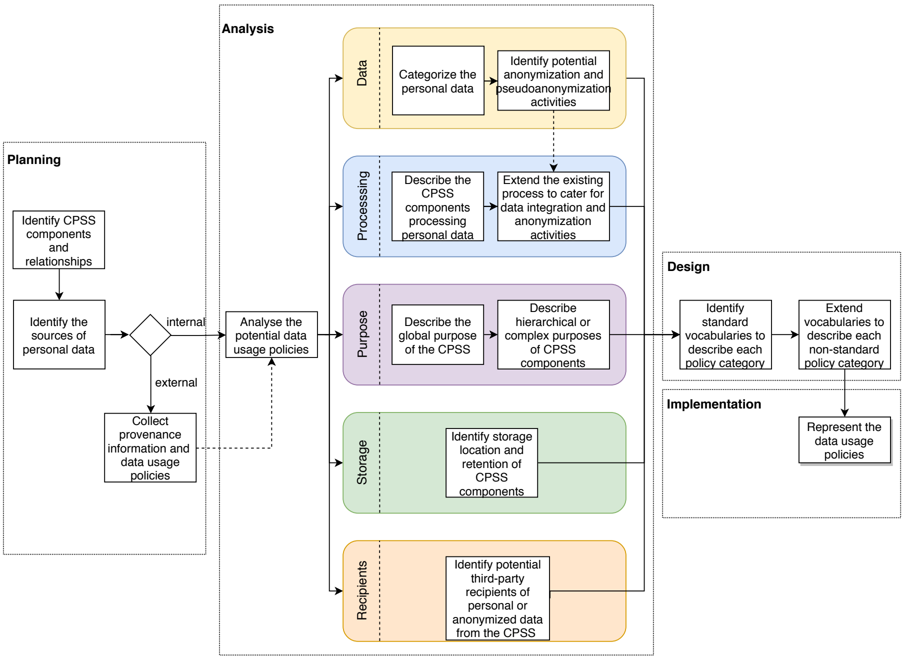

Planning
Identify CPSS components and relationships
Identify the sources of personal data
internal
external
Collect provenance information and data usage policies
Analysis
Analyse the potential data usage policies
Data
Categorize the personal data
Identify potential anonymization and pseudoanonymization activities
Processing
Describe the CPSS components processing personal data
Extend the existing process to cater for data integration and anonymization activities
Purpose
Describe the global purpose of the CPSS
Describe hierarchical or complex purposes of CPSS components
Storage
Identify storage location and retention of CPSS components
Recipients
Identify potential third-party recipients of personal or anonymized data from the CPSS
Design
Identify standard vocabularies to describe each policy category
Extend vocabularies to describe each non-standard policy category
Implementation
Represent the data usage policies

Fig. 6 Practical workflow to define CPSS data subjects' consent and data usage policies.

Notifications consist in feedback provided to the system users in diverse situations. Depending on the level of risk and danger in these situations, notifications can range from informative notes and messages to warnings (e.g., *HealthWarning*) and alerts. E.g., health warnings are provided to asthma patients depending on registered levels of pollen and air pollution in [14].

In the following section, we describe a practical workflow to define CPSS data subjects' consent and data usage policies.

6 A Practical Workflow for Conceptualizing CPSS Data Usage Policies

Our practical workflow, depicted in Figure 6, is aimed at supporting CPSS owners to analyze their CPSSs and to establish the terms that will be used to represent the CPSS data usage policies. These policies (i) are used to ask for data subjects' consent, and (ii) they shall be integrated in those CPSS components processing personal data in order to facilitate transparency and compliance. The workflow consists of a sequence of steps that take into account the SPECIAL usage policy model (cf. Section 3.2) and the general guidelines of the privacy by design [10] philosophy. Note that we group the series of steps following a typical planning, analysis, design and implementation life cycle, described in the following.

6.1 Planning

In a first phase, we identify the CPSS components, relationships and sources of data that will guide the rest of the process.

6.1.1 Identify CPSS components and relationships

The first step aims at identifying all CPSS components that manage data, as well as the different relationships among them. CPSSs are often complex systems composed of components of diverse nature [45], from physical world entities (e.g. sensors, vehicles, robots, smart meters, etc.) to socio-technical systems (crowdsourcing, collective intelligence systems, etc.) and cyber components (recommenders, decision support, etc). Thus, this introductory phase must clearly reveal and describe the components and the expected flow of data. Special attention should be paid to the description of inputs and user feedback loops, a key aspect in CPSSs that will be reflected in the MCM components (e.g., in the processing and purpose categories).

6.1.2 Identify the sources of personal data

Once the components and their relationships are clearly described, this step aims to identify all sources of personal data. The concept of *personal data* is defined in the GDPR as *"any information relating to an identified or identifiable natural person ('data subject')"*. This step is of particular importance given that most CPSSs integrate different data sources, with a strong social component.

Application in the CitySPIN smart mobility use case. As a practical example of the first two steps, we identified the CPSS components and relationships as well the internal/external data for the CitySPIN use case (Section 4). The result of the analysis is shown in Figure 7.

Our component identification process consists of the following steps: First, we extract terms (e.g. *'collect'*,*'integrate'*, *'aggregate'*, etc.) from the use case as CPSS component candidates. Next, we filter out duplicates and unnecessary terms, and finally we classify the remaining terms according to the minimum core model for usage policies (cf. 2) and data subjects.

We identify two types of data subjects: WienMobil users (e.g., Doris), and WStW planners (e.g., Eva). We focus on the first type, WienMobil users, as they are the source of personal data in the use case. In particular, they share up to five types of data: *routing requests, search history, location data, event attendance*, and *real-time feedback*. In addition, we identify other non-personal data sources, external (city events, weather data, etc.) and internal (e.g. public transport data). These data are gathered in a first stage (Figure 7.1).

The collection of personal data will go through a personal data collection process (Figure 7.2) to keep track of data provenance. From this point on, the data might be processed further to create profiles (Figure 7.3) and used

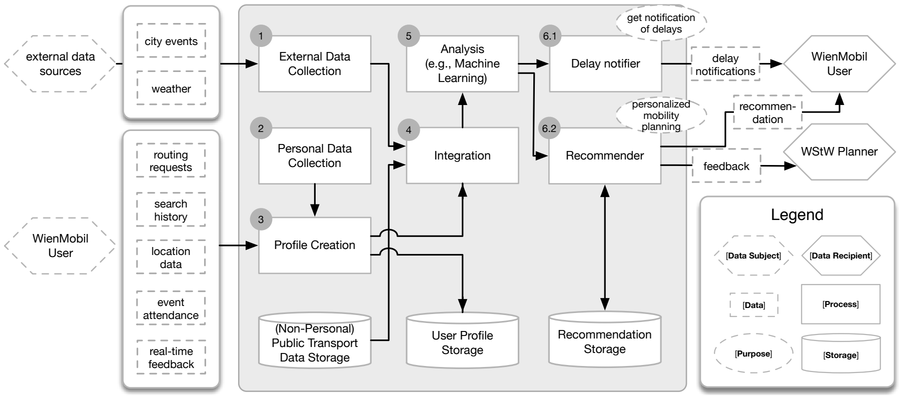


Fig. 7 CitySPIN CPSS privacy-related components.

directly within the integration process with external data (Figure 7.4), depending on the user consent. The profiles may also be stored in the profile storage. The main processing component of the use case is the analysis (Figure 7.5). This component is responsible for producing analysis results for various data recipients, e.g., delay notifications (Figure 7.6.1) and different kinds of recommendations, i.e. personalized planning (Figure 7.6.2), for WienMobil users. In addition, the feedback of users for such recommendations (if consented) will be processed by WStW planners.

6.1.3 Collect provenance information and data usage policies

At this point, after the first two steps, we must categorize the sources of personal data in two categories, *external* and *internal*. On the one hand, external personal data refers to personal data that is not generated in the CPSS. Note that processing personal data gathered from public sources (e.g. open data) or third-party companies is also subject to the GDPR, as the company behind the CPSS should be able to demonstrate that the data was collected and managed in compliance with the GDPR. This aspect is covered in the following phase. On the other hand, internal personal data refers to personal data generated within the CPSS. In this case, the usage policies should be represented (described below) and the appropriate data subjects' consent associated to the data should be obtained.

In this step, we first focus on the external personal data, where all provenance information (data sources, third-party contracts and terms, etc.) and data usage policies are collected. In this case, the CPSS company should take care of linking the provenance information with the policies and the concrete data that adhere to such policies. This process of linking provenance information, external policies and actual data is out of scope of this paper. In the future, we plan to extend SPECIAL to consider this aspect, implementing the concept commonly referred to as "sticky policies" [29].

6.2 Analysis and design of the data usage policies

As depicted in the workflow in Figure 6, (a) the internal data usage policies should be represented, which will then provide the basis to ask data subjects' for their consent to manage such data, or (b) the external policies and provenance information should be collected, in order to link it to the actual data and keep track of the process. Note that the second case can be simplified (and be limited to the linkage of data and policies) if the usage policy is already provided in a standard format by the data source provider, e.g. using the Resource Description Format (RDF) [39].

When it comes to representing CPSS data usage policies using the SPECIAL model, a first step involves a deeper analysis of the potential data usage policies for the use case. This implies to analyze the concrete terms that we need to specify in each of the five elements (data, processing, purpose, storage and recipients) of the SPECIAL minimum core model (cf. see Section 3.2), summarized below. Then, a second step consists of (i) identifying standard vocabularies to represent such terms (within the SPECIAL auxiliary vocabularies or others existing and reusable vocabularies) or (ii) extending the SPECIAL auxiliary vocabularies to cover the CPSS use case needs. Examples of existing and reusable vocabularies related to CPSS are the taxonomy for planning and designing smart mobility services [12], the Road Traffic Management ontology [3] and other smart city ontologies [15], to name but a few.

In the following, we describe the analysis and design for each of the five elements (data, processing, purpose, storage and recipients) of the SPECIAL minimum core model.

- The element '*Data*' describes the personal data collected from a data subject. First, the already identified CPSS elements and data sources must be analyzed further to categorize such data. In this step, rather than the actual data, the category of the data and the potential *skeleton* (i.e., structure) of typical data items should be identified.
  In a second step, following the privacy by design [10] philosophy, potential anonymization and pseudoanonymization activities shall be identified. This step plays an important role as the GDPR does not concern the processing of anonymous information, i.e., *"information which does not relate to an identified or identifiable natural person or to personal data rendered anonymous in such a manner that the data subject is not or no longer identifiable"*. However, if the CPSS company is in charge of the collection of personal data and their anonymization, then the appropriate data subject's consent should be obtained. Thus, this step is the place where we can identify whether anonymization, pseudonymization and aggregation of data (e.g.,

applying techniques such as k-anonymity and l-diversity [18,1]) is applied or can be applied in the CPSS. The strength of the anonymization techniques and possible attacker models based on combining anonymous data and other knowledge is out of scope for this paper. In our CitySPIN use case, we focus on consent, given the description and the activities in Figure 7 and the fact that anonymization impacts utility.
Then, the category of the final relevant personal data in the CPSS (i.e., data that cannot be further anonymized) needs to be represented in the SPECIAL policy language. Thus, we must identify standard vocabularies to describe the personal data categories, extending or providing new vocabularies if needed. As mentioned before, this step completely depends on the particular scenario, hence it is expected that the initial auxiliary vocabulary presented in SPECIAL (cf. see Table 1) needs to be extended with use case specific ontologies.

Application in the CitySPIN smart mobility use case. In our previous *smartMobility* example, following from the description and the activities in Figure 7, the analysis would reveal that the CPSS needs to store (i) location data, consisting of (potentially real-time) GPS locations of the user, (ii) routing requests, including source and target destination at a particular moment in time, (iii) a history of lookups in the wien Mobile APP, which is basically a log of user queries to the APP, (iv) event attendance, which is a particular case of a search of a route request to attend a specific event, and (v) real-time feedback of alternative routes. In the following, we show the analysis and vocabulary selection/creation for each of them. Note that we use the `wm` prefix to denote the use-case specific `Wien Mobile` namespace.

* Location data, i.e., GPS locations of the user, can be directly represented with the existing `svd:location` in SPECIAL. Nonetheless, note that the company might decide to have a more informative consent, stating that the data is collected from the GPS location of the APP. In such case, a specific `wm:WienMobileGPSData` category could be created, as a subclass of `svd:location`. We consider this as an instantaneous location when searching for a route, while we further refer to `wm:WienMobileGPSDataRealTime` as a continuous location stream.
* Event data, i.e., event records for a particular user and time, are underrepresented in SPECIAL, hence a particular extension is needed. In this case, we can make use of the proposed CPSS ontology to describe data source (see Table 2), e.g., using the `svd-cpss:ScheduledActivity` data category.
* Route requests and history of lookups, i.e., user's lookups in the APP. This data category could potentially be covered by the categories defined in the SPECIAL project, `svd:activity`, which represents data concerning user's activities, and its subcategory `svd:online-activity`, considering *data describing online activities such as browsing, liking on social networks, posting, etc.* [5]. Although these categories should cover several scenarios, fine-grained, company-specific categories can be pre-

ferred. For instance, in our example, we decide to create the class `wm:SmartMobilityHistory` extending `svd:Activity` to represent the history of lookups in the WienMobil APP. In addition, given that a mobility pattern can be extrapolated from the data, which will be part of the profile as specified in our diagram in Figure 7, we also create the `wm:SmartMobilityPattern` category, extending `svd:PhysicalActivity`.

  * Service ratings, i.e. information to reflect the user's satisfaction with the Wien Mobile service. In this case, the SPECIAL auxiliary vocabulary provides the general category `svd:preference`, which stands for *data about an individual's likes and dislikes - such as favorite color or musical tastes* [5]. An organization designing a CPSS may need to provide further details on the information collected. In that case, the `svd:preference` class should be extended to cope with the respective needs. In our particular scenario, we can make use of our aforementioned CPSS extensions (Table 2). Thus, we define the novel `wm:TransportSatisfaction` and `wm:UserComplaint` categories as subclasses of `svd-cpss:OpinionsAndFeedback`.

- *Processing.* The element '*Processing*' specifies the operations that are performed on the personal data. Given the inherent complexity of CPSSs, where multiple components are often organized in a 'pipeline' architecture, the first step is to analyze the information flow and the components identified in the planning phase, and to describe the CPSS components processing personal data. Given that the identification of components could be incomplete, as some processing activities could be implicit (e.g., a machine learning component can have several input sources and an implicit data integration process could be required), a second step considers to extend the previous analysis of components to identify and describe such potential implicit processes. Special attention shall be paid to describing (i) potential data anonymization and pseudoanonymization activities emerging from the previous 'data' phase (as represented with a dashed arrow in Figure 6), and (ii) integration activities, which are often present while combining different data sources in a CPSS scenario.
  Once all components and activities have been identified, similarly to the previous case, standard vocabularies to represent the concrete CPSS processing must be identified, or new concepts must be provided if needed. Note that, given the broad spectrum of CPSS applications and components, CPSS processing could cover all potential processing activities of an information system. SPECIAL provides a set of processing concepts (summarized in Table 1) that are more closely related to data protection, such as `svpr:Aggregate`, `svpr:Analyze`, `svpr:Collect`, etc.

Application in the CitySPIN smart mobility use case. In the following, we review the most important CPSS stages/activities emerging from our CitySPIN use case and the description and the activities in Figure 7:

  * Data collection - can be directly mapped to the SPECIAL `svpr:Collect` concept.

  * Profile creation - is not directly present in the SPECIAL auxiliary vocabularies. However, it is explicitly recommended to provide a use case specific concept as a subclass of `svpr:Analyze` [5]. In our case, we created the `wm:Profiling` concept.
  * Data integration - can be (partially) mapped to the SPECIAL `svpr:Derive` and `svpr:Aggregate` concepts. Given that the mapping could be inaccurate, a new concept extending the general `spl:AnyProcessing` class could be provided. In our case, we use `wm:Integration`.
  * Data analysis - can be directly mapped to the SPECIAL `svpr:Analyze` concept.
  * Proactive recommendations and notifications. In this case, SPECIAL considers such cases rather a 'purpose' (described below). Then, the 'processing' leading to the concrete recommendation and notification could be seen as a result of the previous steps, in particular `svpr:Analyze`. In our CitySPIN example, we follow this approach. Note that other use cases could need further processing steps, which could be included as processing if needed.

- *Purpose.* The element '*Purpose*' specifies the objective that is associated with data processing. In CPSSs, we could establish a two-phase identification of (a) describing the global purpose of the CPSS, and (b) analyzing and describing a hierarchical structure of the identified purposes of the CPSS components. The rationale behind this approach is that, whereas the final purpose can be almost extracted from the textual description of the use case, CPSSs often involve complex components and relationships that might be re-purposed for a specific goal, hence further analysis is required. Once these purposes are identified, standard vocabularies, or extensions, to describe CPSS purposes must be put in place.

Application in the CitySPIN use case. In the following, for exemplary purposes, we review the purposes identified in our CitySPIN use case:

  * Notifying of delays is under represented in the SPECIAL auxiliary vocabularies. Note, however, that it would be possible to make use of the `svpu:Current` concept (i.e., completion and support of activity for which data was provided), as a general concept if the main goal of the data collection and the CPSS is to provide such notifications to the user. In our CitySPIN use case, we create a specific `wm:DelayNotification` extending the proposed `svd-cpss:Notification` in the CPSS ontology to describe purposes (see Table 3).
  * Recommendations are only implicitly represented in SPECIAL, as part of marketing purposes (`svpu:Marketing`). Thus, proactive recommendations (`svd-cpss:Recommendation`) are specifically considered in the CPSS ontology (see Table 3). In our use case, additionally, we also reuse the existing `smbf:JourneyPlanning` category and we define a specific recommendation for non-profit partners (`wm:RecommendationNonProfitPartner`, which extends the proposed `svd-cpss:Recommendation`).

  * Providing feedback, for our last scenario, can be represented with the SPECIAL `svpu:Feedback` concept. In addition, given that the final objective is to optimize the transport infrastructure, we can consider the CPSS ontology to describe an optimization purpose (`svd-cpss:Optimizing`). In particular, in our CitySPIN use case, we make use of the existing `smts:improvingTransportInfrastructure` category.

- *Storage.* The element '*Storage*' specifies the location and temporal retention policy for the CPSS data. In the particular case of a CPSS, and given its potential distribution, this implies to identify the storage location and the required data retention of the individual CPSS components. Data retention periods can be then simply represented as a numeric range in the SPECIAL policy language (cf. Section 3.2). In turn, storage locations can be listed with the SPECIAL auxiliary vocabulary (e.g., using concepts such as `svl:EU` or `svl:ThirdParty`) or be extended if finer details are needed by the use cases. Note that the former should cover most CPSS use cases as the SPECIAL vocabulary for locations is designed to cover the GDPR requirements of specifying (i) whether the information is stored in the EU or in countries with similar data protection legislation, and (ii) whether the information is kept by the data controller or stored outside its boundaries [5].

  Application in the CitySPIN use case. In our CitySPIN use case, we only need to specify that data are stored on the company servers in Austria. Thus, we make use of both the SPECIAL `svl:OurServers` concept and the well-established `dbpedia:Austria` term. As for temporal retention, we just need to specify the number of days, from a single day up to 2 years, depending on the scenario.
- *Recipients.* Finally, the element '*Recipients*' specifies who can receive the results of the CPSS personal data processing. In this case, potential third-party recipients of personal data from the CPSS should be identified. Given the inherent complexity of CPSSs, this step may involve careful inspection of all (potentially distributed) CPSS components, involved partners and stakeholders. Then, as in previous elements, standard vocabularies to describe CPSS recipients must be analyzed, and extended where needed. Similarly to the *storage* element, SPECIAL auxiliary vocabularies (cf. see Table 1) should cover most of the CPSS use cases, while specific fine-grained descriptions may need some extensions, e.g., using the FOAF [9] and PROV [26] vocabularies.

  Application in the CitySPIN use case. In the particular case of CitySPIN, no recipients are needed, hence the use of the SPECIAL `svr:Ours` term.

6.3 Implementation: Representing the data usage policies

As a last phase, the final data usage policies should be represented using the SPECIAL policy language, using the selected terms in the previous phase.

Thus, each concrete scenario should be reviewed carefully, and each component of the SPECIAL MCM model should be represented in a simple but complete way, aiming to reflect the scenario (i.e. the textual policy) precisely. Obviously, the process can reveal some gaps that should be filled (e.g., if the data retention time has not been identified), which could require to repeat some of the previous steps of the proposed workflow.

7 Validation: User Policy Representation in CitySPIN Use Cases

This section presents the results of the practical application of the workflow to establish CPSS data subjects' consent for specific use cases (described in Section 6) to our CitySPIN smart mobility use case (shown in Section 4). Once the main CPSS components and personal data sources have been identified (see Figure 7), and we have carefully selected or extended vocabularies (see a summary of the use case specific extensions in Figure 5) for each of the components of the SPECIAL MCM model (data, processing, purpose, storage and recipients), we then proceed to represent the data usage policies.

In the following, we summarize the final policies for each of the "personal data" scenarios in the aforementioned CitySPIN smart mobility use case. Note that we do not specify a policy for scenario 3, as it is built upon the previously defined policies to exemplify the use of the privacy dashboard, providing transparency to data subjects.

7.1 Scenario 1: A personalized mobility planning

The study and analysis of the first scenario of the *Wien mobile* use case (as shown in Section 4) result in the following textual policy: *"The history of transport routing data and GPS location data (at the moment of the search) can be integrated with other non-personal data sources (city events, environment data, traffic congestions) and analyzed to create a mobility profile, in order to recommend best routes and notify about delays in the future. These profiles are stored for two years on the company servers in Austria"*. This policy is formalized in Listing 4.

Thanks to the previous steps of the workflow, the formalization of the policy is almost straightforward. First, the data category can be represented with a union (`ObjectUnionOf`) of three use-case specific terms (`wm:SmartMobilityHistory`, `wm:MobilityPattern` and `wm:WienMobileGPSData`) that accurately reflect the personal data involved in the scenario. The type of processing, also revealed during the identification of CPSS components, is restricted to profiling and integration, both represented with specific use-case terms (`wm:Profiling` and `wm:Integration`). Note that we also include the data collection process (`svpr:Collect`) although it was not explicit in the policy, as we assume the company is the responsible for collecting the data. As for the purpose, together with the general recommendation (`svd-cpss:Recommendation`), we consider the delay notification goal (`wm:DelayNotification`), and the planning of the journey purpose (`smbf:JourneyPlanning`). Finally, the storage (location and duration) and recipient (just ours) directly follow from the use case description and are encoded according to the SPECIAL policy language (e.g. the two year period is represented with a `xsd:maxinclusive` restriction).

Listing 4 Final policy of the CitySPIN scenario 1 - personalized mobility planning

```
ObjectIntersectionOf(
   ObjectSomeValueFrom( spl:hasData
      ObjectUnionOf(
         wm:SmartMobilityHistory wm:MobilityPattern wm:WienMobileGPSData ))
   ObjectSomeValueFrom( spl:hasProcessing
      ObjectIntersectionOf( wm:Profiling wm:Integration svpr:Collect ))
   ObjectSomeValueFrom( spl:hasPurpose
      ObjectUnionOf(
         svd-cpss:Recommendation smbf:JourneyPlanning wm:DelayNotification ))
   ObjectSomeValueFrom( spl:hasStorage
      ObjectIntersectionOf(
         ObjectSomeValuesFrom( spl:hasLocation
            ObjectIntersectionOf( svl:OurServers dbpedia:Austria ))
         DataSomeValuesFrom( spl:durationInDays
            DatatypeRestriction( xsd:integer
              xsd:maxInclusive "730"^^xsd:integer ))))
   ObjectSomeValueFrom( spl:hasRecipient svr:Ours ) )
```

7.2 Policies of scenario 2: Event Partnership

The analysis of the second scenario results in two different policies. The first one extends scenario 1 adding the processing of demographic data (from the transport annual pass) to receive partnership recommendations for non-profit cultural and heritage events, related to user's mobility patterns. Listing 5 shows the formalization of this extended policy.

Listing 5 Final policy of the CitySPIN scenario 2 - Event Partnership

```
ObjectIntersectionOf(
   ObjectSomeValueFrom( spl:hasData
      ObjectUnionOf(
         wm:SmartMobilityHistory wm:MobilityPattern wm:WienMobileGPSData
         wm:AnnualPass ))
   ObjectSomeValueFrom( spl:hasProcessing
      ObjectIntersectionOf( wm:Profiling wm:Integration svpr:Collect ))
   ObjectSomeValueFrom( spl:hasPurpose
      ObjectUnionOf(
         wm:RecommendationNonProfitPartner smbf:JourneyPlanning
         wm:DelayNotification ))
   ObjectSomeValueFrom( spl:hasStorage
      ObjectIntersectionOf(
         ObjectSomeValuesFrom( spl:hasLocation
            ObjectIntersectionOf( svl:OurServers dbpedia:Austria ))
         DataSomeValuesFrom( spl:durationInDays
            DatatypeRestriction( xsd:integer
              xsd:maxInclusive "730"^^xsd:integer ))))
   ObjectSomeValueFrom( spl:hasRecipient svr:Ours ) )
```

Similarly to the previous case, the representation of the policy follows from all previous steps. In this case, the only modifications are the inclusion of the demographic data (`wm:annualPass`) and the new partnership purpose (`wm:RecommendationNonProfitPartner`). Note that both concepts are represented using specific use-case terms, hence they should include a human-readable comprehensive definition of the actual type of data considered in each case (e.g. cultural and heritage events in the case of the partner recommendations).

Then, a second policy extends the previous one considering live updates and recommendations based on real-time GPS location for attended events. In our scenario, the user only consents for a period of one day. This policy is shown in Listing 6.

Listing 6 Final policy of the CitySPIN scenario 2 - real time recommendations

```
ObjectIntersectionOf(
   ObjectSomeValueFrom( spl:hasData
      ObjectUnionOf(
         svd-cpss:PlannedEventData
         wm:SmartMobilityHistory wm:MobilityPattern
         wm:WienMobileGPSDataRealTime ))
   ObjectSomeValueFrom( spl:hasProcessing
      ObjectIntersectionOf( wm:Profiling wm:Integration svpr:Collect ))
   ObjectSomeValueFrom( spl:hasPurpose
      ObjectUnionOf(
         wm:RecommendationNonProfitPartner smbf:JourneyPlanning
         wm:DelayNotification ))
   ObjectSomeValueFrom( spl:hasStorage
      ObjectIntersectionOf(
         ObjectSomeValuesFrom( spl:hasLocation
            ObjectIntersectionOf( svl:OurServers dbpedia:Austria ))
         DataSomeValuesFrom( spl:durationInDays
            DatatypeRestriction( xsd:integer
              xsd:maxInclusive "1"^^xsd:integer ))))
   ObjectSomeValueFrom( spl:hasRecipient svr:Ours ) )
```

Similarly to the previous cases, the policy represents the data with a union (`ObjectUnionOf`) of terms to capture the real time use-case location data history (`WienMobileGPSDataRealTime`) and the planned event data (in this case using the CPSS upper level ontology via `svd-cpss:PlannedEventData`). The processing, purpose, storage and recipients follow the previous examples.

7.3 Policy of Scenario 4: Decision support for WStW planners

Finally, the last scenario builds upon scenario 1 and considers feedback (potentially in real-time) on personalized alternative routes. In this case, the storage is limited to one year after collection. This policy is formalized in Listing 7.

In this case, the personal data include the use-case specific feedback, which is represented with use-case specific terms (`wm:TransportSatisfaction` and `wm:UserComplaint`). The processing and recipients are similar to the previous case, whereas we reduce the storage to 1 year and, in this case, we adapt the purpose to the general provision of feedback (via the existing svpu:Feedback). To be more transparent, we also include the implicit use-case specific purpose of improving the transport infrastructure (smts:improvingTransportInfrastructure).

Listing 7 Final policy of the CitySPIN scenario 4 - Feedback for decision support

```
ObjectIntersectionOf(
   ObjectSomeValueFrom( spl:hasData
      ObjectUnionOf(
         wm:TransportSatisfaction wm:UserComplaint wm:SmartMobilityHistory
         wm:MobilityPattern wm:WienMobileGPSData ))
    ObjectSomeValueFrom( spl:hasProcessing
      ObjectIntersectionOf(
         wm:Profiling wm:Integration svpr:Collect ))
   ObjectSomeValueFrom( spl:hasPurpose
      ObjectUnionOf(
         svpu:Feedback smts:improvingTransportInfrastructure ))
   ObjectSomeValueFrom( spl:hasStorage
      ObjectIntersectionOf(
         ObjectSomeValuesFrom( spl:hasLocation
            ObjectIntersectionOf( svl:OurServers svl:EU ))
         DataSomeValuesFrom( spl:durationInDays
            DatatypeRestriction( xsd:integer
               xsd:maxInclusive} "365"^^xsd:integer ))))
   ObjectSomeValueFrom( spl:hasRecipient svr:Ours ) )
```

8 Summary and Future Work

Privacy protection is a fundamental but challenging requirement in the context of Cyber-Physical Social Systems (CPSSs), which, by definition collect and make use of user-specific data (from the “social” space). CPSS owners need to ensure compliance with user policies be transparent in terms of how users’ data is being processed. While automated compliance checking and transparency can be achieved based on formally represented usage policies, existing policy languages that enable specifying user consent are domain-agnostic and require adaptation when used in concrete use cases. For example, in this paper we exemplify extending the SPECIAL domain-agnostic policy language for describing user policies in a smart mobility use case provided by Vienna’s largest utility provider.

We relied on an approach which aims to support CPSS owners in general, and WStW in particular, in adapting the policy language for the needs of their own use cases in two ways: (1) by providing the SPECIAL-CPSS core-vocabulary that already extends the domain-agnostic SPECIAL terms towards the domain of CPSS; (2) by proposing a novel practical workflow that can be used to elicit vocabularies for defining CPSS data subjects’ consent and data usage policies. We validate the resulting vocabularies (both core and use case specific) by demonstrating that they can be used successfully to construct usage policies according to the SPECIAL specification.

A current limitation of this work is that the SPECIAL-CPSS core vocabulary and the proposed workflow have been tested on a mobility use case only.

Our ongoing work focuses on reusing and validating these two outcomes on another CPSS use case from WStW in the domain of smart energy grids. Additionally, the coverage of the SPECIAL-CPSS core vocabulary was influenced by the selection of query keywords used in the mapping study by not considering related terms due to practical considerations of the study feasibility. Therefore, we also focus on further improvements of the SPECIAL-CPSS core vocabulary in terms of (1) aligning it with foundational ontologies; (2) grounding it in agency models that better reflect the social aspect of CPSS and (3) planning follow-up studies on related terms, such as "participatory sensing" to make it more comprehensive. In the future, we plan to extend our model with layers dedicated to concrete domains, e.g., smart grid, smart manufacturing, smart home. Finally, we plan to extend our SPECIAL-CPSS approach with the concept of sticky policies for those data coming from external sources.

Acknowledgements This work has been supported by the European Union's Horizon 2020 research and innovation programme under grant 731601 (SPECIAL) and by the Austrian Research Promotion Agency (FFG): grant no. 861213 (CitySPIN).

References

1. Aggarwal, C.C., Philip, S.Y.: A general survey of privacy-preserving data mining models and algorithms. In: Privacy-preserving data mining, pp. 11–52. Springer (2008)
2. Bellare, M., Yee, B.: Forward integrity for secure audit logs. Tech. rep., Computer Science and Engineering Department, University of California at San Diego (1997)
3. Bermejo, A., Villadangos, J., Astrain, J.J., Cordoba, A.: Ontology based road traffic management. In: Proc. of Intelligent Distributed Computing, pp. 103–108. Springer (2013)
4. Bonatti, P., Kirrane, S., Petrova, I., Sauro, L., Kerschbaum, C., Pirkova, E.: Special deliverable 2.6: Formal representation of the legislation v2 (2018). URL `https://www.specialprivacy.eu/images/documents/SPECIAL_D26_M21_V10.pdf`
5. Bonatti, P., Kirrane, S., Petrova, I., Sauro, L., Schlehahn, E.: Special deliverable 2.1: Policy language v1 (2017). URL `https://www.specialprivacy.eu/images/documents/SPECIAL_D2.1_M12_V1.0.pdf`
6. Bonatti, P., Kirrane, S., Polleres, A., Wenning, R.: Transparent personal data processing: The road ahead. In: Proc. of TELERISE, pp. 337–349 (2017)
7. Bonatti, P.A., Coi, J.L.D., Olmedilla, D., Sauro, L.: A rule-based trust negotiation system. IEEE Trans. Knowl. Data Eng. **22**(11), 1507–1520 (2010)
8. Bonatti, P.A., Kirrane, S.: Big data and analytics in the age of the gdpr (2019)
9. Brickley, D., Miller, L.: Foaf vocabulary specification 0.91 (2010)
10. Cavoukian, A.: Privacy by design in law, policy and practice. A white paper for regulators, decision-makers and policy-makers (2011)
11. Chen, S., Liu, T., Gao, F., Ji, J., Xu, Z., Qian, B., Wu, H., Guan, X.: Butler, Not Servant: A Human-Centric Smart Home Energy Management System. IEEE Communications Magazine **55**(2), 27–33 (2017)
12. Cledou, G., Estevez, E., Barbosa, L.S.: A taxonomy for planning and designing smart mobility services. Government Information Quarterly **35**(1), 61–76 (2018)
13. Cranor, L.F.: Web privacy with P3P - the platform for privacy preferences. O'Reilly (2002)
14. Dao, M.S., Pongpaichet, S., Jalali, L., Kim, K., Jain, R., Zettsu, K.: A Real-time Complex Event Discovery Platform for Cyber-Physical-Social Systems. Proc. of ICMR pp. 201–208 (2014)
15. Espinoza-Arias, P., Poveda-Villalón, M., García-Castro, R., Corcho, O.: Ontological representation of smart city data: From devices to cities. Applied Sciences **9**(1), 32 (2019)

16. Falkvinge, R.: Airport: ”we’re tracking every single footstep you take and can connect it to your mail address, but your privacy is safe because we say so” (2017). URL https://falkvinge.net/2017/04/15/schiphol-airport-tracking-every-single-footstep/
17. Fatema, K., Hadziselimovic, E., Pandit, H.J., Debruyne, C., Lewis, D., O’Sullivan, D.: Compliance through informed consent: Semantic based consent permission and data management model. In: Proc of PrivOn (2017)
18. Ghinita, G., Karras, P., Kalnis, P., Mamoulis, N.: Fast data anonymization with low information loss. In: Proc. of VLDB, pp. 758–769. VLDB Endowment (2007)
19. Hildebrandt, M.: Smart technologies and the end (s) of law: novel entanglements of law and technology. Edward Elgar Publishing (2015)
20. Hussein, D., Park, S., Han, S.N., Crespi, N.: Dynamic Social Structure of Things: A Contextual Approach in CPSS. IEEE Internet Computing **19**(3), 12–20 (2015)
21. Iannella, R., Villata, S.: Odrl information model 2.2. W3C Recommendation (2018)
22. Information Commissioner’s Office (ICO) UK: Getting ready for the GDPR (2017). URL https://ico.org.uk/for-organisations/resources-and-support/data-protection-self-assessment/getting-ready-for-the-gdpr/
23. Kagal, L., Finin, T.W., Joshi, A.: A policy language for a pervasive computing environment. In: Proc. of POLICY, pp. 63– (2003)
24. Kitchenham, B.A., Budgen, D., Pearl Brereton, O.: Using mapping studies as the basis for further research - A participant-observer case study. Information and Software Technology **53**(6), 638–651 (2011)
25. Kolovski, V., Hendler, J., Parsia, B.: Analyzing web access control policies. In: Proc. of WWW, pp. 677–686 (2007)
26. Lebo, T., Sahoo, S., McGuinness, D.: Prov-o: The prov ontology. W3C Recommendation, April (2013)
27. Ly, L.T., Maggi, F.M., Montali, M., Rinderle-Ma, S., van der Aalst, W.M.: Compliance monitoring in business processes: Functionalities, application, and tool-support. Information systems **54**, 209–234 (2015)
28. Microsoft Trust Center: Detailed GDPR Assessment (2017). URL http://aka.ms/gdprdetailedassessment
29. Mont, M.C., Pearson, S., Bramhall, P.: Towards accountable management of identity and privacy: Sticky policies and enforceable tracing services. In: Database and Expert Systems Applications, pp. 377–382. IEEE (2003)
30. Motik, B., Patel-Schneider, P.F., Parsia, B.: OWL 2 Web Ontology Language – Structural Specification and Functional-Style Syntax (Second Edition). W3C Recommendation (2012)
31. Nymity: GDPR Compliance Toolkit (2017). URL https://www.nymity.com/gdpr-toolkit.aspx
32. Pandit, H., Lewis, D.: Modelling provenance for gdpr compliance using linked open data vocabularies. In: Proc of PrivOn (2017)
33. Peruzzini, M., Pellicciari, M.: A framework to design a human-centred adaptive manufacturing system for aging workers. Advanced Engineering Informatics **33**, 330–349 (2017)
34. Pulls, T., Peeters, R., Wouters, K.: Distributed privacy-preserving transparency logging. In: Proc. of WPES (2013)
35. Sabou, M., Musil, A.: Cityspin deliverable 2.1: Cyber-physical social systems blueprint (v.1) (2018). URL http://cityspin.net/wp-content/uploads/2017/10/D2.1.pdf
36. Sabou, M., Musil, A., Musil, J., Biffl, S.: Protocol for: A Systematic Mapping Study of Cyber-Physical Social Systems. Tech. Rep. IFS-QSE 18-02, TU Wien, Austria (2018). URL http://qse.ifs.tuwien.ac.at/publication/IFS-QSE-18-02.pdf
37. Sackmann, S., Strüker, J., Accorsi, R.: Personalization in privacy-aware highly dynamic systems. Communications of the ACM **49**(9) (2006)
38. Scherp, A., Saathoff, C., Franz, T., Staab, S.: Designing core ontologies. Appl. Ontol. **6**(3), 177–221 (2011). URL http://dl.acm.org/citation.cfm?id=2351285.2351289
39. Schreiber, G., Raimond, Y.: Rdf 1.1 primer (2014)
40. Smirnov, A., Shilov, N., Gusikhin, O.: Socio-cyberphysical System for Proactive Driver Support - Approach and Case Study. Proc. of ICINCO pp. 289–295 (2015)
41. Sutton, A., Samavi, R.: Blockchain enabled privacy audit logs. In: Proc. of ISWC, pp. 645–660 (2017)

42. Uszok, A. and Bradshaw, J.M. and Jeffers, R. and Suri, N. and Hayes, P.J. and Breedy, M.R. and Bunch, L. and Johnson, M. and Kulkarni, S. and Lott, J.: KAoS policy and domain services: Towards a description-logic approach to policy representation, deconfliction, and enforcement. In: Proc. of POLICY, pp. 93–96 (2003)
43. Wang, F.Y.: The emergence of intelligent enterprises: From CPS to CPSS. IEEE Intelligent Systems **25**(4), 85–88 (2010)
44. Xiao, J., Joseph, S.L., Zhang, X., Li, B., Li, X., Zhang, J.: An Assistive Navigation Framework for the Visually Impaired. IEEE Transactions on Human-Machine Systems **45**(5), 635–640 (2015)
45. Xiong, G., Zhu, F., Liu, X., Dong, X., Huang, W., Chen, S., Zhao, K.: Cyber-physical-social system in intelligent transportation. IEEE/CAA Journal of Automatica Sinica **2**(3), 320–333 (2015)
46. Zyskind, G., Nathan, O., et al.: Decentralizing privacy: Using blockchain to protect personal data. In: Proc. of SPW, pp. 180–184 (2015)

6. Legislative compliance assessment: Framework, model and GDPR instantiation

Bibliographic Information

Agarwal, S., Steyskal, S., Antunovic, F. and **Kirrane, S.**, 2018. Legislative compliance assessment: framework, model and GDPR instantiation. In Privacy Technologies and Policy: 6th Annual Privacy Forum, APF 2018, Barcelona, Spain, June 13-14, 2018. Springer International Publishing.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Writing - Review & Editing, and Supervision.

Copyright Notice

Legislative Compliance Assessment: Framework, Model and GDPR Instantiation

Sushant Agarwal, Simon Steyskal, Franjo Antunovic and Sabrina Kirrane

Vienna University of Economics and Business, Vienna, Austria
firstname.lastname@wu.ac.at

Abstract. Legislative compliance assessment tools are commonly used by companies to help them to understand their legal obligations. One of the primary limitations of existing tools is that they tend to consider each regulation in isolation. In this paper, we propose a flexible and modular compliance assessment framework that can support multiple legislations. Additionally, we describe our extension of the Open Digital Rights Language (ODRL) so that it can be used not only to represent digital rights but also legislative obligations, and discuss how the proposed model is used to develop a flexible compliance system, where changes to the obligations are automatically reflected in the compliance assessment tool. Finally, we demonstrate the effectiveness of the proposed approach through the development of a General Data Protection Regulatory model and compliance assessment tool.

Keywords: Compliance, GDPR, ODRL

1 Introduction

The interpretation of legal texts can be challenging, especially for people with non-legal backgrounds, as they often contain domain-specific definitions, cross-references and ambiguities [29]. Also, generally speaking legislations cannot be considered in isolation, for instance European Union (EU) regulations often contain opening clauses that permit Member States to introduce more restrictive local legislation. Additionally, depending on the legislative domain additional legislations may also need to be consulted. For example, when it comes to data protection in the EU, in addition to the General Data Protection Regulation (GDPR) [4], the upcoming e-privacy regulation (for e-communication sector) [5] or the Payment services (PSD 2) directive (for payments sector) [3] may also need to be consulted. As such, ensuring compliance with regulations can be a daunting task for many companies, who could potentially face hefty fines and reputation damage if not done properly. Consequently, companies often rely on legislative compliance assessment tools to provide guidance with respect to their legal obligations [8].

Over the years, several theoretical frameworks that support the modelling of legislation have been proposed [7, 10, 14, 22, 23, 25, 32], however only some of which were validated via the development of legal support systems [7,10,23,25,32]. One of the major drawbacks of such approaches is the fact that some do not consider concepts like soft-obligations (i.e. obligations that serve as recommendations rather than being mandatory) [22, 25] or exceptions (i.e. scenarios where the obligations are not applicable) [10, 29]. Additionally generally speaking the models are only loosely coupled with the actual legislation text, making it difficult to verify the effectiveness of such systems. More recently, a number of compliance assessment tools have been developed [18, 26, 28]. However, these systems are either composed of a handful of questions that are used to evaluate legal obligations [18] or do not filter out questions that are not applicable for the company completing the assessment [26, 28]. One of the primary drawbacks of existing compliance assessment tools is the fact that they do not currently consider related regulations.

In order to address this gap, we propose a generic legislative compliance assessment framework, that has been designed to support multiple legislations. Additionally, we extend the Open Digital Rights Language (ODRL) [34] (which is primarily used for rights expression) so that it can be used to express legislative obligations. Both of which are necessary first steps towards a context dependent compliance system that can easily be adapted for different regulatory domains.

The contributions of the paper are as follows:(i) we devise a flexible and modular compliance assessment framework, which is designed to support multiple legislations; (ii) we propose a legislative ODRL profile that can be used to model obligations specified in different legislations; and (iii) we develop a dynamic compliance system that can easily be adapted to work with different legislations. The proposed framework is instantiated in the form of a GDPR compliance assessment tool, which is subsequently compared with alternative approaches.

The remainder of the paper is structured as follows: Section 2 presents different approaches that can be used to model data protection legislations, along with compliance assessment tools for the GDPR. Section 3 details our framework that decouples the legislative obligations from the compliance assessment tool. Section 4 introduces our legislative model and illustrates how it can be used to model the GDPR. Section 5 describes the compliance tool. In Section 6 we compare and contrast our proposal with alternative solutions. Finally, Section 7 concludes the paper and presents directions for future work.

2 Related work

Although the modelling of legal text has been a field of study for many years, in this section we discuss those that focus on the modelling of data protection related legislations, and present three different tools that have been developed to help companies to comply with the GDPR.

Barth et al. [7] present a theoretical model for the representation of privacy expectations that is based on a contextual integrity framework [27]. The approach is validated via the modelling of the Health Insurance Portability and Accountability Act (HIPAA)[1]. Broadly speaking, the modelling is based on two kinds of norms, positive (allowed) and negative (denied). Using their framework privacy provisions for the sharing of data with different actors can be represented. However, according to Otto et al. [29] actions and purposes are not well represented. For instance, it is possible to model if a company cannot share personal data with a third party, but it fails to include purposes such as statistical reasons whereby a company may be allowed to share data.

May et al. [25] also illustrate how their approach can be used to model the HIPAA. Conditions and obligations are represented as access control rules that allow/deny operations. Given that they use a formal modelling language called Promela [16], it is possible to leverage existing Promela tools, such as for query execution. However, their model can only represent specific access-control related obligations. Other obligations, which are not related to access-control such as providing information about the processing or ensuring appropriate security measures are difficult to model with their approach.

Apart from legislative texts, policies for privacy notice and data exchange have also modelled. The World Wide Web Consortium (W3C) has undertaken numerous standardisation initiatives which deal with the modelling of data related policies. The Privacy Preferences Project (P3P)[2] is one such initiative which deals with representing privacy preferences in a standard machine-readable format. Using P3P we can model different parts of a privacy notice such as what information is collected, how long is it stored and for what purposes it would be used [12]. Though use of P3P can improve transparency of data processing, it does not support representation of other data protection related obligations [15]. For instance, obligations such as for security, data portability and right to erasure are out of scope for the P3P. Open Digital Rights Language (ODRL) [34] is another W3C initiative which presents a standard language to represent permission and obligations for digital content. The ODRL has also been used for modelling data protection legislations, for example Korba et al. [22] have used it to model the older data protection directive of the EU [1]. They have, however, discussed a high level overview of the modelling process for the directive. As a result, it does not include specific

[1] https://www.gpo.gov/fdsys/pkg/PLAW-104publ191/content-detail.html

[2] https://www.w3.org/P3P/

details to model components of the legislation such as soft-obligations (i.e., obligations that serve as recommendations rather than being mandatory) and exceptions to legal obligations.

In terms of the GDPR, the Information Commissioner's Office (ICO) in the UK have developed an online self-assessment tool [18]. It provides two separate checklists, one for controllers[3] and one for processors[4]. The applicable assessment questions are shown for a set of obligations. For every question the users have an option to see additional information. After the questions are answered, a report can be generated which summarises the compliance levels and suggests actions to ensure full compliance. The primary limitation of the tool is the fact that the questions do not assess the obligations in detail.

Microsoft has also developed a GDPR assessment tool [26]. Unlike the ICO tool, it is a spreadsheet based assessment i.e. users have to provide the input in the provided spreadsheet. The questions include references to the GDPR text for further reference. Questions are organised in a hierarchical way and categorised according to the associated concepts. After the input, a report can be generated.

Similar to the Microsoft's tool, Nymity has also developed a spreadsheet based assessment [28]. Obligations are referred to as *Privacy Management Activities*. Unlike Microsoft, the questions are not categorised but follow the order of the GDPR text, whereby each obligation is linked to the corresponding GDPR paragraph. The spreadsheet is designed to work with their commercial software, *Nymity Attestor*[5], through which a report can be generated.

Each of the aforementioned GDPR compliance assessment tools show a list of questions which do not have any contextual connections between them. For instance, even if consent is not the basis for processing, a user still needs to answer all questions for consent as the relations between the questions are missing. As a result, the user has to go through all the questions (162 questions for the Microsoft's tool), even questions which are not applicable, to finish the assessment. Also, surprisingly none of the tools currently consider related national or domain specific legislation.

3 Framework for a compliance assessment system

Due to the shift towards information and knowledge-driven economies, the use of software intensive information systems is increasing. When it comes to legislations such as the GDPR, companies need to ensure that the data processing and sharing carried out by such systems complies with relevant legal obligations. Ensuring compliance is important, otherwise non-compliance can lead to large penalties and reputation damage. As such, companies often rely on compliance assessment tools that can be used to help them to assess if their existing business processes and systems comply with relevant legal obligations.

From a requirements perspective, it is important that compliance tool vendors are able to demonstrate the exhaustiveness of their tool in terms of legal obligations, as wrong conclusions could potentially be drawn from incomplete assessments. Ensuring traceability i.e. providing references to the legislation text is considered to be important for such tools [9, 11, 29]. References, for instance, allow companies to consult the legislations in case of confusion or if they need to verify an assessment. Also, it is important that such tools are kept up-to-date and are capable of taking into account updated legal interpretation of the relevant regulations [9, 11, 20, 29]. For instance, the GDPR mentions *appropriate measures* for security (Article 32.1) where the measure of appropriateness can change over time.

To address these requirements, we propose a framework for compliance assessment, as depicted in Figure 1, which can be used to support multiple legislations as well as to manage changes in interpretation over time, by decoupling the data component from the compliance system.

[3] https://ico.org.uk/for-organisations/resources-and-support/data-protection-self-assessment/controllers-checklist

[4] https://ico.org.uk/for-organisations/resources-and-support/data-protection-self-assessment/processors-checklist

[5] https://www.nymity.com/solutions/attestor/

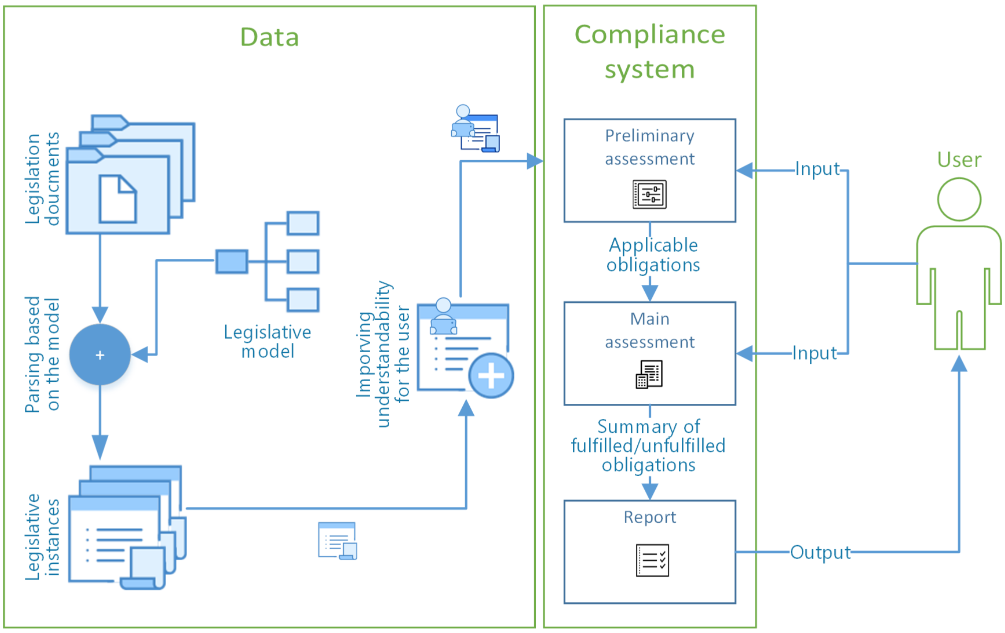


Fig. 1. Framework of the compliance tool

For the data component, a generic legislative model is used to represent legislative obligations and relations. For the *parsing process*, first the obligations are extracted from the legislations. Next the relations are identified between obligations and represented according to the legislative model. Following on from this the modelled obligations are translated into a format that can be read by the compliance system, referred to as *Legislative instance*. Finally, additional data is added to the instance such that the user can also understand the obligations.

The legislative instance is passed as input to the compliance system which assesses compliance based on the user-input and the legislative instance. In order to ensure that irrelevant questions are not shown to the user, the assessment process is divided into two steps: (i) preliminary assessment; and (ii) main assessment. In the first step, the legislative instance is read and input from the user is used to analyse the actions which govern the applicable obligations. Based on the input, the system shortlists the applicable obligations and presents the assessment to the user. In the main assessment the user provides input regarding the fulfilment of the obligations within their company. Once the required input is received, the system generates a report with a list of fulfilled and unfulfilled obligations.

Specific details on our implementation of the data component and the compliance system can be found in Sections 4 and 5 respectively.

4 Data modelling and the GDPR instance

In this section, we provide an overview of the proposed Open Digital Rights Language (ODRL) profile that can be used to model legislative obligations. Following on from this we provide a sequence of steps that are required in order to represent existing legislative text using the proposed model.

4.1 Legislative model

Like Korba et al. [22] we chose ODRL [34], which was released as a W3C Recommendation in February 2018, for modelling the regulation. ODRL provides a standard means to define policy expressions and licenses for digital content. The primary motivation for choosing ODRL is the fact

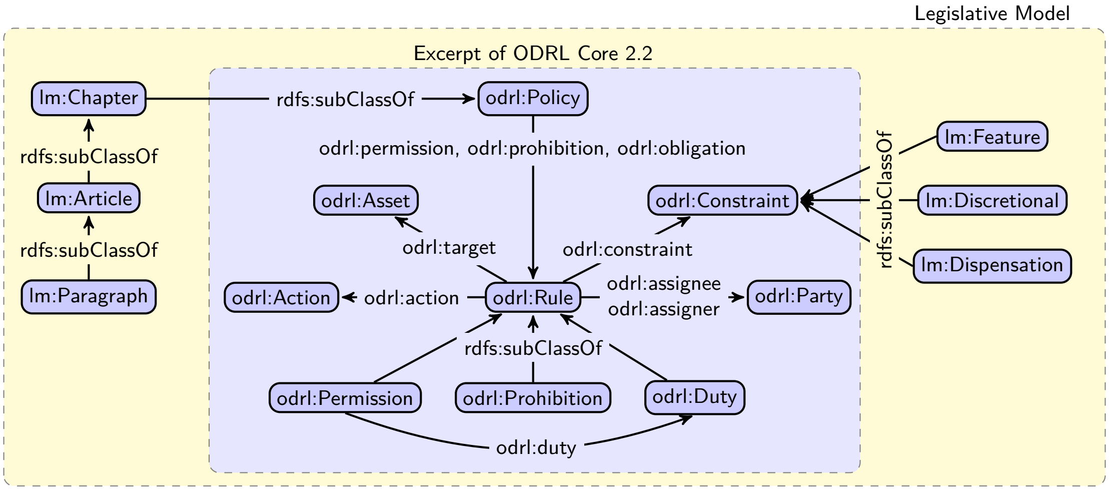
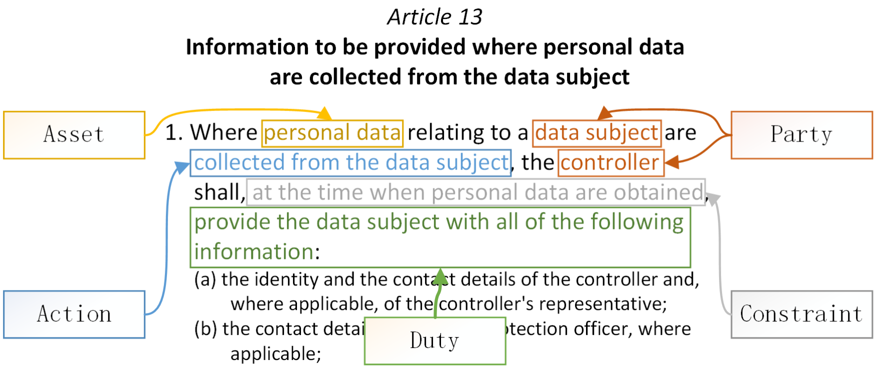


Fig. 2. The Legislative Model: based on an excerpt from ODRL Core 2.2 [34]



Article 13

Information to be provided where personal data are collected from the data subject

1. Where personal data relating to a data subject are collected from the data subject, the controller shall, at the time when personal data are obtained, provide the data subject with all of the following information:

(a) the identity and the contact details of the controller and, where applicable, of the controller's representative;

(b) the contact detai tection officer, where applicable;

Fig. 3. Breaking down Article 13.1 of the GDPR according to the ODRL model

that it can easily be extended for other use-cases such as representation of legislations by defining additional profiles[6].

The central entity of the ODRL model, as depicted in Figure 2, is a *Policy* which is used to specify *Rules* that are used to represent *Permissions*, *Prohibitions* and *Duties*. A *Permission* to perform an *Action* is granted if the associated *Duty* is fulfilled. While, an *Action* would not be allowed if any *Prohibition* is associated with it. Finally, a *Party* is an entity which participates in policy related transactions and an *Asset* is something which can be a subject to the policy under consideration.

Legal obligations are conceptually similar to ODRL duties. Consider Article 13 para 1 as depicted in Figure 3. In this example, personal data can be considered as an *Asset*, the controller and the data subjects are the involved *Parties*. While, the collection of personal data from the data subjects would be the *Action* for which the *Duty* is defined. Also, for this *Duty*, a *Constraint* is defined, which indicates that the *Duty* should be fulfilled at the time when personal data is obtained.

Unfortunately, it is not possible to represent the following concepts using the core ODRL model and vocabulary:

Soft obligations. The term soft-obligation refers to obligations which are non-mandatory. These are similar to recommendations in the sense that they represent best-practices. For instance,

[6] https://www.w3.org/TR/poe-ucr/

consider Example 1 where such a recommendation related to the use of icons is described. Here the text includes "*may be used*", which indicates that the use of icons is optional. As a result, it should not be represented as a *Duty*.

Example 1: Example of an optional constraint from the GDPR
Article 12.7: The information to be provided to data subjects pursuant to Articles 13 and 14 *may be provided* in combination with standardised icons....

Exceptions. Legislations also consist of exceptions, which if present take precedence over the *Duty*. Example 2 illustrates one such exception scenario where obligations defined in certain paragraphs are not applicable if the data subject already has the information.

Example 2: Example of an exception scenario from the GDPR
Article 13.4: Paragraphs 1, 2 and 3 *shall not apply* where and insofar as the data subject already has the information.

Characteristics. There are additional constraints defined in the legislations which describe the features or characteristics of an obligation. Such features should also be fulfilled, along with the corresponding obligations. Example 3 shows constraints such as *conciseness* and *transparency* which should be ensured in order to comply with the *duty* defined in Article 13, depicted in Figure 3).

Example 3: GDPR text defining characteristics
Article 12.1: ...provide any information referred to in Articles 13 ...*in a concise, transparent, intelligible and easily accessible form...*

References to the legislation text. Additionally concepts are also required in order to represent relations with the corresponding legal text, such that it is possible to provide a link to the actual legislative text.

In order to represent these concepts, we define a legislative profile and extend the core ODRL model, as illustrated in Figure 2. We use *Discretional* for the soft-obligations, *Dispensation* for representing exceptions and *Feature* for the characteristics. Also, in order to support referenceability, we define sub-components *Chapter*, *Article* and *Paragraph* under the *Policy* component.

4.2 Instantiation process

Considering the proposed ODRL legislative model, we now discuss the instantiation process that can be used to represent existing legislations in a standard format. The created instance is used as input for the compliance system. The process, as shown in Figure 4 is divided into 5 main steps - (a) filtration of text that relates to obligations; (b) identification of interconnections in the text; (c) normalisation of the text; (d) representation of text in a machine-readable format; and (e) enhancing the readability for the user. In the following, we elaborate on these steps.

(a) Filtration of text that relates to obligations Along with obligations, legislations usually discuss other topics such as the scope of the legislation, relevant definitions and fines for not adhering to the legislation. For a compliance assessment, we focus on the obligations for the stakeholder under consideration, like controllers and processors in the case of the GDPR. Thus, as the first step, the text which is not related to the obligations can be filtered out. For instance, in the GDPR, articles such as Articles 68-76 which define the working of the *European Data Protection Board* can be excluded as these do not introduce any obligations for the controllers or processors.

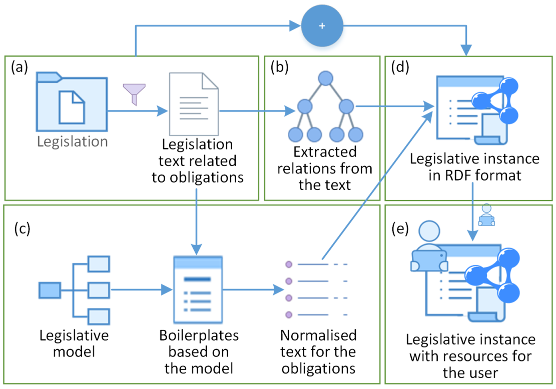

+
(a)
Legislation
Legislation text related to obligations
(b)
Extracted relations from the text
(d)
Legislative instance in RDF format
(c)
Legislative model
Boilerplates based on the model
Normalised text for the obligations
(e)
Legislative instance with resources for the user

Fig. 4. Steps involved for the instantiation process

(b) Identification of interconnections in the text To represent the filtered legal text as per the legislative model, we have to identify text related to the different components such as *Duty*, *Feature* and *Dispensation*. However, legislations consist of several references within the text to other paragraphs and articles [31]. Example 4 shows text stating connections with Article 13, 14, 15-22 and 34 defined in Article 12 para 1 of the GDPR.

Example 4: Example of the interconnections defined in GDPR
Article 12.1: The controller shall take appropriate measures to provide any information referred to in *Articles 13 and 14* and any communication under *Articles 15 to 22 and 34* relating to processing to the data subject in a concise, transparent....

Thus, connected components are defined in different paragraphs and articles. In order to include all such references for the legislative instance, we extract and document all of the defined relations.

(c) Normalisation of the text Next, we need to represent the legislation text according to the legislative model. To achieve this, it is necessary to manually identify and code parts of the text as components of the legislative model such as *Duty* and *Feature*. However, legislations often represent obligations in different legal styles, which increases the complexity of the coding process. Examples 5 and 6 illustrates two of the many different styles used in the GDPR.

Example 5: Example of the following style: <*processing*> *is lawful if*...<*condition*>
Article 8.1:...*processing* shall be lawful only if and to the extent that *consent is given or authorised by the holder of parental responsibility over the child*...

Example 6: Example of the following style: <processing> is prohibited unless...<condition>
Article 9.1: Processing of personal data revealing racial..origin...shall be prohibited.
Article 9.2: Paragraph 1 shall not apply if...: (a) the data subject..explicit consent...

In the case of Example 5, if <*processing*> would be the *Action* then <*condition*> i.e. authorising consent by the holder of parental responsibility would represent the *Duty*. Similarly, considering Example 6, if <*processing*> would be the *Action* then corresponding *Duty* would be to not perform the *action* as described in Article 9.1. Based on Article 9.2, <*condition*> i.e *explicit consent* would then be the dispensation scenario for the duty. However, this example can also be interpreted in a way similar to Example 5 where for the *Action* of <*processing*>, <*condition*> can also be considered as a *Duty*. Thus, different possibilities may exist for the representation of the text according to the components of the legislative model.

To overcome the confusion which arises due to different writing styles, in the field of requirements engineering, the use of boilerplates has been recommended which help in representing the text in

Table 1. Boilerplates used for expressing obligations in a standard style

Type	Boilerplate
Main	`Party` to perform `Action` on a given `Asset` should fulfil `Duty` in order to ensure compliance
Feature	`Duty` has additional requirement of `Feature` which must also be ensured
Dispensation	If `Dispensation` scenario for a `Duty` is true then that `Duty` is not applicable
Discretional	If `Discretional` for a `Duty` or `Feature` is true then that `Duty` or `Feature` is not compulsory

a standard form [6, 17, 24]. A boilerplate is defined as a natural language pattern that restricts the syntax of the sentences to pre-defined linguistic structures [6]. Example 7 illustrates a boilerplate to represent the previous examples in a standard format.

Example 7: Illustration of a boilerplate to represent Example 5 and 6 in a standard form
Boilerplate: <`Party`> to perform <`Action`> on a given <`Asset`> should fulfil <`Duty`>
- *Controller* to perform *Processing* on *Minors' data* should *Obtain consent by their parents*
- *Controller* to perform *Processing* on *Sensitive data* should *Obtain explicit consent for it*

This way, based on a boilerplate, we first represent the text in a standardised format. As we are interested in identification of components like *Action*, *Duty* and *Feature*, the boilerplates are based on the components of the legislative model and are listed in Table 1.

(d) Representation of text in a machine-readable format After the use of boilerplates, the obligations need to be expressed in a format which can be easily read by the compliance system and is standardised such that the data model can be reused for other systems as well. We chose, the Resource Description Framework (RDF) format [7] for the representation, which is also currently used for the exchange of legislation data in Europe[8]. To represent the obligations as RDF, Protege (an open-source ontology editor)[9] was used as it provides a simple GUI for accomplishing the task. Listing 1 shows a snippet of the text related to Article 13.1 of the GDPR in the RDF format. Using RDF, each triple, which is composed of a *subject-predicate-object* expression, asserts a binary relationship between two pieces of information. These triples are placed in common *namespaces*, referenced via *prefixes*. The prefix `odrl` represents the components from the ODRL model `<http://www.w3.org/ns/odrl/2/>`. The prefix `rdf` is used for the RDF built-in vocabulary, `lm` to denote the legislative vocabulary `<http://privacylab.at/vocabs/lm/>`, and `gdpr` for the GDPR instantiation `<http://privacylab.at/vocabs/gdpr/>`.

Listing 1: Snippet of the GDPR instance based on the duty from Article 13.1

```
gdpr:P13_1 rdf:type lm:Paragraph .
gdpr:P13_1 odrl:duty gdpr:ProvideInfo .
gdpr:ProvideInfo rdf:type odrl:Duty .
gdpr:ProvideInfo odrl:action gdpr:DirectCollection .
gdpr:ProvideInfo lm:dispensation gdpr:DataSubjecthasInfo .
gdpr:ProvideInfo lm:feature gdpr:Transparency .
gdpr:ProvideInfo lm:feature gdpr:Conciseness .
gdpr:ProvideInfo lm:discretional gdpr:Icons .
```

[7] https://www.w3.org/TR/rdf11-concepts/
[8] http://www.eli.fr/en/
[9] https://protege.stanford.edu/

In Example 4 we had illustrated an interconnection between Article 13 and 12. In Listing 1, along with representing the duty from Article 13.1, we also include connections to other articles and paragraphs. For instance, line 6 and 7 of the listing represent connections to *transparency* and *conciseness* from Article 12.1 as illustrated in Example 4. Similarly, line 5 of the listing represents the connection to the dispensation defined in Article 13.4 (see Example 2). Also, line 8 represents the discretional task of using privacy icons, illustrated in Example 1 from Article 12.7. Thus, the duty based on Article 13.1 is related to other parts of the text such as to Article 12.1, 12.7 and 13.4. These relations were established with the help of identified interconnections in step (b).

(e) Enhancing readability for the users In the RDF model, additional information such as legal definitions can be added by defining new data fields for the components. For instance, in the GDPR, Article 4 is dedicated for such definitions which can be added to a GDPR instance. Along with the resources such as definitions, in order to take input from the user, questions need to be added to the instance. This way, the compliance system can present the data model in form of a questionnaire. Example 8 illustrates some templates used for creating such questions. Using, the template, the *Duty* for providing the required information to the data subject (Article 13.1) would correspond to a question: "*Does your organisation ensure that the required information is provided to the data subject?*".

Example 8: Example for the structure of the questions
Action: Does your organisation (perform) <`Action`>?
Duty: Does your organisation (ensure) <`Duty`>?
Feature: Does your organisation (ensure) <`Feature`>?

Listing 2 illustrates how questions can be added to the instance. While, Listings 3 and 4 illustrate *Action* and *Feature* questions respectively.

Listing 2: Snippet of the GDPR instance from Listing 1 with the added question

```
gdpr:ProvideInfo rdf:type odrl:Duty .
gdpr:ProvideInfo odrl:action gdpr:DirectCollection .
gdpr:ProvideInfo lm:dispensation gdpr:DataSubjecthasInfo .
gdpr:ProvideInfo lm:feature gdpr:Transparency .
gdpr:ProvideInfo lm:feature gdpr:Conciseness .
gdpr:ProvideInfo lm:discretional gdpr:Icons .
gdpr:ProvideInfo lm:hasquestion "Does your organisation ensure that the
                 required information is provided to the data subject?" .
```

Listing 3: Illustration of an *Action* with added question

```
gdpr:DirectCollection rdf:type odrl:Action .
gdpr:DirectCollection lm:hasquestion "Does your organisation collect
               personal information directly from the data subjects?" .
```

Listing 4: Illustration of a *Feature* related to the duty from Listing 2

```
gdpr:Transparency rdf:type lm:Feature .
gdpr:Transparency lm:hasquestion "Does your organisation ensure
                          transparency for the provided information?" .
```

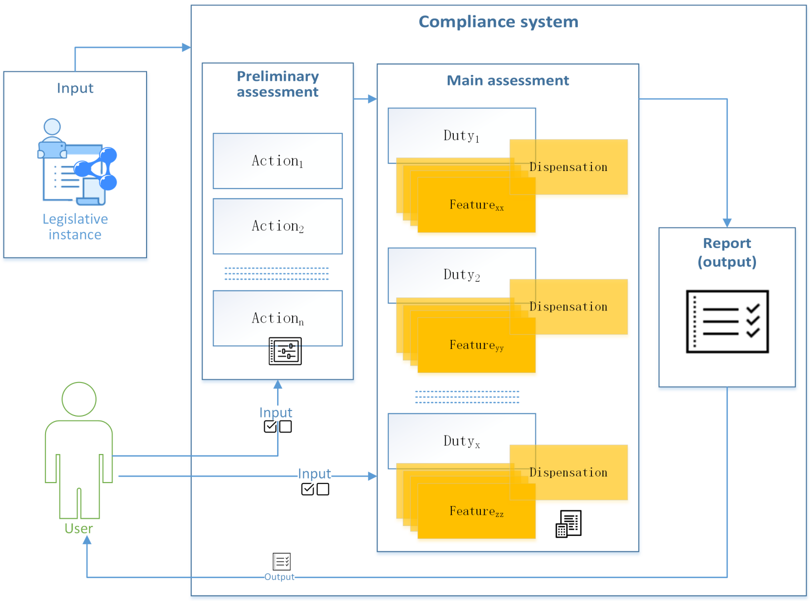
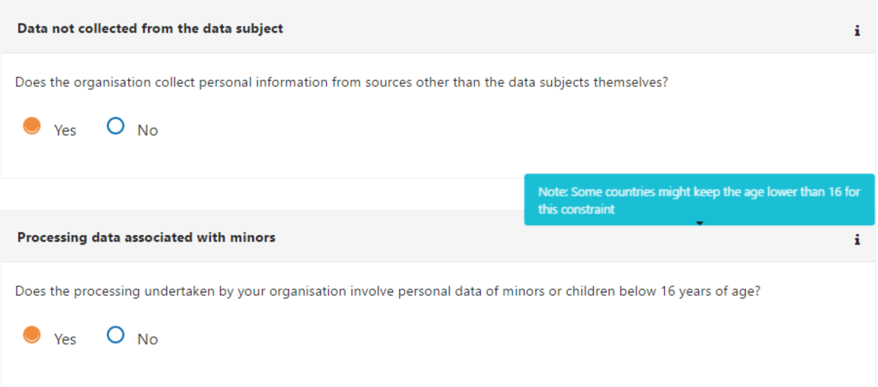


Fig. 5. Detailed process for the assessment of compliance



Data not collected from the data subject i

Does the organisation collect personal information from sources other than the data subjects themselves?

● Yes ○ No

Note: Some countries might keep the age lower than 16 for this constraint

Processing data associated with minors i

Does the processing undertaken by your organisation involve personal data of minors or children below 16 years of age?

● Yes ○ No

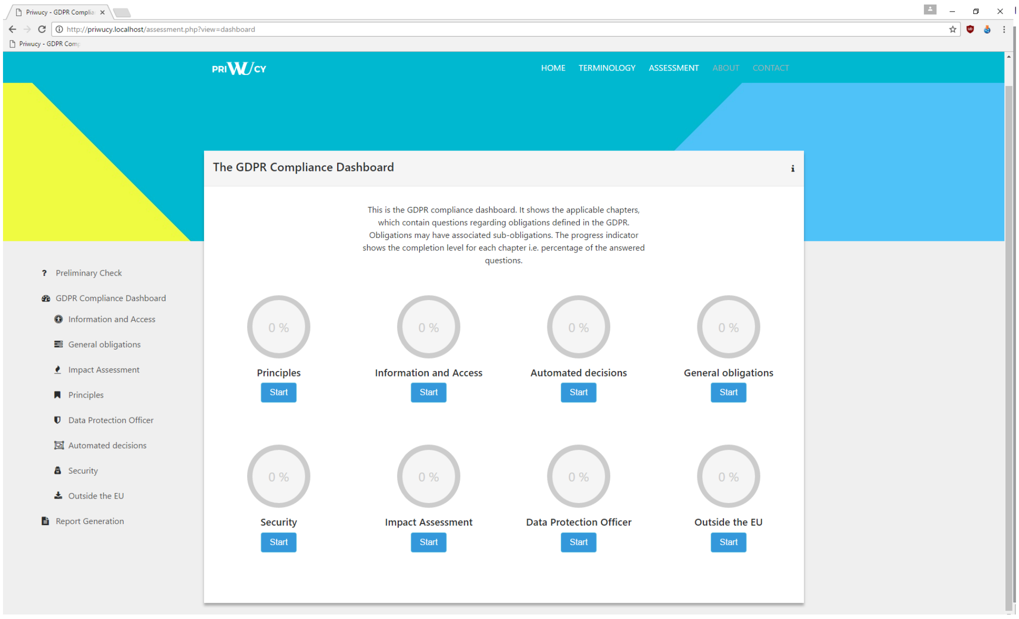
Fig. 6. A screenshot showing some questions from the preliminary analysis. The blue bubble shows additional information related to the question

5 The compliance system

After the definitions and questions are added to the legislative instance, it can be passed as input for the compliance system as shown in Figure 5. We now elaborate on the compliance system and discuss how it can be used for GDPR compliance assessment. For the assessment, we split the process into three parts: (i) preliminary assessment; (ii) main assessment; and (iii) report.

5.1 Preliminary assessment

The aim for the preliminary assessment is to find out the applicable obligations such that user does not have to identify and mark the non-applicable obligations similar to the existing tools [18, 26]. Based on the legislative model, as depicted in Figure 2, in order to perform *Action*, the associated *Duty* must be fulfilled Hence, the component *Action* can be used for the preliminary analysis to filter the applicable obligations. For instance, consider the *Action* illustrated in Listing 3. The *Duty*



The GDPR Compliance Dashboard

This is the GDPR compliance dashboard. It shows the applicable chapters, which contain questions regarding obligations defined in the GDPR. Obligations may have associated sub-obligations. The progress indicator shows the completion level for each chapter i.e. percentage of the answered questions.

Fig. 7. Dashboard based on the GDPR chapters for the main assessment

shown in Listing 2, based on the connection with the considered *Action*, would only be applicable if that *Action* is performed. As shown in Figure 6, a list of questions are presented to the user which can be answered as *Yes* or *No*. For every question, there exists a title to give some context for the question. In addition, on the top right corner of every question, "i" button has been provided to display the additional resources such as definitions or external links for further reference. Once the user submits all the answers, the system then uses this information to select the applicable parts which are associated with the actions where the user responds with a *Yes*.

5.2 Main assessment

Based on the selected *Actions*, all the associated *duties* are extracted from the instance. These *duties* are the basis for the main assessment. Referring back to Figure 2, the *Duty* component is connected to the constraints: *Feature*, *Dispensation* and *Discretional*. Thus, along with the *Duty*, other connected components are also presented to the user. Considering Listing 2, the assessment would also show the question for the *Duty* as well as for the connected components such as transparency, shown in Listing 4. Even after eliminating the non-applicable parts, the number of duties can be overwhelming to show as a flat list. Thus, in an attempt not to overwhelm the user with 100+ questions on a page, we group the questions, by clustering the questions according to the *chapters* as shown in Figure 7.

The user can start the assessment with any of the displayed chapters. Based on the preliminary assessment, the number of *chapters* shown may vary as the dashboard is dynamically created based on the applicable obligations. After the user selects a chapter, a list of questions is shown which is based on *duties* belonging to the selected *chapter*. Like the questions for the preliminary analysis, all questions for the main analysis have a short title and one "i" button on the top right corner. Initially, only the questions based on the *Duty* are shown. If the user selects *No* then nothing happens. However, if *Yes* is selected, a cascaded list of questions is displayed. These questions are based on the connected *Dispensation* and *Features*. By putting questions in a cascaded format, the user only sees the relevant parts. For instance, for duty illustrated in Listing 2, in case the user selects *No* for the question related to the *Duty* then the questions for the associated *features* like transparency, depicted in Listing 4 are not relevant and are not shown to the user. Only when the user selects *Yes* for the *Duty*, the related questions are shown. The user has the option to go back to the dashboard even when the all the questions have not been answered. The progress is saved and reflected as percentage complete on the dashboard.

5.3 Report

The last part for the compliance system is the report which provides a list of all the fulfilled and unfulfilled obligations. An obligation is considered to be fulfilled if a *Duty* is fulfilled along with all of the associated *Features*. *Duties* and *Features* represented as *Discretional* are also documented in the report. Along with the fulfilment status, references to the source (based on the *Articles* and *Paragraphs* which are defined in the legislative instance) are provided, such that users can refer to the legislation for additional information. Furthermore, fulfilled components (*Duty* and *Feature*) are shown in green boxes, *Discretional* components in orange and unfulfilled components are shown in red boxes.

6 Discussion

Our legislative model overcomes several of the challenges discussed in Section 2. It can represent both actions and purposes using the *Action* component of the model, which is one of the shortcoming for Bath et al's approach [7]. Also, as compared to May et al's approach [25] it can represent specifications for the obligations by using the *Feature* component. We have also considered soft-obligations and exceptions, which we refer to in our model as *Discretional* and *Dispensation* respectively.

To compare the capabilities of the compliance tools, we analyse 3 different capabilities: support for exceptions, management of evolving law and traceability. For the compliance tools, similar to legal modelling, **support for exceptions** is also important. For instance, in the GDPR, paragraphs like 17.3 define scenarios where obligation related to "right to be forgotten" is not applicable. Secondly, as law is considered to be dynamic where the interpretation involves based on amendments as well as on important judicial decisions [9, 11, 20, 29], the GDPR tools should support **management of evolving law** by ensuring provisions for updating the obligations according to the changes in the law. Lastly, **traceability** i.e. ensuring traceable references between the legal text and obligations is considered to be important [9, 11, 29]. References provide an overview of the articles and the paragraphs which a tool covers for the evaluation. With such traceable links, changes in the law can also be easily traced to the corresponding obligations defined for the tool.

Based on these criteria, in the following, we compare the GDPR compliance tools. The capabilities have been summarised in Table 2.

ICO The checklist for data protection self assessment provided by ICO [18] does not consider the exceptions. However, the questions can be answered as *not applicable* for cases where a user is aware of the exceptions. Also, as the checklist is web-based the updation of obligations can only be managed by the ICO. In terms of traceability, references to the GDPR text are missing which makes it difficult to analyse how much of the GDPR is covered by their tool.

Microsoft Microsoft's GDPR detailed assessment toolbox [26] also does not support exceptions but like ICO's tool provide an option to answer a question as *n/a*. As the tool is spreadsheet based, the users have an option to modify or update questions if any interpretation changes. The tool also provides references to the GDPR text. However, the references are not defined per obligation but rather for a group of obligations which makes it difficult to identify the reference of a single obligation.

Nymity Nymity's GDPR readiness spreadsheet [28] also does not support exceptions but the questions are framed in a way to exclude the exception scenarios. For instance, for obligation related to "right to be forgotten" the question includes "*where required by law*". The references are then provided to the corresponding article and paragraph and a user can then refer to the GDPR text to check if that obligation is applicable or not. Also, as this tool is also based a spreadsheet the user has the option to modify or update obligations if required.

Table 2. Comparison of the compliance tools

Tool	**Support for exceptions**	**Manage evolving law**	**Traceability**
ICO	**No** manual selection as N/A	**Limited** controlled by ICO	**No** references are absent
Microsoft	**No** manual selection as N/A	**Yes** editing the spreadsheet	**Limited** not defined individually
Nymity	**Limited** has conditional questions	**Yes** editing the spreadsheet	**Yes** references to paragraphs
PriWUcy	**Yes** represented as dispensation	**Limited** requires self-hosting	**Yes** references to paragraphs

PriWUcy In the data model as we defined a component *Dispensation* the exceptions are supported by the tool. For an obligation, if the dispensation is answered as *Yes* then that obligation would not be considered for the analysis. Like ICO's tool, PriWUcy is also web-based and users would not be able to change the obligations unless they self-host the tool. However, as the data component is decoupled from the user interface, updating the obligations based on the changes in the law would not be difficult. Also, by introducing *Chapter*, *Article* and *Paragraph* to the model, we were able to represent the references for all the obligations.

Currently, for the questions used for PriWUcy, we have used the terms as defined in the GDPR. For instance, consider the term transparency defined in Article 12.1 where the corresponding question in the tool is "Does your organisation ensure transparency for the provided information?" The use of the term *transparency* in the question introduces certain limitations regarding ambiguities. The question does not have a precise interpretation and for the user it is difficult to measure if transparency is ensured. Questions with such ambiguities can be confusing to answer. As a result, removing ambiguities is described as an important prerequisite for defining requirements for a system in the field of Requirements Engineering [2, 13, 33]. However, on the other hand, according to the legal literature, ambiguity in the legal texts can be intentional and should not be removed or resolved from the legal texts [29]. Moreover, resolving ambiguities can possibly result in wrong specification of the obligations [19]. So, in case if we do not resolve ambiguities then users may have different interpretations and might answer incorrectly. Also, if we resolve ambiguities, for instance describing transparency is some measurable form then we face of risk of misrepresentation of the GDPR text. This can lead to including a wrong question for the assessment which would lead to a wrong report. Either way, we risk ending up with a wrong assessment of compliance. Therefore, it is crucial to find a right balance for ambiguity in order to ensure correctness of the assessment.

7 Conclusions

In this paper, we described a flexible and modular compliance assessment framework, where changes to the legislative instances are automatically reflected in the compliance assessment tool. In addition we proposed a general legislative model and vocabulary based on the Open Digital Rights Language. In order to assess the effectiveness of the proposed framework and model we discuss how it can be used to model the General Data Protection Regulation. Additionally, we compare our compliance assessment tool with those provided by the Information Commissioner's Office (ICO) in the UK, Software vendor Microsoft, and a company called Nymity who provide tools and consultancy to privacy officers worldwide. Learning from one of the main shortcoming of the P3P [30] i.e. high complexity, we know that companies would also not adopt a compliance tool unless the complexity is kept to the minimum. Thus as a next step, we would work on the ambiguity issue such that the questions can be simplified without affecting the correctness of the questions from a legal perspective.

Also, although in this paper we focus on modelling the GDPR, in future work we plan to demonstrate how our legislative model can be used to express related legislative obligations, such as those found in the e-Privacy regulation or the Payment Services Directive. Additionally, we plan to explore automation techniques such as those investigated by Kiyavitskaya et al. [21], which are designed to automatically extract obligations from legal texts. Such techniques could potentially help in reducing the manual efforts required for the modelling process.

References

1. Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data. OJ **L 281**, 0031–0050 (1995-10-24), http://data.europa.eu/eli/dir/1995/46/oj
2. IEEE recommended practice for software requirements specifications: Approved 25 June 1998, IEEE Std, vol. 830-1998. IEEE, New York, NY (1998)
3. Directive (EU) 2015/2366 of the European Parliament and of the Council of 25 November 2015 on payment services in the internal market, amending Directives 2002/65/EC, 2009/110/EC and 2013/36/EU and Regulation (EU) No 1093/2010, and repealing Directive 2007/64/EC. OJ **L 337**, 35–127 (2015-12-23), http://data.europa.eu/eli/dir/2015/2366/oj
4. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). OJ **L 119**, 1–88 (2016-05-04), http://data.europa.eu/eli/reg/2016/679/oj
5. Proposal for a Regulation of the European Parliament and of the Council concerning the respect for private life and the protection of personal data in electronic communications and repealing Directive 2002/58/EC (Regulation on Privacy and Electronic Communications). COM(2017) **2017/03 (COD)** (2017-10-1)
6. Arora, C., Sabetzadeh, M., Briand, L.C., Zimmer, F.: Requirement boilerplates: Transition from manually-enforced to automatically-verifiable natural language patterns. In: 2014 IEEE 4th International Workshop on Requirements Patterns (RePa). pp. 1–8. IEEE (2014)
7. Barth, A., Datta, A., Mitchell, J.C., Nissenbaum, H.: Privacy and contextual integrity: Framework and applications. In: 2006 IEEE Symposium on Security and Privacy. p. 15. IEEE (2006)
8. Biasiotti, M., Francesconi, E., Palmirani, M., Sartor, G., Vitali, F.: Legal informatics and management of legislative documents. Global Center for ICT in Parliament Working Paper **2** (2008)
9. Boella, G., Humphreys, L., Muthuri, R., Rossi, P., van der Torre, L.: A critical analysis of legal requirements engineering from the perspective of legal practice. In: Requirements Engineering and Law (RELAW), 2014 IEEE 7th International Workshop on. pp. 14–21. IEEE (2014)
10. Breaux, T.D., Vail, M.W., Anton, A.I.: Towards Regulatory Compliance: Extracting Rights and Obligations to Align Requirements with Regulations. In: 14th IEEE International Requirements Engineering Conference (RE'06). pp. 49–58 (2006)
11. Breaux, T.D.: Legal requirements acquisition for the specification of legally compliant information systems. North Carolina State University (2009), http://www.lib.ncsu.edu/resolver/1840.16/3376
12. Cranor, L.F.: P3p: Making privacy policies more useful. IEEE Security & Privacy **99**(6), 50–55 (2003)
13. Génova, G., Fuentes, J.M., Llorens, J., Hurtado, O., Moreno, V.: A framework to measure and improve the quality of textual requirements. Requirements Engineering **18**(1), 25–41 (2013)
14. Ghanavati, S., Amyot, D., Peyton, L.: Towards a framework for tracking legal compliance in healthcare. In: International Conference on Advanced Information Systems Engineering. pp. 218–232. Springer (2007)
15. Grimm, R., Rossnagel, A.: P3P and the privacy legislation in Germany: can P3P help to protect privacy worldwide? In: Proc. ACM Multimedia, Nov (2000)
16. Holzmann, G.J.: Design and validation of protocols: A tutorial. Computer Networks and ISDN Systems **25**(9), 981–1017 (1993)
17. Hull, E., Jackson, K., Dick, J.: Requirements engineering. Practitioner series, Springer, London, 2nd ed. edn. (2005)
18. Information Commissioner's Office (ICO) UK: Getting ready for the GDPR (2017), https://ico.org.uk/for-organisations/resources-and-support/data-protection-self-assessment/getting-ready-for-the-gdpr/
19. Kamsties, E., Berry, D.M., Paech, B.: Detecting ambiguities in requirements documents using inspections. In: Proceedings of the first workshop on inspection in software engineering (WISE01). pp. 68–80. Citeseer (2001)

20. Kiyavitskaya, N., Krausová, A., Zannone, N.: Why eliciting and managing legal requirements is hard. In: Requirements Engineering and Law, 2008. RELAW'08. pp. 26–30. IEEE (2008)
21. Kiyavitskaya, N., Zeni, N., Breaux, T.D., Antón, A.I., Cordy, J.R., Mich, L., Mylopoulos, J.: Automating the extraction of rights and obligations for regulatory compliance. In: International Conference on Conceptual Modeling. pp. 154–168. Springer (2008)
22. Korba, L., Kenny, S.: Towards meeting the privacy challenge: Adapting drm. In: ACM Workshop on Digital Rights Management. pp. 118–136. Springer (2002)
23. Massacci, F., Prest, M., Zannone, N.: Using a security requirements engineering methodology in practice: The compliance with the Italian data protection legislation. Computer Standards & Interfaces **27**(5), 445–455 (2005)
24. Mavin, A., Wilkinson, P., Harwood, A., Novak, M.: Easy approach to requirements syntax (EARS). In: 17th IEEE International Requirements Engineering Conference. pp. 317–322. IEEE (2009)
25. May, M.J., Gunter, C.A., Lee, I.: Privacy APIs: Access control techniques to analyze and verify legal privacy policies. In: 19th IEEE Computer Security Foundations Workshop. p. 13. IEEE (2006)
26. Microsoft Trust Center: Detailed GDPR Assessment (2017), http://aka.ms/gdprdetailedassessment
27. Nissenbaum, H.: Privacy as Contextual Integrity Symposium - Technology, Values, and the Justice System. Washington Law Review **79** (2004)
28. Nymity: GDPR Compliance Toolkit, https://www.nymity.com/gdpr-toolkit.aspx
29. Otto, P.N., Antón, A.I.: Addressing legal requirements in requirements engineering. In: 15th IEEE International Requirements Engineering Conference (RE 2007). pp. 5–14. IEEE (2007)
30. Schwartz, A.: Looking back at P3P: Lessons for the future. Center for Democracy & Technology (2009), https://www.cdt.org/files/pdfs/P3P_Retro_Final_0.pdf
31. Sushant Agarwal, Sabrina Kirrane, Johannes Scharf: Modelling the General Data Protection Regulation. In: 20. Internationales Rechtsinformatik Symposion (IRIS) 2017, Feb 23, 2017 - Feb 25, 2017, Salzburg (2017)
32. Toval, A., Olmos, A., Piattini, M.: Legal requirements reuse: a critical success factor for requirements quality and personal data protection. In: Proceedings IEEE Joint International Conference on Requirements Engineering. pp. 95–103. IEEE (2002)
33. van Lamsweerde, A.: Requirements engineering: From system goals to UML models to software specifications, vol. 10. Chichester, UK: John Wiley & Sons and Wiley and Chichester : John Wiley [distributor], Hoboken, N.J. (2009)
34. W3C ODRL Community Group: ODRL Information Model 2.2 (2018), https://www.w3.org/TR/odrl-model/

7. The linked legal data landscape: linking legal data across different countries

Bibliographic Information

Filtz, E., **Kirrane, S.** and Polleres, A., 2021. The linked legal data landscape: linking legal data across different countries. Artificial Intelligence and Law, 29(4), pp.485-539.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Writing - Review & Editing, and Supervision.

Copyright Notice

Noname manuscript No.
(will be inserted by the editor)

The Linked Legal Data Landscape

Linking Legal Data Across different Countries

Erwin Filtz · Sabrina Kirrane · Axel Polleres

Received: date / Accepted: date

Abstract The European Union is working towards harmonizing legislation across Europe, in order to improve cross-border interchange of legal information. This goal is supported for instance via standards such as the European Law Identifier (ELI) and the European Case Law Identifier (ECLI), which provide technical specifications for Web identifiers and suggestions for vocabularies to be used to describe metadata pertaining to legal documents in a machine readable format. Notably, these ECLI and ELI metadata standards adhere to the RDF data format which forms the basis of Linked Data, and therefore have the potential to form a basis for a pan-European legal Knowledge Graph. Unfortunately, to date said specifications have only been partially adopted by EU member states. In this paper we describe a methodology to transform the existing legal information system used in Austria to such a legal knowledge graph covering different steps from modeling national specific aspects, to population, and finally the integration of legal data from other countries through linked data. We demonstrate the usefulness of this approach by exemplifying practical use cases from legal information search, which are not possible in an automated fashion so far.

Keywords Linked Data · legal knowledge graph · legal ontology · law identifier

1 Introduction

The law can be seen as a framework that consists of a set of orders defining the rules that govern society. There rules are set by an authority (legislative branch, eg. parliament), enforced by another authority (executive branch, eg. law enforcement authorities) and are defended and interpreted by yet another authority (judicial branch, eg. courts). In order to enable citizens to comply with the law it must be made publicly available. In former times laws were posted on official bulletin boards. Nowadays, legal information systems publicly accessible via the web are used for this purpose. For instance, the Austrian legal information system *Rechtsinformationssystem des Bundes (RIS)*[1] provided by the *Federal Ministry*

Erwin Filtz, Sabrina Kirrane, Axel Polleres
Vienna University of Economics and Business
Institute for Data, Process and Knowledge Management Vienna, Austria
E-mail: {firstname.lastname}@wu.ac.at, E-mail: {erwin.filtz}@siemens.com

[1] `https://www.ris.bka.gv.at/`

for Digital and Economic Affairs (BMDW)[2] is a central, publicly available, free of charge, web-accessible platform containing legal documents, such as legislations and court decisions, published by various Austrian authorities (e.g. legislative bodies on both a federal and a state level, courts and tribunals). In addition, jurisdictions have an official manner in which they publish legally binding amendments to existing laws or the abrogation of a law. These publications are usually called bulletins, law gazettes or have other specific names depending on the country.

Yet, despite having legal information publicly available, the documents contained in RIS (or, likewise, other national legal information systems) are not entirely linked with each other. That is, while legal professionals are able to infer links between legal documents and to understand cross-references within those documents by reading the text, the documents and the corresponding metadata are often stored in separate databases, making them hard to access – in particular for non-experts. The lack of integration often results in a tedious time-consuming legal information search process, for instance information may need to be retrieved from the judiciary database for the court decision, and the federal law database for legal provisions. This problem gets even worse when legal documents from other jurisdictions are involved, such as legislative acts from the EU that influence national law, or in the case of cross-boarder cases.

Representing legal information as Linked Data such that legal documents are linked across databases could therefore be highly beneficial, as such linking could speed up the legal information search process significantly and make legal information more accessible, by enabling structured queries and automated aggregation of and navigation through legal information interlinked in a machine-readable manner. Semantic technologies and Linked Data principles have already proven their effectiveness when it comes to data integration, and thus it is not surprising that researchers from the legal domain have already shown interest in the technology (Casanovas et al 2016). Based on the *Resource Description Framework (RDF)*[3], a data model that can be used to link data in a standardized, machine-interpretable manner, these technologies allow for the interlinking of data and metadata, making it possible to answer questions that cannot be answered easily at present – due to missing links in legal documents, missing integration of other available legal datasets (e.g. from other authorities not integrated in a legal information system or from other jurisdictions), etc.

The problem of tedious legal information search is obviously not unique to Austria. Other countries, governments and non-governmental initiatives, are also looking into linking legal data and enhancing their national legal information systems using semantic technologies. For instance, Finland provides access to legal information via the *Finlex Data Bank*[4], which has a web-based search interface and also allows for parts of the legal data to be downloaded in RDF (Oksanen et al 2019). Other countries, like Greece have set up programs[5] to increase transparency in the legal system and make it more accessible. However, additional steps are required in order to ensure that these separate national initiatives are interoperable. Towards this end, the European Union is working towards enuring better access and exchange of legal information across different countries. While each country is encouraged to set up or continue their own legal information systems – the EU proposes a common set of metadata for legislative and judiciary documents. The *European Legislation Identifier (ELI)*[6] and the

[2] https://www.bmdw.gv.at/en.html

[3] https://www.w3.org/RDF/

[4] https://www.finlex.fi/en/

[5] https://diavgeia.gov.gr/

[6] https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52012XG1026(01)

European Case Law Identifier (ECLI)[7] are non-binding proposals by the EU Council[8] to foster the exchange of legal information by providing legal documents with a minimum set of metadata. In light of increasing globalization and harmonization activities within the European Union it is important that all member states not only adopt the proposed ELI and ECLI ontologies, but also provide national extensions and schemes where required. To this end, our work is guided by the following hypothesis:

> *Interlinking national and international legal information from various sources and representing them as Linked Data in a Legal Knowledge Graph will enhance the legal information search process by extending querying possibilities that are not possible at the moment.*

The above hypothesis leads to the following research questions:

(i) Can existing ontologies be combined and extended in order to construct a legal knowledge graph?
(ii) Which approaches are needed in order to automatically populate the legal knowledge graph?
(iii) Is it possible to enhance the legal inquiry and search process by linking legal knowledge graphs from other countries?

In order to answer the aforementioned research questions it is necessary to compare the existing ontologies and their properties with the national requirements to determine where extensions are required. Furthermore, the sources of the entities required for the legal knowledge graph population need to be extracted from the document text using state of the art methods. Linking legal data across borders with data from other countries requires an analyses of the current situation regarding (linked) legal data in these countries. Towards this end, in this paper we make the following contributions[9]:

- We provide an overview of the knowledge graph construction process for our Legal Knowledge Graph (LKG), based on requirements derived from the Austrian legal system, and its current legal information system RIS;
- We propose several legal knowledge graph population methods and exemplify them using our Austrian use case scenario;
- We perform a comparison of rule based and deep-learning based approaches for the automatic extraction of legal entities from legal documents; and
- We provide a comparative analysis of the European legal knowledge graph landscape and identify key challenges and opportunities when it comes to integration across Europe.

The remainder of the paper is structured as follows: Section 2 presents the necessary background information on RDF and legal ontologies. The motivating use cases scenario and corresponding requirements used to guide our work are presented in Section 3. Our proposed legal knowledge graph construction and population process for Austrian legal data is presented in Section 4. Section 5 contains an overview of the current European legal knowledge graph landscape along with key challenges and opportunities when it comes to the integration of these different efforts. A critical discussion of different use case examples is provided in Section 6, followed by the discussion of related work in Section 7. Finally, Section 8 concludes the paper and discusses directions for future work.

[7] `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52011XG0429(01)`

[8] Body of the European Union composed of national ministers of each EU member state.

[9] Additional material is available under: `https://github.com/efiltz/legal-knowledge-graph`

```
1  PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2  PREFIX rdfs: <http //www.w3.org/2000/01/rdf-schema#>
3  PREFIX eli: <http://data.europa.eu/eli/ontology#>
4  PREFIX xsd: <http://www.w3.org/2001/XMLSchema#date>
5  PREFIX frbroo: <http://iflastandards.info/ns/fr/frbr/frbroo/>
6  <http://data.europa.eu/eli/dir/2014/92/oj>
7       rdf:type
8           eli:LegalResource ;
9       eli:type_document
10          <http://publications.europa.eu/resource/authority/resource-type/DIR> ;
11      eli:date_publication
12          "2014-08-28"^^xsd:date .
13 <http://data.europa.eu/eli/ontology#LegalResource>
14      rdfs:subclassOf frbroo:F1_Work .
```

Listing 1 RDF snippet for EU Directive 2014/92/EU (serialized in Turtle)

2 Background

Knowledge Graphs (Hogan et al 2020) are a trending topic, which is attracting increased interest in various domains: in order to organize and link information in a flexible manner, such knowledge graphs typically contain both factual and schematic (or, resp., ontological) information, in a flexible and extensible graph structure. Open standards and technologies to create, represent, interchange and process Knowledge Graphs origin from the Semantic Web and Data activities within the World Wide Web Consortium (W3C)[10]. In this section we provide background information on respective standards and principles, such as the Resource Description Framework (RDF) and Linked Data, and discuss existing legal ontologies that serve as a basis to create our legal knowledge graph.

2.1 Semantic Web and Linked Data

Legal information is typically represented as natural text with the information contained inside documents is not readily available in a machine readable format. When it comes to machine-readability the *Resource Description Framework (RDF)*[11] can be used to make metadata statements about a particular resource (e.g. in our case a legal provision or a court decision) which is identified by a *Unique Resource Identifier (URI)*. Listing 1 shows an RDF snippet about the EU directive 2014/92/EU. In the first five lines URI prefixes used to appreviate *namespaces* are defined, such that for instance `eli:LegalResource` turns into `http://data.europa.eu/eli/ontology#LegalResource` (line 8). An overview of the used namespaces in this paper is presented in Listing 2. Web URIs are represented using the *Hypertext Transfer Protocol (HTTP)*[12]. Besides URIs also typed and untyped *Literals* are used in RDF to describe properties of a certain resource. While *untyped* literals are always interpreted as text strings, *typed* literals may have a datatype that tells us how to interpret the information, for instance whether a string is to be interpreted as a textual string (`xsd:string`) or as a date (`xsd:date`), as shown in the example for property `eli:date_publication` in line 12. An RDF statement consists of the three components *subject, predicate, object* and is called a

[10] `https://www.w3.org/2001/sw/`

[11] `https://www.w3.org/TR/rdf11-concepts/`

[12] `https://tools.ietf.org/html/rfc2616`

```
prefix lkg: <https://data.wu.ac.at/legal/lkg#>
prefix av: <https://data.wu.ac.at/legal/austrovoc#>
prefix owl: <http://www.w3.org/2002/07/owl#>
prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
prefix rdfs: <http//www.w3.org/2000/01/rdf-schema#>
prefix dcterms: <http://purl.org/dc/terms/>
prefix skos: <http://www.w3.org/2004/02/skos/core#>
prefix xsd: <http://www.w3.org/2001/XMLSchema#date>
prefix cdm: <http://publications.europa.eu/ontology/cdm#>
prefix frbroo: <http://iflastandards.info/ns/fr/brbr/frbroo/>
prefix eli: <http://data.europa.eu/eli/ontology#>
prefix ev: <http://eurovoc.europa.eu/>
prefix gn: <http://sws.geonames.org/>
```

Listing 2 Namespaces used in examples throughout the paper (serialized in Turtle)

triple, which may also be viewed as a directed typed link or edge between subjects and objects. The so connected RDF triples form a graph structure. A collection of triples describing schema and instance data is called *ontology*. Although RDF can be serialized in various formats (e.g. RDF/XML[13], N-Triples[14]) in this paper we use the Terse RDF Triple Language (Turtle)[15] due to its simplicity and readability. Additional formats include *RDF in Attributes (RDFa)*[16], which is used to embed RDF in HTML and XML documents, or JSON-LD[17].

RDF Schema (RDFS)[18] and the *Web Ontology Language (OWL)*[19] are used to describe classes of and properties (relations) between resources. The core features of RDFS are summarized in the ρdf subset (Muñoz et al 2009), which contains properties to define simple taxonomies in terms of class (`rdfs:subClassOf`) and property (`rdfs:subPropertyOf`) hierarchies. In such a hierarchy, implicit superproperties between resources, as well as membership in the superclass from membership in the subclass can be inferred. Likewise, domain (`rdfs:domain`) and range (`rdfs:range`) restrictions can be used to infer the class membership of subjects or objects of particular properties as shown in Listing 1 line numbers 13 and 14 that the ELI class `eli:LegalResource` is a subclass of `frbroo:F1_Work`. OWL caters for the definition of more complex ontological axioms on classes and properties, which can be used for more complex reasoning.

The *SPARQL Protocol and RDF Query Language (SPARQL)*[20] is used to retrieve RDF data. SPARQL queries search for matches of user defined triples (*graph patterns*). A `SELECT` query allows users to define a graph pattern which must match the data and the variables to be returned. Basic graph patterns must match all results in order to be returned, whereas in an `OPTIONAL` query we can also define *optional patterns* that need not occur in all results and return an empty binding if not matched. With *alternative patterns* using `UNION` it is possible to define multiple graph patterns of which at least one must be fulfilled. The number of results can be reduced using a `FILTER` clause, which allows users to restrict results to literals that contain a particular string, or to apply comparison operators such as equals, greater than and so on, for instance as shown in Example 1. Long query result lists can be manipulated using

[13] `https://www.w3.org/TR/rdf-syntax-grammar/`

[14] `https://www.w3.org/TR/n-triples/`

[15] `https://www.w3.org/TR/turtle/`

[16] `https://www.w3.org/TR/rdfa-primer/`

[17] `https://www.w3.org/TR/json-ld11/`

[18] `https://www.w3.org/TR/rdf-schema/`

[19] `https://www.w3.org/TR/owl2-overview/`

[20] `https://www.w3.org/TR/sparql11-overview/`

solution modifiers such as `ORDER BY`, which sorts the results in an ascending or descending order based on the given variable, as well as `LIMIT` and `OFFSET` to restrict the number of results.

Example 1 SPARQL Query: Which EU directives have been published in 2014?

```
PREFIX eli: <http://data.europa.eu/eli/ontology#>
PREFIX eu: <http://publications.europa.eu/resource/authority/resource-type/>
SELECT (?s as ?Directive)
WHERE {
  ?s eli:type_document eu:DIR .
  ?s eli:date_publication ?d .
  FILTER (year(?d) = 2014)
}
```

Directive
<http://data.europa.eu/eli/dir/2014/23/oj>
<http://data.europa.eu/eli/dir/2014/92/oj>
...

In order to make machine-readable data more accessible on the Web, Tim Berners-Lee (Berners-Lee 2006) proposed a set of *Linked Data Principles* for publishing data on the Web, which fundamentally rely on RDF:

1. Use URIs as names for things.
2. Use HTTP URIs so that people can look up those names.
3. When someone looks up a URI provide useful information using the standards RDF and SPARQL.
4. Include links to other URIs, so that they can discover more things.

The *things* mentioned in the first principle refer to resources. Identifying resources with HTTP URIs allows the consumers to retrieve additional information about these resources on the Web. Information about the resources stored in RDF allows them to be retrieved using SPARQL. The fourth rule stipulates that resources should be linked with other resources and shall allow users or agents to browse through different resources by following links.

2.2 Legal Ontologies

We base our modeling on the ELI and ECLI ontologies which are specific to the legal domain, as well as the European EuroVoc thesaurus which is also available as an RDF vocabulary. Both the ELI and ECLI ontologies have been proposed in the form of conclusions of the Council of the European Union which consists of EU member states' ministers of the respective policy area. Conclusions are documents that express a political expression without the intent of legal effects. *EuroVoc* is a standardized thesaurus containing normative terminology used in the context of European administration and publications, not restricted to legislation alone. In addition to ELI and ECLI we also introduce the Common Data Model (CDM) which is used by the EU to model their legal data.

European Law Identifier. The European Law Identifier (ELI) (Council of the European Union 2012) serves as a common system to identify legislative documents and its metadata first proposed in 2011 and is followed by additional Council conclusions in 2017 (Council of the European Union 2017) acknowledging the efforts of the participating countries, introducing an ELI task force and clarifying the three pillars of the ELI system. The three

Table 1 Mandatory properties of the ELI ontology

Property	Description
`eli:realizes`	Describes that a legal expression materializes a legal resource.
`eli:embodies`	Describes that a format represents a legal expression.
`eli:type_document`	Indicates the type of a legal resource.
`eli:language`	The language in which a legal expression is written.
`eli:title`	The title of a legal expression.
`eli:format`	Resource format expressed as URI (e.g. HTML).

pillars (Francart et al 2018) the ELI is built on are: (i) to foster the assignment of unique identifiers for laws; (ii) to use a common ontology that provides a metadata standard; and (iii) to provide said metadata in a machine-readable form. As for classes and properties in the ELI ontology, for instance, the EU is required to publish legal acts in various languages and therefore needs the ability to represent different language versions of the same legal act. The ELI ontology distinguishes between three classes of resources and six mandatory properties. As shown in Table 1, a `eli:LegalResource` is a distinct intellectual creation such as a legal act which is realized by a `eli:LegalExpression` and embodied in a specific `eli:Format`. Hence, a `eli:LegalExpression` has a `eli:title` and `eli:realizes` the base version in a particular language (`eli:language`) of a `eli:LegalResource` which is of a specific `eli:type_document`, for instance a directive. The `eli:LegalExpression` is published in a `eli:Format` which is the actual physical representation, whereas physical includes paper as well as electronic formats such as HTML or PDF.

The ELI (both in terms of identifier syntax and in terms of the usage of metadata properties) is modeled in different ways from country to country depending on the respective legal system. Notably, the Council conclusions defines all of the syntactic components of the ELI being optional, such that national requirements can be fulfilled and not all components need to be implemented in each national legal system. Additional information for the member states as well as reference files for the ELI ontology are provided in HTML[21], XLSX[22] and OWL[23] format. The ELI follows the principles set forth in the Functional Requirements for Bibliographic Records[24] (FRBR) ontology (Publications Office of the European Union 2020b) but uses the object-oriented version of FRBR[25] for the ELI ontology (prefix `frbroo:`), for instance `eli:LegalResource` is a `rdfs:subClassOf frbroo:F1_Work` and `eli:LegalExpression` is a `rdfs:subClassOf frbroo:F22_Self-Contained_Expression`. The ELI syntax is very flexible and can be adjusted to national requirements by adding and removing individual components. The syntax of the ELI identifier is defined as the base URI followed by *eli* with the rest of the components being optional and separated by slashes, for instance the ELI for a EU directive such as `http://data.europa.eu/eli/dir/2014/92/oj` looks different from an Austrian legal provision `https://www.ris.bka.gv.at/eli/bgbl/1979/140/P28a/NOR40180997`.

[21] http://publications.europa.eu/resource/distribution/eli_documentation/html/doc_user_manual/eli_ontology.html

[22] http://publications.europa.eu/resource/distribution/eli/xlsx/owl/eli_ontology.xlsx

[23] http://publications.europa.eu/resource/distribution/eli/owl/owl/eli.owl

[24] https://www.ifla.org/publications/functional-requirements-for-bibliographic-records

[25] https://www.ifla.org/files/assets/cataloguing/FRBRoo/frbroo_v_2.4.pdf

Table 2 Mandatory properties of the ECLI ontology

Property	Description
`dcterms:identifier`	The URL where the resource can be retrieved.
`dcterms:isVersionOf`	Indicates that a resource is a version of another resource.
`dcterms:creator`	Full name of deciding court.
`dcterms:coverage`	Indicates the country in which the court or tribunal has its seat.
`dcterms:date`	The date when a decision has been rendered.
`dcterms:language`	The language in which this particular is written.
`dcterms:publisher`	The organization that is responsible for the publication of the document.
`dcterms:accessRights`	Defines who can access the resource, *public* or *private*.
`dcterms:type`	Defines the type of the rendered decision.

```
<http://eurovoc.europa.eu/2836>
  a skos:Concept;
  skos:broader
    ev:138;
  skos:prefLabel
    "Verbraucherschutz"@de, "consumer protection"@en .
<http://eurovoc.europa.eu/138>
    skos:prefLabel
        "Verbraucher"@de, "consumer"@en .
```

Listing 3 Example EuroVoc (serialized in Turtle)

European Case Law Identifier. The European Case Law Identifier (ECLI) (Council of the European Union 2011) has been created to introduce an identifier for *case law*, and to define a minimum set of metadata for judiciary documents (e.g. court decisions). The ECLI does not define any specific classes and uses the properties of the Dublin Core Metadata Initiative (DCMI)[26] ontology with the prefix `dcterms`. In contrast to the ELI there is no separate formal ontology specification provided by the EU, but rather only a recommendation of nine mandatory (listed in Table 2) and eight optional properties which should be used to describe metadata relating to the documents. Moreover, the ECLI conclusion makes particular suggestions for the use of the `dcterms` vocabulary, for instance that the object of `dcterms:coverage` should be used for the country (or more closely defined location) where the court is seated. Unfortunately, these suggestions are given without explicit ontological commitments or formal axioms, e.g. in terms of explicit range restrictions.

The syntax of the ECLI identifier is more restricted compared to the ELI as it consists of five components separated by a double colon, for instance `ECLI:AT:OGH0002:2016:0100OB00012.16M.1220.000` for a decision of the Austrian Supreme Court. The order of the components is fixed and starts with the abbreviation *ECLI* and is followed by a country code (or code of an international organization). The third component is the court code of the deciding court which is individually assigned by each participating country and the year of the decision. The last component is an unique ordinal number of the decision.

EuroVoc. The *EuroVoc* thesaurus[27] is a multi-domain and multi-lingual thesaurus provided by the Publications Office of the European Union (OP) used to classify EU documents into categories for easier information search. It is based on the Simple Knowledge Organiza-

[26] `https://dublincore.org/`

[27] `https://op.europa.eu/s/n3kP`

tion System (SKOS)[28], a well-known standard[29] to represent information using RDF. The individual terms in the EuroVoc thesaurus are of type skos:Concept and a collection of concepts is aggregated in a skos:ConceptScheme. Concepts are linked using the properties skos:narrower and skos:broader to represent the hierarchical structure of terms and skos:related for associative relations. EuroVoc is organized in 21 domains, for instance *Law, Economics, Trade* and 127 microthesauri. In total, EuroVoc contains more than 6,000 concepts and each concept has one preferred term (skos:prefLabel) and (optional multiple) non-preferred terms (skos:altLabel), i.e. synonyms. All concepts are available in the languages of the 23 EU member states and in addition three languages of EU membership candidate countries. The concepts are arranged in a way to avoid polihierarchies except for the Geography domain. Listing 3 shows a snippet of concept ev:2836 with its preferred labels in German (*Verbraucherschutz@de*) and English (*consumer protection@en*) having a skos:broader concept *ev:138* which is labeled *Verbraucher@de* and *consumer@en*.

Common Data Model. The Publications Office of the European Union (OP) uses the Common Data Model (CDM)[30] for their published resources which is based on FRBR (Francesconi et al 2015; Publications Office of the European Union 2020a). The resources that can be accessed via the Eur-Lex SPARQL endpoint are represented using the CDM ontology rather than the ELI and ECLI ontology. An RDF dump of the Eur-Lex data using ELI, up until 2018, is available on the EU Open Data Portal[31]. The usage of the CDM ontology results in using a different identifier for the documents in the Eur-Lex database CELLAR, the repository of the EU Publications Office, instead of the ELI identifier. A mapping between CELLAR and ELI identifiers is however provided using the predicate owl:sameAs.

3 Use Case & Requirements: a Case for Legal Linked Data in Austria

The work presented herein is based on a project commissioned by the Austrian Ministry for Digital and Economic Affairs (BMDW)[32]. The goal of this project was to investigate how the current Austrian legal information system RIS could be improved in terms of searchability and accessibility by: (i) transforming the metadata from RIS into a legal knowledge graph; (ii) further enriched with information extracted from document texts stored within RIS; and (iii) automatically interlinking these legal documents. In the following we provide an overview of the Austrian legal information system and the challenges, requirements and scenarios addressed in the course of the project.

Austrian legal information system. The *Rechtsinformationssystem des Bundes (RIS)*[33] is the legal information system of the Republic of Austria. RIS serves as a single point of information from which legal documents issued by various authorities can be searched and accessed. In addition to the web interface, RIS also provides access to its data via a REST API[34] enabling users to access RIS data in JSON[35]. Through the web interface different backend

[28] https://www.w3.org/2004/02/skos/
[29] https://www.w3.org/2009/08/skos-reference/skos.html
[30] https://op.europa.eu/en/web/eu-vocabularies/model/-/resource/dataset/cdm
[31] http://data.europa.eu/88u/dataset/eli-european-legislation-identifier-eurlex
[32] https://www.bmdw.gv.at/
[33] https://www.ris.bka.gv.at/
[34] https://data.bka.gv.at/ris/api/v2.5/
[35] https://tools.ietf.org/html/rfc8259

databases – subdivided into different parts of the legislation – such as *Bundesrecht* (federal law), *Landesrecht* (state law of the nine Austrian states) or *Judikatur* (judiciary) and many more – can be accessed. Documents in RIS can be retrieved in different formats like HTML, XML, RTF and PDF. Although the RIS web interface gives the impression that it is a single database containing all legal information, it is in fact a collection of independent databases which are not currently connected nor interlinked underneath.

Use case. Currently the search process is mainly based on basic keyword search with the possibility to add filters to restrict the search space for instance to timeframes by setting dates. The objective of the project was threefold: (i) develop a legal information system that is capable of also representing related information, i.e. links to other legal documents referenced within a document, to classify documents based on a classification schema; (ii) to allow for enhanced search capabilities by making certain information contained in documents explicit, for instance linking entities mentioned in the documents to external knowledge bases such as Geonames or DBpedia; and (iii) to support cross-jurisdictional search requests by integrating legal data from other countries and the European Union. The end goal being to allow us to seamlessly get answers to complex search queries such as the following:

- Which documents are referenced in a specific court decision?
- Over which districts does a court have competent jurisdiction?
- What are the national transpositions of a specific EU directive?
- Which legal documents regulate a specific legal area searched with keywords in a foreign language?

Challenges. Primary challenges in the context of the project and the use case in order to facilitate the answering of such complex questions in a more automated manner include the following:

Unstructured/missing information. Information about legal documents can be contained in both structured metadata but also within unstructured text, for instance law references in court decisions are not contained in metadata. Further, some connections between documents are only implicitly available in the text and while these can be detected by a human reader, a machine would struggle with the same task. In addition, the mandatory and optional properties within the ELI and ECLI ontologies can only be partially constructed from the document metadata alone.

Data silos. The Council identified the need to disseminate legal information and that the identification and exchange of legal information from national authorities supports access to legal information[36]. At the moment these legal information systems are still separate silos. Our objective is that Linking legal data first nationally across so far disconnected backend databases *and*, as a second step, across Europe will help to reduce the problem of data silos. It is worth noting that automatic extraction from and linkage of existing databases should avoid any need to maintain the same information at multiple places, while also allowing the data to be easily integrated with other sources.

Redundant data storage. Considering that legal documents contain references to each other, the legal information search process typically involves the need to search across the different databases. At the moment, additional information that should be made available for full text search but is not part of the particular database is stored in an additional column.

[36] 2011/C127/01, 2012/C325/02: Identification of needs

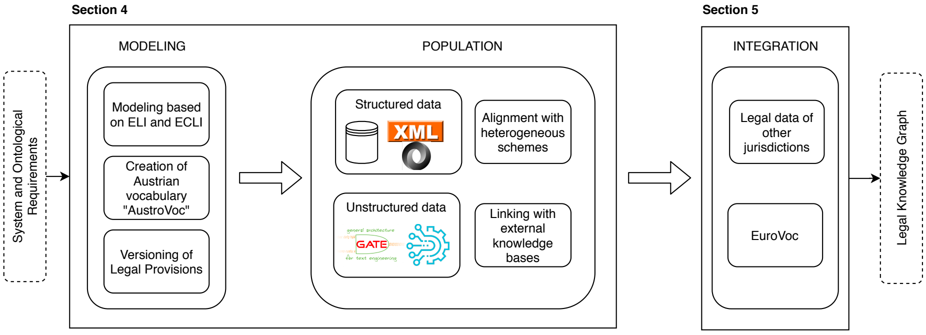

Section 4
Section 5
System and Ontological Requirements
MODELING
Modeling based on ELI and ECLI
Creation of Austrian vocabulary "AustroVoc"
Versioning of Legal Provisions
POPULATION
Structured data
XML
Alignment with heterogeneous schemes
Unstructured data
GATE
Linking with external knowledge bases
INTEGRATION
Legal data of other jurisdictions
EuroVoc
Legal Knowledge Graph

Fig. 1 Legal Knowledge Graph Creation Methodology

Still, this leads to redundant data storage and does not add any beneficial additional information except enabling search. Furthermore, this situation results in anomalies which must be considered on insert, update and deletion operations. Linked data helps to avoid these anomalies as it does not require to store the same information redundantly at multiple places and therefore provides more flexibility.

Requirements. From the challenges outlined above we derive three core requirements. It must be possible to **extract** information that is missing in the metadata from the document text. We need to **integrate** legal data from various national and international data sources into a single knowledge base. **Normalization** by assigning unique identifiers instead of plain text references should be used to avoid redundancies and inconsistencies.

Legal Knowledge Graph Creation Methodology. The aforementioned legal ontologies and use case requirements serve as an input for the legal knowledge graph creation process, which is depicted in Figure 1. In the first step we model the ontology to represent the Austrian legal system based on ELI and ECLI and create a national thesaurus *AustroVoc* for the representation of Austrian specific terms, not covered in existing terminologies such as EuroVoc. Since ELI and ECLI are only describing a minimum set of metadata in order to be applicable to all EU member states, we needed to create additional classes and properties for our legal knowledge graph to reflect Austrian specific requirements. Content Ontology Design Patterns can help to create (legal) domain-specific ontologies, for instance already shown for the modeling of licensing (Rodríguez-Doncel et al 2013) and consumer complaints (Santos et al 2016), and provide building blocks to ensure reusability (Presutti and Gangemi 2008): in our particular case we can build on the already existing ELI and ECLI ontologies. However, on the one hand the existing ontologies are in parts not fine-grained enough and on the other hand legal documents and their metadata provide us with additional required information on the missing parts. Therefore, we extended these ontologies in a middle-out fashion, which seems appropriate in combining top-down and bottom-up approaches and helps us to keep an adequate level of detail (Uschold and Gruninger 1996). In the bottom-up phase we analyzed the available metadata and which additional, relevant data could be extracted from the Austrian legal documents (using Natural Language Processing techniques) in order to populate classes and properties that need to be added, keeping in mind our primary goal is inter-linking of the documents, rather than describing the actual content of the documents. In the top-down phase we reused the existing ontologies and refined and extended classes, properties, as well as taxonomic terminologies/thesauri, where needed. This approach has also been described to be effective in the legal domain in a similar setting with existing legal ontologies that are extended based on underlying legal documents (Ghosh et al 2016). Based on the resulting combined ontological schema, the resulting model has been populated with data from RIS and linked to external knowledge bases. Both steps are described in Section 4. In a final step, described in Section 5, we integrate external legal data from the European Union, the European thesaurus *EuroVoc* containing terms from different domains in the official languages of the EU member states and also legal data from selected other countries.

4 The Austrian Legal Knowledge Graph

In this section we describe how we map explicit metadata information in **the Austrian legal information system** RIS as well as implicit information contained within the RIS documents to the ELI and ECLI ontological models introduced in Section 2.2. This mapping is used to form the foundations of our legal knowledge graph. Furthermore, we introduce a national vocabulary *AustroVoc* which is mapped to EuroVoc where possible. Finally, the model is populated with data from RIS and linked with external knowledge bases.

4.1 Legal Knowledge Graph Modeling

Given that our project was commissioned by the Austrian Ministry of Digital and Economic Affairs, who are interested in participating in the European linked legal data initiatives, we model our Austrian legal knowledge graph based on the ELI and ECLI ontologies. This decision was motivated by the fact that: (i) By doing so the ministry contribute towards the goals of ELI and ECLI as laid out in the Council conclusions for the introduction of ELI and ECLI; (ii) The EU is a supranational system that aims to provide easier access to and interlinking of legal information across Europe, which can only be successful if the various member states participate and use the same system; and (iii) It is possible to accommodate specific national requirements by extending the ELI and ECLI ontologies with classes and properties specific to the Austrian legal system, such that information contained in RIS for which ELI and ECLI do not provide properties can be represented. Such an approach is also common practice in other countries, for instance the Finnish Semantic Finlex Legislation Ontology or the Greek Nomothesia ontology.

When it comes to alternative modeling approaches, Francesconi et al (2015) highlight the disadvantages of coupling resources with the corresponding FRBR classes stating that such a coupling leads to complex queries that are needed in order to retrieve metadata for all FRBR levels (resource, expression, etc...). Although the proposed alternative modeling reduces complexity it does so at the cost of interoperability, which is one of the core requirements underpinning our work. Considering, that linking is necessary to support the legal inquiry process across different jurisdictions, the proposed optimization needs to be built into the ELI and ECLI standards. The incorporation of the proposed optimization and others coming from the research community will be discussed later in Section 5.

4.1.1 Modeling the Austrian legal system based on ELI and ECLI

Since both ELI and ECLI are targeting a variety of different legal systems within the EU member states, they only provide two classes of legal documents, which we extended in order to represent specific legal document types used in Austria's national legal publication

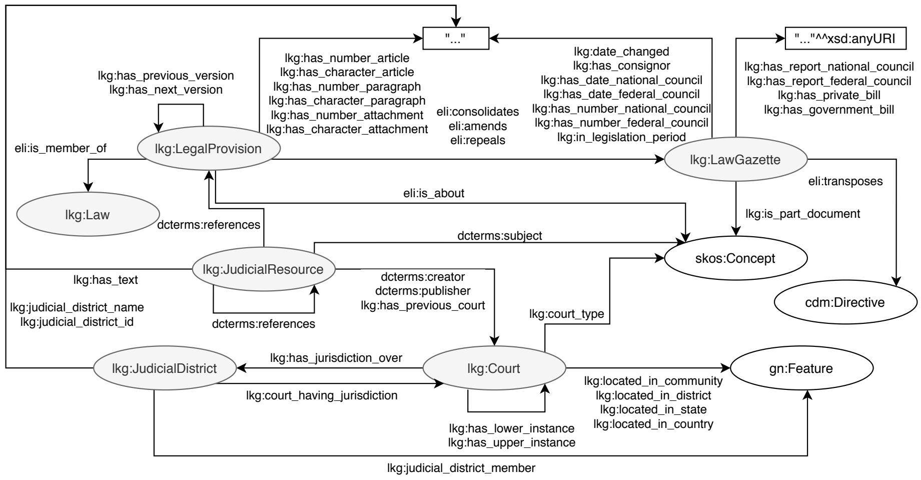


Fig. 2 Legal Knowledge Graph Model

process, such as law gazettes and legal provisions. In our examples herein we exemplify our legal knowledge graph with a focus on federal law as well as jurisdiction by the justice branch, which includes decisions of the Supreme Court and lower courts. Figure 2 depicts our legal knowledge graph model with the specific classes we added colored gray. Nodes denote classes and edges properties connecting their respective domain and range classes.

Law Gazette. The law gazette is used to publish new laws or any changes to existing laws, which happen in editorial instructions (e.g. *[...] in § X change amount Y to Z [...]*). We represent the law gazette with class `lkg:LawGazette` (subclass of `eli:LegalResource`).p We introduce new properties to provide background information about the legislative process which is a useful source to solve legal interpretation problems. These properties cover dates when law changes have been discussed in the councils (`lkg:has_date_national_council`, `lkg:has_report_national_council`) and links to the reports about the parliamentary discussion[37] which are available on the web (`lkg:has_report_national_council, lkg:has_report_federal_council`). These reports are useful in case there is a loophole in the law and the will of the parliament needs to be discovered. Bills initiate the legislative process and are linked using the properties `lkg:has_private_bill` and `lkg:has_government_bill`. The authority bringing in a bill is indicated with the property `lkg:has_consignor`. We use `lkg:is_part_document` to determine the type of the law gazette such as *constitutional law* or *order*. The legislation period in which a law gazette has been published is included for legal analysis and is indicated using the property `lkg:in_legislation_period`.

Legal Provision and Law. A `lkg:LegalProvision` (subclass of `eli:LegalResource`) is a resource containing the actual norm. In Austria each legal provision is an individual document with a *NOR* number as an unique technical identifier, for instance *NOR40180997* (see Listing 4) and a label used in legal practice, for instance *§ 28a KSchG* (Paragraph 28a of the Consumer Protection Law). Figure 3 shows the legal provisions *Artikel 2 B-VG* (Art. 2 of the

[37] Publicly available at the Austrian parliament's website: `https://www.parlament.gv.at/`

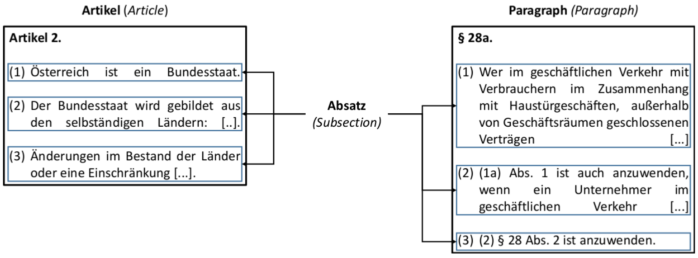

Artikel (Article)
Artikel 2.
(1) Österreich ist ein Bundesstaat.
(2) Der Bundesstaat wird gebildet aus den selbständigen Ländern: [..].
(3) Änderungen im Bestand der Länder oder eine Einschränkung [...].
Absatz
(Subsection)
Paragraph (Paragraph)
§ 28a.
(1) Wer im geschäftlichen Verkehr mit Verbrauchern im Zusammenhang mit Haustürgeschäften, außerhalb von Geschäftsräumen geschlossenen Verträgen [...]
(2) (1a) Abs. 1 ist auch anzuwenden, wenn ein Unternehmer im geschäftlichen Verkehr [...]
(3) (2) § 28 Abs. 2 ist anzuwenden.

Fig. 3 Legal provision naming convention

```
<https://www.ris.bka.gv.at/eli/BGBl/1979/140/P28a/NOR40180997>
        lkg:has_number_paragraph
                28 ;
        lkg:has_character_paragraph
                "a" ;
        lkg:has_next_version
                ris:eli/BGBl/1979/140/P28a/NOR40192489 ;
        lkg:has_previous_version
                ris:eli/BGBl/1979/140/P28a/NOR40173437 .
```

Listing 4 Legal Provision §28a Consumer Protection Law (shortened, serialized in Turtle)

Constitution) and *§ 28a KSchG*. A legal provision can be labeled *Artikel* (article) or *Paragraph* (paragraph) and is always seen in its entirety for modeling, irrespective of whether there is only one *Absatz*[38] (subsection) or multiple subsections.

Listing 4 depicts an RDF snippet for legal provision *§ 28a KSchG* with the new properties we introduced in our extended `lkg:` ontology highlighted in red. Besides the *Artikel* and *Paragraph* there is also a *Anlage* (attachment) usually used for transitional provisions which combines both *Artikel* and *Paragraph*, for instance *Artikel 1 § 1*. We introduce new properties to model numbers as well as characters in the labels of legal provisions, for instance `lkg:has_number_paragraph` and `lkg:has_character_paragraph`. Analogously, for legal provisions named by article or attachment we use the properties `lkg:has_number_article`, `lkg:has_character_article` and `lkg:has_number_attachment`, `lkg:has_character_attachment` respectively. Two temporally subsequent legal provisions are linked with `lkg:has_next_version` and `lkg:has_previous_version`. We create the class `lkg:Law` because legal provisions can be a part of a *law book* which is a collection of legal provisions containing regulations about the same topic. The membership between a `lkg:LegalProvision` and `lkg:Law` is indicated with the ELI property `eli:is_member_of`.

Legal provisions are the basis for court decisions and it is therefore important to link a judgment with the correct version of a legal provision. The linking between judgments and legal provisions is achieved by following a *date-based linking approach* which links a judgment to the legal provision that is in force at the decision date because this will be the correct version most of the time. Furthermore, a specific version of a legal provision is always the sum of the initial version with all its amendments over time.

Judicial Resource. The class `lkg:JudicialResource` (subclass of `frbroo:F1_Work`) is used for judiciary documents which are modeled based on the ECLI suggestions. We add the text

[38] The English translation of *Absatz* is *paragraph*, but we call the *Absatz* subsection to avoid confusion, as the word *Paragraph* in Austrian/German legal language rather refers to law articles.

of a court decision with the property lkg:has_text. The EU Publications Office (OP) provides *Named Authority Lists (NAL)* which are vocabularies to standardize the inter-institutional legal data exchange. Some of these NAL can be used by all countries, for instance the NALs for languages or countries, while other NAL are very EU-specific, for instance court-types which contain EU courts only and therefore cannot be used for national courts. We use these NALs for the ECLI properties that indicate in which country the deciding court is seated (dcterms:coverage), the language of the decision (dcterms:language) and the access rights (dcterms:accessRights). Properties populated with Austrian specific values, such as dcterms:type, dcterms:publisher, lkg:previousCourt, are linked with concepts contained in the *AustroVoc* thesaurus we created for this purpose.

Court and Judicial District. A judgment in the judiciary branch is rendered by a lkg:Court of a specific type indicated with lkg:court_type. Furthermore courts are organized in a hierachical manner and have a higher instance indicated with the property lkg:has_upper_instance and a lower instance (lkg:has_lower_instance). A court is located in a community (lkg:located_in_community), district (lkg:located_in_district), state (lkg:located_in_state) and country (lkg:located_in_country). A district court also lkg:has_jurisdiction_over a lkg:JudicialDistrict[39]. Similarly, the property lkg:court_having_jurisdiction indicates the court having spatial competent jurisdiction. The competent jurisdiction is assigned to the lowest level of authorities, hence district courts. Since we know that a district court has competent jurisdiction over a particular area and that court has an upper instance we can also infer that a higher court has competent jurisdiction over all areas of all lower courts assigned to the higher court. To represent spatial information we us the publicly available database *Geonames*[40], which provides identifiers and spatial information for locations in multiple languages as well as a small ontology (prefix gn:) describing these properties. Figure 4 illustrates the difference between political and judicial districts for the capital of Austria, Vienna which is divided into 23 political districts but only 12 judicial districts. The two political districts *Leopoldstadt* (gn:2772614) and *Brigittenau* (gn:2781400) are the members (lkg:judicial_district_member) of the single judicial district named (lkg:judicial_district_name) *Leopoldstadt.*

4.1.2 The Austrian Vocabulary - AustroVoc

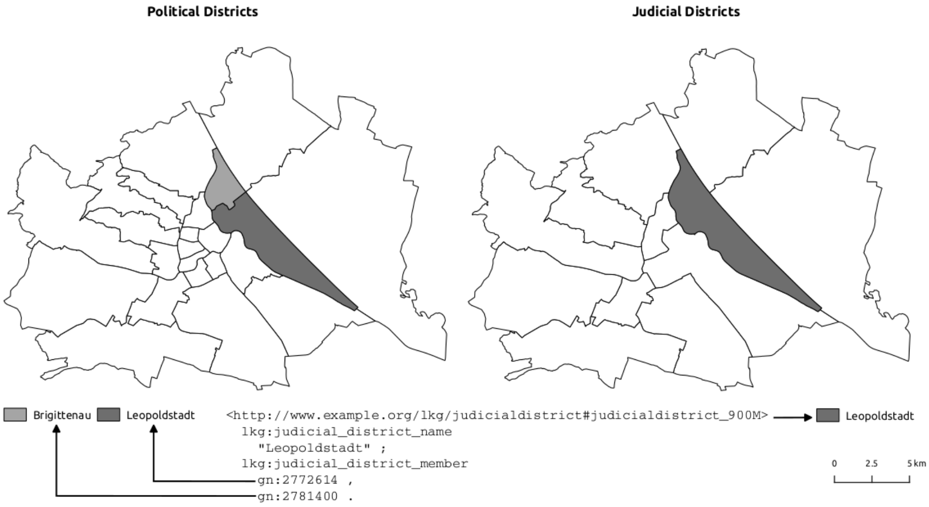
We propose a SKOS-based thesaurus *AustroVoc* containing Austrian specific terminology. ELI and ECLI encourage member states to create their own schema for the properties indicating a document type (eli:type_document and dcterms:type) and a document classification to describe the content or legal area of a document (eli:is_about and dcterms:subject). We create three different schemes for *Gericht-typ* (court type), *Bundesrechtindex* (law index) and *Resource-typ* (resource-type).

Gericht typ. The court types provided in the *Named Authority Lists (NAL)*[41] of the EU Publications Office cannot be used ‘as is’ since they only contain EU courts. That is why we create an additional *court-type* scheme which contains the different types of Austrian courts.

[39] https://www.statistik.at/web_de/klassifikationen/regionale_gliederungen/gerichtsbezirke/index.html

[40] https://www.geonames.org/

[41] https://op.europa.eu/en/web/eu-vocabularies/at-dataset/-/resource/dataset/court-type



```
<http://www.example.org/lkg/judicialdistrict#judicialdistrict_900M>
  lkg:judicial_district_name
    "Leopoldstadt" ;
  lkg:judicial_district_member
    gn:2772614 ,
    gn:2781400 .
```

Fig. 4 Illustration of political and judicial districts for the Austrian capital Vienna.

```
<https://www.ris.bka.gv.at/eli/BGBl/1979/140/P28a/NOR40180997>
    eli:is_about :bri2006 .
:bri2006 a skos:Concept;
    skos:broader :bri20;
    skos:prefLabel "Konsumentenschutz"@de ;
    rdfs:seeAlso ev:2836 .
```

Listing 5 Law index example (shortened, serialized in Turtle)

We distinguish between public tribunals, for instance the Constitutional Court (`av:vfgh`), and ordinary courts, for instance the Supreme Court (`av:ogh`), which are responsible for different legal areas and are organized in a hierarchical way. Adding this information enables a search for judgments rendered by courts of a particular type and superior or subordinate courts and legal analysis.

Bundesrechtindex. The *law index* is an index for Austrian federal law[42] provided by RIS organizing the law in a hierarchical manner. As shown in Listing 5 every legal provision is assigned to an entry in this index with the property `eli:is_about` which allows users to search for legal provisions belonging to a specific legal area, for instance §28a KSchG linked to the law index `av:bri2006`. We also use the law index to indicate the legal area of judgments dependent on the legal provisions they are based on using `dcterms:subject`. Finally, where possible (for details, see Section 4.2.3 below) we link the national law index items with corresponding items to the European thesaurus *EuroVoc* using the property `rdfs:seeAlso` to enable a multi-lingual search across jurisdictions. For instance, the AustroVoc law index `av:bri2006` (*Konsumentenschutz@de*) is linked to the EuroVoc concept `ev:2836` (*Verbraucherschutz@de*).

[42] `https://www.ris.bka.gv.at/UI/Bund/Bundesnormen/IndexBundesrecht.aspx?TabbedMenuSelection=BundesrechtTab`

Resource typ. As with the court-types mentioned above, the resource-types contained in the NAL[43] are EU specific and incomplete with regards to missing specific resources used and required in Austria. We again created our own schema for such specific *resource-types* in RIS. These mainly include different document types, for instance judiciary documents can be subdivided into *Entscheidungstext* (decision text) or a *Rechtssatz* (legal rule) which is a case summary from which general legal rules can be inferred. The properties used to indicate the document types are already available in ELI (`eli:type_document`) and ECLI (`dcterms:type`). These properties to indicate the document types are not to be confused with the property `rdf:type` that is used to indicate to which class a document belongs to, for instance judiciary documents are of type `lkg:JudicialResource` and legislative documents are of type `lkg:LawGazette` or `lkg:LegalProvision`.

4.2 Legal Knowledge Graph Population

We describe different approaches to populate our legal knowledge graph with structured data from the RIS database. While some entities and their relationships can directly be extracted from structured metadata within RIS, for the population from unstructured (text) data we make use of Natural Language Processing (NLP) tools and techniques and provide a comparison of different (rule-based as well as machine-learning based) legal entity extraction approaches exemplified with a dataset of manually annotated court decisions.

4.2.1 Population from structured data

For the population from structured data we were provided with a dump of the relational RIS database which contains the metadata as well as the text of the legal documents contained in RIS. The database schema used does not satisfy the ELI or ECLI metadata requirements upfront. In addition, each RIS application is currently stored in a separate relational database.

Direct population. A direct mapping (in analogy with the terminology used in R2ML (W3C Recommendation 2012))[44] of the legal knowledge by mapping attributes to URLs is possible where the required metadata is available. This is typically applicable to properties that have a literal as an object and preprocessing of the data is limited to a minimum, for instance transforming a date from datetime to date format, for instance for the properties `dcterms:date`, `dcterms:issued`, `eli:first_date_entry_in_force`, `eli:date_no_longer_in_force`, `eli:date_document` and `eli:date_publication` in ISO 8601[45] format (YYYY-MM-DD). Other properties that have a literal as their object, such as `eli:title`, `eli:title_short` and `eli:title_alternative`, are transformed without modification.

Indirect population. This approach is used when there is data available in a structured format that cannot be directly fed into the legal knowledge graph, for instance in case of resource types represented as simple strings in the database which need to be mapped to/replaced with the AustroVoc vocabulary terms based on mappings between the input and the output data, or where linking requires additional lookups or conditionals. In more detail, RIS

[43] `https://op.europa.eu/en/web/eu-vocabularies/at-dataset/-/resource/dataset/resource-type`

[44] However as opposed to the strict definition in the R2RML standard, note that we speak herein also about direct mapping, when minor, straightforward syntactic literal transformations are applied.

[45] `https://www.iso.org/iso-8601-date-and-time-format.html`

```
<https://data.wu.ac.at/legal/court#court_8>
    rdf:type
        lkg:Court ;
    rdfs:label
        "Bezirksgericht Leopoldstadt" ;
    lkg:cout_type
        av:bg ;
    lkg:located_in_community
        <http://sws.geonames.org/2772614/> ;
    lkg:located_in_country
        <http://sws.geonames.org/2782113/> ;
    lkg:located_in_district
        <http://sws.geonames.org/2761333/> ;
    lkg:located_in_state
        <http://sws.geonames.org/2761367/> ;
    rdfs:seeAlso
        <https://www.openstreetmap.org/relation/1651546> .
```

Listing 6 Example *Bezirksgericht Leopoldstadt* (shortened, serialized in Turtle)

document types are indicated as strings or integers in the database but we created a concept scheme `av:resource-types` as suggested by the ELI and ECLI ontologies in AustroVoc. For instance, a legal provision of type *"BG"* (federal law) is replaced with the AustroVoc concept `av:leg_bg`, where the resource can be linked to its type using the properties `eli:type_document` for legislative documents and `dcterms:type` for judiciary documents. We proceed similarly when it comes to mapping the law index of legal provisions using the property `eli:is_about`. The law index item is also replaced with the corresponding `av:bundesrechtindex`. To assign judiciary documents a class we use the legal provisions mentioned in the text, look up the law index for each of the found legal provisions and assign the law index to the judiciary document in order to populate the `dcterms:subject` property for each judiciary document. Furthermore, references extracted from the document text are strings which need to be replaced with the actual URI of the referenced documents and linked using the `dcterms:references` and `eli:cited_by_case_law` properties.

Population by interlinking external sources. Although the RIS database contains relevant legal information – for instance, legal provisions and court decisions – it does not provide additional structured background information that could also be interesting in terms of enhancing the legal search process by adding respective search attributes as well as enabling advanced analysis of the legal system. Such background information includes for instance spatio-temporal information about geographic entities or events mentioned in court decision, for instance the deciding courts or case relevant dates. Similar techniques for enhancing search by interlinking information from spatio-temporal knowledge graphs have already proven successful for Open Data search (Neumaier and Polleres 2019). As for geo-references, we enhance the court information with external data from Nominatim[46], the search engine of OpenStreetMap (OSM)[47], and Geonames[48] from which we get an RDF dump we import in our legal knowledge graph. In order to get information about the Austrian courts we compile a list of court names and query Nominatim for address information, for instance for

[46] `https://nominatim.openstreetmap.org/`

[47] `https://www.openstreetmap.org/`

[48] `https://www.geonames.org/`

Bezirksgericht Leopoldstadt[49]. The result has an entry *display_name* containing address information such as street, community, district, state and country. We extract this information and use Geonames in order to populate the properties `lkg:located_in_community`, `lkg:located_in_district`, `lkg:located_in_state` and `lkg:located_in_country` as shown in Listing 6, where the new information is highlighted in red. In addition we also include the OSM court information page using `rdfs:seeAlso` which allows users of the legal information system to retrieve location and contact information for the respective authorities.

4.2.2 Population from unstructured data

While some of the structured information contained in the RIS metadata is incomplete or not all attributes we are interested in are covered as metadata fields, some of this missing information can be extracted from the document text using Natural Language Processing (NLP) tools and techniques. Extracting entities from a text and classifying them into a set of classes (e.g. person, organization, etc...) is called Named Entity Recognition (NER) (Grishman and Sundheim 1996). In our case we extract legal entities, such as courts, legal provisions and law gazettes. For instance, court decisions contain references to other documents that are not available in the metadata, such as legal provisions and legal rules mentioned in the court decision text. We note though, that rather than structured hyperlinks, the references used in legal practice are oriented on the use by humans and therefore use simple textual labels such as *§ 28a KSchG* rather than URIs like `https://www.ris.bka.gv.at/eli/BGBl/1979/140/P28a/NOR40180997` to reference a legal provision. In order to transform such unstructured references to machine-readable links in our KG we therefore extract such textual entities to find corresponding ELI or ECLI identifiers of referenced documents, linking both documents with the properties `dcterms:references` (`lkg:JudicialResource` –> `lkg:LegalProvision`) and vice versa `eli:cited_by_case_law` (`lkg:LegalProvision` –> `lkg:JudicialResource`). Multiple approaches are available to extract information from document text, which could help us to link the documents with each other. We herein specifically compare a *rule-based approach* used in combination with *gazetteers* with more advanced approaches such as *conditional random fields* and *deep learning*. A comparative assessment of these orthogonal approaches helps to increase confidence in the extraction results in the legal domain.

Corpus. For a performance comparison between the different approaches we need an annotated training corpus of legal documents. To the best of our knowledge, there is no gold standard Austrian legal corpus available, thus we manually annotate 50 randomly selected decision texts from the Justice branch. The documents have are quite varied in length with an average of 11,669 tokens with ± 7,741.88 tokens standard deviation (SD), and 260.12 (± 262.71 SD) sentences. For the population of our knowledge graph we extract the following legal entities: *Case reference* is a reference to another decision text which is used to refer to decisions taken or arguments brought up in previous cases. In the corpus a document contains on average 33 (± 23 SD) case references. *Contributor* contains the names of the judges involved in a decision. The number of judges involved in a decision amounts 5 (± 2 SD) which is caused by the different compositions of the senates. *Court* is mentioned in the decision text to indicate the court taking the decision, but there are also courts in the appeal stages. courts are mentioned 15 (± 6 SD) times in a document. *Legal rule* is a summarizing

[49] `https://nominatim.openstreetmap.org/search/Bezirksgericht Leopoldstadt?polygon_geojson=1&format=json&countrycode=AT&type=administrative`

```
Input: Token
Rule: rs
(
 {Token.string == "RS"}
 {Token.kind == "number"}
):rs
-->
:rs.LegalRule = {legalrule = :rs@string}
```

Listing 7 Example snippet JAPE rule for the extraction of legal rules

statement of a ruling from which general rules are inferred and are often cited in decision texts to back up the decision. Legal rules are cited 23 (± 22 SD) times on average in the documents of the corpus. *Legal provision* is mentioned in the decision text and forms the legal basis on which the decision is grounded. Court decisions must be based on the law, it is therefore not surprising that 87 (± 72 SD) legal provisions are cited on average. *Law Gazette* is cited in cases where the court wants to refer to a specific version of law. A law gazette is usually cited together with a legal provision to indicate the specific version the court is referring to. Given the purpose of citing a law gazette in a court decision the number of citations is on average 4 (± 6 SD) per document. *Literature* is used to cite legal literature used to back up the decision. We also extract these references as they are with 50 (± 36 SD) citations on average an thus constitute a very important source. However, the literature is mostly (at least in Austria) only available against a paid subscription from various legal publishers.

Rule based approach. Given that legal documents follow a relatively regular structure and citation style we apply a rule-based approach for the information extraction using the Java Annotation Pattern Engine (JAPE) (Cunningham et al 1999) which is part of the General Architecture for Text Engineering (GATE)[50]. An example of how we can exploit the standardized citation style in legal documents is shown in Listing 7, which illustrates a (shortened) JAPE rule used to extract references to legal rules in a court decision. A JAPE rule has a left hand side where the rule is defined and a right hand side that defines what to do with the extracted information, with both sides separated with a -->. After a tokenizer (splitting the text into its individual parts) has been applied, the JAPE rule takes a *Token* as an input and looks for the defined pattern in the *Rule* section. In this example a legal rule must start with a token with a string *RS* directly followed by a token of kind *number*. The returned result is the complete the legal rule string, for instance *RS0042781* which we can look up in the database in order to replace the literal text with its actual URI, thus generating a link between the two documents. Rules can easily be supported by gazetteers, which are lookup lists that are very suitable for static, recurring entities, hence entities that do not change frequently. We use gazetteers to assist with the detection of contributors (a list with most common names and academic degrees), courts, legal provision (a list with all law abbreviations) and literature (a list with the most common legal journals used in Austria). Note that for the rule based approach we included a score for a strict and a lenient evaluation. The strict evaluation of rules only counts occurrences as correct when the annotation of the rule matches the gold standard annotation exactly. Lenient results also count occurrences as correct when both annotations overlap with the rule (adding or omitting some words).

[50] `https://gate.ac.uk/`

Table 3 Evaluation results of legal entity extraction. (*P=Precision, R=Recall, F=F-score. Best results highlighted in boldface.*)

			Case reference	Contributor	Court	Legal provision	Law gazette	Legal rule	Literature
Rule based	**Rules strict**	P	0.9782	0.7631	0.9892	0.8742	0.9150	1	0.6814
		R	0.9817	0.9406	0.9659	0.9074	0.9683	1	0.7865
		F	0.9799	0.8426	0.9774	0.8905	0.9409	**1**	0.7302
	Rules lenient	P	0.9806	0.7631	0.9919	0.8923	0.9200	1	0.8095
		R	0.9842	0.9406	0.9685	0.9262	0.9735	1	0.9343
		F	**0.9824**	0.8426	**0.9801**	0.9090	0.9460	**1**	0.8674
CRF	**CRF**	P	0.9868	0.9161	0.9852	0.9452	0.9638	0.9994	0.9145
		R	0.9710	0.9557	0.9416	0.9483	0.9364	1	0.8611
		F	0.9787	0.9328	0.9616	0.9459	**0.9473**	0.9997	**0.8866**
Deep Learning	**Flair**	P	0.9783	0.9187	0.9455	0.9324	0.9263	1	0.8596
		R	0.9800	0.9780	0.9486	0.9526	0.9245	1	0.8671
		F	0.9791	0.9435	0.9456	0.9414	0.9215	**1**	0.8629
	BERT	P	0.9687	0.9481	0.9557	0.9447	0.9546	0.9971	0.8497
		R	0.9738	0.9710	0.9762	0.9536	0.9336	1	0.8409
		F	0.9712	**0.9583**	0.9654	0.9489	0.9396	0.9986	0.8448
	DistilBert	P	0.9759	0.9316	0.9407	0.9446	0.9392	0.9979	0.8663
		R	0.9786	0.9878	0.9784	0.9600	0.9529	1	0.8604
		F	0.9772	0.9551	0.9586	**0.9521**	0.9437	0.9989	0.8626

Conditional Random Fields. An alternative, common approach to label textual sequence data using probabilistic models are Conditional Random Fields (CRF) (Lafferty et al 2001). We use the implementation of the sklearn-crfsuite[51]. The features of a token, for instance position and casing, are used to calculate the probabilities of tokens following each other. In the legal domain CRF have already been used in the context of entity extraction tasks where it has shown good results (e.g. Dozier et al (2010); Cardellino et al (2017); Leitner et al (2019)).

Deep learning approach. For experiments involving embeddings and deep learning we use the Flair framework[52] which provides all the necessary functionality required for our evaluation and in addition also supports importing *pretrained German language models*, which we were hoping to boost the accuracy for our German legal document corpus. We compare the following language models: (i) *Flair*, which uses contextualized character level embeddings (Akbik et al 2018) trained on a mixed corpus of web and Wikipedia documents; (ii) Language models using a transformer based architecture (Vaswani et al 2017) provided by HuggingFace[53] (Wolf et al 2019) known as *Bidirectional Encoder Representations from Transformers (BERT)* (Devlin et al 2019) trained on German Wikipedia, German open legal data and news articles; and (iii) *DistilBERT* (Sanh et al 2019) a faster and smaller version of BERT also trained on Wikipedia articles and web documents. DistilBERT uses a teacher-student setting to distill the knowledge from the teacher (the BERT model) to the student (DistilBERT model).

[51] `https://sklearn-crfsuite.readthedocs.io/en/latest/`

[52] `https://github.com/flairNLP/flair`

[53] `https://huggingface.co/`

Evaluation. For the evaluation of the individual results we measure *Precision (P)* as the share of relevant from the retrieved documents, *Recall (R)* as the share of retrieved documents to all documents that should be retrieved and *F-score (F)* as the harmonic mean of P and R (Manning et al 2008). For our experiments we did not apply any preprocessing to the documents and apply a 5-fold cross-validation approach using a train/test/validation split of 80%/10%/10%. All models have been trained with default settings, in particular the deep learning models with a maximum of 150 epochs, starting learning rate of 0.1, patience 3 and an anneal factor of 0.5. The training stops automatically when the learning rate becomes too small.

Table 3 shows the results for the different legal entities, whereby approaches with the best F-scores are highlighted in boldface. Looking at the evaluation results we can see at first glance that there is no single clear best approach outperforming all other approaches on all legal entities. Furthermore, it can also be noted that the results of all extraction methods are comparable across all methods for the individual legal entities. In particular, the numbers show that rules perform well when the entities under investigation are highly structured and always follow the same pattern, for instance case reference (e.g. *14Os108/20v)* and legal rule (e.g. *RS0042781*) which are very easy to recognize. Moreover, we use gazetteers to support rules with the extraction of the contributors. The rule looks for a degree (from a gazetteer) followed by a last name (from a gazetteer) within the head of the document. The inclusions of additional sources already decreases the performance of the rule based approach and automatic approaches perform better. When adding more variations and more complexity to the legal entities the performance of the rule-based and gazetteer supported approach deteriorates and machine learning based approaches perform better. The numbers of the legal provision, law gazette and literature show this effect. The citations of legal provisions can be simpler (e.g. *§ 41 ZPO* and more complex (e.g. *§§ 41, 43 Abs 2 erster Fall und § 50 ZPO*) which adds a lot of complexity to the rules and as a result makes the result much harder to create. The citations of the law gazettes changed over time by adding additional information (e.g. from *BGBl. 1969/207* to *BGBl. I Nr. 134/2015*). The most complex entity to extract is the literature as there are various types of literature (e.g. commentaries, books, articles,..) and citation styles. The higher complexity for literature is also reflected in the evaluation results. While the best F-scores for the other legal entities are somewhere in the 94% range, the F-score for literature is achieved by CRF with only 88%. The numbers also show that the gap between the rules and automatic approaches is bigger the more complex the rules (with gazetteer support) need to be. However, the gap between the individual approaches is very small. The F-scores of the three deep learning approaches (Flair, BERT, DistilBERT) are within 2% across all legal entities, thus we cannot nominate a clear winner in this segment. Also the difference across all approaches and legal entities falls within a range of 4%.

While the evaluation results show that the extraction approaches perform mostly equally well, we also should take into account the effort that is required to set up such a system for the extraction of legal entities. Rules can be easily and quickly created with only a few sample documents that cover the possible variations in which legal entities can appear. In addition, rules are easy to interpret and explain. The outcome of a rule is clear from the beginning, as a rule either matches a sequence of tokens or not. Gazetteers are suitable for entities that do not change frequently, for instance courts or names, but have a maintenance requirement and might need to be updated on a regular basis, otherwise rules using these gazetteers will start to fail over time. By contrast, approaches using (deep) machine learning promise to be more flexible and are also able to cover variations in patterns where a rule would fail. However, these approaches are less explainable and predictable, hence working with probabilities of the results and selecting the right algorithm for the right task is necessary.

In addition, we remark that it requires considerable effort to annotate documents required for training machine learning approaches as well as computational power and resources to perform both training and model fine-tuning. In our case, the experiments with our corpus of only 50 documents used the full capacity of our machine with 16GB of memory and requires a powerful GPU (a GTX 1080 Ti with 16GB memory) to perform the computations in a timely manner.

Summarizing the results shown by the experiments there is no clear best approach to extract legal entities from text. Thus the approach should be chosen based on the requirements, the available data from the legal information system acting as a data source and human resources. We conclude in particular that rules, in combination with gazetteers, are a viable alternative and can keep up with state of the art NLP techniques using complex neural networks for the relatively well-structured texts in our domain, offering maintainability and explainability of extraction results.

4.2.3 Alignment of heterogeneous schemes

Last, but not least, our AustroVoc vocabulary, which is composed of terms specific to the Austrian legal system, contains for instance a law index which is very suited to be linked with related terms in EuroVoc, thereby, directly enabling a multi-lingual search (given that EuroVoc is available in multiple languages). As the main obstacle herein, legal language is diverse even within German speaking countries, plus EuroVoc contains "German" German whereas Austria often uses specific "Austrian" German terms, for instance we use the term *Konsumentenschutz* while EuroVoc contains the term *Verbraucherschutz* for "customer protection". Since we want to link the concepts of the Austrian law index with EuroVoc concepts, we adopt the approach described in Filtz et al (2018). The simplest way to find a match is a direct lookup of the Austrian term in EuroVoc, if no match is found we also include external knowledge bases such as *DBpedia*[54], *Wikidata*[55] and the *Standard Thesaurus Wirtschaft (STW)*[56] and search for additional language version of the term there. In case a match is found we can link the AustroVoc term with the corresponding EuroVoc term using the property `rdfs:seeAlso`, for instance we find a match from *Konsumentenschutz* to *Verbraucherschutz* and add the triple `av:bri2006 rdfs:seeAlso ev:2836` to AustroVoc as shown in Listing 5.

5 The European Legal Knowledge Graph

Our final objective is to integrate the Austrian legal knowledge graph with other national legal knowledge graphs, which should enable interlinkage across different countries. We herein analyze the current situation regarding the provision of linked legal data as well as legal databases in other EU member states and perform a comparitive analysis. In addition to the legal information provided by governments, we also include a selection of non-governmental initiatives[57] and summarize challenges and opportunities we faced during this process.

[54] `https://wiki.dbpedia.org/`

[55] `https://www.wikidata.org/`

[56] `http://zbw.eu/stw/version/latest/about`

[57] We do not include commercial solutions.

5.1 Legal information provided by Governments

We include the EU member states without the United Kingdom and EU candidate countries in our analysis of whether and how they make legal information available in machine-readable form. We use the EU e-Justice portal[58] as a starting point for our research process, which includes overview pages on which EU member states can provide additional information about their implementation, for all EU member states for ELI[59] and ECLI[60]. While the country-specific ECLI information page contains all EU member states, the ELI information page only has information for 17 countries. Typically an explanation and examples are included as well as links to national legal databases. Some countries provide detailed information about their deployed ELI/ ECLI structure while others do not provide any information or, respectively, only in the national language which needs to be translated using a translation service. When available, we followed the links provided, otherwise we used a search engine to manually find additional national legal databases and examples for legislative and judiciary documents (cf. Tables 10 and 11 (Appendix A.4) for links to databases and examples). In the first step we examine whether ELI/ECLI identifiers are visible in the document and in the second step we also scan the source code of the (HTML) document, searching the metadata for keywords such as *eli, ontology, dc, dcterms, creator* and *date*. We provide an overview of the properties used in the Appendix for ELI (Table 9) and ECLI (Table 7). Where we find metadata embedded in the document we parse the URL using EasyRdf[61] to automatically retrieve RDF triples per document. We also check whether countries use national Named Authority Lists (NALs), i.e. determine whether national information pages about the used NAL are provided. In addition to this search process on the national level we also queried the EU Open Data Portal[62] for national legal data. We also record per country the type of available search interfaces, available document formats, languages and availability of judiciary documents in the EU ECLI search engine.

Table 4 provides a comprehensive overview of the national ELI and ECLI implementation initiatives of the EU member states with a focus on the ELI/ ECLI implementation status. The columns *Implementation ELI* and *Implementation ECLI* describe the implementation status with *Identifier* referring to the situation where documents are given an ELI identifier and *Identifier/Metadata* indicates that the particular country also provides metadata for the documents. The general assumption is that all countries use the ELI ontology for legislative documents (and ECLI for judiciary documents respectively), but some countries provide national extensions in order to represent legal information based on national requirements. These additional ontology extensions are indicated in brackets, for instance Finland defined its own extensions for ELI in the *Semantic Finlex Legislation Ontology (SFL)*[63] and the *Semantic Finlex Case Law Ontology (SFCL)*[64] ontology for judiciary documents. Luxembourg also uses an additional ontology called *JOLUX*[65] in their *Casemates* project[66] incorporating the ELI ontology and extending it. Special cases are Latvia and Slovenia who do not partici-

[58] https://e-justice.europa.eu/

[59] https://eur-lex.europa.eu/eli-register/implementation.html

[60] https://e-justice.europa.eu/content_european_case_law_identifier_ecli-175-en.do?init=true

[61] http://www.easyrdf.org/

[62] https://data.europa.eu/euodp/en/home

[63] http://data.finlex.fi/schema/sfl/

[64] http://data.finlex.fi/schema/sfcl/

[65] https://data.public.lu/en/datasets/r/53aa1301-2a42-465a-8803-c0cb5a3589e7

[66] http://www.legilux.lu/editorial/casemates

Table 4 Linked legal data feature comparison of EU member states (*(+) indicates the usage of additional or other ontologies)*

Country	Implementation ELI	Implementation ECLI	Data Availability	Information ELI / ECLI / NAL	Thesaurus
Austria	Identifier	Identifier	-	✓/ - / -	✓
Belgium	Identifier	Identifier	-	✓/ ✓/ -	-
Bulgaria	-	Identifier	-	- / ✓/ -	-
Croatia	Identifier	Identifier	-	✓/ - / -	-
Cyprus	-	-	-	- / ✓/ -	-
Czech Republic	-	Identifier	-	- / ✓/ -	-
Denmark	Identifier/Metadata	-	RDF	✓/ ✓/ ✓	-
Estonia	-	Identifier		- / ✓/ -	✓
Finland	Identifier/Metadata (+)	Identifier/Metadata (+)	RDF	✓/ ✓/ ✓	✓
France	Identifier/Metadata	Identifier	RDFa	✓/ ✓/ -	-
Germany	-	Identifier/Metadata	-	- / ✓/ -	-
Greece	-	Identifier	-	-	-
Hungary	-	-	-	✓/ - / -	-
Ireland	Identifier/Metadata	-	RDFa, RDF	✓/ ✓/ -	-
Italy	Identifier/Metadata	Identifier	RDFa, RDF	✓/ ✓/ ✓	-
Latvia	- (+)	Identifier	-	- / ✓/ -	-
Lithuania	-	-	-	- / ✓/ -	✓
Luxembourg	Identifier/Metadata (+)	-	RDFa	✓/ - / ✓	-
Malta	-	Identifier	-	✓/ - / -	-
Netherlands	- (+)	Identifier/Metadata	RDFa, RDF	- / ✓/ -	-
Poland	-	-	-	-	-
Portugal	Identifier/Metadata	Identifier (+)	RDFa	✓/ - / -	-
Romania	-	Identifier	-	- / ✓/ -	-
Slovakia	-	Identifier	-	- / ✓/ -	✓
Slovenia	- (+)	Identifier	-	- / ✓/ -	-
Spain	Identifier/Metadata	Identifier	RDFa	✓/ - / ✓	-
Sweden	-	-	-	-	-

pate in the ELI and therefore also do not assign ELI identifiers to their legislative documents but do provide a basic set of metadata (which is less than and different to ELI) using the *Open Graph Protocol (OGP)*[67]. Portugal assigns an ECLI identifier to judiciary documents, but uses OGP for the metadata. The Netherlands use for their legislative documents the *dcterms* and *Overheid* ontologies. We can see that 11 out of 27 countries implemented at least the first pillar of the ELI ontology (i.e. assigning an ELI identifier to the documents), hence giving an ELI identifier to legislative documents. Participation/Implementation is better in terms of ECLI, where 19 countries assign an ECLI identifier to judiciary documents, but the number of countries providing machine-readable metadata (i.e.,3) is lower compared to ELI (i.e.,9). Compared to a study conducted in 2017 (van Opijnen et al 2017b) the participation in ECLI increased in the last years with additional seven countries now participating in ECLI with at least providing an ECLI identifier. The column *Data Availability* describes how the data is provided to the public with the majority of participating countries opting to use the RDFa format and embed the metadata in the source code of the document. Denmark, Finland, Ireland and Italy also allow users to download the data in RDF either from a national website or the European Open Data Portal. The Netherlands provide a web service[68] that can be used to download the data in RDF. We indicate whether information about the national implementation of ELI and ECLI as well as the usage of NAL is provided either using dedicated pages on the EU e-Justice portal or a national website. Some properties are very suitable for the usage of NAL, for instance `eli:language` or `eli:type_document`. An overview of the used NALs is provided in Appendix A.2, Table 8. We notice that there

[67] https://ogp.me/

[68] https://linkeddata.overheid.nl/front/portal/services

Table 5 Features of legal databases of EU member states (* *denotes a subset)*

Country	Central Interface	ECLI Search	Search Interface	Document Format	Languages
Austria	✓	-	Keyword	HTML, PDF, RTF, XML	DE, EN*
Belgium	-	✓	Keyword	HTML	FR, NL, DE
Bulgaria		✓	Keyword	HTML, PDF	BG
Croatia	-	✓	Keyword	HTML	HR
Cyprus	✓	-	Keyword	PDF	EL
Czech Republic	-	✓	Keyword	PDF	CZ
Denmark	-	-	Faceted	HTML, PDF	DK
Estonia	✓	✓	Keyword	HTML, PDF, TXT, XML	EE, EN*
Finland	✓	✓	Keyword, SPARQL	HTML	FI, SE
France	✓	✓	Keyword	HTML, PDF	FR, EN*, DE*, IT*, ES*
Germany	-	✓	Keyword	HTML	DE, EN*
Greece	-	✓	Keyword	PDF	EL
Hungary	-	-	Keyword	HTML	HU, EN*
Ireland	-	-	Keyword	HTML, PDF	EN
Italy	-	✓	Keyword	HTML	IT
Latvia	-	✓	Keyword	HTML, PDF	LV, EN*, RU*
Lithuania	-	-	Faceted	HTML, PDF	LT
Luxembourg	-	-	Faceted, SPARQL	HTML, PDF, XML, RDF	FR
Malta	-	-	Keyword	PDF	MT, EN
Netherlands	-	✓	Both	HTML, PDF, RDF	NL, FR, EN*
Poland	-	-	Keyword	PDF	PL
Portugal	✓	✓	Faceted	HTML, PDF	PT, EN*
Romania	-	-	Keyword	HTML	RO
Slovakia	✓	-	Keyword	HTML, PDF	SK
Slovenia	-	✓	Keyword	HTML, PDF, DOCX	SI, EN*
Spain	-	✓	Both	HTML, PDF, XML, EPUB	ES
Sweden	-	-	Keyword	HTML	SE

are more countries using NALs, however they do not all provide an information page. A thesaurus, such as EuroVoc or a national index of legal terms, is used by five countries as indicated in column *Thesaurus*.

We show the features of the EU member states' legal databases in Table 5. Central search interfaces are very convenient as users can find all the required information in the same place. However, as legal systems are typically divided into legislation and judiciary the information for both branches falls under the responsibility of different authorities and therefore provided at distinct places. The column *Central Interface* shows if there is a central interface available that enables users to access legislation as well as judiciary documents from different authorities even if they are stored in separated backend systems. The EU e-Justice portal contains an ECLI search engine[69] which enables users to search for ECLI identifiers and keywords in judiciary documents from multiple countries, but not all countries providing an ECLI identifier are also participating in the ECLI search engine. The *Search Interface* column indicates how the search process can be performed by users with the majority of countries providing a keyword-based search interface, which might be enhanced with additional filters, for instance to restrict dates to a certain time frame or select only special types of documents. Faceted search interfaces are implemented by a minority of countries only, *Both* means that one legal database provides a keyword-based search and the other legal database supports faceted search. We can also see that Finland and Luxembourg set up a public SPARQL endpoint which allows users to run structured queries on the data directly. The standard way to represent legal documents on the web is HTML as shown in column *Document Format*. While the content is displayed using HTML, the majority of legal information systems also allow users to download documents in PDF format. However, some countries provide documents in

[69] `https://e-justice.europa.eu/content_ecli_search_engine-430-en.do`

Table 6 Non-governmental initiatives using ELI and ECLI.

Project	Type	Using ELI / ECLI	Extension ELI / ECLI	Data Availability	Thesaurus	Open Data Linking	SPARQL
Legal Knowledge Graph	Linking	✓/ ✓	LKG / LKG	RDF	EuroVoc, Other	✓	✓
Semantic Finlex	Linking	✓/ ✓	SFL / SFCL	RDF	EuroVoc, Other	✓	✓
Nomothesia	Linking	✓/ -	Nomothesia / -	RDF	-	✓	✓
EUCases	Linking	✓/ ✓	-	-	EuroVoc, Other	-	-
Lynx	Linking	✓/ -	Lynx-LKG	RDF	EuroVoc, Other	✓	✓
GDPRtEXT	Linking	✓/ -	GDPRtEXT	RDF	-	-	✓
Linkoln	Extraction	✓/ -	-	-	-	-	-
BO-ECLI	Extraction	- / ✓	-	-	-	-	-

PDF only. A popular structured format is XML, supported by Austria, Estonia, Luxembourg and Spain. The EPUB format is only used in Spain. While it is clear that countries provide their documents in their official language(s), Austria, Estonia, France, Germany, Hungary, Latvia, Netherlands, and Slovenia publish a subset of their documents, mainly the documents considered to be most important such as the constitution or the civil code, also in English.

5.2 Non-governmental initiatives

Besides linked legal data initiatives driven by governments there are also efforts by academia and industry in this direction often conducted in collaboration with and funded by governments. We are particularly interested in non-governmental initiatives working with ELI and ECLI providing a linked legal data framework or focusing on special legal areas.

Table 6 shows an overview of several non-governmental initiatives across Europe based on the information provided by the project websites, publications or namespaces used in RDF data retrieved via a SPARQL endpoint. The column *Project* shows the title of the project. We classify the projects as indicated in column *Type* into the classes *Linking* which means that this project aims to link legal data with other legal other data or external knowledge bases and *Extraction* means that the project is focusing on the extraction of specific information contained in legal documents. Column *Using ELI / ECLI* indicates whether a the project uses ELI, ECLI or both. In cases where the project results in extensions to the ELI and ECLI ontologies the name of these extensions is listed in column *Extension ELI / ECLI*. In cases where data is made available for download the format is shown in column *Data Availability*. Column *Thesaurus* indicates whether the European thesaurus EuroVoc or other thesauri (e.g. a national thesaurus) is used. When the data used in the project is linked with other external data such as DBpedia or Geonames this is indicated in column *Open Data Linking*. The column *SPARQL* shows whether a SPARQL endpoint is available to retrieve the data from that project.

The *Legal Knowledge Graph* project that aims to integrate legal data from disparate legal databases into a knowledge graph is described in Section 4. The *Semantic Finlex Project*[70] (Oksanen et al 2019) carried out by the University of Aalto is, similar to our Austrian research project, based on the national legal database of Finland which contains legislative and judiciary documents, and transforms the data into linked legal data based on the ELI and ECLI ontologies. The results of this Finnish project are also visible in Table 4 as they are available to the public via the official Finlex website[71], as well as via a SPARQL endpoint[72]. Fin-

[70] https://seco.cs.aalto.fi/projects/lawlod/

[71] https://data.finlex.fi/

[72] https://www.ldf.fi/sparql-services.html

lex extends the ELI with the Semantic Finlex Legislation Ontology[73] (SFL) and ECLI with the Semantic Finlex Case Law ontology[74] (SFCL). The greek project *Nomothesia* (Chalkidis et al 2017) by the University of Athens focuses on legislation only and is based on legal documents published in PDF format which are transformed into linked legal data based on the ELI which is incorporated in the Nomothesia ontology[75]. The data produced by the Nomothesia project is available for download as well as via a SPARQL endpoint[76] and includes DBpedia as an external knowledge base, for instance to link persons that are mentioned in legal acts. In the *EUCases* project (Boella et al 2015) a first effort effort was made trying to link national and EU legislation and case law, which is no longer accessible because a login is required and there is no response to email requests[77]. This project also includes a proposal to link legal documents with the EuroVoc thesaurus and incorporates the Legal Taxonomy Syllabus (LTS) (Ajani et al 2007). The EU funded *Lynx* project[78] aims at creating a legal knowledge graph with a special focus on compliance (Montiel-Ponsoda et al 2017). This project includes Spanish legislation and jurisdiction as well as documents from selected countries and extends ELI and ECLI with the Lynx-LKG ontology[79]. The Lynx data can also be accessed via a SPARQL endpoint[80]. A legal domain-specific work is *GDPRtEXT*[81] extending the ELI to provide the General Data Protection Regulation (GDPR)[82] as a linked data resource together with a taxonomy of GDPR terms using SKOS (Pandit et al 2018). The linked legal data version of the GDPR extends the ELI ontology with the GDPRtEXT ontology. The data and the ontology are available for download[83] and can be accessed via a SPARQL endpoint[84]. The Italian *Linkoln project* focuses on the automatic extraction of references from legal documents of the Italian Senate and is also able to extract ELI references (Bacci et al 2019). The EU funded *BO-ECLI project*[85] running from 2015 to 2017 focused on the ECLI and investigated the implementation of the ECLI in selected countries resulting in a proposal of a new version of the ECLI due to discovered drawbacks (van Opijnen et al 2017a).

6 Use Case Revisited

With an Austrian legal knowledge graph in place and a more complete picture of other similar international initiatives, we are now able to assess the potential benefits of linked legal knowledge both nationally and internationally. For instance, in terms of providing enhanced capabilities in terms of legal analyses, or in enabling us to answer complex search queries that would entail tedious manual research otherwise. Yet, we still herein have only made initial steps towards an EU wide linked legal data graph, wherefore we also discuss additional required steps and a respective roadmap.

[73] `http://data.finlex.fi/schema/sfl/`

[74] `http://data.finlex.fi/schema/sfcl/`

[75] `http://legislation.di.uoa.gr/data/ontology`

[76] `http://legislation.di.uoa.gr/endpoint`

[77] `http://www.eucases.eu`

[78] `http://www.lynx-project.eu/`

[79] `http://lynx-project.eu/doc/lkg/`

[80] `http://sparql.lynx-project.eu/`

[81] `https://openscience.adaptcentre.ie/projects/GDPRtEXT/`

[82] `https://eur-lex.europa.eu/eli/reg/2016/679/oj`

[83] `https://old.datahub.io/dataset/gdprtext`

[84] `http://openscience.adaptcentre.ie/sparql`

[85] `https://bo-ecli.eu/`

6.1 Benefits of an integrated legal knowledge graph

Let us revisit the questions from Section 3: indeed, we can demonstrate the benefits of an integrated legal knowledge graph by underpinning them with example SPARQL queries providing answers to such questions.

- *Which documents are referenced in a specific court decision?*

Court decisions are based on the law and therefore reference legal provisions but also other court decisions and legal rulings. Users nowadays typically need to query the respective database, e.g. the law database for legal provisions, and manually search the referenced document in order to get the content. In a knowledge graph we can combine several involved steps into a single query that returns a court decision with all referenced documents, their texts, plus types of the documents. This leads to a more **efficient legal information search process**. To enable such a query we need to extract the referenced documents from the court decision and replace them with the respective URIs as well as a schema of document types. Example 2 shows the convenience of such a query when a lawyer is interested in a particular court decision and gets all referenced documents with their text and sorted by their types as result.

Example 2 SPARQL Query: Which documents are referenced in the Supreme Court decision with case number 10Ob12/16m?

```
SELECT DISTINCT ?Reference ?Text ?Type
WHERE {
  ?justiz rdfs:label "10Ob12/16m" .
  ?justiz dcterms:references ?ref .
  {
    ?ref rdf:type lkg:LegalProvision ;
         rdfs:label ?Reference ;
         eli:is_realized_by ?realization .
    ?realization lkg:has_text ?Text .
    ?ref eli:type_document ?type_document .
    ?type_document skos:prefLabel ?type .
  } UNION {
    ?ref rdf:type lkg:JudicialResource ;
         dcterms:type av:jud_rs ;
         rdfs:label ?Reference ;
         lkg:has_text ?Text .
    av:jud_rs skos:prefLabel ?type .
  } UNION {
     ?ref rdf:type lkg:JudicialResource ;
         dcterms:type av:jud_te ;
         rdfs:label ?Reference ;
         lkg:has_text ?Text .
     av:jud_te skos:prefLabel ?type .
  }
    FILTER (lang(?type) = 'de')
}
ORDER BY ?type
```

Reference	Text	Type
"§ 500 ZPO"	"§ 500. (1) Das Urteil oder der Beschluß [...]"	"Bundesgesetz"
"§ 28a KSchG"	"§ 28a. (1) Wer im geschäftlichen Verkehr [...]"	"Bundesgesetz"
"4OB89/88"	"Ein Veröffentlichungsbegehren im Sinne [...]"	"Rechtssatz"
...	...	...

- *Over which districts does a court have competent jurisdiction?*

Legal databases are typically domain-specific and focus on legal matters only without additional contextual references that would be useful to be included for scoping search (such as explicit spatio-temporal references). For instance, a lawyer has a client who is facing a lawsuit regarding a property. Therefore, the lawyer needs to know which court has spatial competent jurisdiction, in order to find related cases in a regional context. At the moment this information is not made explicit in the legal information system and the lawyer would need to look through various websites of the authorities to find out about the regionally competent jurisdiction. This problem can be addressed by integrating external data in our legal knowledge graph and leads to **enriched information content and better user experience**. Since, in our knowelgedge graph, we have readily linked the information about the Austrian courts and the judicial districts from the respective authorities with a geospatial hierarchy, also taking the court hierarchy into account, we can easily provide such information again by a straightforward SPARQL query. As shown in Example 3 the lawyer is now able to query the court having competent jurisdiction, just by providing the name of a community.

Example 3 SPARQL Query: Which court has competent spatial jurisdiction for the market town *Krieglach*?

```
select ?court where {
 ?geo gn:name "Krieglach" .
 ?jd lkg:judicial_district_member ?geo ;
      lkg:court_having_jurisdiction ?c .
    ?c rdfs:label ?court
}
```

Court
"Bezirksgericht Mürzzuschlag"

– *What are the national transpositions of a specific EU directive?*

Legal systems differ across countries but still we need to consider legal information from other countries from time to time, especially in a European context with the EU's harmonization activities through issuing commmon regulations, but also directives, which need to be transposed into national legislation. For companies wanting to expand their businesses abroad it is necessary to know the legal situation and standards in these foreign countries. So far, a lawyer needs to search for the legal information system of the other country and find out how a particular directive, that is relevant for the company, has been transposed.[86] Also, the Eur-Lex search interface is not always helpful here, because it does not provide the transposed texts. Integrating legal data across countries in a legal knowledge graph thus would enable **cross-jurisdictional search of legal information**. In our example, we demonstrate how this can be achieved, across countries that follow the proposed ELI and ECLI standards for legal data (cf. Section 5). As shown in Example 4 the company lawyer is able to find the concrete national transpositions of a given directive with the actual transposed texts, across national legislations, again with a single query.

Example 4 SPARQL Query: What are the national transpositions of EU directive 2014/92/EU (with links to the resp. documents)?

```
select ?country ?title ?document where {
```

[86] Further tedious search would be needed to find out about and compare respective jurisdictions across countries

```
      VALUES ?format {
          <http://www.iana.org/assignments/media-types/text/html>
          <http://www.iana.org/assignments/media-types/application/html> }
  ?n ?p <http://data.europa.eu/eli/dir/2014/92/oj> ;
              eli:relevant_for ?c ;
              eli:is_realized_by ?r .
      ?r eli:title ?title ;
          eli:is_embodied_by ?document .
      ?document eli:format ?format .
      ?c skos:prefLabel ?country .
     FILTER (lang(?country) = 'en')
}
```

Country	Title	Document
"Ireland"	"European Union (Payment Accounts) Regulations 2016."	Document 1
"Austria"	"Bundesgesetz, mit dem ein Bundesgesetz über [...]"	Document 2
"Austria"	"Verordnung der Finanzmarktaufsichtsbehörde (FMA) über [...]"	Document 3
...	...	

Document 1: `http://www.irishstatutebook.ie/eli/2016/si/482/made/en/html`
Document 2: `https://www.ris.bka.gv.at/Dokumente/BgblAuth/BGBLA_2016_I_35/BGBLA_2016_I_35.html`
Document 3: `https://www.ris.bka.gv.at/Dokumente/BgblAuth/BGBLA_2018_II_60/BGBLA_2018_II_60.html`

Further integrating and harmonizing existing legal knowledge graphs across countries, as discussed in Section 5 would further enable comparison of the respective jurisdiction for a particular directive, across countries.

- *Which legal documents regulate a specific legal area searched with keywords in a foreign language?*

Legal systems are not only different in their structure but legal documents are typically penned in the official language(s) of a country, which puts an additional language barrier in the legal information search process. Additional sources such as the EuroVoc thesaurus, ideally aligned with national thesauri, which contain terms in multiple languages to the legal knowledge graph enables **multi-lingual search of legal information**. Linking legal documents with concepts instead of language-specific labels allows users to search in their language for documents written in another language. For instance, a lawyer is researching in a lawsuit covering another country and wants to know which legal provisions cover a specific legal area and is able to search in his language as shown in Example 5. Different languages are a barrier and supporting multi-lingual search is a step towards improved, more transparent access to legal information.

Example 5 SPARQL Query: Which documents belong to the category consumer protection searched by an Italian?

```
select ?law ?legalprovision ?document where {
    ?ev skos:prefLabel "protezione del consumatore"@it .
    ?austrovoc rdfs:seeAlso ?ev .
  ?lp eli:is_about ?austrovoc ;
          eli:jurisdiction <http://publications.europa.eu/resource/authority/
               country/AUT> ;
          eli:in_force eli:InForce-inForce ;
          eli:is_realized_by ?le ;
          lkg:has_number_paragraph ?number ;
          rdfs:label ?legalprovision .
    ?le eli:title_alternative ?law ;
          eli:is_embodied_by ?document .
    ?document eli:format <http://www.iana.org/assignments/media-types/
           application/html>
}
ORDER BY ASC(?law) ASC(?number)
```

Law	Legal Provision	Document
"KSchG"	"§ 1 KSchG"	Document 1
"KSchG"	"§ 42 KSchG"	Document 2
"VKrG"	"§ 1 VKrG"	Document 3
...	...	

Document 1: `https://www.ris.bka.gv.at/Dokumente/Bundesnormen/NOR12041200/NOR12041200.html`
Document 2: `https://www.ris.bka.gv.at/Dokumente/Bundesnormen/NOR40050352/NOR40050352.html`
Document 3: `https://www.ris.bka.gv.at/Dokumente/Bundesnormen/NOR40117826/NOR40117826.html`

6.2 Roadmap towards a linked legal knowledge graph

The current situation towards a truly interconnected legal knowledge graph on a European level looks promising, with many good starting points, but some challenges lie ahead to be addressed. On the one hand, providers of legal information, typically governments, would need to help to ease the access to law and support non-governmental initiatives to provide and obtain legal information. On the other hand, these providers are confronted with resource restrictions and other priorities, which slows down this process. We discuss some of the related challenges in the following.

Licensing and access policies. The publication of and access to legal information might be hindered by licensing and access policies, or lack thereof. Open (government) data is a goal of the European Union as laid out in the PSI-Directive[87], which stipulates that documents from the public sector should be made available free of charge in machine-readable and open formats which also includes possibilities for a mass download. The PSI directive goes hand in hand with the *8 Open Government Data Principles*[88] to provide data in a machine-readable, license free, complete and accessible format in a timely manner. Following open government data publication methodologies such as COMSODE (Kucera et al 2015) helps governments to set up respective publication strategies. The terms and conditions should be communicated in a clear manner and data provided ideally under a permissive license which also allows private initiatives to use the data for their business model by providing additional services, e.g. build on the data and restrict access to certain parts of the knowledge graph such as linked legal commentaries.

Support of linked legal data initiatives. Our analysis of the legal landscape (cf. Section 5) shows that documents are provided in various formats with structured formats being the minority. The problem of having documents in an unstructured format as a starting point (e.g. Chalkidis et al (2017)) might slow down the process of the providing linked legal data. It is therefore desirable that legal documents are provided in a structured format from the very beginning in order to enable the transition to and participation in an EU-wide linked legal data ecosystem. Hence, following the Linked Data Principles together with using appropriate linked data formats such as JSON-LD (W3C JSON for Linking Data Community Group 2012) or RDF serializations or XML standards for legal documents, such as Akoma-Ntoso[89] enables easy access to the data for linked legal data initiatives. The EU can help member states in activities towards the provision of linked legal data by providing detailed guidelines on how to use the proposed ELI and ECLI standards or software tools supporting the transition. Furthermore, the provision of dedicated vocabularies in addition to the existing named

[87] EU 2019/1024`http://data.europa.eu/eli/dir/2019/1024/oj`

[88] `https://public.resource.org/8_principles.html`

[89] `http://www.akomantoso.org/`

authority lists and EuroVoc thesaurus, which do not really fit the requirements of member states, are beneficial as it reduces the barrier of participating in ELI and ECLI.

We emphasize here, that despite the resulting documents are typically plain text documents, in many countries – including Austria – the legal document preparation process is regulated by clearly defined processes where, as opposed to extracting unambiguous metadata on hindsight only - such metadata and linked data creation could and should be directly included into these processes. Respective tools that rely entirely on Open Web Standards could replace and improve the legal document creation process Beno et al (2019).

Information provision. The lack of coordination in terms of ELI and ECLI implementation concerns the European Union as well as EU member states. Currently, it is a very time-consuming task to find any information about ELI and ECLI implementations in different countries. At the moment the information is cluttered with some countries using the EU e-Justice portal or others providing respective information only on national websites. Furthermore, implementation details can often only be inferred from studying the source code of example documents, rather than by available documentation. Positive examples of countries providing extensive information are, for instance, Denmark[90], Finland[91], and Luxembourg[92] who run national websites with implementation information about the ELI. The same applies to the usage of NAL which is encouraged by the ELI and ECLI ontologies. Without additional information about the used NAL it is a tedious task for outsiders to find information which NAL are used. In addition to missing information websites about the NAL some countries use NAL but these NAL cannot be retrieved from the internet or dereferenced. As argued herein, aligning the ELI and ECLI pages at EU level, hence integrating ELI into the EU e-Justice portal, and providing templates for member states about their ELI and ECLI implementation status as well as the usage of national NAL could be highly beneficial. More consistent best practices would also help other, not yet participating countries to investigate what and how to implement ELI and ECLI in an overall more aligned manner, which in turn might lower the barrier to participate.

Search interfaces. Access to legal information should be as easy as possible for end users as well as data processing professionals. Centralized web search interfaces serving as a *one-stop-shop* with a graphical user interface enabling the access to legal documents from various authorities eases the search process for the end user, citizens and legal professionals. Linked legal data initiatives enable such centralized aggregation of legal information, and can also support common application programming interfaces (API) – such as, e.g. access through the SPARQL protocol – as well as indexes to access and retrieve legal data for subsequent processing.

Multilinguality. Legal data is typically presented in the official language(s) of the respective country, some of the legal information systems provide some laws (e.g. civil code and the constitution) in English. As demonstrated herein, one approach to enable better multi-lingual search is to link national indexes with the multi-lingual EuroVoc thesaurus which then acts as a connecting point between legal information provided in different countries and languages. Yet, we also emphasize the importance of national extensions (such as AustroVoc which we proposed in this paper) to cover countrywise specifics, or for keeping ambiguous

[90] `https://www.retsinformation.dk/eli/about`

[91] `https://data.finlex.fi/en/datamodeling`

[92] `http://www.legilux.lu/editorial/casemates`

language use in different legislations/jurisdictions (e.g. Germany and Austria) separate. We envision the creation of similar national extensions, for instance SpainVoc or IrishVoc, by other member states. Another emerging approach to the multilinguality challenge is to create graph-based Linked Data native dictionaries that include lexical knowledge and overcome the disadvantages of tree-based dictionaries (Gracia et al 2017). Others enrich the underlying ontology with linguistic information, for instance as proposed by the Ontolex-lemon model (McCrae et al (2017); W3C Ontology-Lexica Community Group (2016)). Finally, multilinguality could be further supported by adding linguistic and lexical information to enable NLP applications working with this information contained in an ontology.

Modeling standards. In order to achieve the overarching ELI and ECLI goals EU member states should follow the modeling standards outlined in these proposals. Both ELI and ECLI describe a minimum set of non-country specific metadata and are therefore very well suited for national extensions where needed. Our comparison of the linked legal data features in the EU member states (cf. Table 4) shows that most of the participating countries follow the proposed modeling standards. Some countries, for instance Luxembourg provide their JOLUX ontology in their own as well as the ELI format. Individual deviations from these standards undermine the fundamental ideas of easier access to legal information across borders. One of the drawbacks of the current modeling standard, is the need to write queries in order to retrieve certain data as shown by Francesconi et al (2015). The proposed solution, which involves decoupling the ELI and FRBR ontologies, needs to be approached and initiated in a centralized manner, for instance via a stakeholder engagement process whereby national experts who know their legal system and experts from the responsible EU institutions work together in order to shape future ELI and ECLI enhancements.

7 Related Work

The exchange of legal information was already a concern before the advent of (legal) knowledge graphs and started with the standardization of (XML-based) formats that would allow the exchange of legal information across different jurisdictions. Furthermore, also ontologies to model legal information have been proposed. The goal of this section is to present other semantic technology based initiatives in the legal domain beyond work on legal knowledge graphs.

Several formats have been proposed enabling or simplifying the exchange of legal information in a structured and standardized manner. Boer et al (2002) described the XML standard MetaLex which can be used to encode the structure and the content of legal documents. Another open and extensible XML standard for the exchange of legislative and judiciary documents is Akoma Ntoso[93] providing schemes for the structure and metadata of legal documents. Other standards for the XML-based exchange of legal information are for instance LegalDocML TC[94] based on Akoma-Ntoso aiming at the creation of a standard for a worldwide exchange of legal information using a standardized set of metadata. LegalRuleML (Palmirani et al 2011; Athan et al 2013) focuses on the expression of rules and constraints in the legal domain in XML format. The Legal Knowledge Interchange Format (LKIF) proposed by Hoekstra et al (2007) is an ontology aiming at interchanging legal information between different legal systems modeling the semantics contained in the text of legal documents (Boer et al 2008).

[93] http://www.akomantoso.org/

[94] https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=legaldocml

With respect to legal ontologies there has been research work in the past years mainly dealing with legal domain specific ontologies. A summary of existing legal ontologies has been published by Breuker et al (2009) listing 23 ontologies and categorizing them by application (information retrieval, general language for expressing legal knowledge,...), type (knowledge representation) or character (general vs domain-specific). A recent extensive study conducted by de Oliveira Rodrigues et al (2019) analyses legal ontologies found in various digital libraries based on multiple dimensions such as formalization, legal theories, semantic problems and ontology engineering problems in a systematic manner. The study shows that a large number of legal ontologies have been proposed over time and are available for reuse. Leone et al (2019) classifies legal ontologies according general, modeling and semantic information. Ajani et al (2016) proposed the European Legal Taxonomy Syllabus (ELTS) as a lightweight ontology that should help to relate national and European legal terminology to represent the differences in the national legal systems of the EU member states. A legal knowledge management system based on ELTS to semi-automatically classify and interlink documents has been proposed by Boella et al (2019). Besides the generic legal ontologies used in this paper, many domain-specifc legal ontologies have been proposed tailored for the usage in a narrow legal domain. For instance the Open Digital Rights Language (ODRL)[95] (Steyskal and Polleres 2014; Vos et al 2019), Linked Data Rights (LDR)[96] and the Media Contract Ontology (MCO) (Rodríguez-Doncel et al 2016) to model policies, LOTED2 (Distinto et al 2016) and PPROC (Muñoz-Soro et al 2016) for the procurement domain. Ontologies related to data protection are for instance GDPRtEXT (Pandit et al 2018) which is an extension of the ELI ontology to model the GDPR, PrivOnto (Oltramari et al 2018) and PrOnto (Palmirani et al 2018) to model privacy policies and a similarly named ontology to represent product information called PRONTO (Vegetti et al 2011). Ontologies are subject to improvement over time. In the legal domain, Francesconi et al (2015) highlight drawbacks in the modeling of the CDM ontology used by the EU leading to unnecessarily complex queries and show how they could be resolved.

Lastly, ontology design patterns have been proposed to help with the creation of ontologies in a more systematic manner, for instnace based on patterns found in domain-specific documents. An overview of legal ontology design patterns is provided by Gangemi (2007). Examples for specific ontology design patterns in the legal domain are the Complaint Ontology Pattern (COP) by Santos et al (2016) and the License Linked Data Resources Pattern proposed by Rodríguez-Doncel et al (2013). Our middle-out ontology engineering method used to extend the existing ontologies described herein can likewise be used and applied alongside ontology design patterns.

8 Conclusion

In this paper, we describe the creation of a legal knowledge graph for Austria and propose the LKG ontology based on a real-world project funded by the Austrian Ministry for Digital and Economic Affairs. We provide detailed information about the modeling of the Austrian legal system using ELI and ECLI and propose different ontology population methods including rule-based and machine learning based approaches. Our comparative evaluation shows that rule-based as well as machine learning based approaches work similarly well for the extraction of legal entities. Furthermore, we enhance our Austrian LKG by linking to external spatial knowledge bases such as Geonames and Open Street Map, thus enabling more

[95] `https://www.w3.org/community/odrl/`

[96] `http://vocab.linkeddata.es/ontologies/purl.oclc.orgNETldrns.html`

fine grained spatial search. We also performed an depth analysis into the existing linked legal data initiatives by the various EU member states, and extended the analysis by presenting the predominant non-governmental linked legal data initiatives that are based on ELI and ECLI. Finally we demonstrated how said initiatives can enhance search possibilities and eases access to legal information by providing example SPARQL queries over several linked legal knowledge sources. The findings show that although the existing initiatives have already started to bear fruit when it comes to making all legal information machine-accessible we have barely scratched the surface.

Future work includes the extension of the corpus used for the evaluation of the legal entities extraction approaches with a study whether these results could be further boosted, for instance by training a state of the art language model based on Austrian legal documents or hyperparameter optimization. Furthermore, analyzing the content of legal documents and including the outputs in our legal knowledge graph, e.g. the automatic extraction of rules and constraints of legal provisions, or in analyzing the semantic content of court decisions to predict the outcome of future court decisions. Another possible route for further work involves an extensive linkage of our legal knowledge graph to external knowledge bases, for instance general knowledge bases, news sources, etc. Lastly, while we have shown that integrating the EuroVoc thesaurus supports search across multiple languages, it would be worth investigating the semantic meaning, differences, ambiguities, and similarities of legal expressions across different languages and jurisdictions.

Acknowledgements We thank the anonymous reviewers for their valuable suggestions and inputs. This work was partially funded by the Austrian Research Promotion Agency (FFG) under the "ICT of the Future" program (project "AI@Work", contract #874111), the Federal Ministry of Digital and Economic Affairs of the Republic of Austria and the Jubilee Fund of the City of Vienna. Sabrina Kirrane is funded by the FWF Austrian Science Fund and the Internet Foundation Austria under the FWF Elise Richter and netidee SCIENCE programmes as project number V 759-N. This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 860801.

References

Ajani G, Lesmo L, Boella G, Mazzei A, Rossi P (2007) Terminological and ontological analysis of european directives: Multilinguism in law. In: Proceedings of the 11th International Conference on Artificial Intelligence and Law, Association for Computing Machinery, New York, NY, USA, ICAIL '07, p 43–48, DOI 10.1145/1276318.1276327, URL https://doi.org/10.1145/1276318.1276327

Ajani G, Boella G, Caro LD, Robaldo L, Humphreys L, Praduroux S, Rossi P, Violato A (2016) The european taxonomy syllabus: A multi-lingual, multi-level ontology framework to untangle the web of european legal terminology. Applied Ontology 11(4):325–375, DOI 10.3233/AO-170174, URL https://doi.org/10.3233/AO-170174

Akbik A, Blythe D, Vollgraf R (2018) Contextual string embeddings for sequence labeling. In: COLING 2018, 27th International Conference on Computational Linguistics, pp 1638–1649

Athan T, Boley H, Governatori G, Palmirani M, Paschke A, Wyner A (2013) Oasis legalruleml. In: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Law, Association for Computing Machinery, New York, NY, USA, ICAIL '13, p 3–12, DOI 10.1145/2514601.2514603, URL https://doi.org/10.1145/2514601.2514603

Bacci L, Agnoloni T, Marchetti C, Battistoni R (2019) Improving public access to legislation through legal citations detection: The linkoln project at the italian senate. Knowledge of the Law in the Big Data Age 317:149

Beno M, Filtz E, Kirrane S, Polleres A (2019) Doc2RDFa: Semantic annotation for web documents. In: Alam M, Usbeck R, Pellegrini T, Sack H, Sure-Vetter Y (eds) Proceedings of the Posters and Demo Track of the 15th International Conference on Semantic Systems (SEMANTiCS 2019), CEUR-WS.org, Karlsruhe, Germany, CEUR Workshop Proceedings, vol 2451, URL http://ceur-ws.org/Vol-2451/paper-06.pdf

Berners-Lee T (2006) Linked data design issues. https://www.w3.org/DesignIssues/LinkedData.html, accessed: 2020-03-15

Boella G, Caro LD, Graziadei M, Cupi L, Salaroglio CE, Humphreys L, Konstantinov H, Marko K, Robaldo L, Ruffini C, Simov KI, Violato A, Stroetmann VN (2015) Linking legal open data: breaking the accessibility and language barrier in european legislation and case law. In: Sichelman T, Atkinson K (eds) Proceedings of the 15th International Conference on Artificial Intelligence and Law, ICAIL 2015, San Diego, CA, USA, June 8-12, 2015, ACM, pp 171–175, DOI 10.1145/2746090.2746106, URL https://doi.org/10.1145/2746090.2746106

Boella G, Caro LD, Leone V (2019) Semi-automatic knowledge population in a legal document management system. Artif Intell Law 27(2):227–251, DOI 10.1007/s10506-018-9239-8, URL https://doi.org/10.1007/s10506-018-9239-8

Boer A, Hoekstra R, Winkels R, Van Engers T, Willaert F (2002) Metalex: Legislation in xml. Legal Knowledge and Information Systems (Jurix 2002) pp 1–10

Boer A, Winkels R, Vitali F (2008) Metalex XML and the legal knowledge interchange format. In: Casanovas P, Sartor G, Casellas N, Rubino R (eds) Computable Models of the Law, Languages, Dialogues, Games, Ontologies, Lecture Notes in Computer Science, vol 4884, Springer, pp 21–41, DOI 10.1007/978-3-540-85569-9_2, URL https://doi.org/10.1007/978-3-540-85569-9_2

Breuker J, Casanovas P, Klein MCA, Francesconi E (2009) The flood, the channels and the dykes: Managing legal information in a globalized and digital world. In: Breuker J, Casanovas P, Klein MCA, Francesconi E (eds) Law, Ontologies and the Semantic Web - Channelling the Legal Information Flood, IOS Press, Frontiers in Artificial Intelligence and Applications, vol 188, pp 3–18, DOI 10.3233/978-1-58603-942-4-3, URL https://doi.org/10.3233/978-1-58603-942-4-3

Cardellino C, Teruel M, Alemany LA, Villata S (2017) A low-cost, high-coverage legal named entity recognizer, classifier and linker. In: Keppens J, Governatori G (eds) Proceedings of the 16th edition of the International Conference on Artificial Intelligence and Law, ICAIL 2017, London, United Kingdom, June 12-16, 2017, ACM, pp 9–18, DOI 10.1145/3086512.3086514, URL https://doi.org/10.1145/3086512.3086514

Casanovas P, Palmirani M, Peroni S, van Engers TM, Vitali F (2016) Semantic web for the legal domain: The next step. Semantic Web 7(3):213–227, DOI 10.3233/SW-160224, URL https://doi.org/10.3233/SW-160224

Chalkidis I, Nikolaou C, Soursos P, Koubarakis M (2017) Modeling and querying greek legislation using semantic web technologies. In: Blomqvist E, Maynard D, Gangemi A, Hoekstra R, Hitzler P, Hartig O (eds) The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I, Lecture Notes in Computer Science, vol 10249, pp 591–606, DOI 10.1007/978-3-319-58068-5_36, URL https://doi.org/10.1007/978-3-319-58068-5_36

Council of the European Union (2011) Council conclusions inviting the introduction of the European Case Law Identifier (ECLI) and a minimum set of uniform metadata for case law. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52011XG0429(01), online; accessed 17 May 2020

Council of the European Union (2012) Council conclusions inviting the introduction of the European Legislation Identifier (ELI). https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:52012XG1026(01), online; accessed 17 May 2020

Council of the European Union (2017) Council conclusions of 6 November 2017 on the European Legislation Identifier. https://eur-lex.europa.eu/legal-content/GA/TXT/?uri=CELEX:52017XG1222(02), online; accessed 17 May 2020

Cunningham H, Cunningham H, Maynard D, Maynard D, Tablan V, Tablan V (1999) Jape: a java annotation patterns engine

Devlin J, Chang M, Lee K, Toutanova K (2019) BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein J, Doran C, Solorio T (eds) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), Association for Computational Linguistics, pp 4171–4186, DOI 10.18653/v1/n19-1423, URL https://doi.org/10.18653/v1/n19-1423

Distinto I, d'Aquin M, Motta E (2016) LOTED2: an ontology of european public procurement notices. Semantic Web 7(3):267–293, DOI 10.3233/SW-140151, URL https://doi.org/10.3233/SW-140151

Dozier C, Kondadadi R, Light M, Vachher A, Veeramachaneni S, Wudali R (2010) Named entity recognition and resolution in legal text. In: Francesconi E, Montemagni S, Peters W, Tiscornia D (eds) Semantic Processing of Legal Texts: Where the Language of Law Meets the Law of Language, Springer, Lecture Notes in Computer Science, vol 6036, pp 27–43, DOI 10.1007/978-3-642-12837-0_2, URL https://doi.org/10.1007/978-3-642-12837-0_2

Filtz E, Kirrane S, Polleres A (2018) Interlinking legal data. In: Khalili A, Koutraki M (eds) Proceedings of the Posters and Demos Track of the 14th International Conference on Semantic Systems co-located with the 14th International Conference on Semantic Systems (SEMANTiCS 2018), Vienna, Austria, September 10-13, 2018, CEUR-WS.org, CEUR Workshop Proceedings, vol 2198, URL `http://ceur-ws.org/Vol-2198/paper\_118.pdf`

Francart T, Dann J, Pappalardo R, Malagon C, Pellegrino M (2018) The european legislation identifier. In: Peruginelli G, Faro S (eds) Knowledge of the Law in the Big Data Age, Conference 'Law via the Internet 2018', Florence, Italy, 11-12 October 2018, IOS Press, Frontiers in Artificial Intelligence and Applications, vol 317, pp 137–148, DOI 10.3233/FAIA190016, URL `https://doi.org/10.3233/FAIA190016`

Francesconi E, Küster MW, Gratz P, Thelen S (2015) The ontology-based approach of the publications office of the EU for document accessibility and open data services. In: Ko A, Francesconi E (eds) Electronic Government and the Information Systems Perspective - 4th International Conference, EGOVIS 2015, Valencia, Spain, September 1-3, 2015, Proceedings, Springer, Lecture Notes in Computer Science, vol 9265, pp 29–39, DOI 10.1007/978-3-319-22389-6_3, URL `https://doi.org/10.1007/978-3-319-22389-6\_3`

Gangemi A (2007) Design patterns for legal ontology constructions. In: Casanovas P, Biasiotti MA, Francesconi E, Sagri M (eds) Proceedings of the 2nd Workshop on Legal Ontologies and Artificial Intelligence Techniques June 4th, 2007, Stanford University, Stanford, CA, USA, CEUR-WS.org, CEUR Workshop Proceedings, vol 321, pp 65–85, URL `http://ceur-ws.org/Vol-321/paper4.pdf`

Ghosh ME, , Naja H, Abdulrab H, Khalil M (2016) Towards a middle-out approach for building legal domain reference ontology. International Journal of Knowledge Engineering 2(3):109–114, DOI 10.18178/ijke.2016.2.3.063, URL `https://doi.org/10.18178\%2Fijke.2016.2.3.063`

Gracia J, Kernerman I, Bosque-Gil J (2017) Toward linked data-native dictionaries. In: Electronic Lexicography in the 21st Century: Lexicography from Scratch. Proceedings of the eLex 2017 conference, pp 19–21

Grishman R, Sundheim B (1996) Message understanding conference- 6: A brief history. In: 16th International Conference on Computational Linguistics, Proceedings of the Conference, COLING 1996, Center for Sprogteknologi, Copenhagen, Denmark, August 5-9, 1996, pp 466–471, URL `https://www.aclweb.org/anthology/C96-1079/`

Hoekstra R, Breuker J, Bello MD, Boer A (2007) The LKIF core ontology of basic legal concepts. In: Casanovas P, Biasiotti MA, Francesconi E, Sagri M (eds) Proceedings of the 2nd Workshop on Legal Ontologies and Artificial Intelligence Techniques June 4th, 2007, Stanford University, Stanford, CA, USA, CEUR-WS.org, CEUR Workshop Proceedings, vol 321, pp 43–63, URL `http://ceur-ws.org/Vol-321/paper3.pdf`

Hogan A, Blomqvist E, Cochez M, d'Amato C, de Melo G, Gutierrez C, Gayo JEL, Kirrane S, Neumaier S, Polleres A, Navigli R, Ngomo ACN, Rashid SM, Rula A, Schmelzeisen L, Sequeda J, Staab S, Zimmermann A (2020) Knowledge graphs. `2003.02320`

Kucera J, Chlapek D, Klímek J, Necaský M (2015) Methodologies and best practices for open data publication. In: Necaský M, Pokorný J, Moravec P (eds) Proceedings of the Dateso 2015 Annual International Workshop on DAtabases, TExts, Specifications and Objects, Neprivec u Sobotky, Jicin, Czech Republic, April 14, 2015, CEUR-WS.org, CEUR Workshop Proceedings, vol 1343, pp 52–64, URL `http://ceur-ws.org/Vol-1343/paper5.pdf`

Lafferty JD, McCallum A, Pereira FCN (2001) Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Brodley CE, Danyluk AP (eds) Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001, Morgan Kaufmann, pp 282–289

Leitner E, Rehm G, Schneider JM (2019) Fine-grained named entity recognition in legal documents. In: Acosta M, Cudré-Mauroux P, Maleshkova M, Pellegrini T, Sack H, Sure-Vetter Y (eds) Semantic Systems. The Power of AI and Knowledge Graphs - 15th International Conference, SEMANTiCS 2019, Karlsruhe, Germany, September 9-12, 2019, Proceedings, Springer, Lecture Notes in Computer Science, vol 11702, pp 272–287, DOI 10.1007/978-3-030-33220-4_20, URL `https://doi.org/10.1007/978-3-030-33220-4\_20`

Leone V, Di Caro L, Villata S (2019) Taking stock of legal ontologies: a feature-based comparative analysis. Artificial Intelligence and Law DOI 10.1007/s10506-019-09252-1, URL `https://doi.org/10.1007/s10506-019-09252-1`

Manning CD, Raghavan P, Schütze H (2008) Introduction to information retrieval. Cambridge University Press, DOI 10.1017/CBO9780511809071, URL `https://nlp.stanford.edu/IR-book/pdf/irbookprint.pdf`

McCrae JP, Bosque-Gil J, Gracia J, Buitelaar P, Cimiano P (2017) The ontolex-lemon model: development and applications. In: Proceedings of eLex 2017 conference, 2017, pp 19–21

Montiel-Ponsoda E, Rodríguez-Doncel V, Gracia J (2017) Building the legal knowledge graph for smart compliance services in multilingual europe. In: Rodríguez-Doncel V, Casanovas P, González-Conejero J (eds) Proceedings of the 1st Workshop on Technologies for Regulatory Compliance co-located with the 30th International Conference on Legal Knowledge and Information Systems (JURIX 2017), Luxembourg, December 13, 2017, CEUR-WS.org, CEUR Workshop Proceedings, vol 2049, pp 15–17, URL http://ceur-ws.org/Vol-2049/02paper.pdf

Muñoz S, Pérez J, Gutiérrez C (2009) Simple and efficient minimal RDFS. J Web Semant 7(3):220–234, DOI 10.1016/j.websem.2009.07.003, URL https://doi.org/10.1016/j.websem.2009.07.003

Muñoz-Soro JF, Esteban G, Corcho Ó, Serón FJ (2016) Pproc, an ontology for transparency in public procurement. Semantic Web 7(3):295–309, DOI 10.3233/SW-150195, URL https://doi.org/10.3233/SW-150195

Neumaier S, Polleres A (2019) Enabling spatio-temporal search in open data. Journal of Web Semantics 55:21–36, DOI https://doi.org/10.1016/j.websem.2018.12.007

Oksanen A, Tamper M, Tuominen J, Mäkelä E, Hietanen A, Hyvönen E (2019) Semantic finlex: Transforming, publishing, and using finnish legislation and case law as linked open data on the web. Knowledge of the Law in the Big Data Age 317:212–228

de Oliveira Rodrigues CM, de Freitas FLG, Barreiros EFS, de Azevedo RR, de Almeida Filho A (2019) Legal ontologies over time: A systematic mapping study. Expert Systems with Applications 130:12 – 30, DOI https://doi.org/10.1016/j.eswa.2019.04.009, URL http://www.sciencedirect.com/science/article/pii/S0957417419302398

Oltramari A, Piraviperumal D, Schaub F, Wilson S, Cherivirala S, Norton TB, Russell NC, Story P, Reidenberg JR, Sadeh NM (2018) Privonto: A semantic framework for the analysis of privacy policies. Semantic Web 9(2):185–203, DOI 10.3233/SW-170283, URL https://doi.org/10.3233/SW-170283

van Opijnen M, Palmirani M, Vitali F, van den Oever J, Agnoloni T (2017a) Towards ECLI 2.0. In: Parycek P, Edelmann N (eds) 2017 Conference for E-Democracy and Open Government, CeDEM 2017, Krems, Austria, May 17-19, 2017, IEEE Computer Society, pp 135–143, DOI 10.1109/CeDEM.2017.17, URL https://doi.org/10.1109/CeDEM.2017.17

van Opijnen M, Peruginelli G, Kefali E, Palmirani M (2017b) On-line Publication of Court Decisions in the EU. Report of the policy group of the project 'building on the european case law identifier', BO-ECLI, URL http://bo-ecli.eu/uploads/deliverables/Deliverable%20WS0-D1.pdf

Palmirani M, Governatori G, Rotolo A, Tabet S, Boley H, Paschke A (2011) Legalruleml: Xml-based rules and norms. In: Olken F, Palmirani M, Sottara D (eds) Rule-Based Modeling and Computing on the Semantic Web, 5th International Symposium, RuleML 2011- America, Ft. Lauderdale, FL, Florida, USA, November 3-5, 2011. Proceedings, Springer, Lecture Notes in Computer Science, vol 7018, pp 298–312, DOI 10.1007/978-3-642-24908-2_30, URL https://doi.org/10.1007/978-3-642-24908-2_30

Palmirani M, Martoni M, Rossi A, Bartolini C, Robaldo L (2018) Pronto: Privacy ontology for legal reasoning. In: Ko A, Francesconi E (eds) Electronic Government and the Information Systems Perspective - 7th International Conference, EGOVIS 2018, Regensburg, Germany, September 3-5, 2018, Proceedings, Springer, Lecture Notes in Computer Science, vol 11032, pp 139–152, DOI 10.1007/978-3-319-98349-3_11, URL https://doi.org/10.1007/978-3-319-98349-3_11

Pandit HJ, Fatema K, O'Sullivan D, Lewis D (2018) Gdprtext - GDPR as a linked data resource. In: Gangemi A, Navigli R, Vidal M, Hitzler P, Troncy R, Hollink L, Tordai A, Alam M (eds) The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings, Springer, Lecture Notes in Computer Science, vol 10843, pp 481–495, DOI 10.1007/978-3-319-93417-4_31, URL https://doi.org/10.1007/978-3-319-93417-4_31

Presutti V, Gangemi A (2008) Content ontology design patterns as practical building blocks for web ontologies. In: Li Q, Spaccapietra S, Yu ESK, Olivé A (eds) Conceptual Modeling - ER 2008, 27th International Conference on Conceptual Modeling, Barcelona, Spain, October 20-24, 2008. Proceedings, Springer, Lecture Notes in Computer Science, vol 5231, pp 128–141, DOI 10.1007/978-3-540-87877-3_11, URL https://doi.org/10.1007/978-3-540-87877-3_11

Publications Office of the European Union (2020a) Common Data Model (CDM). https://op.europa.eu/en/web/eu-vocabularies/cdm/, online; accessed 27 Nov 2020

Publications Office of the European Union (2020b) European Legislation Identifier (ELI). https://op.europa.eu/en/web/eu-vocabularies/eli/, online; accessed 27 Nov 2020

Rodríguez-Doncel V, Suárez-Figueroa MC, Gómez-Pérez A, Poveda-Villalón M (2013) License linked data resources pattern. In: Gangemi A, Gruninger M, Hammar K, Lefort L, Presutti V, Scherp A (eds) Proceedings of the 4th Workshop on Ontology and Semantic Web Patterns co-located with 12th International Semantic Web Conference (ISWC 2013), Sydney, Australia, October 21, 2013, CEUR-WS.org, CEUR

Workshop Proceedings, vol 1188, URL http://ceur-ws.org/Vol-1188/paper_7.pdf

Rodríguez-Doncel V, Delgado J, Llorente S, Rodríguez E, Boch L (2016) Overview of the MPEG-21 media contract ontology. Semantic Web 7(3):311–332, DOI 10.3233/SW-160215, URL https://doi.org/10.3233/SW-160215

Sanh V, Debut L, Chaumond J, Wolf T (2019) Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. CoRR abs/1910.01108, URL http://arxiv.org/abs/1910.01108, 1910.01108

Santos C, Pruski C, Silveira MD, Rodríguez-Doncel V, Gangemi A, van der Torre L, Casanovas P (2016) Complaint ontology pattern - COP. In: Hammar K, Hitzler P, Krisnadhi A, Lawrynowicz A, Nuzzolese AG, Solanki M (eds) Advances in Ontology Design and Patterns [revised and extended versions of the papers presented at the 7th edition of the Workshop on Ontology and Semantic Web Patterns, WOP@ISWC 2016, Kobe, Japan, 18th October 2016], IOS Press, Studies on the Semantic Web, vol 32, pp 69–83, DOI 10.3233/978-1-61499-826-6-69, URL https://doi.org/10.3233/978-1-61499-826-6-69

Steyskal S, Polleres A (2014) Defining expressive access policies for linked data using the ODRL ontology 2.0. In: Sack H, Filipowska A, Lehmann J, Hellmann S (eds) Proceedings of the 10th International Conference on Semantic Systems, SEMANTICS 2014, Leipzig, Germany, September 4-5, 2014, ACM, pp 20–23, DOI 10.1145/2660517.2660530, URL https://doi.org/10.1145/2660517.2660530

Uschold M, Gruninger M (1996) Ontologies: principles, methods and applications. Knowl Eng Rev 11(2):93–136, DOI 10.1017/S0269888900007797, URL https://doi.org/10.1017/S0269888900007797

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. In: Guyon I, von Luxburg U, Bengio S, Wallach HM, Fergus R, Vishwanathan SVN, Garnett R (eds) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, pp 5998–6008, URL http://papers.nips.cc/paper/7181-attention-is-all-you-need

Vegetti M, Leone HP, Henning GP (2011) PRONTO: an ontology for comprehensive and consistent representation of product information. Eng Appl Artif Intell 24(8):1305–1327, DOI 10.1016/j.engappai.2011.02.014, URL https://doi.org/10.1016/j.engappai.2011.02.014

Vos MD, Kirrane S, Padget JA, Satoh K (2019) ODRL policy modelling and compliance checking. In: Fodor P, Montali M, Calvanese D, Roman D (eds) Rules and Reasoning - Third International Joint Conference, RuleML+RR 2019, Bolzano, Italy, September 16-19, 2019, Proceedings, Springer, Lecture Notes in Computer Science, vol 11784, pp 36–51, DOI 10.1007/978-3-030-31095-0_3, URL https://doi.org/10.1007/978-3-030-31095-0_3

W3C JSON for Linking Data Community Group (2012) JavaScript Object Notation for Linking Data (JSON-LD). https://www.w3.org/community/json-ld/, online; accessed 17 May 2020

W3C Ontology-Lexica Community Group (2016) Lexicon Model for Ontologies: Final Community Group Report, 10 May 2016. https://www.w3.org/2016/05/ontolex/, online; accessed 27 Nov 2020

W3C Recommendation (2012) A Direct Mapping of Relational Data to RDF. https://www.w3.org/TR/2012/REC-rdb-direct-mapping-20120927/, online; accessed 17 May 2020

Wolf T, Debut L, Sanh V, Chaumond J, Delangue C, Moi A, Cistac P, Rault T, Louf R, Funtowicz M, Brew J (2019) Huggingface's transformers: State-of-the-art natural language processing. CoRR abs/1910.03771, URL http://arxiv.org/abs/1910.03771, 1910.03771

A Appendix

The appendix contains overview tables for the properties used in ELI and ECLI in different EU member states as well as links to the legal databases and example documents we used in this work.

A.1 ECLI properties used in different countries

Table 7 contains all properties from the ECLI ontology and shows which countries use which ECLI properties. Countries for which we use the non-governmental initiatives are highlight gray.

A.2 Overview of used Named Authority Lists

Table 8 shows for which properties NAL are used by different countries. We only list countries which provide metadata in RDF or RDFa format and are using NAL for legislative and judiciary documents. Furthermore,

Table 7 Overview of used ECLI properties of countries providing metadata using ECLI

ECLI Property	Austria	Finland	Germany	Netherlands
Data based on	*LKG*	*Finlex SPARQL Endpoint*	*RDFa*	*RDFa*
dcterms:abstract		✓		✓
dcterms:accessRights	✓		✓	
dcterms:contributor	✓	✓		
dcterms:coverage	✓		✓	
dcterms:creator	✓	✓	✓	✓
dcterms:date	✓	✓	✓	
dcterms:description		✓		
dcterms:identifier	✓		✓	✓
dcterms:isReplacedBy				
dcterms:issued		✓		✓
dcterms:isVersionOf	✓	✓	✓	
dcterms:language	✓	✓	✓	✓
dcterms:publisher	✓	✓	✓	✓
dcterms:references	✓			
dcterms:subject	✓			✓
dcterms:title				✓
dcterms:type	✓		✓	✓

not all countries do provide a dedicated NAL information page although they use NAL and they cannot be retrieved from the internet.

Table 8 Overview of the used NAL in different countries for legislative and judiciary documents. Countries for which we use the non-governmental initiatives are highlight gray.

NAL for property	Austria	Denmark	Finland	France	Italy	Luxembourg	Netherlands	Portugal	Spain
Data based on	*LKG*	*RDF*	*Finlex SPARQL Endpoint*	*RDFa*	*RDF*	*RDF*	*RDFa*	*RDFa*	*RDFa*
dcerms:type	✓	-	-	-	-	-	✓	-	-
dcterms:subject	✓	-	-	-	-	-	✓	-	-
eli:is_about	✓	-	-	-	-	✓	-	-	-
eli:jurisdiction	✓	-	-	-	-	-	-	-	✓
eli:language	✓	-	-	✓	✓	✓	-	-	✓
eli:passed_by	-	✓	✓	-	-	-	-	-	-
eli:publisher_agent	-	-	-	-	-	✓	-	✓	-
eli:relevant_for	✓	✓	-	-	-	-	-	-	-
eli:responsibility_of_agent	-	-	-	-	-	✓	-	✓	-
eli:rightsholder_agent	-	-	-	-	-	✓	-	✓	-
eli:type_document	✓	✓	-	-	✓	✓	-	✓	✓
eli:version	-	-	-	-	✓	-	-	-	✓

A.3 ELI properties used in different countries

Table 9 contains all properties from the ELI ontology and shows which countries use which ELI properties. Countries for which we use the non-governmental initiatives are highlight gray.

Table 9 Overview of used ELI properties of countries providing metadata using ELI including non-governmental initiatives

ELI Property	Austria	Denmark	Finland	France	Greece	Ireland	Italy	Luxemburg	Portugal	Spain
Data based on	*LKG*	*RDF*	*Finlex SPARQL Endpoint*	*RDFa*	*Nomothesia SPARQL Endpoint*	*RDF*	*RDFa, RDF*	*RDF*	*RDFa*	*RDFa*
eli:amended_by	✓		✓							
eli:amends	✓		✓		✓			✓		
eli:applied_by										
eli:applies										
eli:based_on		✓				✓		✓		
eli:basis_for		✓								
eli:changed_by	✓	✓	✓							
eli:changes	✓	✓	✓		✓	✓				
eli:cited_by			✓							
eli:cited_by_case_law	✓									
eli:cited_by_case_law_reference										
eli:cites			✓	✓	✓			✓	✓	
eli:commenced_by										
eli:commences										
eli:consolidated_by		✓								✓
eli:consolidates	✓	✓						✓		
eli:corrected_by										✓
eli:corrects								✓		
eli:date_applicability								✓		
eli:date_document	✓	✓		✓		✓	✓	✓		✓
eli:date_no_longer_in_force	✓	✓	✓					✓		
eli:date_publication	✓		✓	✓	✓		✓	✓	✓	✓
eli:description						✓		✓	✓	
eli:embodies	✓	✓	✓	✓		✓		✓	✓	✓
eli:first_date_entry_in_force	✓		✓		✓			✓		
eli:format	✓	✓	✓	✓		✓	✓	✓		✓
eli:has_another_publication										
eli:has_member	✓		✓							✓
eli:has_part			✓		✓	✓				
eli:has_translation										
eli:id_local	✓	✓	✓	✓	✓		✓	✓	✓	✓
eli:implemented_by										
eli:implements										
eli:in_force	✓	✓	✓						✓	
eli:is_about	✓		✓						✓	
eli:is_another_publication_of										
eli:is_embodied_by	✓	✓	✓	✓		✓	✓	✓		✓
eli:is_exemplified_by										
eli:is_member_of	✓		✓					✓		✓
eli:is_part_of			✓	✓				✓		
eli:is_realized_by	✓	✓	✓			✓	✓		✓	✓
eli:is_translation_of										
eli:jurisdiction	✓									✓
eli:language	✓	✓	✓	✓		✓	✓	✓	✓	✓
eli:legal_value		✓		✓		✓			✓	
eli:licence						✓		✓	✓	
eli:media_type										
eli:number	✓					✓			✓	✓
eli:passed_by		✓	✓	✓			✓			
eli:published_in		✓		✓	✓	✓				
eli:published_in_format						✓		✓	✓	
eli:publisher				✓		✓	✓	✓	✓	✓
eli:publisher_agent							✓		✓	
eli:publishes										
eli:realized_by	✓				✓					
eli:realizes	✓	✓	✓	✓		✓	✓	✓	✓	✓
eli:related_to			✓			✓				
eli:relevant_for	✓	✓				✓				
eli:repealed_by	✓		✓							
eli:repeals	✓		✓					✓		
eli:responsibility_of				✓				✓	✓	
eli:responsibility_of_agent								✓	✓	
eli:rights								✓		
eli:rightsholder						✓				
eli:rightsholder_agent								✓	✓	
eli:title	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
eli:title_alternative	✓	✓	✓					✓		
eli:title_short	✓	✓								
eli:transposed_by	✓				✓					
eli:transposes	✓	✓	✓			✓	✓	✓		
eli:type_document	✓	✓	✓	✓		✓	✓	✓	✓	✓
eli:uri_schema		✓				✓	✓		✓	
eli:version	✓		✓			✓	✓			✓
eli:version_date			✓							✓

A.4 Overview of legal databases and example documents

Tables 10 (Legislation) and 11 (Jurisdiction) provide an overview over the legal databases and example documents for all EU member states we used for our analysis in Section 5.

Table 10 Overview of legal databases and example documents for legislation (URLs of example documents are shortened)

Country	Legislation	Example Document
Austria	https://ris.bka.gv.at/	https://bit.ly/2UEq4E9
Belgium	http://www.ejustice.just.fgov.be/	https://bit.ly/30zHeGx
Bulgaria	https://dv.parliament.bg/	https://bit.ly/2MQ7q83
Croatia	http://nn.hr/	https://bit.ly/3hnXy34
Cyprus	http://www.cylaw.org/	https://bit.ly/3hew1Bc
Czech Republic	https://aplikace.mvcr.cz/sbirka-zakonu/	https://bit.ly/2XVLajg
Denmark	https://www.retsinformation.dk/	https://bit.ly/2YwzBhs
Estonia	https://www.riigiteataja.ee/	https://bit.ly/2XUxLIf
Finland	https://www.finlex.fi/	https://bit.ly/2UEbRXA
France	https://www.legifrance.gouv.fr/	https://bit.ly/2XUy4Tp
Germany	http://www.bgbl.de/	https://bit.ly/3cV7dLh
Greece	http://www.et.gr/	https://bit.ly/2B4bApT
Hungary	http://njt.hu/	https://bit.ly/3d2iQQN
Ireland	http://www.irishstatutebook.ie/	https://bit.ly/2XUhVxd
Italy	https://www.normattiva.it/	https://bit.ly/30zls5Z
Latvia	http://www.likumi.lv/	https://bit.ly/2UDUJBl
Lithuania	https://www.e-tar.lt/	https://bit.ly/2XVquIj
Luxembourg	http://legilux.public.lu/	https://bit.ly/30ycd5Q
Malta	https://legislation.mt/	https://bit.ly/2XSrgpq
Netherlands	https://www.officielebekendmakingen.nl/	https://bit.ly/2Ooq5IV
Poland	http://isip.sejm.gov.pl/	https://bit.ly/3hkOc8i
Portugal	https://dre.pt/	https://bit.ly/3gOtNrn
Romania	http://legislatie.just.ro/	https://bit.ly/37lNJhA
Slovakia	https://www.slov-lex.sk/	https://bit.ly/2XUz4a7
Slovenia	http://www.pisrs.si/Pis.web/	https://bit.ly/3cVWu2Y
Spain	https://boe.es/	https://bit.ly/2AjLPCk
Sweden	http://rkrattsbaser.gov.se/	https://bit.ly/3d2jQV3

Table 11 Overview of legal databases and example documents for jurisdiction (URLs of example documents are shortened)

Country	Judiciary	Example Document
Austria	https://ris.bka.gv.at/	https://bit.ly/37maTo6
Belgium	http://jure.juridat.just.fgov.be/	https://bit.ly/2AjPLTB
Bulgaria	https://legalacts.justice.bg/	https://bit.ly/3hpfdI2
Croatia	https://sudskapraksa.vsrh.hr/home	https://bit.ly/3fiabv0
Cyprus	http://www.cylaw.org/	https://bit.ly/30BXxD0
Czech Republic	http://www.nsoud.cz/	https://bit.ly/2Ywztyu
Denmark	https://domstol.dk/	https://bit.ly/2MQrqan
Estonia	https://www.riigiteataja.ee/	https://bit.ly/2UFChIK
Finland	https://www.finlex.fi/	https://bit.ly/3cS3w93
France	https://www.courdecassation.fr/	https://bit.ly/30CQjyq
Germany	http://www.bundesverfassungsgericht.de/	https://bit.ly/2BXbHUN
Greece	http://www.adjustice.gr/	https://bit.ly/3feYwwT
Hungary	https://birosag.hu/birosagi-hatarozatok-gyujtemenye/	Direct download
Ireland	https://beta.courts.ie/	https://bit.ly/2YpEwRm
Italy	http://www.italgiure.giustizia.it/	Registration required
Latvia	https://manas.tiesas.lv/eTiesas/	https://bit.ly/30Bettm
Lithuania	https://www.lat.lt/	https://bit.ly/30BHfdc
Luxembourg	https://justice.public.lu/	https://bit.ly/3fnuJ5n
Malta	https://justice.gov.mt/	https://bit.ly/2XVMmmK
Netherlands	https://data.rechtspraak.nl/	https://bit.ly/3hlxaqN
Poland	http://orzeczenia.nsa.gov.pl/	https://bit.ly/2zuAq22
Portugal	https://jurisprudencia.csm.org.pt/	https://bit.ly/3dQNspr
Romania	http://www.rolii.ro/	https://bit.ly/2YtigWL
Slovakia	https://obcan.justice.sk/	https://bit.ly/2MOOBDX
Slovenia	http://www.sodnapraksa.si/	https://bit.ly/2XRJEia
Spain	http://www.poderjudicial.es/	https://bit.ly/3fm2ALX
Sweden	https://rattsinfosok.domstol.se/	https://bit.ly/3fckCjq

8. TempCourt: Evaluation of temporal taggers on a new corpus of court decisions

Bibliographic Information

Navas-Loro, M., Filtz, E., Rodríguez-Doncel, V., Polleres, A. and **Kirrane, S.**, 2019. TempCourt: evaluation of temporal taggers on a new corpus of court decisions. The Knowledge Engineering Review, 34, p.e24. 10.1017/S0269888919000195.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Writing - Review & Editing, and Supervision.

Copyright Notice

The Knowledge Engineering Review, Vol. 00:0, 1–24. © 2004, Cambridge University Press
DOI: 10.1017/S0000000000000000 Printed in the United Kingdom

TempCourt: Evaluation of Temporal Taggers on a new Corpus of Court Decisions

María Navas-Loro, Erwin Filtz, Víctor Rodríguez-Doncel, Axel Polleres and Sabrina Kirrane[1]

María Navas-Loro and Víctor Rodríguez-Doncel
Universidad Politécnica de Madrid, Montegancedo Campus
D3204 – Ontology Engineering Group
Madrid, Spain
E-mail: {mnavas, vrodriguez}@fi.upm.es
ORCiD: 0000-0003-1011-5023, 0000-0003-1076-2511

Erwin Filtz, Axel Polleres and Sabrina Kirrane
Vienna University of Economics and Business
Institute for Information Business
Vienna, Austria
E-mail: {firstname.lastname}@wu.ac.at
ORCiD: 0000-0003-3445-0504, 0000-0001-5670-1146, 0000-0002-6955-7718

Abstract

The extraction and processing of temporal expressions in textual documents has been extensively studied in several domains, however for the legal domain it remains an open challenge. This is possibly due to the scarcity of corpora in the domain and the particularities found in legal documents that are highlighted in this paper. Considering the pivotal role played by temporal information when it comes to analyzing legal cases, this paper presents TempCourt, a corpus of 30 legal documents from the European Court of Human Rights, the European Court of Justice and the United States Supreme Court with manually annotated temporal expressions. The corpus contains two different temporal annotation sets that adhere to the TimeML standard, the first one capturing all temporal expressions and the second dedicated to temporal expressions that are relevant for the case under judgment (thus excluding dates of previous court decisions). The proposed gold standards are subsequently used to compare ten state-of-the-art cross-domain temporal taggers, and to identify not only the limitations of cross-domain temporal taggers but also limitations of the TimeML standard when applied to legal documents. Finally, the paper identifies the need for dedicated resources and the adaptation of existing tools, and specific annotation guidelines that can be adapted to different types of legal documents.

Keywords: legal corpus, temporal annotation, case law, legal NLP, evaluation

1 Introduction

Legal information systems are indispensable tools for many legal practitioners. An emerging area of research is the use of text analytics to derive structured data from legal text (e.g. norms, opinions, recommendations or court decisions). In this context, one of the most relevant activities is the automatic extraction and processing of events and temporal expressions with a view to creating timelines.

In this context, a *temporal expression* (TE) is a word or sequence of words making reference to a time instant (e.g. 'seven o'clock') or a time interval (e.g. 'from seven to ten'). Temporal expressions frame events or happenings implicitly or explicitly mentioned in the document. *Temporal relations* bind TEs to events and determine the relative position of some events with respect to other events (through relations such as 'after' or 'before').

The example below is a text excerpt from a court decision of the European Court of Human Rights describing the facts of the *Aras v. Turkey* case (no. 21824/07, 20 July 2017). The text contains three TEs

[1] The two first authors equally contributed to this work.

(in bold below), two of them being in an absolute form (e.g. *11 December 2002*) and one in a relative form (*same day*).

> ”On **11 December 2002** the applicant’s statement was taken by the public prosecutor and, on the **same day**, the judge at *Istanbul State Security Court* ordered her detention on remand. On **7 December 2002** the applicant was arrested on suspicion of membership of a terrorist organisation.”

This temporal information is related to three events, namely, the public prosecutor taking the statement, the judge ordering a detention, and the applicant being arrested. Each of the events is related to the other entities, either named (*Istanbul State Security Court*) or not (the applicant). Although the two absolute dates in the text above appear in the same format, this is not always the case and very often different formats are used even within the same document. Although our exemplary legal case can be used to motivate an investigation into both temporal and event extraction (e.g. (39, 46)), in this paper we focus specifically on temporal expressions.

Temporal taggers operate on texts like the one above, performing different tasks, namely TE *identification*, *normalization*, and *classification*. *Identification* (also called *detection* or *extraction*) is a task which involves finding TEs and their start and end position in the text. *Normalization* (or *anchoring*) is a task that interprets TEs to obtain specific instants and intervals represented in a standard format. This task resolves relative TEs (as ‘the same day’) from context information, localizes time formats (i.e. mm/dd/yy vs dd/mm/yy), considers timezones and enables the reformatting of the TEs into a standard format (e.g. ISO 8601[2]). In contrast, a *classification* task is used to determine which kind of TEs have been found. For instance, **on 7 December 2002** is most likely a time point, while **from 7 December 2002 to 12 December 2002** is a time interval. The temporal expressions found by the temporal taggers are usually represented in domain-agnostic formats, such as TimeML[3]. TimeML is the most widely accepted mark-up language for temporal expressions, and its use is justified over domain-specific formats (e.g. Akoma Ntoso[4] in the legal domain) for it permits representing more details and nuances specific to the temporal terms.

Although several temporal taggers have been proposed and investigated in different domains, the suitability of existing methods to extract temporal information from legal texts has been relatively unexplored to date as being only a side effect for other tasks, for instance document classification or reasoning over documents. Additionally, the lack of temporal resources in the domain is a major drawback when it comes to research in this direction. To the best of the authors’ knowledge, there is no preexisting temporal annotation gold standard based on legal text corpora. Consequently, there is no previous evaluation of how well standard temporal tagging tools perform in this domain. To this end, this paper makes the following contributions:

- an analysis of the particularities of temporal annotation in the legal domain;
- the provision of a temporally tagged corpus (named TempCourt, freely available online[5]), composed of legal documents from three sources, namely the European Court of Human Rights, the European Court of Justice and the United States Supreme Court; and
- a broad comparison of state-of-the-art cross-domain temporal taggers using the proposed gold standard.

The remainder of this paper is structured as follows: Section 2 describes existing work on temporal information extraction. Section 3 examines the particularities of dealing with temporal expressions in the legal domain. Section 4 presents the methodology used for the construction of the TempCourt corpus. Section 5 introduces several existing temporal taggers. Section 6 evaluates ten state-of-the-art temporal taggers over documents from three different legal sources, namely the European Court of Human Rights,

[2] `https://www.iso.org/iso-8601-date-and-time-format.html`
[3] `http://www.timeml.org`
[4] `http://www.akomantoso.org/`
[5] `https://tempcourt.github.io/TempCourt/`

the European Court of Justice and the United States Supreme Court. Finally, Section 7 presents our conclusions and discusses future work.

2 Related Work

Temporal tagging is a mature area of research that has been applied in different contexts, but scarcely in the legal domain. This section reviews several corpora with temporal annotations, along with the work done previously in temporal annotation of legal texts and in other domains.

The temporal information of a text document can be represented in structured, ad-hoc formats such as TIDES TIMEX2 (10) or TimeML (35). TimeML is the ISO standard[6] for time and event markup and annotation. Other general-purpose annotation standards can also be used to represent TEs, such as the W3C Web Annotations[7] or the NLP Interchange Format[8] (NIF) (15). TimeML uses TIMEX3 tags (modelled on previously mentioned TIMEX2) for marking TEs, and distinguishes between different `types` (namely, `DATE, DURATION, TIME` and `SET`, the latter being the type associated with sets of recurrent times). Other attributes in TIMEX3 tags allows for the expression of temporal information as a normalized value, for instance the actual date instead of relative expressions such as `yesterday`, following the ISO 8601 standard (`value`). TIMEX3 can also mark the presence of modifiers (`mod`) such as `END` or `LESS_THAN`, or specific information for each type, such as the frequency (`freq`) for `SET`.

Thus, for the analysis of temporal expressions, the following three domains received the most attention: medical texts (e.g. the THYME corpus (44)), news (e.g. the Timebank corpus (34) and the MEANTIME corpus (29)) and historical documents (e.g. the Wikiwars corpus (28)). Corpora have also included texts in different language registers, such as tweets (45), colloquial texts (43) or scientific abstracts (43). However, to the best of the authors' knowledge, there are no temporally annotated legal corpora publicly available that relate to English language court decisions. Although annotation challenges (both in general and also in different specific domains) have been identified in literature (17, 43, 44), very little work has been conducted in connection with the legal domain. A description of the different approaches adopted by existing temporal taggers, including the identification of several state-of-the-art temporal taggers, can be found in Section 5.

In the legal domain, previous research work by Schilder (39) already pointed out the relevance of the temporal dimension of information in legal documents. In this work, an analysis of the different types of legal documents and the temporal information that can be found in them was outlined. Schilder distinguished between dates in transactional documents (namely, documents written by lawyers for specific transactions, such as contracts or agreements), constraints in statutes or regulations, and legal narratives in case law. While the first two types of documents received dedicated attention, narratives in case law were assimilated to narratives present in news. An alternative approach proposed by Isemann et al. (16), used both Named Entity Recognition (NER) and temporal processing to extract temporal dependencies from regulations with no narrative-structure. The authors also described some of the recurrent pitfalls temporal taggers have to deal with, such as the confusion between legal references (e.g. 'Directive *2009*/28/EC') and actual dates, as shown in Table 2, or the distinction between *episodic* and *generic* events —the former referring to a specific moment (e.g. 'the rescission of the contract was done on 7 December 2017') and the latter referring to an event in general truths, laws, rules or expectations (e.g. 'Every rescission implies the following actions'). Other approaches in the legal domain include works on transactional documents by Naik et al. (30), where a first framework for dealing with temporal information in that kind of texts is proposed. Also additional efforts focused on reasoning with legal evidence (burden of proof) and coherence of narratives (e.g. plausibility and completeness) were made (49), using temporal information but without extracting it from scratch.

[6] ISO 24617-1 Language Resource Management - Semantic Annotation Framework (SemAF) - Time and Events (SemAF Time and ISO-TimeML)

[7] `https://www.w3.org/TR/annotation-model/`

[8] `http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core#`

Works in other fields, such as the medical domain, are also of interest since they share common requirements, such as the need of domain knowledge for identifying specific events[9] and for dealing with the existence of several timelines in the same text, among others. The analysis by Styler et al. (44) in the clinical domain identifies the need of specific guidelines for temporal annotation, which require domain-specific temporal knowledge and the definition of general phases in clinical processes (some kind of commonsense domain knowledge). Furthermore, new tags not included in the temporal annotation standard TimeML, commonsense information and events are defined in the same work, along with annotation needs and different timelines (such as discussions with other colleagues and notes about risks in treatment) were redesigned for fitting the medical particularities. We work under the assumption that most of these considerations and challenges can also be present in a similar form in legal documents, requiring therefore a dedicated approach. We conclude that one of the primary limitations of existing work is the fact that no special consideration has to date been given to both the narrative structure and the particularities of the legal domain (see Section 3 for additional details).

3 Particularities of Legal English

Temporal information is a very important aspect of legal cases. It has an effect on the version of the applicable law and it creates a chronological order of events in a legal case. Sometimes it is important to know whether event *A* or event *B* happened first. In addition, temporal information is also used to assess whether past events may be time-barred.

When it comes to the automatic extraction of temporal information from legal documents, it is important to highlight that legal documents, and in particular court decisions, slightly differ in structure and writing style from documents from other domains. These differences include deeper parse trees, differences in part-of-speech distributions and more words per sentence (8).

Judgments are usually framed in legal processes following specific procedures, events and timings, whose mention in the judgment constitutes context information that should not be lost in the annotation process. An example of this is the concept of *preliminary ruling*, a legal term referring to a phase previous to the decision when the European Court of Justice is asked how law should be interpreted, being therefore a reference to this period and a hint for temporal localization of other events. Also specific events happening in legal frameworks must be considered when processing legal texts, as done in other domains such as the medical domain (44).

3.1 Structure of Judgments

Table 1 illustrates the differences in document structure for judgments made by the European Court of Justice (ECJ) and the United States Supreme Court (USC), and preliminary assessments of applications submitted to the European Court for Human Rights (ECHR). The court decisions from the European courts follow a similar structure that already hints which categories of TE could be expected in different parts of the texts. In particular, both ECJ and ECHR start with a description of the involved parties (section A) and are then followed by a case summary (B), stating concisely why this case has been brought to the respective court and what happened so far in terms of the legal proceedings. In ECJ decisions, the legal proceedings are followed by the applicable legal framework and then by the case description, whereas the ECHR structure is the other way round. The decisions of the ECJ and ECHR courts conclude with the matching of the law with the facts of the case under the legal basis (E) and the resulting judgment (F). In contrast to ECHR documents, the ‘Legal framework’ section (D) in ECJ documents cites European and local legislation, without any direct references to the case itself, and as such this information was excluded from the final documents in the corpus presented in this paper. Although TEs corresponding to other related events such as prior decisions could be extracted from these sections, we focus on case-related temporal information and leave the extraction of events for future work. Apart from beginning with the involved

[9] For instance, *diagnosis* such as *tumors* or medical tests are relevant events that should appear in a timeline of a medical doctor, as stated by Styler et al. (44), but not in other types of texts. Similarly, specific legal events such as *preliminary rulings* (explained in Section 3) in European judgments are always relevant to lawyers, although they never appear in other kinds of texts.

Table 1 Structure of ECJ, ECHR and USC decisions.

Section	ECJ	ECHR	USC
A	Involved parties	Involved parties	Involved parties
B	Case summary	Procedure	Syllabus
C	Legal framework	Circumstances of the case	Main Opinion
D	Circumstances of the case	Legal framework	Concurring and dissenting opinions
E	Court assessment	Court assessment	
F	Judgment	Judgment	

parties (A) in a particular case, the structure of USC decisions is quite different. The second section (B) is called 'syllabus' and contains a short summary of the case. It is followed by the main opinion (C), that includes the final decision of the court and explains how the court came to this decision, by referring to the legal foundations. The last part of a decision states, where applicable, the concurring and dissenting opinions of the involved justices (D). An opinion is called 'concurring' if a justice follows the main opinion but grounds the decision on a different legal rationale. A dissenting opinion is issued in cases where a justice disagrees with the main opinion and the underlying legal rationale. Following a consistent structure makes legal documents comparable, and fulfills the expectations of readers who are used to find a specific kind of information always at the same place in the same kind of legal document. Furthermore, the consistent structure of legal documents (from the same authority or within a jurisdiction) leads to expectations with respect to the type of temporal information which could be expected in each section of the document. We expect temporal references describing the facts of the case (*what happened when?*), which could be used for generating timelines for document summarization, to be present in the *case summary* (ECHR), *Circumstances of the case* (ECJ) or *syllabus* (USC) sections in the judgment of the respective deciding court, but mentions to general temporal events to appear throughout the entire document. The structural properties of legal documents could also be exploited for the automatic creation of timelines as legal documents can be very long. For the analysis of a judgment it is necessary to understand the order of the events as this can affect the legal proceedings. The easier understanding could be supported with a visual representation of the order of events, hence a timeline that shows the important events and provides a visual summary of the case.

Dates are used in virtually every domain. In contrast to posts published in social media, e.g. Facebook or Twitter, where every user might write dates in different formats, documents from official authorities, such as courts, usually use the same format to represent dates in all documents. Differences in date representation that can be noticed are for instance the order of day and month or the used separators. Therefore, the differences in date representation are seldom found within a document, but may vary from court to court.

3.2 Mistaken or Misleading Temporal Expressions in Legal Documents

References to legal documents often include some sort of temporal information, usually forming a text pattern prone to be confused with a true temporal expression (see examples in Table 2). Typical references containing temporal information are references to previous court cases, laws or legal literature, where the temporal information indicates a point in time when the respective reference has been decided or published. However, temporal information contained in references is not considered relevant for a specific case in terms of describing *what happened when?*. For example, the expressions in Table 2 convey some temporal information, e.g., four-digit sequences that could be recognized as years, but which only in some cases indeed refer to actual years. Tagging these kind of expressions as TEs may become a major problem and lead to additional errors —for instance nearby dates in the text can be normalised from these wrong references leading to further errors in the entire text. Additionally, references to other legal documents often present their creation date, that must be differentiated from dates in the document timeline of referred case events. An example of this, where the given date refers to the date of a Council Directive of the European Union and thus is irrelevant for the narrative of the text, is the excerpt below:

Table 2 Examples of mistaken and misleading temporal information.

Source	Example	Description
ECHR	no. 7334/13, 127 - 128, ECHR **2016**	Reference to another case
ECHR	Timoshin v. Russia (**dec.**)	Reference to a decision (dec.), often confused with the month of December
ECJ	OJ **2008** L 348 p. 98	Reference to official journal of the EU
ECJ	Directive **2008**/115/EC	Reference to a directive published in 2008
USC	See Va. Code Ann. §53.1-165.1 (**2013**)	Law reference
USC	[...] 772 F. 3d **1328**, **1333** (CA10 **2014**)	Precedent case reference

> ”Council Directive 93/13/EEC of **5 April 1993** on unfair terms in consumer contracts must be interpreted as not precluding (...)”

For processing these kinds of expressions, we could first detect and hide them from the temporal tagger (e.g., replacing them for an innocuous expression before the processing and restoring them afterwards) or alternatively we could filter them in a post-processing step.

3.3 Incompleteness of the TimeML Standard for the Annotation of Legal Documents

During the annotation of the corpus presented in this paper, we also detected relevant information that the TimeML standard is not able to represent. The main drawbacks of the TimeML standard applied to legal documents are summarized in the following subsections.

3.3.1 Specific Legal Terminology as Modifiers

Documents in the legal domain are rich in non-colloquial noun phrases representing temporal information. For example, the sentence “the *expiry* of the three-day period” is badly understood by parsers in comparison with “the *end* of the three-day period”.

Similarly, when the extension of a duration is uncertain (e.g, range between two points, such as in “*period of between seven and thirty days*”), there is no way to properly represent the uncertainty. Likewise, when referring to different possibilities frequently found in the legal language such as “*was a year or two more of prison time*”, this information cannot be properly annotated —even if some taggers such as SUTime (4) provide alternative values for similar expressions, i.e. “*from one to two years*”, the standard specification does not allow them.

The standard should be able to represent all these particularities of the legal domain. Similarly, a temporal tagger for the legal domain should be able to reason with this level of granularity.

3.3.2 Missing Levels of Granularity

Temporal expressions are important in the legal domain. Not only points in time which are used to determine the applicability of a particular law, but also durations are of high importance especially in formal laws determining the limitations of time (e.g. to plead the statute of limitations) for actions that must be taken before they preclude. For instance, in the legal domain a different way to count days is often applied. While `DURATION` is sufficient to indicate the absolute lapse of time, TimeML is not sufficient to indicate non-absolute durations such as “*10 working days*”.

Temporal taggers could be enhanced with external knowledge to recognize special constraints being applied to durations, for instance, work calendars where working days are identified. Eventually, also the capability to reason at this level of granularity would be desirable.

3.3.3 Exhaustive List of Attributes

The TimeML attribute `functionInDocument` allows for the marking of some temporal expressions as special reference ones, but just as one among: *‘creation_time’*, *‘expiration_time’*, *‘modification_time’*, *'publication_time'*, *'release_time'*, *'reception_time'* or *'none'*. This is not enough for legal documents, where domain expressions such as *'lodgement_time'*, *'argued_time'* or *'decision_time'* would be more useful. Domain-specific extensions to the TimeML standard could be used to solve this particular problem.

3.3.4 Limited Expressivity of the Existing Format

There are temporal expressions whose anchor time is not the DCT (Document Creation Time) nor are they related to any temporal expressions in the text, but in other legal documents cited in the text, such as in "*The dissent also relies heavily on Missouri v. Frye, 566 U. S. 134 (2012), and Lafler v. Cooper, 566 U. S. 156 (2012). (...) Lafler, decided* **the same day** *as Frye (...)*".

To cover this issue a temporal tagger needs to be combined with a co-reference system in order to find the matching events to which a certain temporal expression relates. This could be addressed by making use of the clear structure of legal documents which usually use the same citation style in all documents such that temporal expressions appearing next to case references can be annotated as belonging to them.

The official TIMEX3 tags cannot properly represent precise intervals on their own. A time interval such as "*between 12.45 and 18.45*" can only be represented as a DURATION (of 4 hours) or as two unrelated datetime points. This is a problem in cases where exact intervals are needed to solve legal problems such as confirming an alibi or evaluating exact timespans.

While some of these limitations could also be found anecdotally in other kinds of texts, they are common in legal documents, and relevant to their temporal dimension. Other non legal issues raised when using the TimeML standard are the correct extent of the tags or how to deal with multiple normalization options (for instance, "*one decade*" can be "`P1DE`" or "`P10Y`", and "*a few weeks later*" can be a duration with a known *beginPoint* or a `FUTURE_REF`).

3.4 Temporal Dimensions

In legal texts temporal expressions can be attributed to different temporal dimensions. We identify three different temporal dimensions and illustrate them based on the example decision *Sophie Mukarubega v Préfet de police and Préfet de la Seine-Saint-Denis* (ECLI:EU:C:2014:2336).

3.4.1 Temporal Dimension of the Legal Process

Each court proceeding is based on some formal rules and new events are added with the gradual advancement of the legal proceeding. This temporal dimension covers events related to the legal process itself, for instance the date a lawsuit has been filed, date of the hearings or the decision date.

> "By a decision of **21 March 2011**, adopted after hearing the person concerned, the Director General of the Office francais de protection des refugies et apatrides (OFPRA) (Office for the protection of refugees and stateless persons) rejected her application for asylum. (...)"

This temporal expression indicates that a certain event has happened, in this case the rejection of asylum.

3.4.2 Temporal Dimension of the Case

This temporal dimension covers factual information about the case which serves as the basis for a judgment.

> "Ms Mukarubega, who was born on 12 March 1986 and is of Rwandan nationality, entered France on **10 September 2009** in possession of a passport bearing a visa. (...)"[10]

The highlighted date refers to a fact of the case, hence the entrance of France.

[10] Please note that the same sentence contains two temporal expressions which are attributed to two different temporal dimensions.

3.4.3 Temporal Dimension of the Legal Context

Temporal information can also affect the legal context and determine the applicable law and the degree of the resulting penalty. This is especially relevant when determining the limitation of liability in time or when checking a legal reference to know the applicable law version. We can illustrate this in the following example of a preliminary ruling request to the European Court of Justice with the dates marked in bold.

> "(...) This request for a preliminary ruling concerns the interpretation of Article 6 of Directive 2008/115/EC of the European Parliament and of the Council of **16 December 2008** (...)"
>
> "Ms Mukarubega, who was born on **12 March 1986** and is of Rwandan nationality, entered France on **10 September 2009** in possession of a passport bearing a visa. (...)"[10]
>
> "By a decision of **21 March 2011**, adopted after hearing the person concerned, the Director General of the Office francais de protection des refugies et apatrides (OFPRA) (Office for the protection of refugees and stateless persons) rejected her application for asylum. (...)"

The first, third and fourth temporal expression refer each to a point in time that is relevant for the legal context. A preliminary ruling for the interpretation of an article requires the article to exist (first date). In the second paragraph, the birth date is general information about the defendant, which does not affect the *temporal dimension of the case* but might influence the *temporal dimension of the legal context*. This is especially important in criminal cases when the birth date in conjunction with the date of the offence constitutes the application of the criminal law relating to juvenile offenders. The third date, on the other hand, refers to a fact of the case, the day of entrance in the host country, being therefore part of the *temporal dimension of the case*. Finally, the fourth date indicates when a decision on the case in the legal process was reached, so this TE corresponds to the *temporal dimension of the legal process*.

3.4.4 Conflict of Temporal Dimensions

One could wonder whether there is the possibility of an overlap of temporal dimensions such that a single event might be part of the *temporal dimension of the legal process* and of the *temporal dimension of the case*. For instance, in cases that go through the entire hierarchy of courts, decisions are reversed by higher courts and referred back to the previous court. In these cases the judgments of the previous courts do have an influence on the following proceedings as courts might be bound to former judgments or receive an order to investigate certain parts of former proceeding in more detail and do more investigation work. However, from our perspective the *temporal dimension of the case* encompass the events inherent to the case, while revisions and case remands do not change anything in the temporal order of events in the original case, instead such information adds context which is relevant for the further proceeding without affecting the *temporal dimension of the case*.

In this section we outlined the particularities of documents in the legal domain which encompass the special structure of judgments, legal terminology, annotation standards such as TimeML and its incompleteness for annotation tasks in the legal domain as well as a classification of temporal dimensions present in judgments.

4 Temporal Annotation

In this section, we aim at evaluating in how far the automatic identification (and normalization) of temporal expressions is feasible using existing taggers, and to test the effectiveness of such tools. In order to enable such an evaluation, we propose two gold standards, one domain focused (LegalTimeML, composed of temporal information important for the facts of the case) and one generic (StandardTimeML, including all temporal information), that can be used to compare the results of temporal taggers and to determine which of them is most suited to be used when working with legal documents. The temporal annotation of all documents used in this work is based on the TimeML annotation language[11]. Figure 1 illustrates the

[11] `https://catalog.ldc.upenn.edu/docs/LDC2006T08/timeml_annguide_1.2.1.pdf`

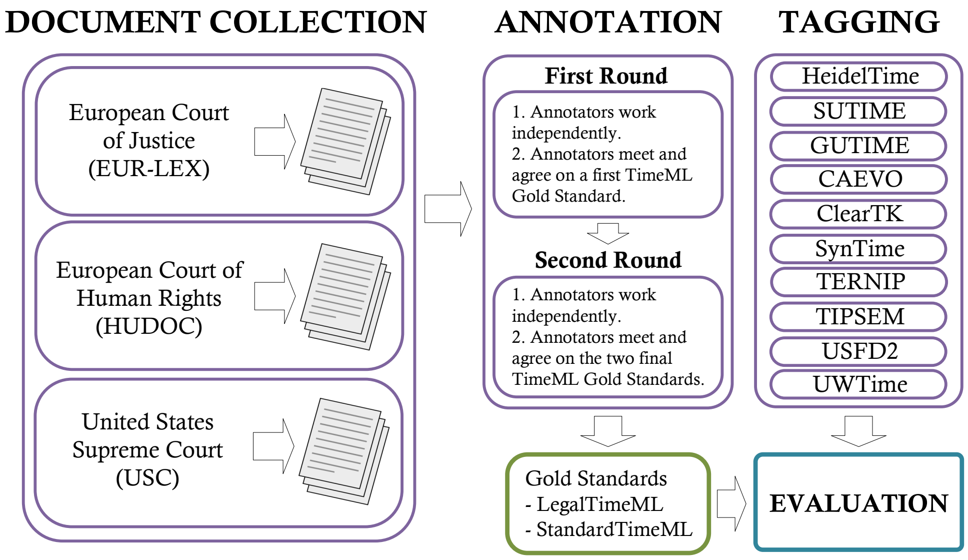

DOCUMENT COLLECTION
European Court of Justice (EUR-LEX)
European Court of Human Rights (HUDOC)
United States Supreme Court (USC)
ANNOTATION
First Round
1. Annotators work independently.
2. Annotators meet and agree on a first TimeML Gold Standard.
Second Round
1. Annotators work independently.
2. Annotators meet and agree on the two final TimeML Gold Standards.
Gold Standards
- LegalTimeML
- StandardTimeML
TAGGING
HeidelTime
SUTIME
GUTIME
CAEVO
ClearTK
SynTime
TERNIP
TIPSEM
USFD2
UWTime
EVALUATION

Figure 1: Outline of our work, including document collection, annotation and evaluation of taggers.

methodology we followed in order to create and evaluate our proposed gold standards. In the document collection phase we retrieve the documents, and in the annotation phase we create in two rounds the gold standards which are then used to compare to the results retrieved from the temporal taggers in the tagging phase.

4.1 Document Collection

Although different types of documents could have been chosen to create a gold standard in the legal domain, our proposed corpus TempCourt is composed of judgments and preliminary assessments of applications as they contain a large number of temporal expressions.

As many of the taggers do not have full support to other languages, we selected court decisions in English to enable a fair comparison of the results of the temporal taggers. Also, in order to increase the variety of ways in which temporal information is represented in different types of courts, we decided to investigate the judgments of courts acting in different jurisdictions and domains. Specifically, we focus on the court decisions of the European Court of Justice (ECJ), which is the highest court of the European Union, and of the United States Supreme Court (USC), and on preliminary assessments of applications submitted to the European Court of Human Rights (ECHR). The documents for the two European courts are available in the respective databases, namely EUR-Lex[12] for the ECJ and HUDOC[13] for the ECHR, while the USC documents were collected from the website of the United States Supreme Court[14]. The corpus created for this work, named TempCourt, consists of thirty court decisions, composed of an even distribution of ten documents per court in each subcorpus. Documents provided by the European Court of Human Rights are allowed to be reproduced for private use or for the purposes of information and education in connection with the Court's activities when the source is indicated and the reproduction is free of charge[15]. The same policy applies to documents retrieved from EUR-Lex whose documents are allowed to be reused in conjunction with the Commission Decision of 12 December 2011 on the reuse of Commission Documents[16] for commercial and non-commercial purposes given the source is

[12] http://eur-lex.europa.eu/
[13] https://hudoc.echr.coe.int
[14] https://www.supremecourt.gov/
[15] https://echr.coe.int/Pages/home.aspx?p=disclaimer&c=
[16] https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32011D0833

Table 3 Corpus statistics

Corpus	# Doc.	# Tokens	Doc. Size (Avg. KB)	Doc. Size (Avg. Tokens)	Sentence length (Avg. Tokens)
ECHR	10	7,252	4	725	13
ECJ	10	53,044	32	5,304	32
USC	10	50,874	25	5,087	18
Total	30	111,170	20	3,705	21

Table 4 Statistics of corpora annotated with TimeML in literature.

Corpus	# Doc.	# Tokens	Doc. Size (Avg. Tokens)
TimeBank[20]	183	78,444 (61,000[21])	428.7
AQUAINT[22]	73	34,154	467.9
TempEval-3 Eval. (*Platinium*) (47)	20	~6,000[23]	~300
WikiWars (42)	22	119,468	5,430.4
Time4SMS (42)	1,000	20,176	20.2
Time4SCI (42)	50	19,194	383.9

acknowledged[17]. Documents published by US governmental institutions (such as the US Supreme Court) are in the public domain[18].

Legal documents often contain names of persons, especially court decisions. The documents in our corpus contain the names of the involved judges and the names of parties in a non-anonymized way. Names are considered personal data and need to respect the General Data Protection Regulation[19] (GDPR) which in the case of public data involves providing transparency with respect to the processing on request (Article 14 GDPR). Consent for the processing of personal data from the data subject is not required for public data.

For the purpose of temporal annotations we are mainly interested in the section of the court decisions describing the facts of a case, because we expect to find the most valuable temporal information about the chronology of a case in this section, whereas temporal information in other sections is expected to be relating to laws or previous cases. Therefore, we omitted in our corpus the “Legal framework” section of the documents from the ECJ.

The figures in Table 3 illustrate differences between documents depending on their source. Although we include documents from three different courts in this paper, the corpus statistics show that the documents in the ECJ and USC subcorpora are similar in terms of document size and length. The documents in the ECHR subcorpus are only one fifth in terms of size in comparison with the other two subcorpora. As stated previously, legal texts often make use of very long and complicated sentences to explain legal details, thus we also included the average sentence length in tokens for each corpus. We show that the sentences of the ECHR are roughly one third of length compared to the USC court decisions, and also tend to be shorter than the ones in the ECJ corpus. These numbers contrast with those relating to corpora from other domains and sources, such as Wikipedia articles (25.1 words per sentence (18)), the CONLL 2007 corpus of documents from the Wall Street Journal (24 and 23.4 tokens per sentence in training and test data, respectively (31)) and the basic corpus of everyday documents (33), including all kind of common texts, such as banking or

[17] `https://eur-lex.europa.eu/content/legal-notice/legal-notice.html#droits`
[18] `https://www.copyright.gov/title17/92chap1.html#105`
[19] Regulation (EU) 2016/679.
[20] `http://www.timeml.org/timebank/documentation-1.2.html`
[21] The website just mentions 61k non-punct tokens, the other figure was extracted from (42).
[22] `http://www.timeml.org/timebank/aquaint-timeml/aquaint_timeml_1.0.tar.gz`
[23] Just approximate figures were provided (47).

shopping documents (with an average of 17.2 words per sentence). Regarding the amount of documents in each corpus, Table 4 provides an overview (extracted from previous literature (47)) of the size of referential corpora manually annotated with TimeML. These figures provide evidence that despite the fact that we have less documents per corpus our corpus is substantially bigger in terms of tokens than most of the previous corpora.

4.2 Annotation

For each subcorpus (ECJ, ECHR and USC), the ten documents were selected at random. In order to compare the results of different temporal annotation tools, all thirty documents have been annotated in multiple steps. In the first part of the annotation process, two different annotators performed the manual annotation of the documents following the TimeML guidelines[24]. Once manual annotation, which was done independently by two persons using General Architecture for Text Engineering (GATE) (7), was completed, they met to create a gold standard with annotations agreed by both annotators. When doubts arose, the TimeML guidelines were consulted specifically looking for similar cases; if the doubt persisted, also the TIDES TIMEX2 guidelines[25] were examined, as referred to in the TimeML annotation guidelines. However, due to the particularities of the legal domain, some annotation decisions needed further discussion as shown in the following examples:

1. The word `now` is heavily used in legal documents and was only annotated when it was not used as an adverb, hence the meaning is not `currently` or `at the moment`. For instance in the case ECJ C-457/12, *[...] so the provision is* `now` *worded as follows [...].*

2. For the annotation of references to the present time, some taggers use the *PRESENT_REF* token as a value, while others normalize to a date (usually the creation date). We decided for the legal domain we should follow the latter approach, since all the documents in the corpus contain this information and humans would also be able to derive it.

3. Legal documents, especially judgments, often contain references to previous court decisions in the legal grounding of a decision. The citation of such preceding cases depends on how decisions of such courts are usually referenced. Typically, a year is contained in the citation and annotated as a temporal reference. Temporal information contained in identifiers used to refer to collections of court decisions (e.g. *2006*I) or included in the document identifier, should not be annotated (e.g. EC:C:*2013*:180).

4. Expressions such as `the date indicated`, appearing for instance in the excerpt "*the application lodged on* ***the date indicated*** *in (...)*" are not considered as temporal references but as co-references, being therefore not annotated in the gold standard, since a temporal tagger would not be expected to do so.

The discussion between the two annotators resulted in the creation of two gold standards *StandardTimeML* and *LegalTimeML*:

1. **StandardTimeML** annotates all the TEs following the TimeML guidelines, and uses the *PRESENT_REF*, *PAST_REF* and *FUTURE_REF* tokens as usually done in the domain.

2. **LegalTimeML** annotates just the TEs relevant to the narratives of the judgment, following the particularities in the legal domain previously discussed (no dates in legal references, normalize to dates...). As per the *StandardTimeML* annotation set, it follows the guidelines but does not annotate all the expressions, being therefore a subset considering domain particularities.

[24] `https://catalog.ldc.upenn.edu/docs/LDC2006T08/timeml\_annguide_1.2.1.pdf`

[25] `https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-timex2-guidelines-v0.1.pdf`

The Inter-Annotator Agreement (IAA) between both gold standards is high (0.95), as well as Cohen's kappa (6) (0.94) and Scott's Pi (40) (0.94), indicating that the normalization of the TE's that are included in both annotation sets have a high agreement. If we check differences between annotations, we find there are an average of 13.1 common TEs per document, 0.3 partial coincidences and about 16.2 TEs that are not in the *LegalTimeML* but appear in the *StandardTimeML*. The recall among both annotation sets is of 0.44 while precision is of 0.90, which confirms that a lot of TEs are not relevant for the case timeline (44% with regard to the ones annotated following the full TimeML standard), but that the way to tag them by the annotators is highly similar.

4.3 Tagging

Once the corpus was collected, the following temporal taggers: HeidelTime (43), SUTime (4) GUTime (which is part of the TARSQI toolkit) (48), CAEVO (3), ClearTK-TimeML (1), SYNTime (50), TERNIP (32), TIPSem (22), USFD2 (9) and UWTime (21) were executed over our legal corpus, as they represent the different approaches available and are the most widely used in literature. These temporal taggers will be introduced in Section 5. HeidelTime was used in its configuration for narrative text. GUTime was used as a part of the TARSQI toolkit, using it alone with the preprocessor in the pipeline. Since the code available online was just able to annotate an specific corpus, USFD2 was slightly modified in order to annotate any input and to generate TIMEX3 tags as output[26]. All other taggers were used with default parametrization.

The output of the taggers which generated offline annotations (such as GUTIME/TARSQI) were modified in order to be comparable with the output of the rest of the taggers and ensure they were readable by GATE. These processes were executed using a new coded converter, which added the temporal annotations to the document and excluded non-temporal entities. Once the outputs of all the taggers were in the same format, they were loaded into the same GATE document, which contained twelve annotation sets (two for the manually-created gold standards and one for each of the ten temporal taggers).

4.4 Final Corpus

The final documents have been generated in several formats.[27] First, as GATE XML documents, that facilitate the storage of different annotation sets and also the visual and numerical comparison of the different sets. Second, a set of TimeML documents (TML) is provided for each of the manual gold standards. These documents contain the same annotations as in the correspondent annotation set in the corresponding GATE document, but makes the comparison with the output of other temporal taggers easier, as it is in the *official* TimeML format. Also a set of TML documents without any tag is provided to facilitate testing. These TML documents have been validated using the TimeML validator[28] from TempEval-3[29], so it is guaranteed that they fulfill the guidelines of the TimeML standard. Finally, all original documents are stored as TXT-files; these documents are of similar size in terms of kilobyte and length in tokens as shown in Table 3.

5 Temporal Taggers

Many of the temporal taggers described in the literature over the last few years are no longer available, not maintained, or just work for previous annotation schemas like the formerly mentioned TIMEX2. Some examples are DANTE (27), TEA (14), JU_CSE (19) or ManTIME (11). Therefore, we focus on the most widely used active temporal taggers which are often cited in literature and report good results on corpora from different domains, or have successfully participated in well-known temporal challenges, such as TempEval-3[30].

Table 5 provides an overview of the temporal taggers under investigation for which an implementation is freely available. The first column is used to refer to particular temporal taggers later on.

[26] The functionality and the rules were not modified.

[27] The final corpus can be downloaded at: `https://tempcourt.github.io/TempCourt/`

[28] `http://www.cs.york.ac.uk/semeval-2013/task1/data/uploads/timeml-validator-1.1a.tar.gz`

[29] `https://www.cs.york.ac.uk/semeval-2013/task1/`

[30] `https://www.cs.york.ac.uk/semeval-2013/task1/`

Table 5 Overview of temporal taggers. (*) Not all the types are covered.

Temporal Tagger	Approach	Identification	Normalization	Events	Relations
HeidelTime (HE)	rule-based	✓	✓	-	-
SUTime (SU)	rule-based	✓	✓	-	-
GUTime (GU)	hybrid	✓	✓	✓	✓
CAEVO (CA)	hybrid	✓	✓	✓	✓
ClearTK (CL)	machine-learning	✓	-	✓	✓
SynTime (SY)	rule-based	✓	-	-	-
TERNIP (TE)	rule-based	✓	✓	-	-
TIPSem (TI)	hybrid	✓	✓	✓	✓
USFD2 (US)	hybrid	*	*	-	*
UWTime (UW)	hybrid	✓	✓	-	-

The following aspects will be discussed for each tagger: supported languages, used approach, covered functionality, parametrization options, implementation language, availability, integration and interoperability with other software and dependencies on other resources and required installations.

5.1 Tasks of Temporal Taggers

The functionalities of temporal taggers can be classified into four categories as shown in Table 5. Some temporal taggers support all functionalities while other taggers require some additional tools.

- *Identification* means that the system is actually able to identify temporal expressions in a text compared to other systems which are only used for normalization of already tagged texts.
- *Normalization* refers to the ability to represent temporal information in the written text into the corresponding standard value following the ISO 8601 norm, which can be further processed. For instance expressions like ‘the next day’ refer to the day before which might be indicated with an explicit date in the text, and the temporal tagger is able to normalize this expression and assign the actual date as the value to the temporal annotation.
- *Events* are real-world situations at a particular time and are classified into seven categories, such as `OCCURENCE, STATE` or `REPORTING`, in the TimeML standard (38).
- *Relations* indicate a certain connection between events, times or a mixture of both usually classified into temporal `TLINK`, subordination `SLINK` and aspectual `ALINK` links (38).

5.2 Approaches

The detection of temporal expressions in a text is based on different approaches. Some taggers use rules for both identification and normalization tasks, while others use Machine Learning for the former. Also hybrid approaches have been proposed in literature. Nevertheless, it must be noted that normalization is generally tackled using rules, even when the identification is done otherwise.

5.2.1 Rule-based Approach

Temporal information is detected based on manually created rules (e.g. regular expressions), which need to cover all possible variations of how temporal information might be expressed. Thoroughly created rules are expected to perform better than other approaches, but come with the disadvantage of being inflexible. A missing or erroneous rule will prevent the temporal tagger from finding a temporal expression.

HeidelTime (43) is a rule-based domain-sensitive temporal tagger. Available for more than 200 languages (just 13 of them based on manually developed resources, the rest of them being automatically created), it offers the option to select from four different text categories: *News, Narratives, Colloquial* and *Scientific*, the last two are only available for English. HeidelTime covers both TE identification and normalization, having different strategies for each domain. HeidelTime, implemented in Java, can be used as a standalone version[31], or integrated in other pipeline environments like the General Architecture for Text Engineering (GATE) (7) or a UIMA[32] pipeline. In spite of being one of the most popular temporal tagging tools, to the best of our knowledge, it has never been used in the legal domain.

SUTime (4) is the Stanford CoreNLP (26) annotator for temporal expressions. SUTime is a rule-based temporal tagger built on the TokenRegex tool (5) (a pattern definition service also part of CoreNLP), able to both identify and normalize TEs. SUTime produces TimeML/TIMEX3 tags with new attributes not included in the standard, such an alternative value more flexible than the one covered by the standard. SUTime presents several related limitations (as analyzed by the authors themselves in (4)) and offers no domain adaptation. SUTime is available as part of the CoreNLP pipeline as a Named Entity Recognition (NER) system for different languages. Still, the tool works better in English than in other languages. The Java code[33] is available online, and also a GATE plugin and a Python wrapper have been developed[34].

SynTime (50) is a rule-based tagger that proposes a *type-based* approach as it defines different types of tokens (*time tokens, modifiers* and *numerals*) with similar syntactic behaviour and builds heuristic rules on these types instead of doing it on strings or regular expressions. As the types are domain independent and the rules work on types, the system is designed to be domain and language independent; nevertheless, to work in different domains or languages, more tokens need to be added for each type. SynTime only performs TEs recognition, and does not normalize them. For initialization, both tokens and regular expressions over them are collected for the independent temporal tagger SUTime (4). It is written in Java and available online[35]. It uses the Stanford CoreNLP library for Part of Speech (POS) disambiguation.

TERNIP (Temporal Expression Recognition and Normalisation in Python) (32) is a rule-based Python 2.7 library that identifies and normalizes TEs. The rules used for both subtasks can be easily extended. It only covers English and provides no domain particularities. It can be used as an API or be integrated as a GATE processing resource, via an XGAPP file (a GATE application file format) available with the code[36]. TERNIP relies on the Natural Language Toolkit library (NLTK) (24).

5.2.2 *Machine-learning-based Approach*

In contrast to rule-based approaches machine-learning based temporal taggers do not rely on previously created rules to identify temporal expressions. Using machine-learning techniques makes temporal taggers much more flexible and enables them to detect temporal expressions in an unexpected form, however it requires a good pre-trained model based on a large annotated corpus that supports a variety of temporal expressions which can be expected later in the document to be tagged with temporal expressions. A poor training set with missing variations of temporal expressions will result in a poor performance of the temporal tagger in terms of *precision*[37] and *recall*[38].

ClearTK-TimeML (1) is a system that identifies temporal information in English texts using external machine-learning tools. It uses specific annotators modelled as a BIO[39] token-chunking (for extent/identification of the expressions) or as a multiclass classification task (for types and attribute classification). The TIMEN normalisation tool (23) is suggested for the normalization task as this is not covered by ClearTK-TimeML. The features used are the ones proved to be the most successful in previous independent temporal taggers, and are extracted by a morpho-syntactic annotation pipeline with tools like OpenNLP and Apache. While ClearTK-TimeML does not offer domain-specific adaptions, the pipeline and the parameters can be customized by the user. It is written in Java and can be found online[40].

[31] https://github.com/HeidelTime/heideltime/

[32] https://uima.apache.org

[33] https://github.com/stanfordnlp/CoreNLP/tree/master/src/edu/stanford/nlp/time

[34] https://nlp.stanford.edu/software/sutime.shtml\#Extensions

[35] https://github.com/xszhong/syntime

[36] https://github.com/cnorthwood/ternip

[37] Fraction of the results identified which were correct.

[38] Fraction of the results that should have been found which were correctly identified.

[39] Beginning of, Inside of, Outside of a time expression.

[40] https://cleartk.github.io/cleartk/docs/module/cleartk_timeml.html

5.2.3 Hybrid Approach

Hybrid approaches combine rules with machine-learning. For instance creating rules of large corpus with machine-learning techniques to be manually refined afterwards.

GUTime (25) was developed at the Georgetown University originally for the temporal annotation of news. GUTime was subsequently incorporated into **TARSQI**, a modular system for automatic temporal annotation (48). The approach of GUTime is different from the temporal taggers previously mentioned, as it does not only use rules to find temporal expressions, but it also applies a hybrid approach of rules and machine-learning techniques. The hand-crafted rules serve in GUTime as a basis for temporal annotations that are extended by additional machine-learning ones discovered using the C4.5 algorithm (36), i.e. rules to support term disambiguation. The TARSQI framework is also able to extract events and relations from English texts. TARSQI is written in Python[41] and well described[42].

CAEVO (3) (CAscading EVent Ordering) is a sieve-based architecture, which uses twelve different classifiers (both rule-based and machine-learning), pipelined in a cascade way, starting with the one with the highest precision. Even when these classifiers work individually, some transitivity constraints are imposed; also the order of the classifiers can be modified, and new sieves can be added. In contrast to other taggers, CAEVO focuses on the extraction of temporal relations for event ordering, producing *dense* temporal graphs where events and temporal expressions are heavily connected. CAEVO is an expansion of NavyTime (2) and reuses part of the code of ClearTK-TimeML (1) for part of its sieves. It works just for English texts and has no domain adaptations. It is written in Java, and it is available online[43].

TIPSem (22) (Temporal Information Processing based on Semantic information) is an hybrid temporal tagger able to extract temporal information from English and Spanish. It uses both Semantic Role Labeling (13) and Conditional Random Field (CRF) (20) models. Different features are used by CRF recognition models, such as morphological or syntactic considerations at token level, along with semantic level ones such as the Role, the Governing Verb or Lexical Semantic information for each token. Similar features are used at tag level for classification. Finally, the relation extraction features differ depending on the type of relation. TIPSem tackles therefore all the temporal tasks. The Java code is available online[44], but it requires installation of additional software, and also optional libraries for certain languages (such as Spanish).

USFD2 (9) is a temporal tagger focusing on TEs and relations, using a rule-based approach for TEs and both rules and the NLTK's Maximum Entropy classifier for relations. USFD2 obtains a good recall with a smaller set of rules when compared with other taggers, since they consider specific heuristics for scpecific tags, such as `DATE`s and `DURATIONS`s as Temporal Expression types, that are the most common. It only works for English. The Python code of USFD2 is available online[45], but it must be noted that it is developed for the evaluation of specific datasets, so it must be slightly modified for custom use. This has been done for the results on our corpus described in this paper.

UWTime (21) follows a hybrid approach, using a Combinatory Categorial Grammar (CCG) (41) parser with hand-crafted rules and learning. UWTime just tackles the recognition and normalization of temporal expressions. It uses features such as surrounding tokens and POS, lexical and dependency information, and relies on techniques such as AdaBoost (12) for optimization. UWTime is only available in English with no domain particularities. It can be downloaded online[46], used as an API or as a server. UWTime relies on Stanford CoreNLP software.

6 Evaluation and Results

The final step of our research methodology involved a comparison of the effectiveness of all ten taggers on the two gold standards, along with the analysis of the results.

[41] https://github.com/tarsqi/ttk
[42] http://timeml.org/tarsqi/index.html
[43] https://github.com/nchambers/caevo
[44] https://github.com/hllorens/otip
[45] https://github.com/leondz/usfd2, https://code.google.com/archive/p/usfd2/
[46] https://bitbucket.org/kentonl/uwtime-standalone

6.1 Evaluation Methodology

After having all documents annotated with the ten different temporal taggers we evaluated the results, for which we used the typical *precision*, *recall* and *F-measure* metrics, which are commonly used in literature for the evaluation of extraction and normalization of temporal annotations (43). Precision is defined as the share of correctly identified items in percent compared to all identified items; whereas recall is defined as the number of items correctly found compared to the number that should have been found. The third measure we included in the evaluation, the *F-measure*, describes a weighted average between precision and recall (37). It is worth noting that we elected to provide both the *strict*-F-measure (which only considers completely correct and ignores partially correct annotations) and the *lenient*-F-measure, that admits partial annotations. The reason to do so is that while it is important to identify the complete temporal expression, it is also true that some taggers normalize correctly an expression even if they do not fully cover it. It also must be taken into account that in some cases the correct extent of a temporal expression is not clearly derivable from the guidelines, for this reason we decided that providing both measures would allow for the evaluation of both the degree of support with respect to the guidelines and the actual detection capabilities.

The evaluation process was designed in a way to avoid a bias or preference towards a particular temporal tagger. Therefore, the results of all taggers are consolidated in a single document with individual annotation sets for each tagger containing the temporal annotations and respective features. Each evaluation involves a key set (the correct reference) and a response set (the annotations to evaluate). Since the goal is to create gold standards for the legal domain, the manually annotated temporal expressions in both annotation sets, *LegalTimeML* (LTML) and *StandardTimeML* (STML), serve as the key sets. The annotation sets of each tagger act as the response set for each evaluation run. We therefore evaluated each automatic tagger for all three sections of the corpus (i.e. the documents from the three different legal sources) against each of the manually created gold standards *LegalTimeML* and *StandardTimeML* and calculated the *lenient* and *strict* precision, recall and F-measure.

All the temporal taggers were applied to the corpus with the standard configuration and there were no domain-specific modifications to achieve better results specifically for the legal domain[47]. The standard configuration was chosen so as to evaluate the out-of-the-box performance of each annotator and the suitability when applied to the legal domain. The average number of annotations per corpus in both Gold Standards (STML and LTML) and the various taggers are shown in Table 6, which illustrates the occurrences of different TIMEX3 annotation types (`DATE`, `DURATION`, `TIME`, `SET`) for each analysed corpus. It is clearly shown that the most used annotation type in court decisions is DATE. This result is not surprising as the date is considered to be sufficient in most cases as the actual time of the day is not relevant. Furthermore, deadlines in the legal domain usually indicate the end of the day and it is not important if an action is taken in the morning or in the afternoon. It must also be noted that the pattern of appearances of each of the TIMEX3 types does not fit any of those of the domains analyzed by Strötgen et al. (43) (news, narratives, colloquial and scientific).

Table 7 clearly shows that most taggers perform well on the short ECHR subcorpus and tend to find the same number of annotations as in the gold standard, especially if we focus on the *lenient* figures, showing that the errors are mostly in the extension of the tagging more than in its identification. In the ECJ and USC subcorpora (Tables 8 and 9 respectively) the number of annotations by the taggers differs from the gold standards, especially HeidelTime draws attention to its annotations in the ECJ corpus. When looking into the documents, the reason for this significant difference becomes obvious. The designators of European legal acts such as regulations and directives follow a special scheme which also includes the year when the legal act has been agreed. A typical designator of an EU directive is therefore, for instance ***2016/679***, which is considered to be a designator of a legal act but it is not a valuable temporal reference within a court decision.

[47] Except USFD2.

Table 6 Average number of annotation types per document for each corpus (***D**ate,**Dur**ation,**S**et,**T**ime*).

	ECHR				ECJ				USC			
Tagger	**D**	**Dur**	**S**	**T**	**D**	**Dur**	**S**	**T**	**D**	**Dur**	**S**	**T**
StandardTimeML	11.6	1.3	1	0	31.5	4.3	2	2.7	35.7	5.6	3.5	4
LegalTimeML	10.1	1.3	1	0	16.8	4.3	1.5	3	9.1	5.4	1.5	0
HeidelTime	11.4	1.7	1	0	68.1	5.3	1	1	41.6	5.6	1.5	2
SUTime	11.3	2	0	0	39.1	3.9	1.3	1.3	46.9	7.9	1.5	2.7
GUTime	11.7	0	0	0	31.4	1	0	0	37.3	2	0	0
CAEVO	11.1	1.8	0	0	36.7	5.8	1	1.5	39.9	9.4	1.5	3
ClearTK	10.2	1	0	0	38.6	3.4	0	0	36.1	5.1	1	2
Syntime	11.5	0	0	0	39.1	0	0	0	47.8	0	0	0
TERNIP	11.7	1.7	0	0	30.3	3.6	0	0	33.3	5.6	1	0
TIPSem	13	1	0	0	38.4	2.6	0	0	-	-	-	-
USFD2	13.9	2	0	0	66.6	3.3	0	0	28.4	3.8	0	0
UWTime	11	2.5	0	0	-	-	-	-	-	-	-	-

Table 7 Evaluation results for the ECHR corpus for each temporal tagger, both for identification (two first columns, *lenient* and *strict*) and normalization (two last columns, *lenient* and *strict*). The first row (in white) corresponds to results against the *StandardTimeML* gold standard, while the second (in gray) corresponds to the *LegalTimeML* gold standard.

	lenient			**strict**			**lenient+ value**			**strict+ value**		
A	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
HE	**0.99**	**0.99**	**0.99**	0.84	0.84	0.84	0.78	0.78	0.78	0.78	0.78	0.78
	0.88	**0.99**	**0.93**	0.71	0.80	0.75	0.67	0.75	0.71	0.64	0.72	0.68
SU	0.88	0.87	0.88	0.85	0.84	0.84	0.78	0.78	0.78	0.76	0.75	0.75
	0.76	0.85	0.80	0.71	0.80	0.76	0.66	0.74	0.79	0.64	0.72	0.68
GU	0.96	0.93	0.94	**0.95**	0.92	**0.93**	**0.86**	0.84	0.85	**0.86**	0.84	**0.85**
	0.84	0.92	0.88	**0.83**	**0.92**	**0.87**	0.74	0.82	0.78	**0.74**	0.82	**0.78**
CA	0.88	0.87	0.87	0.83	0.82	0.82	0.78	0.78	0.78	0.75	0.75	0.75
	0.75	0.85	0.80	0.70	0.79	0.74	0.65	0.74	0.69	0.64	0.72	0.67
CL	0.92	0.78	0.85	0.34	0.32	0.35	-	-	-	-	-	-
	0.80	0.77	0.78	0.33	0.32	0.33	-	-	-	-	-	-
SY	0.98	0.93	0.96	0.83	0.79	0.81	0	0	0	0	0	0
	0.86	0.93	0.90	0.70	0.76	0.73	0	0	0	0	0	0
TE	0.94	0.95	0.95	0.92	**0.93**	0.92	**0.86**	**0.88**	**0.87**	0.85	**0.86**	**0.85**
	0.83	0.95	0.89	0.80	**0.92**	0.85	**0.75**	**0.86**	**0.80**	0.72	**0.83**	0.77
TI	0.78	0.85	0.81	0.64	0.70	0.67	0.64	0.71	0.67	0.63	0.69	0.66
	0.69	0.86	0.76	0.62	0.77	0.69	0.64	0.79	0.71	0.61	0.76	0.68
US	0.73	0.61	0.67	0.69	0.58	0.63	0	0	0	0	0	0
	0.65	0.62	0.64	0.61	0.58	0.60	0	0	0	0	0	0
UW	0.90	0.53	0.67	0.51	0.30	0.38	0.55	0.33	0.41	0.51	0.30	0.38
	0.86	0.58	0.69	0.48	0.32	0.38	0.51	0.34	0.41	0.48	0.32	0.38

6.2 Results

From the results shown in Tables 7 (ECHR), 8 (ECJ) and 9 (USC), we can see that the performance of the individual temporal taggers is quite similar for each section of the corpus. Furthermore, the numbers for all three measures that have been calculated are unexpectedly high for some taggers in comparison to the application of temporal taggers (out of the box without any domain-specific modifications) in the case of non legal text. They tend to be nevertheless less performant than results previously reported by taggers in general evaluations[48] (4).

[48] `https://github.com/HeidelTime/heideltime/wiki/Evaluation-Results`

Table 8 Evaluation results for the ECJ corpus for each temporal tagger, both for identification (two first columns, *lenient* and *strict*) and normalization (two last columns, *lenient* and *strict*). The first row (in white) corresponds to results against the *StandardTimeML* gold standard, while the second (in gray) corresponds to the *LegalTimeML* gold standard.

	lenient			**strict**			**lenient+ value**			**strict+ value**		
A	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
HE	0.48	0.95	0.64	0.47	0.94	0.63	0.47	**0.94**	0.62	0.47	**0.93**	0.62
	0.27	0.97	0.42	0.26	**0.96**	0.41	0.26	**0.94**	0.40	0.26	**0.93**	0.40
SU	0.81	0.97	0.88	0.79	**0.95**	0.86	0.78	0.93	0.85	0.77	0.92	0.84
	0.44	0.95	0.60	0.43	0.93	0.58	0.41	0.90	0.57	0.41	0.89	0.56
GU	**0.97**	0.87	0.91	**0.97**	0.86	**0.91**	0.94	0.84	0.89	0.94	0.84	0.88
	0.51	0.82	0.63	0.50	0.82	0.62	0.48	0.78	0.60	0.48	0.78	0.60
CA	0.89	0.74	0.81	0.85	0.70	0.77	0.86	0.71	0.77	0.85	0.70	0.77
	0.49	0.74	0.59	0.46	0.70	0.56	0.46	0.70	0.56	0.46	0.69	0.55
CL	0.77	0.88	0.82	0.32	0.36	0.34	-	-	-	-	-	-
	0.42	0.88	0.57	0.18	0.37	0.24	-	-	-	-	-	-
SY	0.89	**0.99**	**0.93**	0.81	0.90	0.85	0	0	0	0	0	0
	0.49	**0.98**	0.65	0.46	0.92	0.61	0	0	0	0	0	0
TE	**0.97**	0.88	0.92	0.96	0.88	**0.91**	**0.96**	0.87	**0.91**	**0.95**	0.87	**0.91**
	0.54	0.89	**0.67**	**0.53**	0.88	**0.66**	**0.53**	0.88	**0.65**	**0.52**	0.87	**0.65**
TI	0.72	0.81	0.76	0.64	0.72	0.68	0.62	0.70	0.65	0.61	0.69	0.65
	0.41	0.83	0.54	0.37	0.75	0.49	0.35	0.71	0.47	0.34	0.70	0.46
US	0.31	0.54	0.39	0.29	0.51	0.37	0.02	0.04	0.03	0.02	0.03	0.02
	0.20	0.65	0.31	0.19	0.61	0.29	0.02	0.06	0.03	0.02	0.05	0.02
UW	-	-	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-	-	-

On the ECHR corpus most taggers perform equally well when *strictly* evaluated, while GUTime provides the best results, closely followed by TERNIP. On the contrary, TIPSem, USFD2 and UWTime are not as performant. This is because the ECHR uses fully qualified dates (e.g. 10 January 2017) and does not include many references to other court decisions. These results fall when we look at the normalization values. It also must be noted that most taggers (except of GUTime, SynTime and TERNIP) struggle with identifying dates denoting the birthdates of the persons involved in the cases and case numbers, with some also normalizing them. It must be noted how big differences between *lenient* and *strict* values such as those of UWTime or ClearTK-TimeML do not always affect in terms of differing in the extent of the tag, but it also impacts in the normalization values. For instance, if instead of marking up ‘October 13’, just ‘October’ is marked, the *lenient* score will count it as positive, the *strict* will not, but the normalization will for sure be wrong.

In the ECJ corpus one outlier in the figures can be spotted immediately, which is the precision of the HeidelTime annotations that is significantly different from its other precision values across each section of the corpus. The much better performance of GUTime in the ECJ corpus can be explained by the fact that it does not annotate numbers referring to collections of judgments (such as TIPSem or ClearTK-TimeML do).

The USC corpus is slightly different to ECHR and ECJ as it uses a different date format and it also repeats part of the text in the judgment, which leads to poorer performance as incorrect annotations are also repeated.

Different date formats are a typical challenge which occur when applying temporal taggers to a corpus. Typically dates found across all evaluated documents are fully qualified dates containing a day, the month in full and a year. The format in which these dates are provided are different for European and American sources of legal documents. The date in Europe is usually indicated in the format Day, Month, Year (e.g. 10 January 2017), whereas the American date format is “Month DD, YYYY” (e.g. January 10, 2017). This particular difference in the date format has been processed correctly by some taggers, such as HeidelTime

Table 9 Evaluation results for the USC corpus for each temporal tagger, both for identification (two first columns, *lenient* and *strict*) and normalization (two last columns, *lenient* and *strict*). The first row (in white) corresponds to results against the *StandardTimeML* gold standard, while the second (in gray) corresponds to the *LegalTimeML* gold standard.

	lenient			**strict**			**lenient+ value**			**strict+ value**		
A	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
HE	0.83	0.94	0.88	0.81	0.92	**0.86**	0.79	**0.90**	**0.84**	0.79	**0.89**	**0.83**
	0.29	0.97	0.44	0.26	0.88	0.40	0.20	0.67	0.31	0.19	0.64	0.29
SU	0.75	**0.99**	0.85	0.72	**0.95**	0.82	0.67	0.88	0.76	0.66	0.86	0.75
	0.25	**0.98**	0.40	0.23	**0.90**	0.36	0.18	0.72	0.29	0.17	**0.69**	0.28
GU	0.84	0.78	0.81	0.71	0.66	0.69	0.67	0.62	0.65	0.65	0.60	0.62
	0.25	0.69	0.36	0.16	0.45	0.23	0.12	0.34	0.18	0.10	0.27	0.14
CA	0.77	0.90	0.82	0.72	0.84	0.77	0.73	0.85	0.78	0.71	0.83	0.76
	0.23	0.82	0.36	0.21	0.72	0.32	0.21	**0.73**	0.33	0.20	**0.69**	0.30
CL	0.85	0.84	0.84	0.81	0.79	0.80	-	-	-	-	-	-
	0.30	0.89	0.45	0.26	0.78	0.39	-	-	-	-	-	-
SY	0.85	0.98	**0.91**	0.78	0.91	0.84	0	0	0	0	0	0
	0.28	**0.98**	0.44	0.24	0.84	0.37	0	0	0	0	0	0
TE	**0.93**	0.86	0.90	**0.90**	0.83	**0.86**	**0.86**	0.79	0.83	**0.85**	0.78	0.81
	0.32	0.90	**0.48**	**0.29**	0.81	**0.43**	**0.25**	0.69	**0.37**	**0.23**	0.64	**0.34**
TI	-	-	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-	-	-
US	0.50	0.30	0.37	0.39	0.23	0.29	0.08	0.02	0.08	0.02	0.01	0.02
	0.16	0.28	0.21	0.07	0.13	0.09	0.08	0.14	0.10	0.03	0.05	0.04
UW	-	-	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-	-	-

and SUTime, annotating both versions as a single date. GUTime however was not reliable in this context, despite the fact that it is the best tagger in the other corpora. It either detected only one part of the American-formatted date (e.g. January 10) or it treated both parts of the same date as two different annotations.

The performance of GUTime in terms of precision, recall and F-measure is pretty good over all three subcorpora. However, GUTime performs poorly on the USC corpus. Inspecting the GUTime annotations in this corpus confirms the fact that GUTime has a hard time recognizing dates in the American format, as already pointed out above, an issue that is also reflected in normalization figures (where TERNIP maintains the performance from the other subcorpora).

In summary, although the results of the evaluation are promising it is worth noting that legal documents, especially court decisions, have some particularities (such as those highlighted in Section 3) which cause some stumbling blocks for automatic temporal taggers being applied out-of-the-box. An example of this would be the case of *'dec.'*, a non-temporal expression that appears when citing *decisions on admissibility*[49] that most taggers (such as CAEVO or SUTime) normalize as *December*.

With regard to the comparison between the two reference standards, if we check the differences between figures and focus on the recall (since the taggers are not trained for the particularities of this annotation set, the precision is obviously not expected to be high and does not indicate the tagger's usefulness), we see that the best taggers remains more or less the same (GUTime, TERNIP, SUTime and HeidelTime, since although SynTime performs well in terms of recognition it does not provide a value).

6.3 Comparative Analysis of Several Temporal Taggers

The thorough analysis of the corpus documents and the manual inspection of the most frequent errors of the taggers led to the synthesis of a collection of test cases that present the phrases prone to cause errors.

[49] `http://www.echr.coe.int/Documents/Note_citation_ENG.pdf`

The most salient results are described below, where the output of the tagger is represented in bold and the correct tag is underlined.

HeidelTime is able to identify temporal modifiers (e.g. *at least five years*) automatically and add the feature to the annotation. However, it fails to detect the correct date format (e.g. DD/MM/YYYY vs MM/DD/YYYY) as well as failing to recognize the indication of the age of mentioned persons (e.g. "*a* ***62-year-old*** *woman*"). It does not normalize expressions like *today* and annotates them with the value `PRESENT_REF`. In legal texts, it tags as TEs references to other documents or IDs (e.g. *"No **1612/68**"*, *"No. 15-**1031**"*, *"See Pet. for Cert. **5-7**"*). It also has an interval option that does not work well in this kind of document.

SUTime also fails to identify the correct date representation form (e.g. DD/MM/YYYY vs MM/DD/YYYY). In addition, SUTime exhibits inconsistencies when parsing the same expression in different paragraphs, and it also wrongly annotates expressions like *"**fall**"*, *"**may**"* as temporal expressions although they refer to an action *"to fall"*, *"may"* instead of the season. SUTime also has some limitations with respect to ambiguity resolution or non-whole numbers recognition.

Although GUTime has a good performance in general, sometimes it does not normalize some expressions and has problems with some ways to represent hours (e.g., it does not recognize "*(...) between 12.15 and 18.45*", nor if it was expressed as "*12:15 and 18:45*", it just recognizes "***12h15*** *and* ***18h45***"). Also some DURATIONs are not recognized, series or dates neither (in "*15 and* ***16 December 2008***" it just recognizes the part in bold) , and sometimes it tags expressions that look like years, such as "*EUR* ***2000***".

CAEVO does normalize DATEs in the format DD/MM/YYYY as MM/DD/YYYY, so it does not even recognize the ones not fitting it, such as "*25/03/2016*". It also partially annotates expressions such as "*On the next* ***day***" (categorizing it as a DURATION) and tags separately "***once a week***", as a `PAST_REF DATE` and a `DURATION`, respectively. It also does not recognize 15 in "*15 and* ***16 December 2008***", and tags "***62-year-old*** *woman*", year-like expressions as *"**§1101**"* and time-like expressions as "*Order in No.* ***2:10****-cv-02698 (WD Tenn.)*". Finally, it also tags separately "*sentenced to* ***a year*** *and* ***a day*** *in prison*".

Similarly to GUTime, ClearTK-TimeML does not recognize TIMEs when expressed as in "(...) between 12.15 and 18.45"; it does not either recognize expressions like "09/01/1981" as DATEs. Some DURATIONs are also not recognized (e.g. "at least five years"), and tags expressions such as "**May**" or "**62-year-old** woman". It just annotates partially expressions such as "23 **January 2013**" or "once **a week**" (that is categorized as a DURATION).

SynTime just normalizes to the date when it is executed. Although it is able to recognize expressions such as "***15*** *and* ***16 December 2008***", it fails when it finds expressions such as "*as amended by Council Regulation (EC) No* ***1791/2006 of 20 November 2006***", where it annotates all in bold, not just the underlined correct part. It also seems to recognize all four-digit expressions as years (e.g.. "*See 10 U. S. C. §**1408**(c)(1).*", "*So. 3d* ***1264, 1269-1272***") and ambiguous expressions as "***may***", "***the second***" or "***fall***", but fails to fully annotate some temporal expressions (e.g. "*per* ***month***", "*May 15,* ***2017***").

TERNIP tags expressions such as "*EUR* ***2000***", "***may***", "***fall***", but fails to identify some DATEs and DURATIONs. It also does not identify 13 in "*13 and 27 October 2008*", but is on the other hand is able to recognize misspelled temporal expressions such as "***eighth months***" (even if it is not correctly normalized). It also tags "*303, 98 Stat.* ***2045****, 21 U. S. C. §853(a)(1),*" as DATEs expressions .

TIPSem is not able to annotate some of the documents in the corpus (namely the ones from the USC subset), and does not recognize the first DATE in the ECJ subset, expressed as in the format `DD Month YYYY`; since it recognizes in the rest of the document without a problem, it is probably due to a lack of a syntactic/semantic context for it. It tags expressions such as "*Directives* ***2004/83, 2005/85*** *and 2003/9]*" or "***Article 5 of Directive 2008/115***", "**Directive 2001/42**" or "***the judgment of 28 February 2012***". It also tags expressions such as "***MON 810***" or random numbers or words as "***4,285***", "*(in* ***euros)i***", that tends to mark as `FUTURE_REF`. On the other hand, it does not recognize some dates, as "*29/02/2016*", but it does so with a similar one like "***28/09/2016***".

USFD2 is unable to parse some of the documents in the corpus, throwing errors when trying to normalize expressions it considers out of the range and warnings for some ASCII codes. It also tags some numbers randomly, such as in "*amending Regulation (EEC) No* ***1612/68*** *and repealing Directives*

*64/**221**/EEC, 68/**360**/EEC*" and always normalizes DATEs to the present day. It does not recognize straightforward dates and tags ambiguous words even when they are a part of another word, such as in "*Soto**may**or*"; TIME expressions are categorized as DATEs.

Finally, UWTime is not able to parse long legal sentences, throwing several errors because of the lack of head rules defined for some of the expressions it finds. In our corpus, it was not able to annotate even a third of the documents.

The most commonly occurring errors in which the taggers fall, whether because they happen frequently in the text or because several taggers incur in them, are the following:

- Separation of whole `SET` expressions as "*Once a week*" into "*Once*" and "*a week*", converting one `SET` into a `PAST_REF` `DATE` and a `DURATION`.
- Not recognizing series of `DATE`s such as "*15 and 16 December*", but detecting the last `DATE` of such a series only.
- Separation of `DURATION`s such as "*One year and one day*" into two different `DURATION`s.
- In some documents (as also happens in other kinds of legal texts, such as in the previously mentioned transactional ones), some information is put into brackets, such as in "*before the expiry of a period of* ***[48] hours***"; usually generic temporal taggers are not able to detect them (for instance tagging in this concrete example just "*hours*").
- Tagging general ambiguous expressions such as "*fall*" or "*may*" or specific ambiguous ones such as the previously described case of "*dec.*".
- Tagging year-like expressions such as "*No **1612**/68*" or "*§**1408***"; most taggers tag every four-digit number as a year.
- Problems with dates expressed in the format "DD/MM/YYYY", frequently in identification but in some cases also in normalization.
- Identification of a currency as a year ("*EUR **2000***").
- Tagging of expressions such as "*62-year-old*".
- Most taggers do not take modifiers (`mod`) into account, probably because of the low ratio of appearance of `SET`s in other domains, despite the fact that they are extremely important in legal documents. Namely, HeidelTime correctly tagged[50] 17 out of 28 modifiers, while TERNIP tagged 10 out of 28. The remaining taggers tagged no modifiers (Fexcept of UWTime, in one of the few documents it tagged, but not correctly).
- The case of the `quant` and `freq` attributes is similar for `SET`s. While HeidelTime marks correctly 2 out of 11 `quant`, and marks incorrectly two `freq` as `1` (when it should be `1X`), TERNIP just marks one `quant` (and incorrectly, since it must be in capital letters) out of 11 and no `freq`.

7 Conclusion

In this paper we pointed out the importance of temporal information contained in legal documents. An extensive state of the art analysis showed that the extraction of temporal information has been investigated for other domains but not for the legal domain.

Considering the specific requirements of temporal annotation in the legal domain, we identified a lack of corpora that can be used for the evaluation of temporal entity extractors. In order to fill this gap, we created a corpus of 30 documents from the European Court of Human Rights, the European Court of Justice and the United States Supreme Court, containing manually annotated temporal expressions. The

[50] Some cases, such as distinctions between *EQUAL_OR_LESS* / *LESS_THAN* (for UWTime) and *LATE* / *END* and *EARLY* / *START* (for TERNIP) were counted as errors.

corpus is presented in two forms: (i) a generic gold standard called *StandardTimeML*; and (ii) a domain-focused gold standard called *LegalTimeML*. The latter was tailored specifically for temporal dimensions that are important for the entailed legal case, namely the *temporal dimension of the legal process* and the *temporal dimension of the case*.

We also preformed an in-depth analysis of several state-of-the-art temporal taggers and performed a comparative evaluation against our corpus. The results of our analysis on the *StandardTimeML* gold standard shows that the best temporal taggers are quite effective when it comes to finding all possible temporal expressions in legal text, however they fail when they encounter misleading references to legal documents. This can generally be attributed to the fact that courts tend to use a clear structured language and absolute date formats. It is not surprising that the performance of the cross domain temporal taggers on the *LegalTimeML* gold standard is much less impressive, highlighting the need for tools and guidelines that are specifically tailored to particularities of the legal domain.

The work presented herein is a prerequisite for future work which focuses on the automatic extraction of timelines from legal text. In this context, it will also be necessary to evaluate existing event extraction techniques, with respect to the particularities of the legal domain. The combination of temporal information and legal events could result in the creation of a temporal events taxonomy, that would help in a better understanding of legal processes. Additionally, based on our analysis and experience working both with temporal expressions and events, we aim to develop a set of guidelines, which will be of benefit for the legal informatics community. Besides the extension of this work towards event extraction and timeline creation, the legal domain is also very language dependent. Documents published in various countries and jurisdictions are typically written in the national language. Therefore, an interesting avenue for future research is to analyze the performance of existing temporal taggers over legal corpora that are written in languages other than English.

Acknowledgement

This work was partially funded by a Predoctoral grant from the Programa Propio de la Universidad Politécnica de Madrid, and from a grant from Consejo Social de la Universidad Politécnica de Madrid. This work was supported by the Republic of Austria's Federal Ministry for Digital and Economic Affairs and the Jubiläumsfonds der Stadt Wien

References

[1] S. Bethard. ClearTK-TimeML: A minimalist approach to TempEval 2013. In *Proceedings of the Workshop SemEval 2013*, pages 10–14. ACL, June 2013.

[2] N. Chambers. Navytime: Event and time ordering from raw text. In *Proceedings of the Workshop SemEval 2013*, volume 2, pages 73–77, 2013.

[3] N. Chambers et al. Dense event ordering with a multi-pass architecture. *Transactions of the ACL*, 2: 273–284, 2014.

[4] A. X. Chang et al. Sutime: A library for recognizing and normalizing time expressions. In *Proceedings of LREC 2012*, 2012.

[5] A. X. Chang et al. TokensRegex: Defining cascaded regular expressions over tokens. Technical Report CSTR 2014-02, Department of Computer Science, Stanford University, 2014.

[6] J. Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

[7] H. Cunningham et al. Getting More Out of Biomedical Documents with GATE's Full Lifecycle Open Source Text Analytics. *PLOS Computational Biology*, 9(2):1–16, 02 2013.

[8] Dell'Orletta et al. The splet–2012 shared task on dependency parsing of legal texts. In *Semantic Processing of Legal Texts (SPLeT-2012) Workshop Programme*, page 42, 2012.

[9] L. Derczynski et al. Usfd2: Annotating temporal expresions and tlinks for tempeval-2. In *Proceedings of the Workshop SemEval*, pages 337–340. ACL, 2010.

[10] L. Ferro, L. Gerber, I. Mani, B. Sundheim, and G. Wilson. Tides 2005 standard for the annotation of temporal expressions. Technical report, Technical report, MITRE, 2005.

[11] M. Filannino and G. Nenadic. Temporal expression extraction with extensive feature type selection and a posteriori label adjustment. *Data & Knowledge Engineering*, 100:19 – 33, 2015. ISSN 0169-023X. doi: https://doi.org/10.1016/j.datak.2015.09.002. URL `http://www.sciencedirect.com/science/article/pii/S0169023X15000725`.

[12] Y. Freund, R. Schapire, and N. Abe. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612, 1999.

[13] D. Gildea and D. Jurafsky. Automatic Labeling of Semantic Roles. *Computational Linguistics*, 28(3):245–288, 2002. doi: 10.1162/089120102760275983. URL `https://doi.org/10.1162/089120102760275983`.

[14] B. Han, D. Gates, and L. Levin. From language to time: A temporal expression anchorer. In *Temporal Representation and Reasoning, 2006. TIME 2006. Thirteenth International Symposium on*, pages 196–203. IEEE, 2006.

[15] S. Hellmann et al. NIF: An ontology-based and linked-data-aware NLP Interchange Format. 2012.

[16] D. Isemann et al. *Temporal Dependence in Legal Documents*, pages 497–504. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-41278-3.

[17] H. Ji, T. Cassidy, Q. Li, and S. Tamang. Tackling representation, annotation and classification challenges for temporal knowledge base population. *Knowledge and Information Systems*, 41(3): 611–646, Dec 2014. ISSN 0219-3116. doi: 10.1007/s10115-013-0675-1. URL `https://doi.org/10.1007/s10115-013-0675-1`.

[18] T. Kajiwara et al. Building a monolingual parallel corpus for text simplification using sentence similarity based on alignment between word embeddings. In *Proceedings of COLING 2016: Technical Papers*, pages 1147–1158, 2016.

[19] A. K. Kolya, A. Kundu, R. Gupta, A. Ekbal, and S. Bandyopadhyay. Ju_cse: A crf based approach to annotation of temporal expression, event and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, volume 2, pages 64–72, 2013.

[20] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In C. E. Brodley and A. P. Danyluk, editors, *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001*, pages 282–289. Morgan Kaufmann, 2001. ISBN 1-55860-778-1.

[21] K. Lee et al. Context-dependent semantic parsing for time expressions. In *Proceedings of the 52nd Annual Meeting of the ACL*, volume 1, pages 1437–1447, 2014.

[22] H. Llorens et al. Tipsem (english and spanish): Evaluating crfs and semantic roles in tempeval-2. In *Proceedings of the Workshop SemEval*, pages 284–291. ACL, 2010.

[23] H. Llorens et al. Timen: An open temporal expression normalisation resource. In *Proceedings of LREC 2012*, pages 3044–3051, 2012.

[24] E. Loper and S. Bird. NLTK: The Natural Language Toolkit. *CoRR*, cs.CL/0205028, 2002. URL `http://arxiv.org/abs/cs.CL/0205028`.

[25] I. Mani et al. Robust temporal processing of news. In *Proceedings of the 38th annual meeting on ACL*, pages 69–76. ACL, 2000.

[26] C. D. Manning et al. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of the 52nd Annual Meeting of the ACL 2014, System Demonstrations*, pages 55–60, 2014.

[27] P. Mazur and R. Dale. The dante temporal expression tagger. In Z. Vetulani and H. Uszkoreit, editors, *Human Language Technology. Challenges of the Information Society*, pages 245–257, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. ISBN 978-3-642-04235-5.

[28] P. Mazur et al. Wikiwars: A new corpus for research on temporal expressions. In *Proceedings of EMNLP 2010*, pages 913–922. ACL, 2010.

[29] A.-L. Minard, M. Speranza, et al. Meantime, the newsreader multilingual event and time corpus. In *Proceedings of LREC 2016, European Language Resources Association*, 2016.

[30] V. Naik, G. Vanitha, and S. Inturi. Reasoning in legal text documents with extracted event information. *International Journal of Computer Applications*, 28:8—13, 08 2011.

[31] J. Nivre et al. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of EMNLP-CoNLL 2007*, 2007.

[32] C. Northwood. TERNIP: temporal expression recognition and normalisation in Python. Master's thesis, University of Sheffield, 2010.

[33] D. Pellow et al. An open corpus of everyday documents for simplification tasks. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)*, pages 84–93, 2014.

[34] J. Pustejovsky et al. The Timebank corpus. In *Corpus linguistics*, volume 2003, page 40. Lancaster, UK, 2003.

[35] J. Pustejovsky et al. TimeML: Robust Specification of Event and Temporal Expressions in Text. In M. T. Maybury, editor, *New Directions in Question Answering, Papers from 2003 AAAI Spring Symposium*, pages 28–34. AAAI Press, 2003. ISBN 1-57735-184-3.

[36] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993. ISBN 1-55860-238-0.

[37] C. J. V. Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA, 2nd edition, 1979. ISBN 0408709294.

[38] R. Saurí, J. Littman, B. Knippen, R. Gaizauskas, A. Setzer, and J. Pustejovsky. TimeML annotation guidelines. *Version*, 1(1):31, 2006.

[39] F. Schilder. Event Extraction and Temporal Reasoning in Legal Documents. In *Annotating, Extracting and Reasoning about Time and Events, International Seminar, Dagstuhl Castle. Revised Papers*, pages 59–71, 2005.

[40] W. A. Scott. Reliability of content analysis: The case of nominal scale coding. *Public opinion quarterly*, pages 321–325, 1955.

[41] M. Steedman et al. *Combinatory Categorial Grammar*, chapter 5, pages 181–224. Wiley-Blackwell, 2011.

[42] J. Strötgen and M. Gertz. Domain-sensitive temporal tagging. *Synthesis Lectures on Human Language Technologies*, 9(3):1–151, 2016.

[43] J. Strötgen et al. Temporal tagging on different domains: Challenges, strategies, and gold standards. In *Proceedings of LREC 2012*, volume 12, pages 3746–3753, 2012.

[44] W. Styler IV et al. Temporal annotation in the clinical domain. *Transactions of the Association of Computational Linguistics*, 2(1):143–154, 2014.

[45] J. Tabassum et al. Tweetime: A minimally supervised method for recognizing and normalizing time expressions in twitter. *arXiv preprint arXiv:1608.02904*, 2016.

[46] N. Uzzaman et al. Event and Temporal Expression Extraction from Raw Text: First Step Towards a Temporally Aware System. *International Journal of Semantic Computing*, 04(04):487–508, 2010.

[47] N. UzZaman et al. SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations. In *Proceedings of the Workshop SemEval 2013*, pages 1–9, 2013.

[48] M. Verhagen et al. Automating Temporal Annotation with TARSQI. In *Proceedings of the ACL 2005 on Interactive Poster and Demonstration Sessions*, ACLdemo '05, pages 81–84. ACL, 2005.

[49] C. S. Vlek et al. Representing and evaluating legal narratives with subscenarios in a bayesian network. In *OASIcs-OpenAccess Series in Informatics*, volume 32. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2013.

[50] X. Zhong et al. Time expression analysis and recognition using syntactic token types and general heuristic rules. In *Proceedings of the 55th Annual Meeting of the ACL*, volume 1, pages 420–429, 2017.

9. Events matter: Extraction of events from court decisions

Bibliographic Information

Filtz, E., Navas-Loro, M., Santos, C., Polleres, A. and **Kirrane, S.**, 2020. Events matter: Extraction of events from court decisions. Legal Knowledge and Information Systems, pp.33-42.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Writing - Review & Editing, and Supervision.

Copyright Notice

Events Matter: Extraction of Events from Court Decisions

Erwin FILTZ [a,b], María NAVAS-LORO [c], Cristiana SANTOS [d], Axel POLLERES [a] and Sabrina KIRRANE [a]

[a] *Vienna University of Economics and Business*
[b] *Siemens AG Österreich*
[c] *Universidad Politécnica de Madrid – Ontology Engineering Group, Madrid, Spain*
[d] *Utrecht University*

Abstract. The analysis of court decisions and associated events is part of the daily life of many legal practitioners. Unfortunately, since court decision texts can often be long and complex, bringing all events relating to a case in order, to understand their connections and durations is a time-consuming task. Automated court decision timeline generation could provide a visual overview of what happened throughout a case by representing the main legal events, together with relevant temporal information. Tools and technologies to extract events from court decisions however are still underdeveloped. To this end, in the current paper we compare the effectiveness of three different extraction mechanisms, namely deep learning, conditional random fields, and rule-based method, to facilitate automated extraction of events and their components (i.e., the event type, who was involved, and when it happened). In addition, we provide a corpus of manually annotated decisions of the European Court of Human Rights, which shall serve as a gold standard not only for our own evaluation, but also for the research community for comparison and further experiments.

Keywords. event extraction, named entity recognition, court decisions

1. Introduction

Court decisions are an important source of law information for legal practitioners: they elaborate on the facts of a case, involved parties, interpretations of the circumstances, the applicable law and legal principles, and finally the legal assessment leading to the decision. Legal professionals constantly extract, interpret and reason with and about prior cases whilst arguing for a decision in a current, undecided case. However, court decisions texts can be long and complex and thus time-consuming to read. Therefore it would be beneficial to find a means to provide a quick overview of a case, thereby helping to turn decisions into operational, consumable and actionable legal knowledge.

In this work we focus specifically on using Natural Language Processing (NLP) techniques to automatically extract the essence of a court case. Besides extracting general legal rules from individual cases, we aim at providing a quick overview of what happened, who was involved and when the event took place. In the terminology of NLP, event extraction can be treated as a *text classification task* aiming at assigning text fragments (typically, paragraphs, sentences or smaller parts of documents) to predefined (event) classes [1]. Another, related NLP task is Named Entity Recognition (NER) which extracts entities referred to in texts and classifies them into categories [2], for instance people, places and organizations; moreover, named entities can also be domain-specific, for instance, courts or laws. Event extraction can benefit from NER, since it can be used to enrich events with relevant information, such as the parties involved. This paper focuses on the extraction of events and their components from court decisions of the European Court of Human Rights (ECHR)[1] based on a sample thereof.

Summarizing our *contributions*, we: (i) provide a corpus of manually annotated ECHR decisions; (ii) perform a comparison of different approaches to automatically extract events and their components – implementations as well as our evaluation results are made available on GitHub; and (iii) introduce a prototypical web interface that can be used to display court decisions along with their extracted timelines.

The remainder of this paper is structured as follows. We outline related works in Section 2. Our corpus as well as the annotation methodology is described in Section 3. Section 4 contains information about the compared event classification and NER approaches, followed by Section 5 discussing evaluation results. Section 6 provides conclusions.

2. Related Work

Recent advances in NLP are often based on embedding text in multidimensional vector space, with neural network architectures being trained on such numeric representations. Such methods yield in re-usable, publicly available language models trained on large corpora of texts, where embeddings can be created on different levels, for instance words, sentences and documents. While pre-training models on large corpora of generic texts is a very time-consuming process [3], adapting (aka fine-tuning) such generic models to domain-specific language is often less demanding.

Overviews on diverse automated event extraction approaches in the general domain can be found in literature [4,5]. Specifically in the legal domain [6], existing work usually involves searching for *ad hoc* definitions of events, ignoring general event annotation schemas such as the ACE 2005 model [7]. Several approaches tend to be supported by patterns, using manually crafted rules or semantic role labeling techniques [8,9,10,11]. Other approaches do not search for events specifically, but target legal case factors [12].

The automated generation of timelines out of annotated documents could help to get a better and faster understanding of the content of a document. Existing work focusing on this task include Linea [13], a system that is able to build and navigate timelines from unstructured text, and TimeLineCurator [14] a system that is primarily designed to allow journalists to generate temporal stories, however can be used to produce a timeline from any free text or url. Furthermore, the creation of timelines has also been investigated in other domains, such as medicine [15,16] and journalism [17]. We refer to [14] for further details on the respective approaches.

Regarding corpora in the legal domain, court decisions of the ECHR have also been used in literature for different tasks [18,19]. Nevertheless, very few annotated corpora from the legal domain have been made available, and to the best of our knowledge none of them considers events.

[1] `https://echr.coe.int/`

3. Corpus and annotation methodology

This section describes the ECHR corpus as well as our annotation methodology.

Description of the corpus. The corpus consists of 30 decisions of the ECHR. The ECHR decisions were chosen because they contain: i) different types of time-related events concerning different actors in comparison with the decisions of the Court of Justice of the EU [6]; and ii) a standard structure in which different legal events are embedded. ECHR decisions are divided into several sections containing specific information according to Rule 74 of the Rules of the Court [20]: the *Preamble* and the *Introduction* are followed by *Facts* which contain information about the formal procedure and the circumstances of the case providing details about what happened. The following *Law* section describes the legal situations and states the alleged violation(s). The document concludes with the *Decision* section. For the purposes of this paper, we use the mentioned document structure excluding the *Law* section and focus on the procedure, circumstances and decision.

Annotation methodology. The corpus was annotated by two legal experts in several iterations. The experts annotated independently and then met with a third person to reach a consensus on the disagreements. In this work, as we focus on event extraction aimed to automated court decision timeline generation, we were interested in information that is relevant to searching for or extracting time-related information, such as events, processes, temporal information, and the parties involved. As time-related events of cases are linguistically expressed, we annotated the most salient candidate passages thereof. The decisions were manually annotated following the frame "who-when-what". To illustrate the applicability thereof, we make use of an annotated paragraph of the case Altay v. Turkey (no. 2), no. 11236/09, 9 April 2019 (a case referring to respect of private life):

"On 29 May 2008 the applicant lodged an application with the Edirne Enforcement Court for the restriction on the conversations between him and his lawyer to be lifted."

"Who" corresponds to the subject of the event, which can either be a subject, but also an object (i.e., an application); in the example, the subject is "(the) applicant". *"When"* refers to the date of the event, or to any temporal reference thereto; in the paragraph considered, the "when" is the "29 May 2008". *"What"* usually corresponds to the main verb reflecting the baseline of all the paragraph (which in this case is "lodged"); additionally, we include thereto a *complementing* verb or object whenever the core verb is not self-explicit or requires an extension to attain a sufficient meaning; in the paragraph considered, the "what" is "lodged an application". Another e.g. is "dismiss an action". *"Event"* relates to the extent of text containing contextual event-related information. The *type* of such annotations can be either *circumstance* – meaning that the event correspond to the facts under judgment; or *procedure*– wherein the event belongs to the procedural dimension of the case. This includes court procedures (legal proceedings stricto sensu), but also actions that trigger procedural effects. A further analysis of this distinction can be found in previous literature [6,18]. In the paragraph at stake, we annotated as *event* the whole sentence, being its type *procedure*. Further, we have annotated events and their temporal dimension (related-time events) with concrete guidelines:

Extension of what event element. One *what* event element can also include two or more close-related verbs, e.g. "divorced" and "agree on custody", instead of annotating two connected verbs autonomously. Moreover, whenever there is some causal relationship between events, we annotate merely one, e.g. “they drink water and they felt unwell".

Repeated events. When there is reference to events happening in several dates (e.g. “the dates of birthday of three applicants, respectively“), we annotate solely one event as the *what*, and add just one annotation that covers all the related dates.

Non-dated events. Events that are not dated, though semantically expressing an implicit time reference, are then annotated under “when”, for example, the time expressions as “the same date”, “this afternoon”, “on unspecified dates”, “in a number of occasions”.

Non-annotated events. Some events were not considered relevant to be depicted in a timeline, and therefore not annotated, e.g. the fact that *X was born in X* seemed irrelevant.

Factuality. Events that are mentioned in the text but do not occur, are yet annotated with the feature “factuality”, but not included in the timeline. When events are negated, this feature equals to “NOT”, for instance, a party does not appeal against a decision.

Difficult and blurred annotations. During the annotation process, some events were difficult to tag, and others sparked discussion about how to do it, challenging the stipulated guidelines and evidencing how complex and subjective annotating tasks can be. Due to space constrains, we only show one sample annotation that triggered discussion on the type of events between procedure/circumstance. Further examples can be found in the corpus webpage. Regarding the paragraph *"On 26 February 2014 the Deputy Town Prosecutor carried out an inspection of remand prison SIZO-6"*, the issue relates to the semantics attributed to the role "Deputy Town Prosecutor" which renders the idea of being a court magistrate, and as such, it would be deemed as a procedural event. Herein, the function instead refers to an inspection task, without procedural effect.

4. Event extraction and named entity recognition

Herein we describe different methods used in our experiments for the extraction of events and their components in the ECHR court decisions. The applied approaches include deep-learning- and embeddings- based, conditional random fields and rule-based methods. The corpus and the code used in this paper is available on Github[2].

4.1. Deep learning

The task of assigning one or multiple classes from a set of classes to a text fragment is called text classification [1]. Fragments in our context are typically sentences that are classified into the types *procedure*, *circumstance* or neither. Hence we deal with a multi-class classification problem. The extraction of the event components is similar to a Named Entity Recognition Problem. We use a state-of-the-art model as a baseline and compare it further with additional approaches selected upon their results on legal texts (cf. [21,22, 23]). As there is no pre-trained legal model available, we apply the common approach

[2] https://mnavasloro.github.io/EventsMatter/

of *fine-tuning* a Universal Language Model for Text Classification (ULMFiT) [3] which takes a generic model and tunes it with a domain-specific corpus (called transfer learning). In terms of preprocesing, we remove very short sentences from the dataset, for instance headings such as *II THE LAW*. The models are:

Flair and Flair-finetuned. We first selected the generic *news-forward-fast* language model from the Flair embedding approach [24], which is pre-trained on a corpus with one billion words as our baseline model. We also fine-tune the pre-trained model with the documents from our corpus for one epoch (which took more than seven hours).

Flair ECHR. There are no specific legal pre-trained models available that we could use for our experiments. On a different classification task, we made good experiences in prior work with using a domain specific model trained on a small corpus of EU legal documents outperforming generic models in a multi-label text classification task [25]. Therefore, we also train a model on a corpus of 13,000 ECHR court decisions acquired from the European Court of Human Rights OpenData project [26] for four epochs.

BERT and BERT-finetuned. The Bidirectional Encoder Representations from Transformers (BERT) [27] is a language model learning the context of words in a bidirectional way and is applicable to many tasks. We use a BERT model (*bert-base-cased*) pre-trained on Wikipedia and a book corpus, plus further add a layer on top fine-tuning the model with the ECHR corpus for two epochs.

DistilBERT and DistilBERT-finetuned. DistilBERT [28] is a lightweight version of BERT that makes use of a teacher-student setup to distill the knowledge of the large model (BERT) to the student (DistilBERT). Our fine-tuned model (two epochs) is based on the pre-trained *distilbert-base-cased* model with an additional ECHR corpus layer.

4.2. Conditional Random Fields

Conditional Random Fields (CRF) are used for the mapping of sequences based on probabilistic models to label sequences [29]. CRF have already been applied in similar tasks in the legal domain for extracting specific legal entities, such as lawyers, courts and legal literature [30]. A CRF model uses features of a token, for instance casing, position of the token and subsequences, to calculate the probability that it is preceded or followed by a particular other token. It also takes the probabilities into account that a specific named entity, for instance a temporal information is followed by a subject.

4.3. Rule-based

Unlike the previous approaches, implemented as a classification task, the rule-based approach is an annotation task based on a search for specific patterns of events in the form of frames. Our approach has two steps: i. the collection of frames (done before the annotation), and ii. the event extraction that uses the frames in order to annotate a text.

1. Frame collection. We listed all *what* event components in the training set, and then identified the main verb, its type and the dependency relations (using the CoreNLP dependency parser [31]), within the *what*, and towards the subject (*who*), including the object for both possible active and passive voices since they are very different. When there are several mentions of the same main verb, all information is gathered and combined into a single frame. Once all the *what* elements are processed, they are stored for later use by the extraction algorithm.

2. Event extraction. Using the previously obtained frames, we look for the relevant events in the text. Since there are events that can appear many times in a text, we just consider events that have a date attached. To find dates and their normalized value (in order to be able to build a timeline), we adapted the Añotador software [32]. Then we used the information from the frames to look for the main verb of the event and for the previously identified dependency relations, as well as some Part-of-Speech considerations (using also CoreNLP). Additionally, some specific events that tend to appear always in the same form in the text (such as the final decision) are identified using regular expressions.

4.4. Use case: Timeline generation

In order to enable an intuitive way to overview a case, we decided to generate timelines from the case. We developed a demonstrator [3] that takes the *id* of a ECHR case and returns its rule-based annotation and generates a timeline. Through this timeline, we can navigate a case going directly to the event mention in the text just by clicking on it in the timeline. The fact that it directly refers to the text allows the user to retrieve the context of the event, as well as surrounding information that might not be reflected in the timeline.

5. Evaluation and Discussion

In this section we present results of our experiments. For experiments based on deep learning approaches, we used the state-of-the-art NLP library Flair[4] which uses contextualized string embeddings (called FlairEmbeddings) that captures the semantics and the context, and therefore, produce different context dependent embeddings for the same words [24]. The pre-trained transformer models (BERT, DistilBERT) are provided by the Huggingface library [33] and can be easily imported into Flair. The Flair ECHR model is created using the Flair library, and fine-tuning of the BERT and DistilBERT models is also based on the transformers library by Huggingface. All models have been trained with the same settings of a maximum of 150 epochs, patience of 3 and an anneal factor set to 0.5 and the training is automatically stopped when the learning rate is too small. We use common evaluation metrics: *Precision (P)*, *Recall (R)* and *F-score (F)*.

The documents have an average size of 2,302 tokens without the legal section (legal framework). Each document includes on average 21 different events, divided into 10 *procedure* and 11 *circumstance* events on average. The number of *who* occurrences amounts to 13.9 on average, while the number of temporal information annotations (*when*) to 17.6, and the number of *core* annotations to 24. We split the dataset into training, testing and validation set on a document level applying 5-fold cross-validation (in the deep learning based approach) such that the training set consists of 24, and the test and validation set of three documents each. The results represent the average of all splits. The results for all approaches are presented in Table 1. When comparing different approaches on event (component) extraction, we can observe that more advanced language models based on

[3] `https://whenthefact.oeg-upm.net/`

[4] `https://github.com/flairNLP/flair`

Table 1. Evaluation results for event classification and event components. (*P=Precision, R=Recall, F=F-score. Best results highlighted in boldface.*)

		Event Types		Event Components		
		Procedure	Circumstance	What	When	Who
CRF	P	82.39	68.78	85.10	89.30	89.09
	R	80.26	47.88	76.91	84.46	70.38
	F	80.80	54.78	**80.50**	86.58	78.34
Flair pretrained	P	83.32	57.21	56.41	90.50	89.93
	R	78.95	32.64	45.50	79.65	76.49
	F	80.31	40.57	50.10	84.35	82.30
Flair finetuned	P	87.07	58.88	60.12	90.87	91.63
	R	81.57	51.12	51.79	80.02	83.71
	F	84.13	53.33	55.58	84.87	87.44
Flair ECHR	P	76.78	41.93	57.94	82.00	40.48
	R	71.21	13.12	15.69	57.88	11.87
	F	73.86	17.92	23.28	66.88	18.23
BERT pretrained	P	81.95	66.70	60.45	85.88	86.37
	R	80.79	49.23	61.17	88.22	89.90
	F	80.56	54.31	60.78	86.98	88.05
BERT finetuned	P	91.44	76.81	65.58	89.45	88.88
	R	90.20	78.94	66.26	91.01	92.22
	F	90.55	77.59	65.83	**90.22**	**90.44**
DistilBERT pretrained	P	83.91	56.53	59.58	81.87	86.67
	R	83.57	51.63	57.45	86.35	85.73
	F	83.26	53.26	58.41	83.95	86.09
DistilBERT finetuned	P	91.64	81.61	62.79	87.31	89.92
	R	93.27	78.65	62.06	89.33	90.12
	F	**92.38**	**79.75**	62.37	88.23	89.98

		Event				Event Components					
		Identification		Type		What		When		Who	
		Len	Str	Len	Str	Len	Str	Len	Str	Len	Str
Rules	P	85.71	80.00	47.14	42.86	80.26	23.68	77.59	72.41	75.00	68.75
	R	77.92	72.73	42.86	38.96	69.32	20.45	63.38	59.15	63.16	57.89
	F	81.63	76.19	44.90	40.82	74.39	21.95	69.77	65.12	68.57	62.86

the transformer architecture [34] (BERT and DistilBERT), in general, provide a better result compared to the standard embedding models (Flair). Furthermore, we can see that the application of the ULMFiT approach to finetune generic language models, with a domain-specific corpus, leads to improved results between less than 1% (Flair pretrained to Flair finetuned for *who*) and 25% (DistilBERT for *circumstance*). The average increase in performance with fine-tuning is 8% for recognizing *procedure* and 21% for *circumstance* events, resp. The results of the CRF approach for the *what* component is unexpected, as it outperforms the more advanced methods by approximately 20%. The results for the extraction of the event components show that recognizing temporal information (*when*) of an event yields better results than the *what* of an event by 27% and the subject (*who*) by 21% (averaged over all approaches). The performance increase for the extraction of the event components of fine-tuned models, compared to generic models, is with 5% (what), 3% (when) and 4% (who) lower compared to the results for event types.

We see that results within the event type detection are within approx. 20% over all approaches, with the worst result being achieved by the Flair ECHR approach (F 73.86%), and the best result by the DistilBERT finetuned approach with an F-score of 92.38%. The results for the *circumstance* event type show a bigger spread between the worst result of the Flair ECHR approach with an F-score of only 17.92%, while the best result is achieved by DistilBERT finetuned (F 79.75%). For the *circumstance* event types we see generally lower results than for *procedure* type detection. We attribute this to the fact that the linguistic variety of the *procedure* events is narrower as they refer to a restricted set of ways of how to express them. The performance of the Flair ECHR model showed the least performance, due to being trained only on 13,000 ECHR documents, while it is common to train language models on much larger corpora to capture the basics of a language.

The performance differences between the *procedure* and *circumstance* event classes are evident with the latter results being worse by 29% on average. *Procedure* events capture formal processes throughout a legal trail and the ways to formulate the same events is somewhat restricted, for instance, *the court upheld the judgment*; in the description of the *circumstance*s of a case, however, the English language is potentially used in its entirety. Similarly, we observe the same behavior with the results for the event components with the results for *when* and *who* being better than the results for *what*. We attribute this to the fact that absolute temporal information (e.g. a date) contained in the court decisions under investigation always follows the structure of *Day Month Year*, and the number of acting subjects is also limited to a certain range of persons (eg. applicant, judge, prosecutor), authorities (eg. Supreme Court, housing authority) or things (eg. application, appeal). Relative temporal information (eg. *X days later*, *between X and Y* or *until X*) is also expressed in a few ways only.

Overall, we can say that fine-tuning an existing language model trained on a large corpus that captures the basic features of a language with a domain-specific corpus performs better than training a new language model with a rather small domain-specific corpus. Moreover, the more restricted the variety of class candidates for classification is, the better the results. The same applies to the information following a specific format, i.e. temporal information in the form of dates.

Regarding the rule-based approach, the evaluation is slightly different. While in the deep learning approach (first table) the number of named entities reflect the results of finding the event arguments *only* in those sentences where there is an event. On the contrary, the rule-based approach (second table) finds the events and the arguments in the same algorithm, so the results of the argument are contingent upon the event results. Additionally, we provide both *strict* and the *lenient* results, meaning that either the extent of our annotation match exactly to the one by the annotators or that it only overlaps (adding or omitting some words), resp. Also, the event evaluation includes finding the extent of the event, and then, over this finding, decide its type. The annotation and evaluations for the rule-based approach were done with the software GATE [35].

From the results of the rule-based approach we see that in the event finding task we got good results, both in the strict and lenient case, meaning that most of the events are correctly found and with the correct extent. Generally speaking, we identify about 4 out of every 5 relevant events, and additionally some that were not marked as relevant (although this does not mean they are not events). Regarding event types, the results for rule-based approaches are not very promising, mainly due to the fact that the same verb can often represent both circumstantial or procedural events, depending on surrounding information that the current rule-based implementation is not able to identify.

Results for detecting event arguments with the rule-based approach, on the other hand, are very different. While the *what* event component has very bad strict results, mainly due to the difficulty to determine the extent of the relevant modifiers of a verb, the *who* and the *when* show very good results, finding correctly most of them (e.g., 68.57% of the *who* taken into account that the limit was less than the 81.63% of the events) and almost always with the correct extent. The lenient results of the core, similar to the ones from the other arguments, demonstrates that besides the extent, the identification is correct.

6. Conclusions and Future Work

This paper presented a new corpus of legal decisions annotated with relevant events, along with a comparison of different approaches for the extraction of events and their components. Moreover, we tested state of the art methods to accomplish this annotation task automatically with promising results. To illustrate the utility of this task, we implemented an online timeline generation service which could be used by lawyers to get a quick overview of a case, thereby helping to turn decisions into operational, consumable and accessible legal knowledge.

To the best of our knowledge there is no previous comparison of event extraction techniques over legal texts in literature, and neither an available legal corpus annotated with events. In future work it would be interesting to validate the results with decisions from other courts such as the European Court of Justice or the United States Supreme Court, which are structured differently.

Acknowledgements. María Navas-Loro[5] work was partially supported by a Predoctoral grant from the I+D+i program of the Universidad Politécnica de Madrid. Sabrina Kirrane is funded by the FWF Austrian Science Fund and the Internet Foundation Austria netidee SCIENCE programme.

References

[1] Sebastiani F. Machine learning in automated text categorization. ACM Comput Surv. 2002;34(1):1–47.

[2] Grishman R, Sundheim BM. Message understanding conference-6: A brief history. In: COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics; 1996. p. 466–471.

[3] Howard J, Ruder S. Fine-tuned Language Models for Text Classification. CoRR. 2018;abs/1801.06146.

[4] Hogenboom F, Frasincar F, Kaymak U, De Jong F. An overview of event extraction from text. In: DeRiVE@ ISWC. Citeseer; 2011. p. 48–57.

[5] Xiang W, Wang B. A Survey of Event Extraction From Text. IEEE Access. 2019;7:173111–173137.

[6] Navas-Loro M, Santos C. Events in the legal domain: first impressions. In: TERECOM@JURIX; 2018. p. 45–57.

[7] The ACE 2005 Evaluation Plan.;. `https://api.semanticscholar.org/CorpusID:10821576`.

[8] Kiyavitskaya N, Zeni N, Breaux TD, Antón AI, Cordy JR, Mich L, et al. Automating the extraction of rights and obligations for regulatory compliance. In: International Conference on Conceptual Modeling. Springer; 2008. p. 154–168.

[9] Maxwell KT, Oberlander J, Lavrenko V. Evaluation of semantic events for legal case retrieval. In: Proceedings of the WSDM'09 Workshop on Exploiting Semantic Annotations in Information Retrieval. ACM; 2009. p. 39–41.

[5] ORCID 0000-0003-1011-5023

[10] Lagos N, Segond F, Castellani S, O'Neill J. Event extraction for legal case building and reasoning. In: International Conference on Intelligent Information Processing. Springer; 2010. p. 92–101.

[11] Navas-Loro M, Satoh K, Rodríguez-Doncel V. Contractframes: Bridging the gap between natural language and logics in contract law. In: JSAI International Symposium on Artificial Intelligence. Springer; 2018. p. 101–114.

[12] Wyner AZ, Peters W. Lexical Semantics and Expert Legal Knowledge towards the Identification of Legal Case Factors. In: JURIX. vol. 10; 2010. p. 127–136.

[13] Etiene T, et al. Linea: Building Timelines from Unstructured Text. In: 28th SIBGRAPI Conference on Graphics, Patterns and Images, SIBGRAPI 2015. IEEE Computer Society; 2015. p. 234–241.

[14] Fulda J, et al. TimeLineCurator: Interactive Authoring of Visual Timelines from Unstructured Text. IEEE Trans Vis Comput Graph. 2016;22(1):300–309.

[15] Styler IV W, et al. Temporal Annotation in the Clinical Domain. Transactions of ACL. 2014;2:143–154.

[16] Jung H, et al. Building timelines from narrative clinical records: initial results based-on deep natural language understanding. In: Proceedings of BioNLP 2011 workshop. ACL; 2011. p. 146–154.

[17] Tannier X, Vernier F. Creation, Visualization and Edition of Timelines for Journalistic Use. In: Proceedings of Natural Language meets Journalism Workshop at IJCAI; 2016. .

[18] Navas-Loro M, Filtz E, Rodríguez-Doncel V, Polleres A, Kirrane S. TempCourt: evaluation of temporal taggers on a new corpus of court decisions. The Knowledge Engineering Review. 2019;34:e24.

[19] Medvedeva M, Vols M, Wieling M. Using machine learning to predict decisions of the European Court of Human Rights. Artificial Intelligence and Law. 2020;28(2):237–266.

[20] Registry of the Court. European Court of Human Rights; 2020. Accessed 2020-09-14. `https://www.echr.coe.int/documents/rules_court_eng.pdf`.

[21] Chalkidis I, Fergadiotis M, Malakasiotis P, Androutsopoulos I. Large-Scale Multi-Label Text Classification on EU Legislation. CoRR. 2019;abs/1906.02192.

[22] Shaheen Z, Wohlgenannt G, Filtz E. Large-scale legal text classification using transformer models. In: Semapro 2020; to appear.. .

[23] Tuggener D, von Däniken P, Peetz T, Cieliebak M. LEDGAR: a large-scale multi-label corpus for text classification of legal provisions in contracts. In: 12th Language Resources and Evaluation Conference (LREC) 2020. European Language Resources Association; 2020. p. 1228–1234.

[24] Akbik A, Blythe D, Vollgraf R. Contextual String Embeddings for Sequence Labeling. In: COLING 2018, 27th International Conference on Computational Linguistics; 2018. p. 1638–1649.

[25] Filtz E, Kirrane S, Polleres A, Wohlgenannt G. Exploiting EuroVoc's Hierarchical Structure for Classifying Legal Documents. In: OTM Confederated International Conferences" On the Move to Meaningful Internet Systems". Springer; 2019. p. 164–181.

[26] Quemy A. European Court of Human Right Open Data project. CoRR. 2018;abs/1810.03115.

[27] Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: NAACL-HLT (1); 2019. p. 4171–4186.

[28] Sanh V, Debut L, Chaumond J, Wolf T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. CoRR. 2019;abs/1910.01108.

[29] Lafferty JD, McCallum A, Pereira FCN. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In: Brodley CE, Danyluk AP, editors. Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001). Morgan Kaufmann; 2001. p. 282–289.

[30] Leitner E, Rehm G, Moreno-Schneider J. Fine-grained Named Entity Recognition in Legal Documents. In: International Conference on Semantic Systems. Springer; 2019. p. 272–287.

[31] Chen D, Manning CD. A fast and accurate dependency parser using neural networks. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP); 2014. p. 740–750.

[32] Navas-Loro M, Rodríguez-Doncel V. Annotador: a temporal tagger for Spanish. Journal of Intelligent & Fuzzy Systems. 2020;39:1979–1991. 2.

[33] Wolf T, Debut L, Sanh V, Chaumond J, Delangue C, Moi A, et al. HuggingFace's Transformers: State-of-the-art Natural Language Processing. arXiv e-prints. 2019 Oct;p. arXiv:1910.03771.

[34] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. In: Advances in neural information processing systems; 2017. p. 5998–6008.

[35] Cunningham H, et al. Getting More Out of Biomedical Documents with GATE's Full Lifecycle Open Source Text Analytics. PLOS Computational Biology. 2013 02;9(2):1–16.

10. A novel model usability evaluation framework for explainable artificial intelligence

Bibliographic Information

Dieber, J. and **Kirrane, S.**, 2022. A novel model usability evaluation framework (MUsE) for explainable artificial intelligence. Information Fusion, 81, pp.143-153.

Applicants contributionn

Conceptualisation, Methodology, Writing - Review & Editing, and Supervision.

Copyright Notice

A Novel Model Usability Evaluation Framework (MUsE) for Explainable Artificial Intelligence

Jürgen Dieber, Sabrina Kirrane*

Institute for Information Systems and New Media,
Vienna University of Economics and Business, Welthandelsplatz 1, 1020 Vienna, Austria

Abstract

When it comes to complex machine learning models, commonly referred to as black boxes, understanding the underlying decision making process is crucial for domains such as healthcare and financial services, as well as when they are used in connection with safety critical systems such as autonomous vehicles. As a result, interest in explainable artificial intelligence (xAI) tools and techniques has increased in recent years. However, the user experience (UX) effectiveness of existing xAI frameworks, especially concerning algorithms that work with data as opposed to images, is still an open research question. In order to address this gap, we examine the UX effectiveness of the Local Interpretable Model-Agnostic Explanations (LIME) xAI framework, one of the most popular model agnostic frameworks found in the literature, with a specific focus on its performance in terms of making tabular models more interpretable. In particular, we apply several state of the art machine learning algorithms on a tabular dataset, and demonstrate how LIME can be used to supplement conventional performance assessment methods. Based on this experience, we evaluate the understandability of the output produced by LIME both via a usability study, involving participants who are not familiar with LIME, and its overall usability via a custom made assessment framework, called Model Usability Evaluation (MUsE), which is derived from the International Organisation for Standardisation 9241-11:2018 standard.

Keywords: Machine learning, Explainable Artificial Intelligence, Model Agnostic Explanations, Usability Study, User Experience

1. Introduction

Since the term was first mentioned in 1956 [1], artificial intelligence (AI), and especially its subset machine learning, has steadily made its way into various kinds of industries and aspects of our lives, like healthcare[1][2], transportation[3] and advertisement[4][5]. While machine learning applications are advancing further, the understanding of how machine learning models work and how decisions are made is not advancing at the same pace. In some applications like recommendation systems or predictive maintenance it may not be necessary to understand the black box decision making, as long as the models' predictions are accurate in the majority of cases. However, in circumstances where human lives are involved, like medical diagnosis or self-driving cars, the ability to understand the decision process is essential in order to establish trust in such systems.

In this context, Arrieta et al. [2] defines understandability, as *"the characteristic of a model to make a human understand its function – how the model works – without any need for explaining its internal structure or the algorithmic means by which the model processes data internally."* Efforts made in the field of Explainable AI (xAI) [3] aim to accomplish just that, by building and using models that generate transparency for their users, thus giving a functional understanding of the model [4]. One approach is to develop powerful and fully explainable models, such as deep k-nearest neighbours [5] and teaching explanations for decisions [6], with an explanation being an accurate proxy of the decision maker, used with the aim to create understandability for humans [7]. Another approach is to tackle the issue of model agnostic post modelling interpretability, hence, the ability to explain the meaning to a person [2], by explaining the output of well established machine learning models, instead of replacing these models entirely (cf., LIME by Ribeiro et al. [8], SHAP by Lundberg and Lee [9], and MAPLE by Plumb et al. [10]).

When it comes to xAI frameworks, the Local Interpretable Model-Agnostic Explanations (LIME) framework is, with 5832 citations[6], one of the predominant tools dis-

*Corresponding author

Email addresses: `juergen.dieber@gmail.com` (Jürgen Dieber), `sabrina.kirrane@wu.ac.at` (Sabrina Kirrane)

[1] https://www.entrepreneur.com/article/341626

[2] https://medicus.ai/de/

[3] https://kodiak.ai/

[4] https://instapage.com/blog/machine-learning-in-advertising

[5] https://www.ezoic.com/

[6] https://bit.ly/3hcv4eS

cussed in the literature. For instance, one highly cited publication, by Selvaraju et al. [11] (with 5083 citations), remarks that the method on assessing trust in models, proposed by Ribeiro et al. [8], motivated them to use a similar approach to assess their own model. Another prominent example, the interpretability SHAP framework, by Lundberg and Lee [9] (with 3587 citations), bases its computational method on LIME and also uses LIME as a benchmark for their performance evaluation.

Another indicator for LIMEs popularity is their activity on the biggest repository hosting service GitHub[7][8]. From August 2016 to July 2021 the project has been bookmarked (*starred*) over 9000 times, has been copied (*forked*) over 1500 times and has been used by over 1300 GitHub users. 45 researchers and developers have contributed to the project with over 526 approved commits, with the most recent update being made in June 2021.[9]

Although existing publications primarily use LIME as a benchmarking framework in order to assess their tools [12, 13, 14], they do not evaluate the effectiveness of LIME from a usability perspective, hence its explainability. No extensive assessment of its effectiveness from a user experience (UX) perspective has been conducted to date, thus the overarching goal of this work is to close this gap.

Summarizing our contributions, we: (i) demonstrate how LIME can be used to supplement conventional performance assessment methods; (ii) evaluate the understandability of the output produced by LIME via a usability study; and (iii) propose an assessment framework, which is derived from the International Organisation for Standardisation (ISO) 9241-11:2018 standard, that can be used not only to evaluate the usability of LIME but also other xAI frameworks. In addition, our code and data are made available in a GitHub repository[10].

The reminder of this article is structured as follows: Section 2 summarizes the state of the art with respect to post-modelling interpretability. Section 3 compares the performance of several machine learning models using conventional methods. Section 4 illustrates the value LIME adds when it comes to understating the models output in comparison to conventional performance assessment methods. Section 5 evaluates LIME from a usability perspective via a user-study and by analyzing the experience we have had via a self-assessment. Finally, our conclusions and interesting directions for future work are presented in Section 6.

2. A comparative analysis of existing work on model agnostic explainablility

Existing work relating to xAI can be grouped into two distinct categories: (i) the development of fully explainable models (cf., [5, 6]), which are interpretable by design, without using another framework, and (ii) the development of model agnostic explainability frameworks (cf., [8, 9, 10]), which are used on a model to make it more interpretable. Considering that model agnostic frameworks can be used with any machine learning algorithm, in this paper we focus specifically on the latter. In particular, our integrative literature review, which is summarised in Table 1, focuses on comparing and contrasting existing work with respect to the scope of the interpretability, the type of data the method is tested with, and the evaluation used to assess or compare the methods performance.

In terms of the scope of interpretability, a framework can either be on a *global level*, meaning it makes different models comparable with each other, by summarizing their performance with respect to specific indicators, or on a *local level*, giving insight into how a classification in the case of a single prediction is made. Although the vast majority of works focus on local interpretability [34, 35, 45, 19, 41, 40, 21, 32, 39, 44, 30, 31, 27, 12, 16, 24, 20, 3, 18, 42, 9, 13, 28, 10, 8, 22, 17, 26, 33, 38, 46, 14, 36, 37], several can also be used for a global comparison [45, 19, 41, 32, 31, 47, 15, 18, 10, 23, 8, 29, 46]. Only the activation maximization method [15] and model distillation [29] are exclusively global. Although each of the papers includes some demonstration of the method using a specific data type, the actual data used is very different: twenty-four methods are applied to tabular data [35, 45, 19, 41, 21, 32, 44, 30, 31, 12, 16, 24, 3, 18, 42, 13, 10, 23, 8, 22, 17, 38, 29, 14], sixteen are applied to image data [35, 40, 39, 27, 20, 3, 9, 15, 28, 17, 38, 46, 36, 37], and eight are applied to textual data [34, 3, 42, 8, 22, 26, 33, 38]. Only four publications, Koh and Liang [3], Ribeiro et al. [8, 22] and Sundararajan et al. [38] include an application of all three data types.

Concerning the evaluation technique, where an assessment is performed two different methods are used: a *baseline evaluation* and a *user interview*. A baseline evaluation is a quantitative evaluation technique, where one or more indicators are used for a comparative assessment. For instance, Plumb et al. [10] uses a self defined causal local explanation metric to compare their framework to LIME. In total, eight of the publications apply some sort of baseline evaluation [34, 21, 12, 20, 42, 13, 10, 14]. The second evaluation technique is a qualitative method, either a survey or user interview. Only three publications use this approach. Lakkaraju et al. [47] and Lundberg and Lee [9] include a survey in their evaluation and Dhurandhar et al. [21] ask two professionals to rate a mixed set of interpretability framework outputs given to them. Out of the ten publications who evaluate their framework, six draw a comparison to LIME [21, 12, 42, 9, 10, 14], from which we can assume that LIME constitutes a benchmark for interpretability frameworks. However, when it comes to the evaluation of LIME itself, none of the publications actually use evaluation techniques to assess LIMEs performance and only Sokol and Flach [25] evaluate LIME as a demonstration of their novel explainability taxonomy.

[7] https://github.com/

[8] https://thenewstack.io/i-dont-git-it-tracking-the-source-collaboration-market/

[9] https://github.com/marcotcr/lime

[10] https://github.com/jdieber/WhyModelWhy

Method	Reference	Scope	Data Type	Evaluation Technique
Activation maximization	[15]	Global	Image	-
Counterfactual	[16], [17]	Local	Tabular, Image	-
Feature importance	[18], [19]	Global, Local	Image, Tabular	-
Fisher kernels	[20]	Local	Image	*Baseline evaluation:* Fisher kernels compared to Influence functions
Frequency map	[21]	Local	Tabular	*Baseline evaluation:* MACEM compared to LIME *User interview:* MACEM compared to LIME
if-then rules	[22], [23]	Global, Local	Image, Tabular, Text	-
Influence function	[3]	Local	Image, Tabular, Text	-
LIME	[8], [24], [22], [25]	Global, Local	Image, Tabular, Text	-
LIME extension	[26], [27], [12], [14], [28], [13]	Local	Image, Tabular, Text	*Baseline evaluation:* SUP-LIME compared to K-LIME; SLIME compared to positive saliency map; DLIME compared to LIME
MAPLE	[10]	Global, Local	Tabular	*Baseline evaluation: MAPLE* compared to LIME
Model distillation	[29]	Global	Tabular	-
Parametric statistical tests	[30]	Local	Tabular	-
Partial dependence plot	[31]	Global, Local	Tabular	-
Prototype and criticism	[32]	Global, Local	Tabular	-
Ranking models	[33]	Local	Text	-
Relevance scores	[34]	Local	Text	*Baseline evaluation:* LRP compared to TFIDF and uniform
Saliency map	[35], [36], [37], [38], [39], [40]	Local	Tabular, Text, Image	-
Sensitive analysis	[41]	Global, Local	Tabular	-
Shapley value	[9], [19], [42], [43], [44]	Local	Tabular, Text, Image	*Baseline evaluation:* true shapley value, classical shapley estimations, LIME and ES values *User interview:* SHAP compared to true shapley Value, LIME and shapley sampling
Surrogate models	[8], [45], [46]	Global, Local	Image, Tabular, Text	-
Visualisation	[19]	Global, Local	Tabular	-

Table 1: Existing model agnostic explainablility approaches

Model agnostic frameworks have also been applied in several domains. Within the medical sector, considering that AI systems are used to support the diagnosis, both Gale et al. [48] and Katuwal and Chen [24] identify the need to enhance model comprehensibility for the professionals using them. In the case of Holzinger et al. 2019 [49] they go beyond simply explaining the models, towards uncovering causality. Within the field of news detection, the automatic understanding or processing of text, xAI helps to shed light on the multi-layer deep learning applications used for advanced applications [34]. While, in the music business, content analysis is supported by model agnostic interpretability frameworks in order to gain a better understanding of how certain tones are identified [13].

Although the LIME framework[11], especially its image explainer, is one of the predominant tools discussed in the literature, its tabular explainer has received limited attention to date. In addition, existing work focuses primarily on using LIME as a benchmark as opposed to assessing the usability of LIME itself. In order to fill this gap in this paper we apply LIME on tabular machine learning models and evaluate LIMEs performance in terms of comparabil-

[11] https://github.com/marcotcr/lime

ity, interpretability and usability.

3. Using machine learning to classify tabular data

We start by presenting four state of the art classification models, namely decision tree [50], random forest [51], logistic regression [52] and XGBoost [53]. Following on from this, we make use of conventional methods (i.e., the classification report [54] and receiver operating characteristic curve [55]) to assess the model performance and identify the best performing algorithm.

3.1. Tabular data pre-processing

For our tabular data analysis we use the *Rain in Australia* data-set from Kaggle[12]. Before the algorithm is trained, we work through the different variables step by step to fully understand their meaning and make them processable by our model. Given that *RISK_MM* has a 100% correlation with the target variable, it is removed. Other variables with too many missing values are also excluded. A summary of the full dataset is given in Table 2, while the features we use for training are denoted with an asterisk.

From a preprocessing perspective, we modify several categorical variables, making them numeric so they can be processed by the models. We further build a scikit learn `pipeline` object, to apply the preprocessor on the data and sequentially build our model based on its structure. This enables us to perform a sequence of different transformations and to give each algorithm a customised setting while being able to cross-validate each setting-combination during the training process.

The scikit-learn `train_test_split` function is used to break our data into different parts, namely training and testing data. We assign 70% of our observation to the training dataset and the remaining 30% to the testing dataset. Once the data is prepared, we train our four models with the same training data. For comparability reasons, we mainly used standard parameter settings for the setup of the algorithms.

3.2. The application and interpretation of the machine learning models

Our choice of algorithms (i.e., decision trees, random forest, logistic regression and XGBoost) is based on the different levels of interpretability they pose. While the decision tree and the logistic regression are interpretable on their own, the random forest and XGBoost, as examples of ensemble methods, are black box models [8][56] that require interpretation by a framework such as LIME.

In order to analyse the models performance on the testing data, we utilise the sklearn classification report. A model comparison using conventional methods is presented in Table 3. *Precision*, *recall* and *f1-score* are calculated based on the classification results true positive, true negative, false positive and false negative. True positive and true negative both indicate that the weather was correctly predicted with either it is going to rain or it is not going to rain, respectively. A false positive however indicates a class that should not have been predicted positive and false negative indicates that a class should have been predicted positive. The scores next to the metrics name in Table 3 either refer to the target variable that it is not going to rain (0) or that it is going to rain (1) as well as the weighted scores (w) and the training baseline value (tr) for the receiver operating characteristic curves (ROC). Taking the decision tree as an example, the values are then calculated as follows:

Accuracy: The accuracy gives an average of how often the model classified the target variable correctly, in the decision trees example in 79% of the time.

Precision: The precision describes how often the model was correct in classifying an observation as positive, and is therefore also known as the *positive predictive value*. It is the result of the true positives, divided by the sum of false positives and true positives, adding up to 91% for the outcome that it is not going to rain and 53% for the outcome that it is going to rain.

Recall: For the recall measurement, the performance of the variables is more similar. It consists of the true positives divided by the sum of true positives and false negatives, 81% and 73%, respectively. A popular synonym for recall is the *true positive rate*.

F1-score: The f1-score tells us what percentage of positive prediction is correct, including the *recall* and *precision* into its measurement. The *f1-score* consists of two times the *precision* * *recall* divided by the sum of *precision* and *recall*. The decision tree delivers a *f1-score* of 86% for the outcome that it is not going to rain and 61% for it is going to rain.

Macro score: The macro score represents the overall performance of the indicator, meaning the average. The *macro precision* reaches 82%, the *macro recall* 71% and the *macro f1-score* 74%.

Weighted average score: The weighted average is the respective score times its number of instances, for example, the 0.85% *weighted average precision* result from the target variable not going to rain, having a score of 91% and 53% of target variable going to rain, respectively.

Another state of the art tool to measure the validity of classification results is the ROC curve [57]. Figure 1 displays one ROC curve per model, each graph showing two curves, the upper one is the ROC curve, posing a

[12] https://www.kaggle.com/jsphyg/weather-dataset-rattle-package

variable name	sample input	type	non-null-values
Date	2008-12-03	categorical	142193
Location	Albury	categorical	142193
MinTemp*	13.4	numerical	141556
MaxTemp*	25.1	numerical	141871
Rainfall*	0.00	numerical	140787
Evaporation	23	numerical	81350
Sunshine	11	numerical	74377
WindGustDir*	W	categorical	132863
WindGustSpeed*	44.0	numerical	132923
WindDir9am*	NW	categorical	132180
WindDir3pm*	W	categorical	138415
WindSpeed9am*	25.0	numerical	140845
WindSpeed3pm*	8.0	numerical	139563
Humidity9am*	25.0	numerical	140419
Humidity3pm*	22.0	numerical	138583
Pressure9am*	1007.7	numerical	128179
Pressure3pm*	1007.1	numerical	128212
Cloud9am	2.0	numerical	88536
Cloud3pm	8.0	numerical	85099
Temp9am*	16.9	numerical	141289
Temp3pm*	21.8	numerical	139467
RainToday*	Yes	categorical	140787
RISK_MM	0.2	numerical	142193
RainTomorrow*	No	categorical	142193

Table 2: An overview of the datasets' features (Variables used for the training of the models are marked with a *)

probability, the lower one is the *baseline*, which separates the ROC and the area under the curve (AUC), which is a measurement for separability. The ROC curve uses the false positive rate, *fall-out*, and the true positive rate, *recall*, for its measurement. Due to its graphical display, the curves of different models can be easily compared with each other. Each point on the curve represents the relation between *fall-out* and *recall*. The further to the upper left corner the curve bends, the better the classification. The AUC measures the general accuracy, meaning how well a model can differentiate between classes. It provides an aggregated measure of performance across all possible classification thresholds, which makes it a quality indicator for a model's prediction regardless of what threshold is chosen. For the AUC the following rule holds true: the closer its value is to 1, the better the model is able to correctly classify. If the value is 0.5 it means that the model is not better than randomly guessing and a value of close to 0 means that the model is doing the classification upside down[13][14][15]. For instance, in the case of our decision tree, the *baseline* performs with 0.85 on our test-data and the model can therefore be interpreted as reliable.

3.3. *An assessment of the machine learning models*

Overall it is notable that the performances of the decision tree, random forest and logistic regression are very similar while the XGBoost performance differs significantly. In this comparison, the XGBoost delivers the highest values with a 85% *accuracy*, *weighted average scores* of 85% *precision*, 85% *recall* as well as 84% *f1-score*. But it's weak performance in classifying that it is going to rain correctly, can be seen in a low *recall (1)* and *f1-score (1)* score with 46% and 58%, respectively. It is worth noting that the high difference in the *recall* scores for the respective target variable might be caused by unbalanced testing data, which is something we would like to further explore in future work. The logistic regression offers the highest *recall (1)*, in the case of 77% of the positive observations it predicts correctly that it is going to rain, with a weighted *recall* of 79%. In terms of *f1-score (1)* the logistic regression and the random forest score equal 62% which is four percent higher than the XGBoost with 58%. Furthermore, comparing the ROC curves shows a similar performance for all models, with XGBoost scoring 88% *ROC baseline*, the logistic regression 87%, the random forest 86% and the decision tree 85%, indicating, that all four models are reasonably reliable when it comes to classifying instances correctly.

To summarize, the decision tree performs worst in all metrics. The random forest and the logistic regression never differ more than two percent in any of the metrics and are therefore performing similarly. Although the XGBoost outperforms the others in several metrics, it scores significantly lower when it comes to predicting the outcome of a positive observation. Thus, in order to decide which model should be deployed, based on this results, requires a trade-off: a higher accuracy and more accurate prediction of true negatives would stand in favor of the XGBoost, while the need for a more accurate prediction of true positives would stand in favor of the random forest or the logistic regression. Furthermore, while the confusion matrix and the ROC give us insight into how the models perform, they do not reveal how the models reach a certain decision.

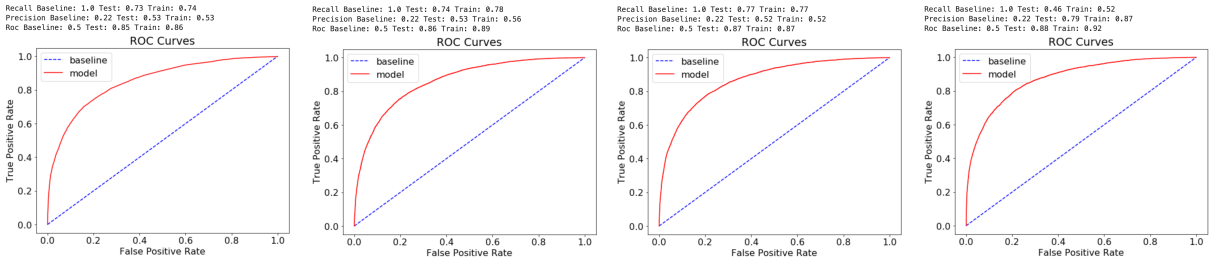
[13] https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-classification-in-python/

[14] https://www.jstor.org/stable/2531595?seq=1

[15] https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc

model	accuracy	precision (0)	precision (1)	recall (0)	recall (1)	f1-score (0)	f1-score (1)	precision (w)	recall (w)	f1-score (w)	ROC (tr)
decision tree	0.79	0.91	0.53	0.81	0.73	0.86	0.61	0.83	0.79	0.80	0.85
random forest	0.80	**0.92**	0.53	0.81	0.75	0.86	**0.62**	0.83	0.80	0.81	0.86
logistic reg.	0.79	**0.92**	0.52	0.80	**0.77**	0.86	**0.62**	0.84	0.79	0.80	0.87
XGBoost	**0.85**	0.86	**0.79**	**0.96**	0.46	**0.91**	0.58	**0.85**	**0.85**	**0.84**	**0.88**

Table 3: A model comparison using conventional methods



(a) Decision Tree

(b) Random Forest

(c) Logistic Regression

(d) XGB

Figure 1: The ROC curves of the models

4. Applying the LIME xAI framework to tabular data

In order to better understand the behaviour of our four classification models we employ the Local Interpretable Model-Agnostic Explanations (LIME) xAI framework. We start by providing a short introduction to LIME and follow on by applying LIME on our four tabular models and describing the output. Finally, we conduct a quantitative analysis of fifty aggregated LIME observations to further compare performance on a global level.

4.1. *A short introduction to LIME*

LIME is an open source framework, published by Ribeiro et al. in 2016 [8], which aims to shed light on the decision-making process of machine learning models and therewith establish trust in their usage. LIME is based on the assumption that every model is linear on a local scale. Therefore, it explains individual predictions by creating new, slightly altered data points around the real data and then applies a local linear model on it. In addition, LIME visualises the output, using coloured megapixels on image data and bar charts for tabular and text.

LIME is an acronym for Local Interpretable Model-Agnostic Explanations. *Local* means that the framework analyses specific observations. It does not give a general explanation as to why the model behaves in a certain way, but rather explains how a specific observation is categorised. *Interpretable* means that the user should be able to understand what a model does. Thus, in image classification it shows which part of the picture it considered when it comes to predictions and when working with tabular data it shows which features influence its decision. *Model-Agnostic* means that it can be applied to any blackbox algorithm we know today or that we might develop in the future. If the model is a glassbox this is not taken into consideration as LIME treats every model like a blackbox. *Explanations* denote the output, which the LIME framework produces. LIME has three core functionalities: the image explainer interprets image classification models, the text explainer provides insight into text based models[16] and the tabular explainer assesses to what extent features of a tabular dataset are considered when it comes to the classification process[17].

Listing 1: The LIME tabular explainer

```
explainer = LimeTabularExplainer(
    convert_to_lime_format(X_train,
        categorical_names).
    values,
    mode="classification",
    feature_names=X_train.columns.tolist(),
    categorical_names=categorical_names,
    categorical_features=categorical_names.keys(),
    discretize_continuous=True,
    random_state=42)
```

4.2. *The application of the LIME Tabular Explainer*

The main function that LIME offers is called the `explainer`. As LIME is model agnostic, the explanation happens exclusively on the data level, hence ignoring the

[16] https://www.tensorflow.org/lite/models/text_classification/overview

[17] https://towardsdatascience.com/pytorch-tabular-binary-classification-a0368da5bb89

process within the model. Therefore, the explainer explains predictions on tabular data by perturbing features based on the statistical properties of the training data [58]. A highlevel overview of the LIME explainer is provided below:

The convert to LIME function: Prior to being able to explain an observation, we need to convert the output into a certain format, which we do by creating a list of all possible categorical values per feature. Then, we use the `convert_to_lime_format` function [59] adopted from Kevin Lemagnen's Pycon presentation in 2019[18], as the one included in the LIME documentation only works with older versions of Python. The function converts all existing string variables to integers, such that they can be interpreted.

The explainer: The explainer itself is included in the LIME library and displayed in Listing 1. We set all parameters manually, as the explainer does not possess any default values. First we call our now formatted dataset and set the mode to classification, then we give a list of all features in our dataset (line 3) and with `categorical_names=categorical_names` we specify which of the variables are categorical (line 4), `Categorical_features` (line 5) lists the index of all features with a categorical type and `discretize_continuous` (line 6) is a mathematical function that simply helps to produce a better output by converting continuous attributes to nominal attributes. The final parameter, `random_state`, brings consistency into the function, otherwise it always picks a different number whenever we reload the function.

Displaying one observation: We choose one observation on which we apply the interpretability framework and subsequently print the classification that each model gives for this instance as well as the true label. We can now convert the output to the LIME format, saving it in the observation variable before defining a standard predict function. The `custom_predict_proba` function, is able to transform very simple models but also more complex input. It converts the data so that it is processible by the `LIMETabularExplainer`, which we carry out for every model we wish to interpret. After this we can apply the LIME framework on our classification models. To create a LIME output, we define the explanation as `explainer.explain_instance` and include the observation we chose above, adding the `lr_predict_proba` and five features as this shows us the factors considered the most influential on predicting the target variable.

Running the code presents us with the first of the four LIME outputs, displayed in Figure 2, consisting of four parts: the prediction probabilities on the left side, the feature probabilities in the center, the feature-value table on the right and the r-squared value on the bottom left. The prediction probabilities graph shows the model's decision on that instance, meaning which outcome it predicts and the corresponding probability. In our example it displays the output of the logistic regression and predicts, that it is not going to rain with 92% probability, represented by the blue bar with the number 0 and that it is going to rain with 8%, represented by the orange bar with the number 1. The feature probabilities graph gives insight into how much a feature influences the given decision. For this observation the variable *Humidity3pm* is the most influential factor and supports the prediction, that it is not going to rain tomorrow. The second most important feature is *WindGustSpeed* which weights towards that it is going to rain tomorrow, represented by the number 1. In this case, we display the top five features in our output, but theoretically all the features could be listed that way, ordered by their importance. The last graph is the feature-value table, which also sorts the features by importance, but instead of showing their weight, is given the actual value that this feature possesses in this observation. For example, the forth feature, *Temp3pm*, shows 35.60 in this table, representing 35.60 degrees Celsius, the temperature at 3pm of the day of the observation. It is coloured orange, as it is influencing the model's decision towards rain. The r-squared indicates how well the model fits the observed data and can take a value between 0 and 1, with 1 constituting a perfect fit. For this instance, the value of 0.50 indicates a moderate fit. As demonstrated in Figure 2, LIME does not differentiate between the machine learning model used but displays each of them the same way.

4.3. Evaluating the models on a global level

In order to analyse the LIME output on a global level, we apply the framework on fifty observations. For this we adopt a simple random sampling methodology [60], which is applied by utilizing a random selection function. We then aggregate the output in an excel file to compare the graphs with each other. As we analyse four models, we end up with 200 interpretations in total. Our simple random baseline approach, could be enhanced with more sophisticated sampling mechanisms, such as Submodular Pick LIME (SP-LIME), which can be used to select a diverse yet representative set of explanations.

LIME allows us to look at individual features in more detail and evaluate their influence, the occurrences of the three most relevant features are summarized in table Table 4. In our analysis the framework displays the top five features per observation resulting in 200 total feature counts and 50 top positions per model. Out of this set, *Humidity3pm* occurs most frequently, except for the XGBoost where it is ranked second after *Pressure9am*. It appears 50

[18] https://speakerdeck.com/klemag/pycon-2019-introduction-to-model-interpretability-in-python

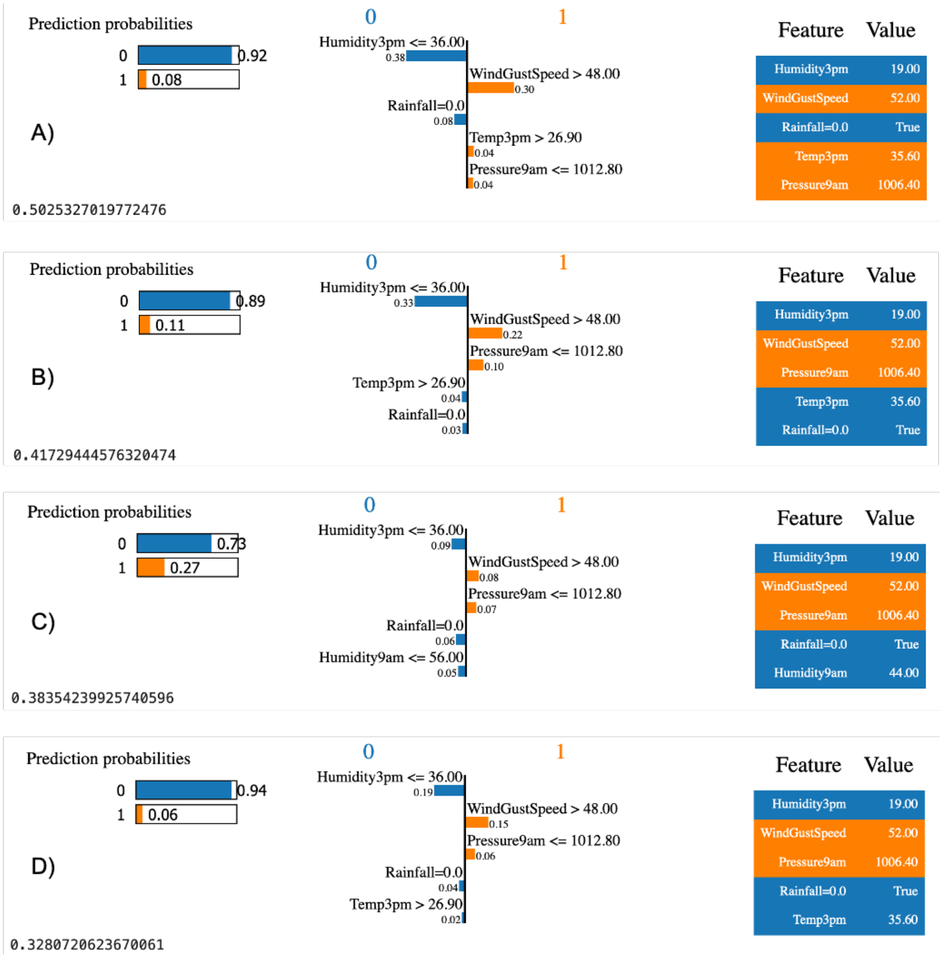


Figure 2: LIME output of the same observation from the (A) Logistic Regression, (B) Decision Tree, (C) Random Forest and (D) XGBoost

times in the analysis of the decision tree and logistic regression, 42 times at the random forest and 48 times at the XGBoost. Furthermore, *Humidity3pm* is not only the most frequent, but is also considered the most important feature, as for the logistic regression it is the most influential feature, meaning it is ranked number one, in all 50 cases and for the decision tree in 42 cases. In case of the random forest, its prediction that it is not going to rain is heavily influenced by `Rainfall`, as whenever it did not rain, it is ranked in first or second position, which happens in 22 and 11 cases, respectively. Nevertheless, *Humidity3pm* is also important for the random forest and occurs in 21 cases on the first rank. In the XGBoost classification *Humidity3pm* is considered the most important feature 38 times. The least considered features are *WindGustDir*, *RainToday* and *Temp9am*, with an occurrence of five, seven and eight times, respectively, none of which are ever ranked within the first or second position. Considering this values, we now know that *Humidity3pm* is highly predictive for our models, bringing us a step closer to developing a usable application.

By displaying the intervals of its classification, LIME enables us to evaluate the accuracy of a single prediction. In terms of a false assessment we calculate the absolute difference between the probabilities assigned to the target variables, measured in percent. This tells us by how much the prediction is wrong and results in another indicator to assess the models. The false classifications are divided into two categories: a wrong prediction with less than 20 percent of absolute difference is called a close miss and a prediction with 20 percent or over more absolute difference is called a far miss. The results are displayed in Table 5. The analysis of all observations results in the following: the decision tree classifies 12 out of 50 instances incorrectly, which are split evenly between close and far misses. The average absolute difference of all wrong classifications is 23 percent. In terms of the amount of incorrect classifications the logistic regression performs better than the decision tree, with eight wrong classifications, of which five are a close and three are a far miss. In absolute differ-

Feature	Decision Tree		Random Forest		Logistic Regression		XGBoost	
	O	TP	O	TP	O	TP	O	TP
Humidity3pm	50	42	42	21	50	50	42	38
Pressure9am	50	4	37	2	20	0	48	5
WindGustSpeed	50	4	29	4	44	11	34	4

Table 4: Summary of the most occurring (O) and highest rated (TP) features

Type	Decision Tree	Random Forest	Logistic Regression	XGBoost
Num. of close Misses (¡ 20%)	6	6	5	1
Num. of far Misses ($\geq$ 20%)	6	3	3	3
Average (in %)	23	15	26	38

Table 5: Summary of close and far misses

ence the logistic regression performs slightly worse, with around 26 percent. The random forest misclassifies nine times, of which six are close and three are far misses and gives us an average of 15 percent. Lastly, the XGBoost predicts incorrectly only four times, one time causing a close miss and three times a far miss, resulting in around 38 percent absolute difference, which is significantly lower in the times of incorrect classifications, but when it fails than by a lot more than other models.

Considering the different evaluations we conducted, XGBoost is superior in the majority of cases. With the highest accuracy of 85%, weighted classification report scores of 85% *precision*, 85% *recall*, 84% *f1-score*, a *ROC-test-baseline* of 88% and the least amount of incorrect classifications, it delivers a better performance than the other models.

5. Evaluating LIME from a usability perspective

After applying LIME on four machine learning models, and testing its local and global functions, we evaluate its usability. This usability assessment consists of two parts: firstly, we perform interviews to get an impression of how LIME is interpreted by people who are not familiar with the concept of explainable AI; secondly, we use a user experience evaluation framework in order to perform a self assessment of LIME's usability based on its criteria.

5.1. The interviews

We interviewed twelve people, equally split between male and female, six with prior knowledge of machine learning, classification models and data modelling, and six with no prior knowledge in these fields. None of them were familiar with the concept of xAI before participating in the interview. The participants were either academics or in the process of pursuing a degree and were chosen for the usability assessment based on the mentioned characteristics. In each interview we wanted to find out how interpretable the LIME output is for a person who has never worked with xAI before. The interviews, which lasted between fifteen and twenty-five minutes, were conducted using the standardised question-catalogue discussed in detail below. An overview of the interview results discussed herein is displayed in Table 6, while the set of anonymous interview notes can in turn be found in our GitHub repository[19].

The interview was split into two sections, both of which started with an explanation from the interviewer. In the first part the interviewees were given a quick introduction into rain prediction, as well as a quick introduction into the applicable machine learning methods. They were subsequently shown the first `LIMETabularExplainer` output graph (cf., Figure 3) and were asked the following four questions.

What do you see in this graph? All interviewees expressed uncertainty about what the illustrations show. All started with identifying the three graphs and tried to make sense of the different numbers. Although a few participants struggled with the prediction-probabilities and the feature-value graph, every participant had difficulties interpreting the feature probabilities as the numbers did not seem to add up and there was too much information given in a badly structured way.

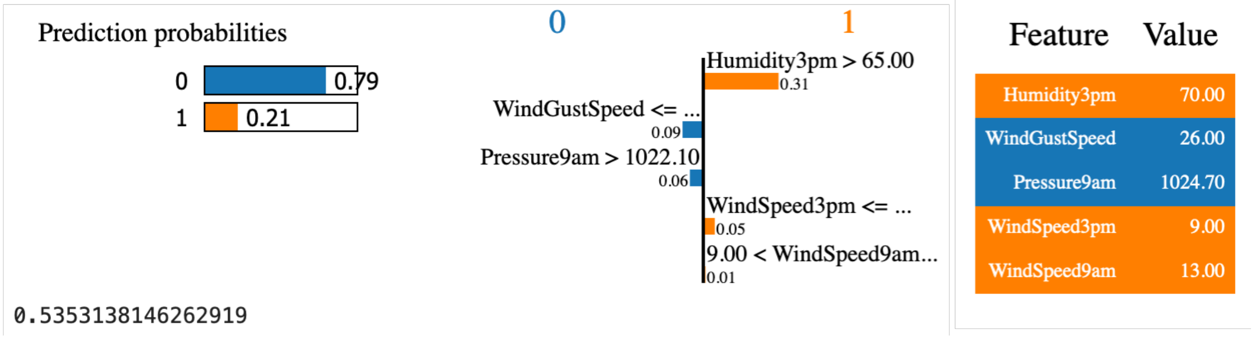
Which feature influences the prediction and how? People without prior machine learning knowledge struggled to see the relation between the prediction probabilities and the classification, but those with prior knowledge in machine learning concluded, that there is a connection between the feature probabilities and the prediction probabilities graph. Five concluded correctly, that the second smaller numbers on the central graph are probabilities, as they are between 0 and 1 and influence the predictability.

Do you know why the model made this prediction? Five out of twelve answered correctly, that the classification is determined by the numbers of the feature probabilities graph.

[19] https://github.com/jdieber/WhyModelWhy

Participant	Prior knowledge	Gender	Understood illustration	Understood prediction	Rating part I	Understanding part II	Rating part II
1	yes	m	yes	yes	3	improved	8
2	no	f	no	no	3	improved	6.5
3	no	m	no	no	4	improved	7.5
4	yes	m	yes	yes	5	improved	9.5
5	no	f	no	no	4	improved	7.5
6	yes	f	no	no	3	improved	7
7	yes	f	yes	yes	8	improved	10
8	yes	m	yes	yes	7	decreased	4
9	yes	m	no	no	5	improved	9
10	no	f	no	no	3	improved	5
11	no	m	yes	yes	4	improved	8
12	no	f	no	no	1	improved	3

Table 6: Summary of the participants' understanding of the LIME output (ratings on a scale from 1-10, increasing)



Feature	Value
Humidity3pm	70.00
WindGustSpeed	26.00
Pressure9am	1024.70
WindSpeed3pm	9.00
WindSpeed9am	13.00

```
0.5353138146262919
```

Figure 3: Example of the interview LIME output

How well can you interpret the results of the prediction of the graph, on an increasing scale from 1-10? The interpretability of the LIME output was rated with an average of 4.16. The rating between the subgroups differed significantly, as the participants without prior knowledge gave an average of 3.16 and the participants with prior knowledge 5.16, respectively.

The second section started with a short explanation of each graph of the LIME output as well as an explanation of the meaning of the r-squared value at the bottom of the output. The participants were subsequently shown another LIME output and were asked four more questions.

What do you see in the second graph? After the participants were given the explanation for each graph the answers improved significantly. Seven understood the graphs correctly, but were still uncertain where the probabilities of the prediction probabilities graph came from. Four of the participants with a machine learning background and one without understood the framework after the explanation. Another six pointed out that the r-squared scores of both models were low, which resulted in concerns about the reliability of the prediction.

How well can you interpret the results of the prediction, on an increasing scale from 1-10? Even though several remarks were made in the previous question the interpretability of the graph after the explanation improved significantly, to an average of 7.08. Participants with prior machine learning knowledge again rated it slightly higher with an average of 7.91, compared to an average of 6.25 by the participants without prior knowledge.

What differences do you see between this one and the other graph? All participants noted the different prediction probabilities. Some participants pointed out that there is a big difference on how the features in the different outputs were rated and that the numbers of the feature value graph had changed.

Is there anything that stands out as strange or unusual? Additionally, nine out of twelve participants stated that the central graph was not very interpretable and four mentioned that they found the choice of colours disturbing. Furthermore, six interviewees suggested a legend, titles or a short explanation should be included in the output visualisation to improve its interpretability.

To sum up, the results produced by the framework are difficult to understand without documentation and/or explanation. Although the participants with a background in machine learning were more effective in terms of interpreting the explanation produced by LIME, usability assessments such as the one described in this paper could be used to significantly improve the user experience.

5.2. *Self assessment of the usability*

To assess LIME's user experience more broadly, we adopt the definition of usability proposed by the International Organisation for Standardisation (ISO)[20] in their ISO 9241-11:2018 report [61]. Therein, usability is defined as the *"extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use"* [61]. As this definition is too broad to be directly applied in our evaluation context, we improve its applicability by taking into consideration the *"New ISO Standards for Usability, Usability Reports and Usability Measures"* produced by Bevan et al. [62] and the *"Usability Meanings and Interpretations in ISO Standards"* guidelines provided by Abran et al. Abran et al. [63]. Combined they constitute our custom made assessment framework, called Model Usability Evaluation (MUsE).

5.2.1. *How effective is LIME in terms of achieving model interpretability?*

In terms of effectiveness, Bevan et al. [62] state that *"effectiveness has been associated with completing a task completely and accurately, but it is also important to take account of the potential negative consequences if the task is not achieved correctly"*. From this we extract three effectiveness factors: measure of completion; measure of accuracy; and negative consequences to rate effectiveness. Abran et al. [63] take a more holistic perspective questioning *"how well do users achieve their goal using the system?"*. Thus, we use both the standard and the guidelines in order to develop four UX effectiveness questions tailored specifically to LIME, and subsequently use them to perform our assessment:

(a) How complete is the explanation on a local level? LIME is a local explainability framework, therefore it calculates the influence of every feature and its importance on a local level (i.e., this is done for each prediction). Nevertheless, the connection between the prediction probabilities and the feature probability graph is incomplete as currently only the feature importance score is shown. Additionally, these scores do not add up to the prediction probabilities. As displayed in Figure 3, the feature *Humidity3pm* with a feature probabilities score of 0.31 alone exceeds the total prediction probability of 0.21 that it is going to rain, while the overall classification was in favor of no rain. This can only be explained by assuming that the displayed prediction probabilities are not the sum of the feature probabilities, but the result of another calculation not obvious to a user.

(b) How complete is the explanation on a global level? While LIME is generally used for local interpretability, in this paper we also assess its performance on a global level. It is not surprising that the `LIMETabularExplainer` is less effective globally, as it does not include a function or interface to allow a global evaluation. Thus, we extract several observation outputs manually and analyse them in an Excel file, as we did in the global analysis of Section 5. Considering the importance of global interpretability and the effectiveness of the simple proof of concept presented in this paper, it would be beneficial to: (i) implement performance indicators that allow for a global comparison with other models; and/or (ii) add a function to extract the local outputs of several random observations as a spreadsheet, so the user can calculate indicators necessary for a global comparison themselves.

(c) Could accurate results be misinterpreted? The interpretations of the local predictions appear to be accurate. But we see a risk of misinterpretation when it comes to the tabular explainer, as no comprehensive explanation of it has been published yet [58]. Therefore, we have to rely on third party explanations like online articles[21] [22] or talks on YouTube[23][24]. Ideally such guidance should be incorporated into the LIME documentation.

(d) What negative consequences arise from a misinterpretation? In case of a misinterpretation of the LIME evaluation the severity of the negative consequences depends on the use-case. For example the implication of the predictions produced by our rain prediction model for Australia and an automated defense system [64] differ greatly. In our case a mistake in the interpretation could lead to a faulty feature importance and therefore a wrong rain forecast. In the automated defense system case an incorrect classification could put lives at risk. As the severity of the consequences is not determined by the developers of LIME but rather lies in the hands of the users, reducing the risk that a misinterpretation occurs should be one of the key evaluation criteria when it comes to usability assessments.

[20] https://www.iso.org/home.html

[21] https://medium.com/analytics-vidhya/explain-your-model-with-lime-5a1a5867b423

[22] https://www.oreilly.com/content/introduction-to-local-interpretable-model-agnostic-explanations-lime/

[23] https://www.youtube.com/watch?v=CY3t11vuuOM

[24] https://www.youtube.com/watch?v=C80SQe16Rao

5.2.2. What resources are consumed in order to achieve interpretability?

In order to evaluate resource efficiency Bevan et al. [62] identify the following factors: task time, time efficiency, cost-effectiveness, productive time ratio, unnecessary actions and fatigue. We aggregate them to a list with mutually exclusive components and conclude with the question raised by Abran et al. *"What resources are consumed in order to achieve the goal?"* [63].

(a) How much time does it take to use LIME? Both, the time to set up LIME as well as the time to analyse the output play a role in this context. The setup works well, however the official `LIMETabularExplainer` setup documentation relates to several old packages[25]. Therefore, the initial process of applying the original notebook and trying to find workarounds consumed a lot of time. Additionally, the analysis of the LIME output took a considerable amount of time, as the documentation of the graphs is non-transparent as stated in the effectiveness evaluation. On the up-side, the time it takes to compute and display an observation is minimal.

(b) What other costs are involved? As LIME is an open source tool, no licensing costs are involved and also the publications, documents and videos to understand the tool (where available) are can be freely accessed.

(c) Does this process cause fatigue? Applying LIME to only a few observations can be performed quickly and therefore is not costly from a performance perspective. However, the global interpretation was a tedious process, which entailed hours of repetitive manual work copying and pasting LIME output from the notebook into an Excel file. Also, given that there is no benchmark on the number of observations necessary to evaluate the models globally it is not clear how many outputs are necessary/sufficient.

5.2.3. How satisfying is the application of LIME?

Satisfaction is the least standardised of the three parameters as it is highly dependent on the user and use-case [62]. Based on Bevan et al. satisfaction aims to take *"positive attitudes, emotions and/or comfort resulting from use of a system, product or service"* [62] into account. The question Abran et al. raise to assess satisfaction is *"How well does the user feel about the use of the system?"* [63], which we include in our analysis. Combining both ideas we come up with the following assessment questions:

(a) Do we have a positive or negative attitude towards the tool? At the start of the implementation our attitude was very positive, as LIME's serves to help users to interpret and trust predictions performed by blackbox algorithms. During the setup our attitude deteriorated due to a lack of documentation and support, which posed an even bigger problem during the analysis. LIME gives insight into a model's processes, but here again it takes a lot of effort to get a clear understanding of the framework, which has a negative influence on our attitude. Naturally, once we learned how to apply and interpret LIME, the process was a much more pleasant one.

(b) What emotions arise from using it? The lack of clear and explicit guideline makes understanding LIME a frustrating process. However, reaching the point of a better overall understanding of our classification models raises positive feelings. Especially LIME's short processing time makes it easy to evaluate several instances in a row, which leads to a very pleasant user experience.

(c) How satisfying is the final result? The output of the `LIMETabularExplainer` unquestionably helps to understand the model's classification process, as it offers insights conventional methods can not provide, which causes satisfaction. However, this satisfaction could be increased by eliminating doubt about the relationships between the local indicators and offering a global analysis.

6. Conclusions

Motivated by the lack of limited evaluation of existing post model interpretability tools, in this paper, we evaluated the UX effectiveness of the LIME framework, via both a usability study and a structured self assessment analysis. In particular, we examined the performance of four state of the art classification algorithms on a tabular dataset that is used to predict rain; applied the `LIMETabularExplainer` to analyse single observations on a local level; and used a random sampling approach in order to evaluate the models on a global level. In order to assess the interpretability of the output produced by LIME, we conducted interviews with individuals who had no prior experience with LIME. Whereas, in order to examine the usability of LIME, more generally, we developed a usability assessment framework, Model Usability Evaluation (MUsE), derived from the ISO 9241-11:2018 standard.

Based on our analysis we conclude that LIME could be further enhanced via self explanatory data visualisations, better support for global interpretability, improved documentation, and contextualised accuracy and reliability insights that limit the potential for negative consequences. Additionally, we can conclude that the visualisations provided by LIME is more suitable for users who already have experience working with classification algorithms. Indicating that post model interpretability tools need to consider how best to present their findings to various stakeholder

[25] https://lime-ml.readthedocs.io/en/latest/lime.html

groups (i.e., developers, theorists, ethicists, and users). Some initial insights with respect the the requirements of the various stakeholders are provided by Preece et al. [65] and Tomsett et al. [66]. Taking a broader perspective on usability, there are a number of surveys that focus on usability, from an analysis [67], a design [68], and an evaluation perspective [69] that could provide be used to inform post model interpretability tool enhancement.

When it comes to verification and validation, more generally, there is a need for additional metrics and methodologies that go beyond the baseline evaluations and user interviews that are normally used to evaluate post model interpretability tools. Here researchers have surveyed tools and techniques that can be used to evaluate the effectiveness of machine learning applications [70], expert systems [71], and cyber physical systems [72], to name but a few, that could potentially be used to inform verification and validation for xAI.

From an impact perspective, considering the lack of formal metrics for assessing the effectiveness xAI proposals in general, MUsE, which has been derived from the ISO 9241-11:2018 standard and usability guidelines provided by Bevan et al. [62] and Abran et al. [63], could serve as a means to examine the usability of various post model interpretability tools, and to compare them to one another.

In terms of future work, interviewing experienced LIME users on their user experience with LIME would add another valuable perspective to the usability study. Additionally, an in-depth performance evaluation of LIMEs tabular explainer could close a gap in current research. Besides proposing strategies for improving the interpretability of the output produced by LIME, and the usability of the framework from a global level perspective, we are interested in using MUsE to benchmark alternative model-agnostic explanation frameworks.

References

[1] P. McCorduck, M. Minsky, O. G. Selfridge, H. A. Simon, History of artificial intelligence, in: Proceedings of the 5th International Joint Conference on Artificial Intelligence. Cambridge, MA, USA, August 22-25, 1977, 1977, pp. 951–954. URL: `http://ijcai.org/Proceedings/77-2/Papers/083.pdf`.

[2] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al., Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai, Information Fusion 58 (2020) 82–115.

[3] P. W. Koh, P. Liang, Understanding black-box predictions via influence functions, 2017. `arXiv:1703.04730`.

[4] G. Montavon, W. Samek, K.-R. Müller, Methods for interpreting and understanding deep neural networks, Digital Signal Processing 73 (2018) 1–15.

[5] N. Papernot, P. McDaniel, Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning, 2018. `arXiv:1803.04765`.

[6] M. Hind, D. Wei, M. Campbell, N. C. F. Codella, A. Dhurandhar, A. Mojsilović, K. N. Ramamurthy, K. R. Varshney, Ted: Teaching ai to explain its decisions, 2018. `arXiv:1811.04896`.

[7] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, D. Pedreschi, A survey of methods for explaining black box models, ACM computing surveys (CSUR) 51 (2018) 1–42.

[8] M. Ribeiro, S. Singh, C. Guestrin, ”why should i trust you?”: Explaining the predictions of any classifier, in: In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 97–101. doi:`10.18653/v1/N16-3020`.

[9] S. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, 2017. `arXiv:1705.07874`.

[10] G. Plumb, D. Molitor, A. S. Talwalkar, Model agnostic supervised local explanations, in: Advances in Neural Information Processing Systems, 2018, pp. 2515–2524.

[11] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 618–626.

[12] L. Hu, J. Chen, V. N. Nair, A. Sudjianto, Locally interpretable models and effects based on supervised partitioning (lime-sup), arXiv preprint arXiv:1806.00663 (2018).

[13] S. Mishra, B. L. Sturm, S. Dixon, Local interpretable model-agnostic explanations for music content analysis, in: ISMIR, 2017, pp. 100–110.

[14] M. R. Zafar, N. M. Khan, Dlime: a deterministic local interpretable model-agnostic explanations approach for computer-aided diagnosis systems, arXiv preprint arXiv:1906.10263 (2019).

[15] A. Nguyen, A. Dosovitskiy, J. Yosinski, T. Brox, J. Clune, Synthesizing the preferred inputs for neurons in neural networks via deep generator networks, in: Advances in neural information processing systems, 2016, pp. 3387–3395.

[16] A.-H. Karimi, G. Barthe, B. Belle, I. Valera, Model-agnostic counterfactual explanations for consequential decisions, arXiv preprint arXiv:1905.11190 (2019).

[17] S. Sharma, J. Henderson, J. Ghosh, Certifai: Counterfactual explanations for robustness, transparency, interpretability, and fairness of artificial intelligence models, arXiv preprint arXiv:1905.07857 (2019).

[18] J. Lei, M. G'Sell, A. Rinaldo, R. J. Tibshirani, L. Wasserman, Distribution-free predictive inference for regression, Journal of the American Statistical Association 113 (2018) 1094–1111.

[19] G. Casalicchio, C. Molnar, B. Bischl, Visualizing the feature importance for black box models, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2018, pp. 655–670.

[20] R. Khanna, B. Kim, J. Ghosh, O. Koyejo, Interpreting black box predictions using fisher kernels, arXiv preprint arXiv:1810.10118 (2018).

[21] A. Dhurandhar, T. Pedapati, A. Balakrishnan, P.-Y. Chen, K. Shanmugam, R. Puri, Model agnostic contrastive explanations for structured data, arXiv preprint arXiv:1906.00117 (2019).

[22] M. T. Ribeiro, S. Singh, C. Guestrin, Nothing else matters: model-agnostic explanations by identifying prediction invariance, arXiv preprint arXiv:1611.05817 (2016).

[23] N. Puri, P. Gupta, P. Agarwal, S. Verma, B. Krishnamurthy, Magix: Model agnostic globally interpretable explanations, arXiv preprint arXiv:1706.07160 (2017).

[24] G. J. Katuwal, R. Chen, Machine learning model interpretability for precision medicine, arXiv preprint arXiv:1610.09045 (2016).

[25] K. Sokol, P. Flach, Explainability fact sheets, Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (2020). URL: `http://dx.doi.org/10.1145/3351095.3372870`. doi:`10.1145/3351095.3372870`.

[26] J. Singh, A. Anand, Exs: Explainable search using local model agnostic interpretability, in: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, 2019, pp. 770–773.

[27] R. Guidotti, A. Monreale, S. Ruggieri, D. Pedreschi, F. Turini, F. Giannotti, Local rule-based explanations of black box decision systems, arXiv preprint arXiv:1805.10820 (2018).

[28] T. Peltola, Local interpretable model-agnostic explanations of bayesian predictive models via kullback-leibler projections, arXiv preprint arXiv:1810.02678 (2018).

[29] S. Tan, R. Caruana, G. Hooker, Y. Lou, Detecting bias in black-box models using transparent model distillation, arXiv preprint arXiv:1710.06169 (2017).

[30] S. García, A. Fernández, J. Luengo, F. Herrera, A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability, Soft Computing 13 (2009) 959.

[31] D. P. Green, H. L. Kern, Modeling heterogeneous treatment effects in large-scale experiments using bayesian additive regression trees, in: The annual summer meeting of the society of political methodology, 2010, pp. 100–110.

[32] J. Elith, J. R. Leathwick, T. Hastie, A working guide to boosted regression trees, Journal of Animal Ecology 77 (2008) 802–813.

[33] J. Singh, A. Anand, Model agnostic interpretability of rankers via intent modelling, in: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, 2020, pp. 618–628.

[34] L. Arras, F. Horn, G. Montavon, K.-R. Müller, W. Samek, " what is relevant in a text document?": An interpretable machine learning approach, PloS one 12 (2017).

[35] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, K.-R. MÃžller, How to explain individual classification decisions, Journal of Machine Learning Research 11 (2010) 1803–1831.

[36] M. D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: European conference on computer vision, Springer, 2014, pp. 818–833.

[37] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2921–2929.

[38] M. Sundararajan, A. Taly, Q. Yan, Axiomatic attribution for deep networks, in: Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017, pp. 3319–3328.

[39] R. C. Fong, A. Vedaldi, Interpretable explanations of black boxes by meaningful perturbation, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 3429–3437.

[40] P. Dabkowski, Y. Gal, Real time image saliency for black box classifiers, in: Advances in Neural Information Processing Systems, 2017, pp. 6967–6976.

[41] P. Cortez, M. J. Embrechts, Opening black box data mining models using sensitivity analysis, in: 2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM), IEEE, 2011, pp. 341–348.

[42] S. Lundberg, S.-I. Lee, An unexpected unity among methods for interpreting model predictions, arXiv preprint arXiv:1611.07478 (2016).

[43] J. Chen, L. Song, M. J. Wainwright, M. I. Jordan, L-shapley and c-shapley: Efficient model interpretation for structured data, arXiv preprint arXiv:1808.02610 (2018).

[44] C. Frye, I. Feige, C. Rowat, Asymmetric shapley values: incorporating causal knowledge into model-agnostic explainability, arXiv preprint arXiv:1910.06358 (2019).

[45] O. Bastani, C. Kim, H. Bastani, Interpretability via model extraction, arXiv preprint arXiv:1706.09773 (2017).

[46] J. J. Thiagarajan, B. Kailkhura, P. Sattigeri, K. N. Ramamurthy, Treeview: Peeking into deep neural networks via feature-space partitioning, arXiv preprint arXiv:1611.07429 (2016).

[47] H. Lakkaraju, E. Kamar, R. Caruana, J. Leskovec, Interpretable & explorable approximations of black box models, arXiv preprint arXiv:1707.01154 (2017).

[48] W. Gale, L. Oakden-Rayner, G. Carneiro, A. P. Bradley, L. J. Palmer, Producing radiologist-quality reports for interpretable artificial intelligence, 2018. arXiv:1806.00340.

[49] A. Holzinger, G. Langs, H. Denk, K. Zatloukal, H. Müller, Causability and explainability of artificial intelligence in medicine, WIREs Data Mining and Knowledge Discovery 9 (2019) e1312. doi:10.1002/widm.1312.

[50] J. Morgan, J. Sonquist, Problems in the analysis of survey data, and a proposal, Journal of the American Statistical Association 58 (1963) 415–434.

[51] T. K. Ho, Random decision forests, in: Proceedings of 3rd International Conference on Document Analysis and Recognition, volume 1, 1995, pp. 278–282 vol.1. doi:10.1109/ICDAR.1995.598994.

[52] J. Berkson, Application of the logistic function to bio-assay, Journal of the American statistical association 39 (1944) 357–365.

[53] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, pp. 785–794.

[54] S. V. Stehman, Selecting and interpreting measures of thematic classification accuracy, Remote sensing of Environment 62 (1997) 77–89.

[55] J. Fan, S. Upadhye, A. Worster, Understanding receiver operating characteristic (roc) curves, Canadian Journal of Emergency Medicine 8 (2006) 19–20.

[56] S. R. Safavian, D. Landgrebe, A survey of decision tree classifier methodology, IEEE transactions on systems, man, and cybernetics 21 (1991) 660–674.

[57] D. L. Streiner, J. Cairney, What's under the roc? an introduction to receiver operating characteristics curves, The Canadian Journal of Psychiatry 52 (2007) 121–128. URL: https://doi.org/10.1177/070674370705200210. arXiv:https://doi.org/10.1177/070674370705200210.

[58] M. T. Ribeiro, Lime tabular package, https://lime-ml.readthedocs.io/en/latest/lime.html, 2016. URL: https://lime-ml.readthedocs.io/en/latest/lime.html, accessed: 2020-04-18.

[59] K. Lemagnen, helpers.py, https://github.com/charlespwd/project-titlehttps://github.com/klemag/PyconUS_2019-model-interpretability-tutorial/blob/master/helpers.py, 2019.

[60] A. S. Acharya, A. Prakash, P. Saxena, A. Nigam, Sampling: Why and how of it, Indian Journal of Medical Specialties 4 (2013) 330–333.

[61] I. O. for Standardisation, Iso 9241-11:2018(en) ergonomics of human-system interaction — part 11: Usability: Definitions and concepts, 2018. URL: https://www.iso.org/obp/ui/#iso:std:iso:9241:-11:ed-2:v1:en.

[62] N. Bevan, J. Carter, J. Earthy, T. Geis, S. Harker, New iso standards for usability, usability reports and usability measures, in: New ISO Standards for Usability, Usability Reports and Usability Measures, volume 9731, 2016, pp. 268–278. doi:10.1007/978-3-319-39510-4_25.

[63] A. Abran, A. Khelifi, W. Suryn, A. Seffah, Usability meanings and interpretations in iso standards, Software Quality Journal 11 (2003) 325–338. doi:10.1023/A:1025869312943.

[64] D. Gunning, D. Aha, Darpa's explainable artificial intelligence (xai) program, 2019. URL: https://www.aaai.org/ojs/index.php/aimagazine/article/view/2850.

[65] A. Preece, D. Harborne, D. Braines, R. Tomsett, S. Chakraborty, Stakeholders in explainable ai, arXiv preprint arXiv:1810.00184 (2018).

[66] R. Tomsett, D. Braines, D. Harborne, A. Preece, S. Chakraborty, Interpretable to whom? a role-based model for analyzing interpretable machine learning systems, arXiv preprint arXiv:1806.07552 (2018).

[67] A. Følstad, E. Law, K. Hornbæk, Analysis in practical usability evaluation: a survey study, in: proceedings of the SIGCHI conference on human factors in computing systems, 2012, pp. 2127–2136.

[68] E. Folmer, J. Bosch, Architecting for usability: a survey, Journal of systems and software 70 (2004) 61–78.

[69] D. A. Bowman, J. L. Gabbard, D. Hix, A survey of usability evaluation in virtual environments: classification and comparison of methods, Presence: Teleoperators & Virtual Environments 11 (2002) 404–424.

[70] S. Masuda, K. Ono, T. Yasue, N. Hosokawa, A survey of software quality for machine learning applications, in: 2018 IEEE International conference on software testing, verification and validation workshops (ICSTW), IEEE, 2018, pp. 279–284.

[71] R. M. O'Keefe, D. E. O'Leary, Expert system verification and validation: a survey and tutorial, Artificial Intelligence Review 7 (1993) 3–42.

[72] X. Zheng, C. Julien, Verification and validation in cyber physical systems: Research challenges and a way forward, in: 2015 IEEE/ACM 1st International Workshop on Software Engineering for Smart Cyber-Physical Systems, IEEE, 2015, pp. 15–18.

11. Blockchain based resource governance for decentralized web environments

Bibliographic Information

Basile, D., Di Ciccio, C., Goretti, V. and **Kirrane, S.**, 2023. Blockchain based Resource Governance for Decentralized Web Environments. Frontiers in Blockchain: Blockchain for Trusted Information Systems. To appear.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Writing - Original Draft, Writing - Review & Editing, and Supervision.

Copyright Notice

Blockchain based Resource Governance for Decentralized Web Environments

Davide Basile [1], **Claudio Di Ciccio** [1], **Valerio Goretti** [1] **and Sabrina Kirrane** [2]

[1] *Sapienza University of Rome, Italy*
[2] *Vienna University of Economics and Business, Austria*

Correspondence*:
Valerio Goretti
valerio.goretti@uniroma1.it

ABSTRACT

Decentralization initiatives such as Solid, Digi.me, and ActivityPub aim to give data owners more control over their data and to level the playing field by enabling small companies and individuals to gain access to data, thus stimulating innovation. However, these initiatives typically employ access control mechanisms that cannot verify compliance with usage conditions after access has been granted to others. In this paper, we extend the state of the art by proposing a resource governance conceptual framework, entitled ReGov, that facilitates usage control in decentralized web environments. We subsequently demonstrate how our framework can be instantiated by combining blockchain and trusted execution environments. Through blockchain technologies, we record policies expressing the usage conditions associated with resources and monitor their compliance. Our instantiation employs trusted execution environments to enforce said policies, inside data consumers' devices. We evaluate the framework instantiation through a detailed analysis of requirements derived from a data market motivating scenario, as well as an assessment of the security, privacy, and affordability aspects of our proposal.

Keywords: Decentralization; Usage Control; Governance; Blockchain; Trusted Execution Environment

1 INTRODUCTION

Since its development, the internet has steadily evolved into a ubiquitous ecosystem that is seen by many as a public utility (Quail and Larabie, 2010). The development of centralized web-based platforms on top of the internet has undoubtedly brought benefits from both an economic and a social perspective. However, the web as we know it today, is dominated by a small number of stakeholders that have a disproportionate influence on the content that the public can produce and consume. The scale of the phenomenon has brought about the need for legal initiatives aimed at safeguarding content producer rights (Quintais, 2020). In parallel, technical decentralization initiatives such as Solid[1], Digi.me[2], and ActivityPub[3] aim to give data owners more control over their data, while at the same time providing small companies as well as individuals with access to data, which is usually monopolized by centralized platform providers, thus stimulating innovation. To this end, the Solid community are developing tools, best practices, and web standards that facilitate ease of data integration and support the development of decentralized social applications based on Linked Data principles. In turn, Digi.me are developing tools and technologies

[1] https://solidproject.org/about. Accessed: Wednesday 12th April, 2023.

[2] https://digi.me/what-is-digime/. Accessed: Wednesday 12th April, 2023.

[3] https://activitypub.rocks/. Accessed: Wednesday 12th April, 2023

that enable individuals to download their data from centralized platforms such that they can store it in an encrypted personal data store and leverage a variety of applications that can process this data locally on the data owners device. These client-side applications are developed by innovative app developers who use the Digi.me software development kit to communicate with the encrypted personal data stores directly. Following the same principles, ActivityPub is a decentralized social networking protocol, published by the W3C Social Web Working Group that offers a client-server application programming interface (API) for adding, modifying, and removing material as well as a federated server-server API for sending notifications and subscribing to content. Social networks implementing ActivityPub can be easily integrated with each other in order to form a larger ecosystem, commonly referred to as the Fediverse[4]. Some of the most popular Fediverse initiatives include Mastodon[5], PeerTube[6], and PixelFed[7].

In order to better cater for use case scenarios that involve data sharing across various distributed data stores underpinning decentralized applications, there is a need for tools and technologies that are not only capable of working with distributed data but are also able to manage data resources that come with a variety of usage terms and conditions specified by data producers. However, the vast majority of decentralized web initiatives, which aim to provide users with a greater degree of control over personal resources, manage data access via simple access control mechanisms (Ouaddah et al., 2016; Toninelli et al., 2006; Tran et al., 2005) that are not able to verify that usage conditions are adhered to after access has been granted (Akaichi and Kirrane, 2022b). For example, access control rules can determine if users can retrieve data or not. However, they cannot express conditions on the type of application that can process them, the geographical area in which they can be treated, when the access grant would expire, or the number of times they can be processed.

When it comes to the realization of usage control in decentralized web environments, Trusted Execution Environments (TEEs) and Distributed Ledger Technologies (DLTs) could serve as fundamental enablers. Trusted execution environments offer data and code integrity to enforce the conditions established by decentralized data providers, directly in consumers' devices. DLTs can store shared policies in a distributed ecosystem in which data usage is governed by smart contracts, while recording an immutable log of usage operations.

To this end, in this paper we propose a resource governance (ReGov) conceptual framework and an instantiation thereof. ReGov combines blockchain applications and trusted execution environments to facilitate usage control in decentralized web environments. The work is guided by a typical decentralized web scenario, according to which data are not stored in centralized servers but rather in decentralized data stores controlled by users. Throughout the paper, we refer to the component for managing the data stored locally on every user's device as a *data node* (or *node* for simplicity).

In terms of contributions, we extend the state of the art by: (i) proposing a generic resource governance conceptual framework; (ii) demonstrating how blockchain technologies and trusted execution environments can together be used to manage resource usage; and (iii) assessing the effectiveness of the proposed framework via concrete quantitative and qualitative evaluation metrics derived from our data market motivating use case scenario.

[4] https://fediverse.party/en/fediverse/. Accessed: Wednesday 12th April, 2023.

[5] https://docs.joinmastodon.org. Accessed: Wednesday 12th April, 2023.

[6] https://peertube.uno. Accessed: Wednesday 12th April, 2023.

[7] https://pixelfed.uno/site/about. Accessed: Wednesday 12th April, 2023.

The remainder of the paper is structured as follows: Section 2 presents the necessary background information regarding data access and usage control, trusted execution environments, decentralized applications, and blockchain oracles. In the same section we also provide an overview of related work. We introduce the motivating scenario used to guide our work in Section 3 and our ReGov conceptual framework in Section 4. Following on from this, we described our DLT and TEE-based instantiation in Section 5 and the results of our quantitative and qualitative in Section 6. Finally, we conclude and outline directions for future work in Section 7.

2 BACKGROUND AND RELATED WORK

This section sets the context for the work being presented, highlighting the significance and relevance of the study. It also gives credit to previous work in the field and identifies gaps in the current understanding that the study aims to fill.

2.1 Background

As we leverage blockchain technologies and trusted execution environments to manage resource usage control, in the following we provide the necessary background information from these fields.

2.1.1 Data Access and Usage Control

Access control is a technique used to determine who or what can access resources in a computing environment (Sandhu and Samarati, 1994). In system infrastructures, access control is dependent upon and coexists alongside other security services. Such technologies require the presence of a trusted reference entity that mediates any attempted access to confidential resources. In order to decide who has rights to specific resources, access control frameworks make use of authorization rules, typically stored inside the system (Koshutanski and Massacci, 2003). A set of rules constitutes a policy. A popular approach of implementing access policies is through Access Control Lists (ACLs) (Grünbacher, 2003). Each protected resource has an associated ACL file, which lists the rights each subject in the system is allowed to use to access objects.

With the evolution of the web and decentralized data ecosystems, there is the need to move beyond managing access to resources via authorizations (Akaichi and Kirrane, 2022b). Authorization predicates define limitations that consider the user and resource credentials and attributes. Usage control is an extension of access control whereby policies take into account obligations and conditions in addition to authorizations (Lazouski et al., 2010). Obligations are constraints that must be fulfilled by users before, during, or after resource usage. Conditions are environmental rules that need to be satisfied before or during usage.

One of the most highly cited usage control models is UCON$_{ABC}$ (Park and Sandhu, 2004). The model represents policy rules by defining specific rights (e.g., operations to be executed) related to sets of subjects (e.g., users who want to perform an operation), objects (e.g., the resource to operate), authorizations, obligations, and conditions. *Attributes* are properties associated with subjects or objects. UCON$_{ABC}$ improves conventional access control mainly through the following two concepts: (i) *attribute mutability*, namely the change of attributes as a consequence of usage actions, and (ii) *decision continuity*, i.e., the enforcing of policies not only as a check at access request time, but also during the subsequent resource usage. Systems implementing usage control through the UCON$_{ABC}$ model require dedicated infrastructure to guarantee policy enforcement and monitoring in order to detect misconduct and execute compensation actions (e.g., penalties and/or right revocations).

The literature offers several alternative approaches that could potentially be used to represent usage control policies. For instance, Hilty et al. (2007) propose a language named Obligation Specification Language (OSL) intended for distributed environments. Bonatti et al. (2020) introduce the SPECIAL usage control policy language, which considers a policy as the intersection of basic entities governing data, processing, purposes, location, and storage of personal data. A comprehensive overview of existing usage control frameworks and their respective languages is provided by Akaichi and Kirrane (2022b) and Esteves and Rodríguez-Doncel (2022).

The overarching goal of our work is to enable usage control in a decentralized environment. We provide a conceptual framework that serves as a blueprint for policy governance in a decentralized setting.

2.1.2 Trusted Execution Environments

A Trusted Execution Environment (TEE) is a tamper-proof processing environment that runs on a separation kernel (McGillion et al., 2015). Through the combination of both software and hardware features, it isolates the execution of code from the operating environment. The separation kernel technique ensures separate execution between two environments. TEEs were first introduced by Rushby (1981) and allow multiple systems requiring different levels of security to coexist on one platform. Thanks to kernel separation, the system is split into several partitions, guaranteeing strong isolation between them. TEEs guarantee the authenticity of the code it executes, the integrity of the runtime states, and the confidentiality of the code and data stored in persistent memory. The content generated by the TEE is not static, and data are updated and stored in a secure manner. Thus, TEEs are hardened against both software and hardware attacks, preventing the use of even backdoor security vulnerabilities (Sabt et al., 2015). There are many providers of TEE that differ in terms of the software system and, more specifically, the processor on which they are executed. In this work, we make use of the Intel Software Guard Extensions (Intel SGX)[8] TEE. Intel SGX is a set of CPU-level instructions that allow applications to create *enclaves*. An enclave is a protected area of the application that guarantees the confidentiality and integrity of the data and code within it. These guarantees are also effective against malware with administrative privileges (Zheng et al., 2021). The use of one or more enclaves within an application makes it possible to reduce the potential attack surfaces of an application. An enclave cannot be read or written to from outside. Only the enclave itself can change its secrets, independent of the Central Processing Unit (CPU) privileges used. Indeed, it is not possible to access the enclave by manipulating registers or the stack. Every call made to the enclave needs a new instruction that performs checks aimed at protecting the data that are only accessible through the enclave code. The data within the enclave, in addition to being difficult to access, is encrypted. Gaining access to the Dynamic Random Access Memory (DRAM) modules would result in encrypted data being obtained (Jauernig et al., 2020). The cryptographic key changes randomly each time the system is rebooted following a shutdown or hibernation (Costan and Devadas, 2016). An application using Intel SGX consists of a trusted and an untrusted component. We have seen that the trusted component is composed of one or more enclaves. The untrusted component is the remaining part of the application (Zhao et al., 2016). The trusted part of the application has no possibility of interacting with any other external components except the untrusted part. Nevertheless, the fewer interactions between the trusted and untrusted part, the greater the security guaranteed by the application.

Our work resorts to trusted execution environments to keep control of resources' utilization by enforcing the usage conditions set by data owners.

[8] https://www.intel.co.uk/content/www/uk/en/architecture-and-technology/software-guard-extensions.html. Accessed: Wednesday 12th April, 2023.

2.1.3 Decentralized Applications and Blockchain Oracles

With second-generation blockchains, the technology evolved from being primarily an e-cash distributed management system to a distributed programming platform for decentralized applications (DApps) (Mohanty, 2018). Ethereum first enabled the deployment and execution of smart contracts (i.e., stateful software artifacts exposing variables and callable methods) in the blockchain environment through the Ethereum Virtual Machine (EVM) (Buterin et al., 2014). The inability of smart contracts to access data that is not stored on-chain restricts the functionality of many application scenarios, including multi-party processes. Oracles solve this issue (Xu et al., 2016).

Oracles act as a bridge for communication between the on-chain and off-chain worlds. This means that DApps should also be able to trust an oracle in the same way it trusts the blockchain. Reliability for oracles is key (Mammadzada et al., 2020; Al-Breiki et al., 2020a). Therefore, the designation and sharing of a well-defined protocol become fundamental for the proper functioning of the oracle's service, particularly when the oracles themselves are organized in the form of networks for the interaction with decentralized environments (Basile et al., 2021). As illustrated by Mühlberger et al. (2020), oracle patterns can be described according to two dimensions: the information direction (inbound or outbound) and the initiator of the information exchange (pull- or push-based). While outbound oracles send data from the blockchain to the outside, inbound oracles inject data into the blockchain from the outside. Pull-based oracles have the initiator as the recipient, oppositely to push-based oracles, where the initiator is the transmitter of the information. By combining the push-/pull-based and inbound/outbound categories, four oracle design patterns can be identified (Pasdar et al., 2022). A push-based inbound oracle (*push-in* oracle for simplicity) is employed by an off-chain component that sends data from the real world. The push-based outbound (*push-out*) oracle is used when an on-chain component starts the procedure and transmits data to off-chain components. The pull-based outbound (*pull-out*) oracle is operated by an off-chain component that wants to retrieve data from the blockchain. Finally, the pull-based inbound (*pull-in*) oracle enables on-chain components to retrieve information outside the blockchain.

We leverage the blockchain's tamper-proof infrastructure to record usage conditions associated with resources. We represent this information via smart contracts running in the blockchain and communicating with off-chain processes through oracles.

2.2 Related work

Several works strive to provide more control and transparency with respect to personal data processing by leveraging blockchain distributed application platforms (Xu et al., 2019). For instance, Ayoade et al. (2018) defines an access control mechanism for IoT devices that stores a hash of the data in a blockchain infrastructure and maintains the raw information in a secure storage platform using a TEE. In the proposed framework, a blockchain based ledger is used in order to develop an audit trail of data access that provides more transparency with respect to data processing. Xiao et al. (2020) propose a system, called PrivacyGuard, which gives data owners control over personal data access and usage in a data market scenario.

The literature offers numerous study cases in which usage control frameworks have been instantiated to increase the degree of privacy and confidentiality of shared data. Neisse et al. (2011) propose a usage control framework in which a Policy Enforcement Point (PEP) keeps track of business operations and intercepts action requests while taking into consideration Policy Decision Point event subscriptions (PDP). Bai et al. (2014) addresses usage control in a Web Of Thing environment by adapting the UCON model for Smart Home ecosystems. Zhaofeng et al. (2020) introduce a secure usage control scheme for Internet of things (IoT) data that is built upon a blockchain-based trust management approach. While, Khan et al. (2020) conceptualizes a distributed usage control model, named DistU, for industrial blockchain frameworks with monitoring procedures that are able to revoke permissions automatically.

Additionally, there are several papers that propose frameworks or architectures that combine blockchain platforms and decentralized web initiatives such as Solid web. Ramachandran et al. (2020) demonstrate how together Solid data stores (namely, *pods*) and blockchains can be used for trustless verification with confidentiality. Patel et al. (2019) propose a fully decentralized protocol named DAuth that leverages asymmetric encryption in order to implement authentication. Cai et al. (2020) introduce a secure Solid authentication mechanism, integrating Rivest–Shamir–Adleman (RSA) signatures into permissioned blockchain systems. In turn, Becker et al. (2021) demonstrate how data stored in Solid pods can be monetized by leveraging a blockchain based payment system. Whereas, Havur et al. (2020) discuss how solid could potentially leverage existing consent, transparency and compliance checking approaches.

Several studies have shown that blockchain and TEEs can profitably coexist. The state of the art proposes numerous cases where the combination of the two technologies leads to advantages in terms of data ownership, availability, and trust. One of these is the work of Liang et al. (2017), that propose a patient-centric personal health data management system with accountability and decentralization. The architecture of the framework employs TEEs to generate a fingerprint for each data access that are immutably maintained by a blockchain infrastructure. Whereas, Lind et al. (2017) designed and implemented a protocol named Teechain that integrates off-chain TEEs for secure and scalable payment procedures, built on top of the Bitcoin blockchain platform.

3 MOTIVATING SCENARIO AND REQUIREMENTS

The motivating use case scenario and the corresponding requirements, discussed in this section, are used not only to guide our work but also to contextualize theoretical notions introduced in the paper.

3.1 Motivating Scenario

A new decentralized data market called DecentralTrading aims to facilitate data access across decentralized data stores. Alice and Bob sign up for the DecentralTrading market, pay the subscription fee, and set up their data nodes. Alice is a research biologist in the area of marine science and is conducting studies on deep ocean animals. Such species are difficult to identify due to the adverse conditions of their ecosystem and the lack of good-quality images. Bob is a professional diver with a passion for photography. He has collected several photos from his last immersion and the most scientifically relevant of them portrays a recently discovered whale species named ‘Mesoplodon eueu’ showed in Fig. 1.

Bob shares his photos with the DecentralTrading market by uploading them to his data node. Once the images are shared, they can be retrieved by the other participants in the market. Moreover, he wants to establish rules regarding the usage of his images. Table 1 illustrates the constraints he exerts on the data utilization, along with the **rule type** they represent (inspired by the work of Akaichi and Kirrane, 2022a). Bob makes his images available only for applications belonging to the scientific domain (this constraint belongs to the type of **domain rules**). Moreover, he sets geographical restrictions by making the images usable only by devices located in European countries (**geographical rule**). Finally, Bob wants his photos to be deleted after a specific number of application accesses (**access counter rule**) or after a specific time interval (**temporal rule**). Therefore, he sets a maximum number of 100 local accesses and an expiry date of 20 days after the retrieval date. Bob gets remuneration from the DecentralTrading market, according to

Table 1. Schematization of the usage policy associated with Bob's 'Mesoplodon.jpg' image. Every rule belongs to a rule type and consists of a subject, an action, an object, and a constraint.

Rule components / Rule type	Subject	Action	Object	Constraint
Domain rule	market members	access the resource	Mesoplodon.jpg	The resource can be processed only by research apps
Geographical rule	market members	access the resource	Mesoplodon.jpg	The resource can be loaded only in European countries
Temporal rule	market members	access the resource	Mesoplodon.jpg	The resource can be stored for up to 20 days
Access counter rule	market members	access the resource	Mesoplodon.jpg	The resource can be opened up to 100 times

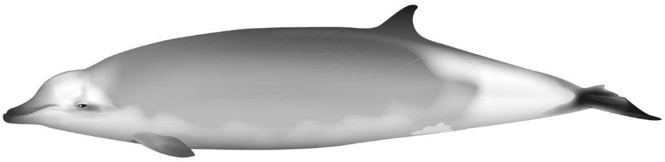


Figure 1. A photographic representation of a Mesoplodon eueu (Carroll et al., 2021)

the number of requests for his resources. At any point in time, Bob can ask the DecentralTrading market to get evidence that the rules associated with his image are being adhered to and check if there were attempts to use his image outside the specified rules.

Bob's images of the Mesoplodon eueu species could be extremely useful for Alice's research, so she requests a specific picture of the gallery through her DecentralTrading node. Alice's node obtains a URL for Bob's node from the market and subsequently contacts Bob's node in order to retrieve a copy of the image, which is stored in a protected location of her device alongside the related usage rules. Data shared in DecentralTrading is used by Alice and Bob through a set of known applications approved by the market community. Alice opens the image through an app called 'ZooResearch', which is used for the analysis of zoological images. 'ZooResearch' belongs to the set of approved applications, and it disables some tasks for data duplication by the operating system (OS) such as screenshots to replicate the image once it is accessed. Since the domain of the application corresponds with the usage constraint set by Bob and her device is located in Ireland, the action is granted by Alice's node. Afterwards, Alice tries to share the image through a social network application named 'Socialgram', which also belongs to the set of supported applications. Then, Alice's node denies the action since it goes against the application domain constraint set by Bob. Alice opens the image through 'ZooResearch' 99 more times and, following the last attempt, the image is deleted from her node since the maximum number of local accesses of 100 has been reached. Therefore, Alice asks her DecentralTrading node to retrieve the image from Bob's node again. Since Alice starts working on a different research project, she stops using the Mesoplodon eueu's image. The image remains stored in the protected location of Alice's node until 20 days from the retrieval date have passed. Subsequently, Alice's node deletes the image from the protected location.

3.2 Requirements

The following concrete requirements are derived from our motivating scenario. The two top level requirements, which are inspired by the seminal work of Akaichi and Kirrane (2022b), are subdivided into more concrete sub-requirements.

(R1) Resource utilization and policy fulfillment must be managed by trusted entities. According to Akaichi and Kirrane (2022b), a usage control framework must provide an enforcement mechanism that ensures usage policies are adhered to both before and after data are accessed. Therefore, the data market must be able to able to handle the access control and additionally the nodes of a decentralized environment must be equipped with a dedicated component managing the utilization of resources owned by other nodes.

(R1.1) The trusted entity must be able to store resources obtained from other entities. Once resources are accessed, they must be kept in a trusted memory zone directly controlled by the trusted entity. This requirement drastically reduces the risks of data theft or misuse. Considering our running example, it allows Alice to not only store Bob's resources but also to protect them from unauthorized access.

(R1.2) The trusted entity must support the execution of programmable procedures that enforce constraints associated with resource usage. Specific procedures must be designed in order to cater for the various usage policy rules types. The trusted entity must execute these procedures in order to enforce policies and control resource utilization. This aspect enables the logic associated with usage control rules, such as those defined in Table 1, to be executed when Alice tries to use Bob's image.

(R1.3) Resources and procedures managed by the trusted entity must be protected against malicious manipulations. The trusted entity must guarantee the integrity of the resources it manages alongside the logic of the usage control procedures. Therefore, Alice should not be able to perform actions that directly manipulate Bob's image or corrupt the logic of the mechanisms that govern its use.

(R1.4) The trusted entity must be able to prove its trusted nature to other entities in a decentralized environment. Remote resource requests must be attributable to a trusted entity of the decentralized environment. Therefore, prior to Bob sending his image to Alice, it must be possible to verify that the data request has actually been generated by Alice's trusted node.

(R2) Policy compliance must be monitored via the entities of a governance ecosystem. According to Akaichi and Kirrane (2022b), usage control frameworks must incorporate a policy monitoring component. The monitoring, performed through one or more services, enables nodes to detect misconduct and unexpected or unpermitted usage. This is, e.g., the mechanism thanks to which Bob can verify that Alice has never tried to open the picture of the Mesoplodon eueu with Socialgram.

(R2.1) The governance ecosystem must provide transparency to all the nodes of the decentralized environment. In order to gain the trust of the various nodes that comprise a decentralized environment, a governance ecosystem must guarantee transparency with respect to its data and procedures. This feature enables Bob to verify at any time that the usage policy associated with his image is being adhered to.

(R2.2) Data and metadata maintained by the governance ecosystem must be tamper-resistant. Once policies and resource metadata are sent to the governance ecosystem, their integrity must be ensured. The inability to tamper with resources and their metadata is crucial for the effective functioning of the governance ecosystem. Therefore, when Bob publishes images and their respective usage policies in the market, his node should be the only entity capable of modifying this metadata.

(R2.3) The governance ecosystem and the entities that the form part of the ecosystem must be aligned with the decentralization principles. It is essential that the governance ecosystem itself respects

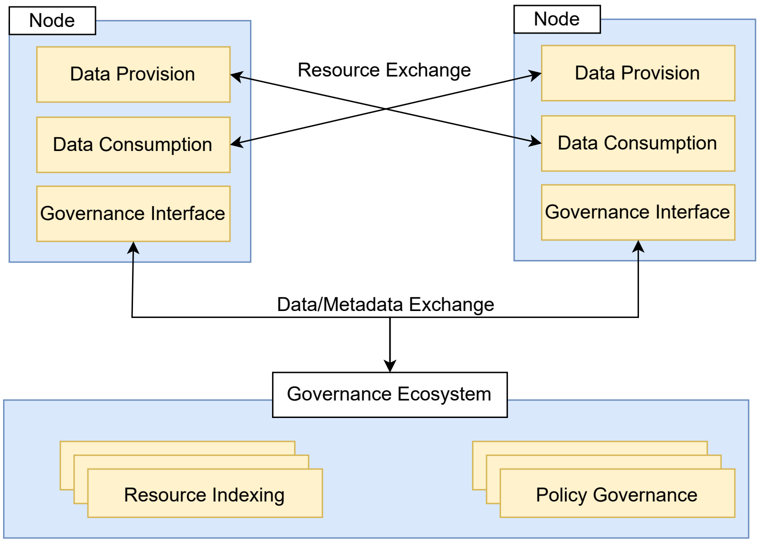


Figure 2. High-level overview of the proposed conceptual resource governance (ReGov) framework.

the decentralization principles, as centralized solutions would establish a central authority in which data and decisional power are accumulated. Hence, the monitoring functionality provided by the previously mentioned market scenario should not rely on centralized platforms and data stores. Bob's policies for the usage of the Mesoplodon eueu's photo are not uploaded on, nor verified by, any third-party service running on a specific server.

(R2.4) The entities that form part of the governance ecosystem must be able to represent policies and verify their observance. In order to provide monitoring functionality, entities in the governance ecosystem should be capable of managing usage policies. These entities should enact procedures for retrieving policy observance information directly from nodes that consume market resources. This feature allows Bob to obtain evidence that Alice is using his image according to the rules stipulated in the usage policy and to detect any misbehavior.

4 CONCEPTUAL RESOURCE GOVERNANCE FRAMEWORK

To cater for our motivating scenario and to meet the derived requirements, we propose a conceptual framework, named ReGov, that enables the governance of usage policies in decentralized web environments. ReGov generalizes the principles of data ownership and control, which constitute the foundations of numerous decentralized web initiatives. The ReGov framework extends these aspects by not only controlling data access but also supporting the continuous monitoring of compliance with usage policies and enforcing the fulfillment of usage policy obligations. The degree of abstraction of the ReGov framework means that it could potentially be instantiated in numerous decentralized web contexts.

4.1 ReGov Framework Entities

According to the decentralization concept, the web is a peer-to-peer network with no central authority. In this scenario, data are no longer collected in application servers, but rather data are managed by nodes that are controlled by users (i.e., data owners determine who can access their data and in what context). Nodes communicate directly with other nodes in order to send and retrieve resources via the decentralized environment.

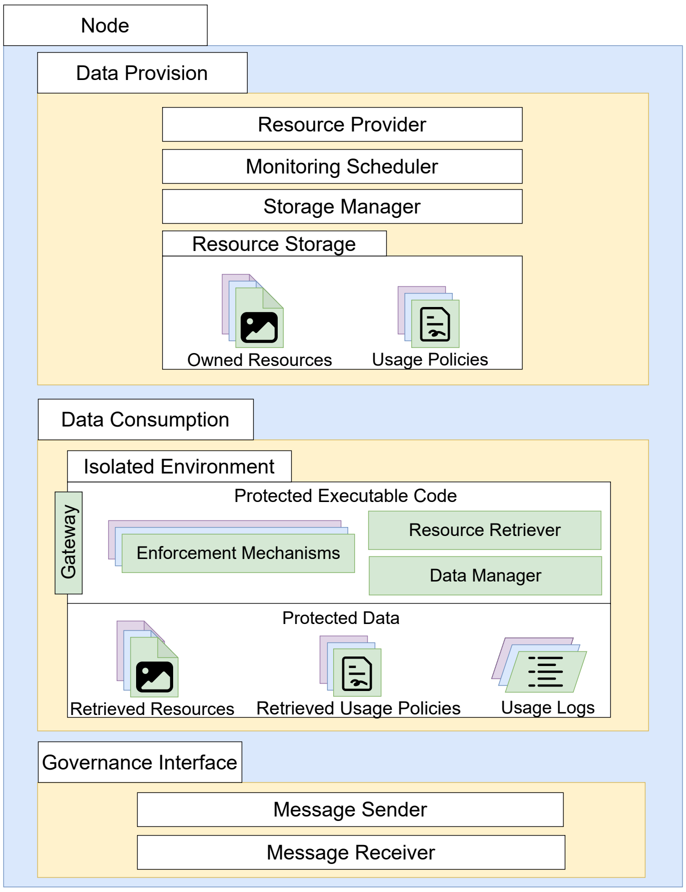


Figure 3. Content of the data provision, data consumption and governance interface components.

Figure 2 depicts a high-level overview diagram of the ReGov framework. Nodes are characterized by the Data Provision, Data Consumption, and Governance Interface components. Governance ecosystems are responsible for indexing web resources, facilitating node and resource discovery, and monitoring resource usage. Thus, in our architecture, a Governance Ecosystem is constituted by the Resource Indexing and Policy Governance components.

4.1.1 Components of a Node

A Node is a combination of hardware and software technologies, running on user devices. As shown in Fig. 3, each Node comprises the following components.

Data provision. The Data Provision component encapsulates the functionality that enable node owners to manage the sharing of their resources with other nodes in the decentralized environment. Users can interact with the Storage Manager to manually upload their data to the Resource Storage that is encapsulated within the Data Provision component. The upload operation also facilitates the definition of usage rules that are collected in usage policies associated with resources. Usage policies are represented in a machine-readable format (e.g., SPECIAL[9] and LUCON[10] policy languages) and stored in the Data Provision component alongside the resources. Additionally, when a new resource is uploaded, the Storage Manager forwards these rules and resource references to the Governance Ecosystem. In

[9] https://ai.wu.ac.at/policies/policylanguage/. Accessed: Wednesday 12th April, 2023.

[10] https://industrial-data-space.github.io/trusted-connector-documentation/docs/usage_control/. Accessed: Wednesday 12th April, 2023.

order to deliver the stored resources, the Data Provision component offers the logic for a Resource Provider that is capable of processing requests that allow other nodes to retrieve data. A data request must contain the necessary information to perform the authentication of the sender node. Therefore, the Resource Provider is able to authenticate resource requests to decide whether to grant or deny access to the requested resource based on the identity of the sender. Several web service protocols could potentially be used to implement the functionality offered by the Resource Provider (e.g., HTTP, FTP, Gopher). Once data are delivered, node owners can plan sessions to monitor the utilization of provisioned resources through the Monitoring Scheduler, which periodically forwards monitoring requests to the Governance Ecosystem.

Referring to our running example, Bob uses the functionality of the Storage Manager inside the Data Provision component to upload the images to his Node. During the upload, he specifies the location where the images must be stored and the rules composing the images' Usage Policy (i.e. the image must be deleted 20 days after the retrieval date, the image can only be used in European countries). Therefore, these pieces of information are delivered to the Governance Ecosystem. The HTTP web service implementing the Resource Provider of Bob's Node enables him to make his resource available to the other participants of the DecentralTrading market. The web service authenticates the requests for his images to determine whether the sender has the rights to access the resource. Finally, Bob can schedule monitoring sessions through the Monitoring Scheduler, in order to get evidence of the usage of his images by other nodes.

Data consumption. The Data Consumption component groups the functionalities that enable nodes to retrieve and use data in the network. Data Consumption is built upon both hardware and software techniques that ensure the protection of sensitive data through an Isolated Environment that guarantees the integrity and confidentiality of protected data and executable code. The Isolated Environment contains the logic of a Resource Retriever that creates authenticable requests for data residing in other nodes. The Resource Retriever supports multiple web protocols (e.g., HTTP, FTP, Gopher) according to the implementation of the Resource Provider inside the Data Provision component. Therefore, if the Resource Provider is implemented as an FTP web service, the Request Retriever must be able to generate authenticable FTP requests. Once resources are retrieved alongside the related usage policies, they are controlled by the Data Manager that stores them in the Isolated Environment. To get access to a protected resource, local applications running in the Node must interact with the Data Manager via the Gateway, which acts as a bridge to the processes running in the Isolated Environment. The Gateway is similarly employed when the Resource Retriever demands new resources from other nodes. In turn, Enforcement Mechanisms governing data utilization are necessary to apply the rules of the usage policies. While controlling resources, the Data Manager cooperates with these mechanisms enabling the rules contained in the usage policies to be enforced. Each operation involving the protected resources is recorded in dedicated usage logs whose administration is entrusted by the Data Manager too. Usage logs facilitate policy monitoring procedures that employ these registers to detect potential misconduct.

As shown in the motivating scenario, Alice uses the Data Consumption component to get Bob's images, which she keeps in her own Node. During the resource retrieval process, the Resource Retriever of Alice's Data Consumption component directly communicates with the Data Provision component of Bob's Node through the Gateway. After the retrieval, the image and the associated policy are maintained in the Isolated Environment and governed by the Data Manager. Considering the geographical rule, when Alice tries to open Bob's image with a local application, the app interacts with the Gateway, which in turn, creates a communication channel with the Data Manager. The latter generates the execution of

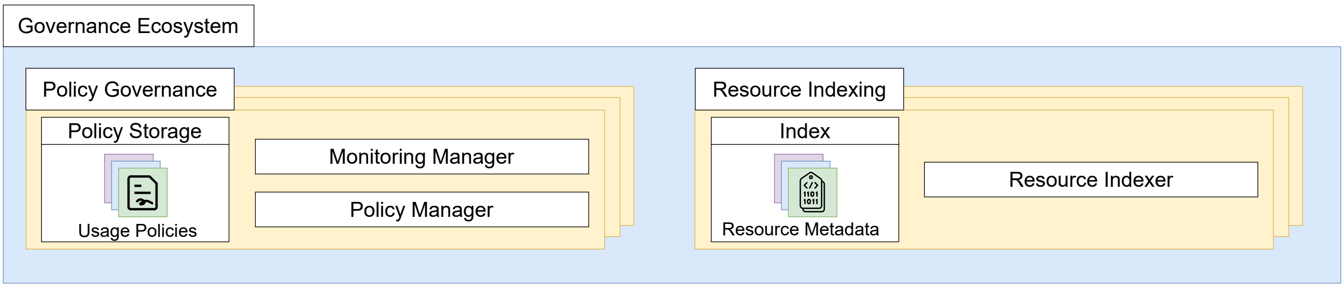


Figure 4. Content of policy governance and resource indexing components inside the governance ecosystem

the `Enforcement Mechanism` of the geographical constraint. This mechanism consults the image's usage policy, retrieves the current geographical position of the `Node`, and decides whether to grant the action.

Governance interface. Nodes facilitate communication with the `Governance Ecosystem` via the `Governance Interface`. As we will see in Section 4.2.2, messages flowing through the `Governance Interface` are crucial for resource usage monitoring. Indeed, the `Governance Ecosystem` can forward the interface messages such as requests for usage logs by remotely interacting with the `Message Receiver`. When a new message is received, the `Governance Interface` interacts with the other components of the `Node` in order to deliver the information. Similarly, the `Data Provision` and `Data Consumption Components` make use of the `Message Sender` to transmit data to the `Governance Ecosystem`. In order to provide continuous communication, the `Governance Interface` must constantly be active and listening for new messages.

4.1.2 Components of the Governance Ecosystem

We extend the typical decentralized model by including the `Governance Ecosystem`, illustrated in Fig. 4. The ecosystem hosts the `Resource Indexing` and `Policy Governance` components, whose multiple instances are able to immutably store data and metadata, execute procedures, and communicate with all the nodes of the decentralized environment.

Policy governance. `Policy Governance` components provide shared `Policy Storage` in which data owners publish applicable usage policies associated with resources. Policies are uploaded and modified through the `Policy Manager` of the component. In addition to their storage capabilities, `Policy Governance` components are able to execute procedures for policy monitoring. This function is supported by the `Monitoring Manager` of the component, containing the logic to verify the compliance of the policies stored inside the `Policy Storage`. Therefore, nodes forward monitoring requests to the `Monitoring Manager` which keeps track of resource usage and detects any illicit behavior.

Resource indexing. Policies are associated with resources through `Resource Indexing` components. They contain metadata about the resources shared in the decentralized environment (e.g., identifiers, web references, owner node). When data owners upload new resources in their node, it interacts with the `Resource Indexer` of these components, in order to serialize the information of the shared data.

Referring to our running example, when Bob uploads his image to his `Node` and specifies the corresponding usage rules in its policy, his `Node` shares the image metadata (e.g., the HTTP reference `https://BobNode.com/images/Mesoplodon.jpg`) and the usage policy with respectively the `Resource Indexing` and `Policy Governance` components running in the `Governance Ecosystem`. After Bob has delivered his 'Mesoplodon.jpg' image to Alice's `Node`, he can demand the verification of the image's

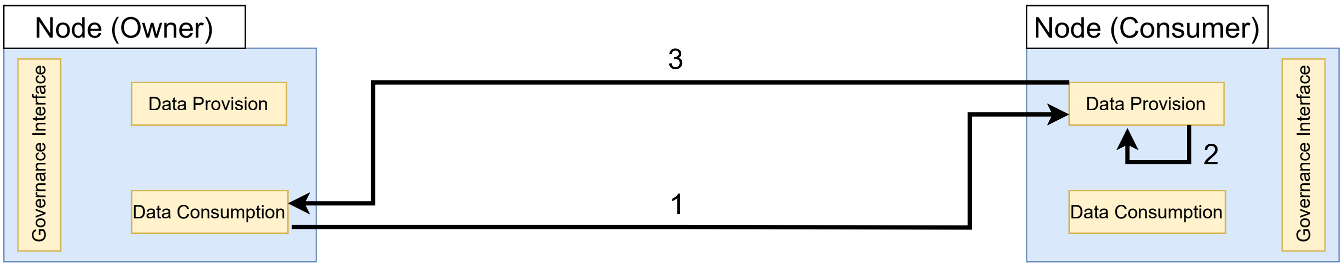


Figure 5. Visualization of the ReGov framework data retrieval process.

utilization to the Policy Governance component holding the image's policy. The Policy Governance component retrieves the usage log of the image from Alice's device, by interacting with her Node. Finally, Alice's usage can be verified based on the content of the usage log.

4.2 Predominant ReGov Framework Operations

Now that we have introduced the entities of our ReGov framework, we detail the predominant framework operations: data retrieval and monitoring. In the following, we simplify the processes by distinguishing owner nodes (i.e., nodes that are assuming the role of data providers) from data consumer nodes (i.e., nodes that are requesting access to and using resources), however, in practice, all nodes are dual purpose.

4.2.1 Data Retrieval

The data retrieval process allows consumer nodes to retrieve a resource from the decentralized environment. Figure 5 depicts a diagram representing the process. In order to obtain a specific resource, the data consumer Node generates a new request and sends it to the owner Node. We assume the consumer Node already has the information needed to contact the owner node (e.g., IP address or web reference). This information can be obtained by reading resource metadata maintained by Resource Indexing components running in the governance ecosystem. The process starts when the Resource Retriever inside the Data Consumption component of the consumer Node formats the request specifying the resource to be accessed and additional parameters intended for verification purposes. Subsequently, the request leaves the Isolated Environment through the Gateway and is received by the Resource Provider inside the Data Provision component of the owner node (**1**). The latter uses the parameters of the request to verify the identity of the sender Node (**2**). At this stage, the Resource Provider also verifies that the request has been generated in the Isolated Environment of a Data Consumption technology. Requests generated by alternative technologies are rejected. Once verified, the Resource Provider decides whether to grant access to the resource, according to the identity of the sender Node. If access is granted, the resource provider interacts with the Storage Manager inside the Data Provision component in order to construct the response, which includes both the requested resource and its usage policy. Finally, the Resource Retriever of the consumer Node obtains the resource, stores it in the Isolated Environment and registers it with the local Data Manager (**3**), as described in Section 4.1.1.

4.2.2 Monitoring

The policy monitoring process is used to continuously check if usage policies are being adhered to. In Fig. 6, we schematize the monitoring procedure. The owner node initiates the process via a scheduled job. Therefore the Monitoring Scheduler in the Data Provision component employs the Message Sender of the Governance Interface (**1**) to send a monitoring request, regarding a specific resource, to a Policy Governance component running in the Governance Ecosystem (**2**). Subsequently, the Policy

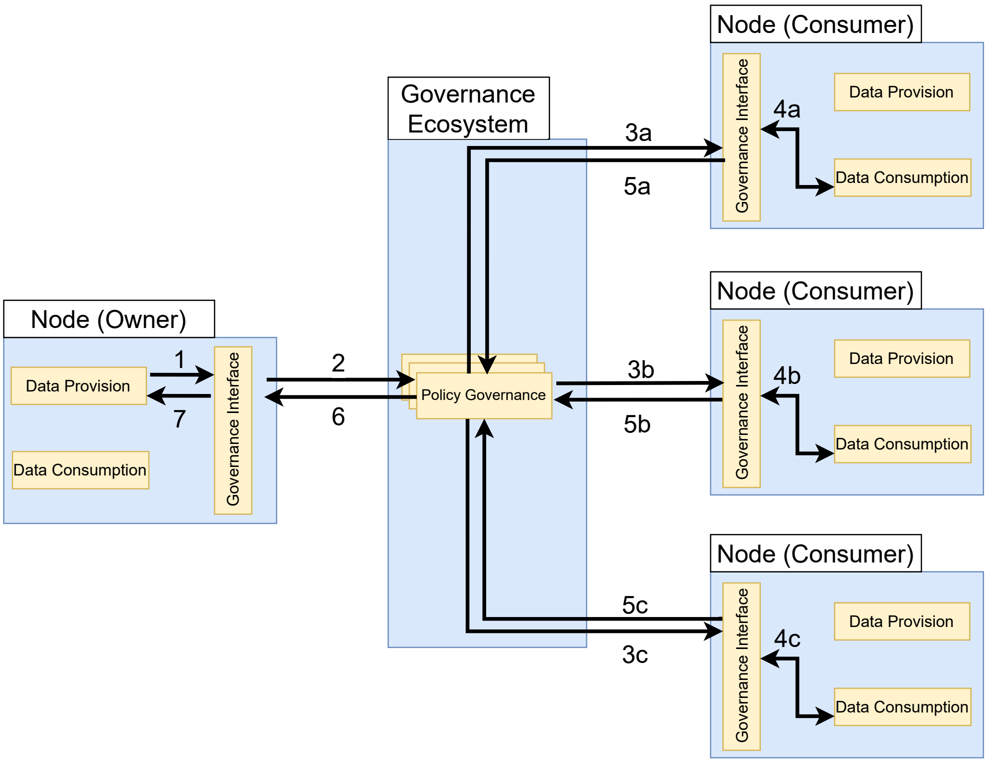


Figure 6. Visualization of the ReGov framework data monitoring routine.

`Governance` component forwards the request to provide evidence of utilization to each consumer `Node` that has a copy of the resource (**3a, 3b, 3c**). In the depicted monitoring routine, we assume the resource whose usage must be monitored is held by three consumer nodes. In each of these nodes, the monitoring request is received by the `Message Receiver` of the `Governance Interface` that forwards, in turn, the request to the `Data Manager` running in the `Isolated Environment` inside the `Data Consumption` component (**4a, 4b, 4c**). The latter retrieves the usage log from the protected data storage and employs the `Message Sender` of the `Governance Interface` to forward the information to the `Governance Ecosystem`, which in turn ensures that all the consumer node responses are collected (**5a, 5b, 5c**). Finally, the evidence are returned to the `Message Receiver` (**6**) of the initiator `Node`, which delivers the information to the `Monitoring Scheduler` (**7**).

5 BLOCKCHAIN AND TRUSTED EXECUTION ENVIRONMENT INSTANTIATION

In this section, we describe an instantiation of the ReGov framework. To this end, we propose a prototype implementation of the DecentralTrading data market illustrated in the motivating scenario. The implementation integrates a trusted application running in a trusted execution environment and blockchain technologies to address usage control needs. The code is openly available at the following address: `https://github.com/ValerioGoretti/UsageControl-DecentralTrading`.

In Fig. 7, we visualize the architecture of our ReGov framework instantiation. As shown in Section 4, the general framework assumes nodes of the decentralized environment are characterized by separate components dealing with `Data Provision` and `Data Consumption`. The `Data Provision` functionality is implemented in a software component we refer to as a `Personal Online Datastore`. We leverage security guarantees offered by the `Intel SGX Trusted Execution Environment` in order to implement

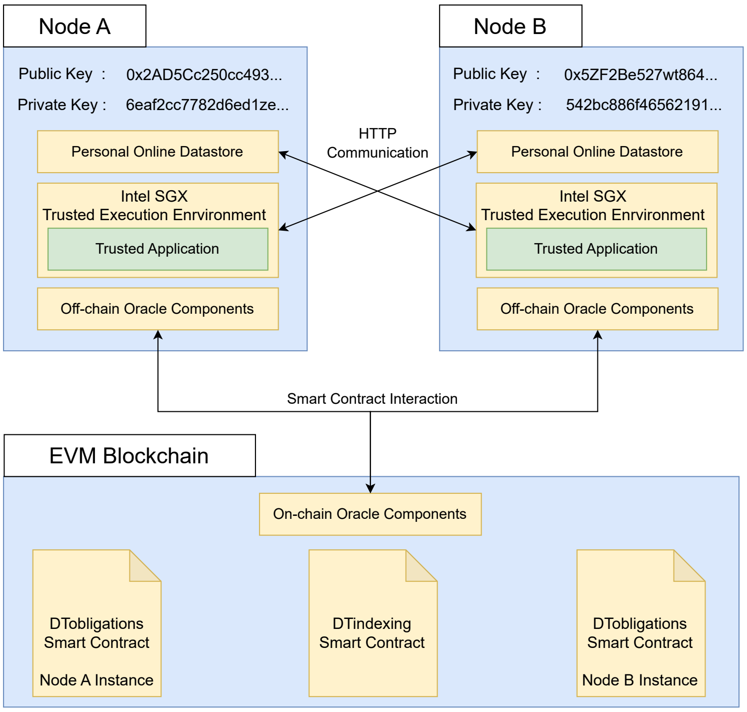

Node A
Public Key : 0x2AD5Cc250cc493...
Private Key : 6eaf2cc7782d6ed1ze...
Personal Online Datastore
Intel SGX
Trusted Execution Enrvironment
Trusted Application
Off-chain Oracle Components
HTTP
Communication
Node B
Public Key : 0x5ZF2Be527wt864...
Private Key : 542bc886f46562191...
Personal Online Datastore
Intel SGX
Trusted Execution Enrvironment
Trusted Application
Off-chain Oracle Components
Smart Contract Interaction
EVM Blockchain
On-chain Oracle Components
DTobligations
Smart Contract
Node A Instance
DTindexing
Smart Contract
DTobligations
Smart Contract
Node B Instance

Figure 7. High-level architectural overview of our ReGov framework instantiation.

a `Trusted Application` containing the logic for `Data Consumption`. The `Governance Ecosystem` is realized by developing blockchain smart contracts that store information and execute distributed procedures. Our implementation involves an `EVM Blockchain`[11] (i.e., a blockchain based on the Ethereum Virtual Machine) which hosts the `DTindexing` and `DTobligations` smart contracts. They fulfill the functions of the `Resource Indexing` and `Policy Governance` components of the general framework, respectively. `DTindexing` is characterized by a unique instance managing the resource metadata of the decentralized environment. Instead, `DTobligations` is designed to be deployed multiple times. Therefore, each `Node` is associated with a specific instance of this smart contract that stores the rules for its resources. The tasks performed by the `Governance Interface` are executed by blockchain oracles that provide a communication channel between the blockchain and the nodes of the decentralized environment. Oracles consist of `On-Chain` components, running in the `EVM Blockchain`, and `Off-Chain` components, operating within each `Node`. We built the resource retrieval process between nodes using the HTTP communication standard. By interacting with smart contracts, nodes exchange metadata necessary for resource indexing and monitoring procedures.

Our implementation employs the asymmetric encryption methodology that underlies the `EVM Blockchain`, in order to provide an authentication mechanism for the environment's nodes. Each `Node` is uniquely related to a public and private key pair that is used to sign authenticable data requests and transactions that transmit information to the blockchain and execute smart contract functions. A private key is a 256-bit number generated through a secure random number generator. The corresponding public key is derived from the private key through the Elliptic Curve Digital Signature Algorithm (Johnson et al., 2001). The public key is connected to a unique account address on the `EVM Blockchain` derived as a 160-bit segment of the hash digest of the public key. In our setting, `Node`s store their private key in an encrypted format to increase the degree of confidentiality of this information.

[11] Ethereum Virtual Machine (EVM): https://ethereum.org/en/developers/docs/evm/. Accessed: Wednesday 12th April, 2023.

In the following, we describe the technical details of the individual aspects of our implementation. In particular, we focus on features inherent to resource governance (data retrieval, enforcement, and monitoring) and avoid the implementation details related to the data market logic (e.g., subscription payments and remuneration mechanisms).

5.1 Usage Policy Instantiation

The first step of the instantiation process involves the definition of rule types that are used to stipulate usage policies. While our approach allows for a wide range of rules, we establish a specific subset of rules to demonstrate the capabilities of our ReGov framework. In particular, we propose four types of rules inspired by the work of Akaichi and Kirrane (2022a). Each rule assumes that the target resource has already been retrieved and stored on the consumer device. In the following, we explain the various rule types that have already been introduced in the motivating scenario detailed in Section 3.1.

Temporal rules. Through a temporal rule, data owners establish the maximum time a resource can be maintained within a consumer device. The rule is parameterized through an integer value representing the duration in seconds. Once the term expires, the rule stipulates that the resource must be deleted.

Access counter rules. An access counter rule specifies a maximum number of local accesses that can be executed for a specific resource, after which, the resource must be deleted. The rule is parameterized with an integer value that defines the maximum number of accesses.

Domain rules. The domain rule represents the purpose for which a resource can be opened. It is characterized by an integer value that identifies groups of applications that share the same domain. Known applications that are part of the domain group can execute local access to the resource.

Geographical rules. A geographical constraint is a limitation on where a resource can be used. It is indicated by an integer code that specifies the territory in which the resource can be utilized.

5.2 Personal Online Data Stores for Data Provision

We develop the `Personal Online Datastore` prototype using the Python language. Python's support for the Web3.py library[12] enables the creation of communication protocols with the blockchain platform acting as the `Governance Ecosystem` of the decentralized environment. Our implementation also includes a graphical user interface developed with the Tkinter library[13]. As shown in Fig. 8, our `Personal Online Datastore` implementation is composed of three main parts: the `Application`, the `Web Service` and the `Resource Storage`. The app module contains the executable code implementing the graphical user interface.

5.2.1 Resource Storage

The resource storage contains the resources of the `Personal Online Datastore`. The storage location is characterized by two meta-files named `DTconfig.json` and `DTobligations.json`. They contain descriptive and confidential information about the `Personal Online Datastore` and its resources. `DTconfig.json` includes various attributes of a `Personal Online Datastore`, such as its unique identifier, its node's public and private keys, the web reference to access data, and a list of the initialized resources. `DTobligations.json` holds rules that apply to the resources of the storage. The user can establish a default policy inherited by all resources in the `Personal Online Datastore`, except those

[12] https://web3js.readthedocs.io/en/v1.8.1/. Accessed: Wednesday 12th April, 2023.

[13] https://docs.python.org/3/library/tk.html. Accessed: Wednesday 12th April, 2023.

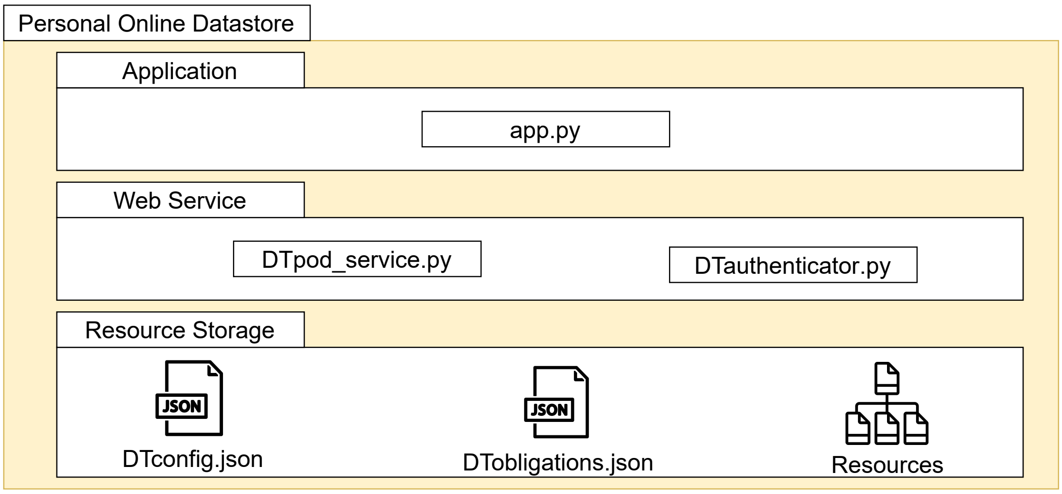


Figure 8. Schematization of the personal online datastore implementation.

with specific policies. Mentioning our running example, Bob interacts with the `Personal Online Datastore` application to upload the ‘Mesoplodon.jpg’ resource in the ‘/images’ location inside the storage. During this process, Bob can establish the rules associated with the image. The initialization of the image generates the metadata to be held in the `DTconfig.json` and `DTobligations.json` metafiles.

5.2.2 Web Service

The implementation of the data provision process is built upon the HTTP web standard. Our `Personal Online Datastore` prototype implements a `Web Service` that listens for HTTP requests, verifies the authenticity of the sender `Node`, and delivers the requested data through HTTP responses. This approach enables the efficient and on-demand provision of initialized data. In Fig. 9, we summarize the main stages of the data provision process, taking place in our `Web Service` implementation. The `DTpod_service` Python class contains the core functionality for resource delivery. The class extends `BaseHTTPRequetsHandler` that enables the processing of GET and POST requests. Due to confidentiality reasons, the `Web Service` of the `Personal Online Datastore` only responds to `POST Requests` and ignores GET ones. The data provision process starts with the `Parameter Extraction`, which takes place when a new `POST Request` is received by the `Web Service`. The parameters inside the body of the `POST Request` are crucial for the authentication and remote attestation procedures. In order to correctly demand a resource, requests must specify a URL composed of the web domain name of the service followed by the relative path of the requested resource inside storage. In the case of the motivating scenario, to retrieve Bob’s image, Alice’s node must generate an authenticable `POST Request`, whose URL is ‘https://BobNode/images/Mesoplodon.jpg’.

Through remote attestation, the `Web Service` can verify that the resource request has been legitimately generated by a `Trusted Application` running a `Intel SGX Trusted Execution Environment` of a `Node`. Therefore, we leverage the `Intel SGX Remote Attestation Verification` to establish a trusted communication channel between the consumer and the owner nodes. Once the attestation procedure ends successfully, the `Web Service` can be assured that the content of its response is managed by a `Data Consumption` technology inside the decentralized environment.

`Sender Authentication` takes place after the successful outcome of the remote attestation verification. The logic of our authentication mechanism is implemented through the `DTauthenticator` class, whose

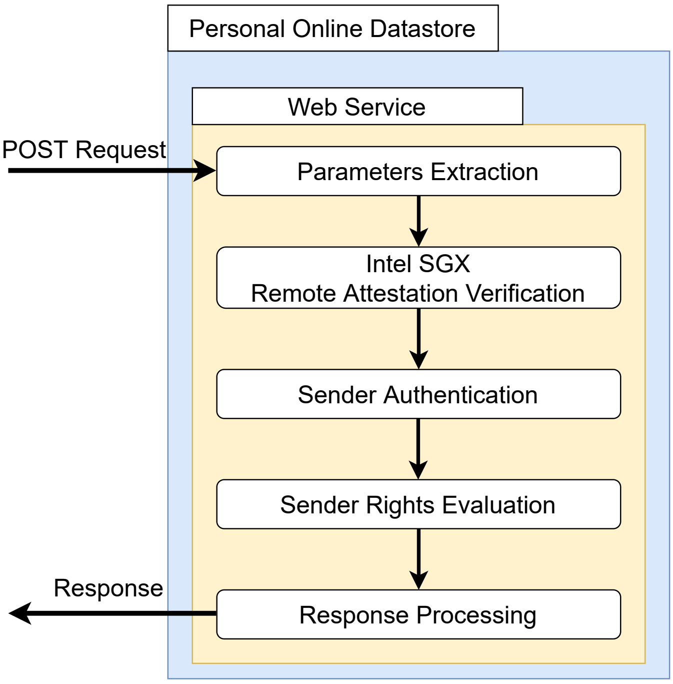


Figure 9. Main stages of the ReGov data provision instantiation process.

purpose is to use the `auth_token` (a message signed with the sender's credentials) and `claim` (the public key of the sender) parameters inside the `POST Request` to determine the sender `Node`'s identity. Specifically, `auth_token` refers to the URL of the resource to be accessed, encrypted with a private key. `DTauthenticator` is able to extract a public key from the `auth_token` parameter when the request is received. If the extracted public key is equal to the `claim` parameter, the identity of the sender `Node` is confirmed. At the end of the authentication procedure, Bob's `Web Service` identifies the sender of the request as Alice's `Node`.

The determined identity is subsequently evaluated by the `Web Service` during the `Sender Rights Evaluation` to determine whether the consumer `Node` can access the resource. Because our instantiation considers the decentralized environment related to the DecentralTrading data market (mentioned in Section 3), this step establishes whether the sender `Node` is associated with an active subscription (e.g., if Alice has an active subscription). However, the evaluation of alternative criteria, such as organization membership, can be freely integrated depending on the specific use case. In all cases, it is crucial to keep track of the consumer nodes that have accessed the `Personal Online Datastore`'s resources by establishing their identity.

Once the `POST Request` has passed the necessary checks, the `Response Processing` takes place. Therefore, the `Web Service` then interacts with the local storage to retrieve the requested resource, which, along with the associated policy, are inserted into the `Response`.

5.3 Trusted Execution Environment for Data Consumption

The `Trusted Execution Environment` manages the resources recovered within the consumer node. In Fig. 10, we propose a schematization of our `Trusted Application` implementation. The trusted application consists of two fundamental components: the `Trusted Part` and the `Untrusted Part`. The `Trusted Part` comprises one or more enclaves. The `Enclave`'s code is in the `enclave.cpp` file. It

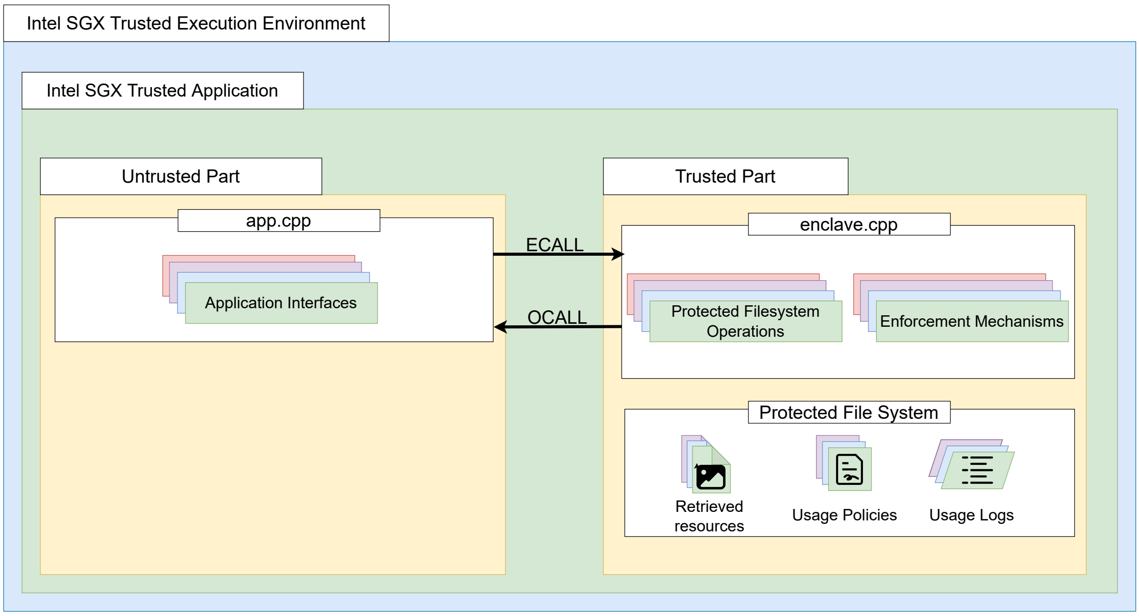


Figure 10. Schematization of our trusted application composed of both trusted and untrusted elements.

includes all the implementations of the `Enforcement Mechanisms` and a set of `Protected File System Operations` to handle the resources stored in it. The `Trusted Part` cannot communicate directly with the outside world. Any pieces of information that enter or leave the `Trusted Part` pass through the `Untrusted Part`. The `Untrusted Part`'s code is in the `app.cpp` file. This application has multiple `Application Interfaces` that are used to expose the application to the outside world. In order to communicate, the two parts use dedicated functions called `Ecall` and `Ocall`. 'Ecall' stands for Enclave Call and represents an invocation made by a function in the `Untrusted Part` to the `Enclave` (`Trusted Part`). The term 'Ocall' (Out Call) refers to a call from the `Enclave` to the `Untrusted Part`.

5.3.1 Data Protection

The main purpose of using the `Trusted Application` is to manage and protect the data of other users obtained from the market. The `Retrieved Resources` are stored within the `Enclave`, more specifically in its `Protected File System`, because in this way they are decrypted only within the processor and only the enclave itself can access the processor in order to decrypt it. Within the enclave, both the `Resources Retrieved` by the user and the `Usage Policies` set by the owner are stored. Storing the `Retrieved Resource` within the `Trusted Part` is essential both from a data protection and a usage control perspective. In addition, the `Usage Policy` chosen by the data owner must also be saved in a secure space, as it could be tampered with by malicious code in order to be bypassed.

Protection of usage data. When a user requests a piece of data, the request is received by the dedicated `Application Interface` in the `Untrusted Part`, and it is retrieved from the market. For instance, when Alice requests a photo of a Mesoplodon eueu from Bob, an identifier is assigned to this data before it is stored in the `Enclave`. The identifier associated with the resource is used to index the retrieved resources and store them within the trusted part. A copy of the policies set by the owner, the rules set by Bob for the photo, is associated with it in order to store all the necessary resource information in the enclave. More specifically, when Alice wants to retrieve a piece of data from Bob, she interacts with the `Untrusted Part` and sends a post HTTP request to Bob's node. Within the request parameters, the resource in which the consumer is interested is specified, and an identifier is provided with which the consumer gets authenticated (as described in Section 5.2.2). Finally, a certificate provided by Intel SGX Remote Attestation is added to the request, providing evidence that the request comes from a `Trusted Application`. Once the `Personal Online Datastore` ensures that the other party involved in the communication is trusted, it sends the resource and policy information via an HTTP reply. Since the `Trusted Part` cannot communicate with the outside world, the response reaches the `Untrusted Part` who forwards it via an `Ecall` to the `Trusted Part`. Once the resource arrives at the `Trusted Part`, it stores the data sent from the `Personal Online Datastore` in the `Enclave` using the `Protected File System Operations` that allow the `Enclave` to manage the `Protected File System`. Based on the example scenario, at this point the photo of the Mesoplodon eueu and the related `Usage Policies` set by Bob, the owner, are stored within Alice's `Enclave`.

Protection of log data. To keep track of the correct use of resources, all actions performed on them within the `Trusted Part` are stored in a `usage log file`. In short, all actions concerning the retrieved resources are stored. The objective is to let the data owner initiate a monitoring procedure through an oracle, to check whether resources are used in accordance with usage conditions. When the `Untrusted Part` receives a monitoring request from the blockchain, it performs an `Ecall` to request a copy of the `Usage Log` file stored in the `Enclave` and returns it to the blockchain through an oracle to perform the monitoring. Referring to the example, all actions performed by Alice are recorded in a `Usage Log` file, and when Bob wants to check that everyone is using their resource correctly, he starts a monitoring procedure that aims to check all the `Usage Log` files of consumers who have retrieved the Mesoplodon eueu photos. When the `Usage Log` file is requested to be monitored, before sending a copy, the `Trusted Part` enters an entry to keep track of the monitoring request.

5.3.2 Implementation of the Enforcement Mechanisms

In order to guarantee that data are accessed and used according to usage policies when a resource from the `Trusted Part` of a `Trusted Application` is requested by an external application, `enforcement mechanisms` must be implemented. These mechanisms are implemented within the `Enclave` to ensure they are executed within a `Trusted Environment`.

Receiving a request for access to a resource stored in the trusted application. Before proceeding with the `Enforcement Mechanisms`, when the external application makes a request to the `Trusted Application`, the latter asks the external application to identify itself in order to check whether the sender is who it declares to be. More specifically, the `Untrusted Part` receives a request for access to a resource via the `Application Interfaces` and forwards it to the `Trusted Part` through an `Ecall` by invoking the `access_protected_resource` function, which verifies the identity of the claimant. Referring to the example, when Alice uses the 'Zooresearch' or 'Socialgram' applications, they have to authenticate themselves.

Retrieval of the requested resource and its usage policy. Once the external application has been authenticated, the `Trusted Application` gathers all the necessary information about it and accepts the request for the data that the external application is interested in and starts checking whether it is possible to access and use the resource. First, the `access_protected_resource` function retrieves the requested data and the associated policies, using the `get_policy` function, set by the owner. Then, the `access_protected_resource` function invokes the different enforcement modules, passing the retrieved policies to it, in order to ensure that the rules are satisfied. In our implementation, four different enforcement modules have been developed. The proposed approach is highly flexible, thus catering for the extension of the existing rule types. The first mechanism in the enforcement process is checking the geographical position of the device.

Geographical rule enforcement. The `enforce_geographical` function is invoked and passed the policy for the requested resource. The `get_geo_location` function (`Ocall`) is then used to retrieve the geographic location of the device from which the resource is being accessed. In the end, the geographic data set by the user and the current location are compared. If the position is correct, a positive result is returned to the `access_protected_resource` function, otherwise access is denied. Referring to the scenario, the `Trusted Application` uses Alice's location to check if it meets the location stipulated by Bob in his usage policy.

Domain rule enforcement. The `access_protected_resource` function invokes the `enforce_domain` function by passing it the policy of the requested resource and information about the requesting application. Following a comparison between the application's domain and the domain set by the resource owner, if the domains are equal, the `enforce_domain` function returns a positive result to the `access_protected_resource` function, which proceeds to the next check. Otherwise, access to the resource is denied. Looking at the example scenario, the domain of the application used by Alice is checked to determine if it satisfies the usage domain set by Bob. If Alice's application domain is correct, a positive result is returned.

Access counter rule enforcement. The `enforce_access_counter` function is called by the `access_protected_resource` function with the policy for the requested resource. If the number of remaining accesses is greater than 1, the function decrements the maximum number of remaining accesses for that resource and returns with success to the `access_protected_resource` function. If the number of remaining accesses is equal to 1, the function removes the resource and related policies from the `Enclave` before returning a positive value, as the resource can no longer be accessed. In the motivating scenario, Bob set 100 as the maximum number of accesses to the resource. Each time Alice makes a request and logs in, the maximum number of hits left decreases. When the counter becomes 1, Alice is allowed a last access to the Mesoplodon eueu's photo, and then the resource is deleted from her `Trusted Application`. Then, having successfully completed all the enforcement, the `access_protected_resource` function forwards the contents of the file to the `Untrusted Part`, which forwards it to the external requesting application. As already mentioned, all actions performed on the resources in the trusted application are saved on a `Usage Log` file, which keeps information and accesses made on the resources from when it is retrieved until it is deleted, maintaining an overview of the use of the resource. This `Usage Log` file makes it possible to prove and check that all resources have been used correctly within the trusted application.

Temporal rule enforcement. When it comes to temporal rules, the `Untrusted Part` periodically invokes the `Ecall` function called `enforce_temporal` to verify that all resources within the trusted part have not expired. The `enforce_temporal` function uses the `get_trusted_time` function to retrieve the current day. It then reads all resource policies stored within the `Trusted Part` and checks whether the date set on the policy is later than the current date. If a resource has expired, the `enforce_temporal` function removes it. Each time this type of check is performed, it is written to the `Usage Log` file, and all deletions are also saved.

5.4 Blockchain as a Governance Ecosystem

In our instantiation, we leverage blockchain smart contracts in order to realize the `Governance Ecosystem`. Transparency, distribution, and immutability are the key features that make this technology highly suitable for our needs. The DecentralTrading implementation leverages the `EVM Blockchain` platform hosting several interconnected smart contracts. Nodes of the decentralized environment that are equipped with confidential blockchain public and private keys, sign authenticate transactions that generate the execution of smart contract functions. Processes that involve data exchange between Nodes and smart contracts are supported by blockchain oracles.

We implemented the smart contracts using the Solidity programming language[14]. The smart contracts have been deployed in a local environment powered by the Ganache tool[15] which enables the execution of a local blockchain replicating the Ethereum protocol and supporting the generation of transactions for testing purposes. In the following, we present the implementation details regarding the DTindexing and DTobligations smart contracts that fulfill the functionality of the Resource Indexing and Policy Governance components respectively.

5.4.1 DTindexing Smart Contract

The DTindexing smart contract caters for the initialization of shared resources in the decentralized environment. The main goal of this component is to keep track of the decentralized environment's data. Owner nodes interact with the smart contract to index their Personal Online Datastore, sharing the necessary metadata for data retrieval. Consumer nodes make use of the smart contract to find references for registered resources through search functionality. Table 2 represents the class diagram of the smart contract. The smart contract saves the following variables in the Pod struct in order to keep track of the information about personal online datastores:

```
struct Pod { int id; address owner; bytes baseUrl; bool isActive; }
```

Similarly, the contract stores information about resources in a Resource struct, which consists of the following:

```
 struct Resource{ int id; address owner; int podId; bytes url; bool isActive; }
```

The Pod and Resource structs are stored in the podList and resourceList array variables, respectively. The contract includes several methods for interacting with online datastores and resources, including the ability to register new ones, deactivate existing ones, and to search for them based on various criteria. For example, the registerPod method allows nodes to initialize new personal online datastores in the network. It takes as input a web reference for the online datastore service and the public key of the owner Node. The function creates a new Pod struct and stores it in the podList. It also deploys a DTobligations smart contract (discussed next in detail), as every Personal Online Datastore is related to one of these contracts. Finally, the function emits a NewPod event containing the identifier and the address of the DTobligations smart contract for the new online datastore. In our running example, Bob's node invokes this function to initialize his new Personal Online Datastore providing the web reference https://BobNode.com/ among the arguments. The function, in turn, generates a new Pod struct. The registerResource method works similarly, generating a new Resource object and storing it in the resourceList state variable. In this case, Bob's Personal Online Datastore employs this function to initialize the 'Mesoplodon.jpg' image providing metadata such as the https://BobNode.com/images/Mesoplodon.jpg url. The deactivateResource and deactivatePod methods ensure that personal online datastores and resources are no longer accessible. Nodes submit metadata referring to new datastores and resources by using push-in oracles, that enable

[14] https://docs.soliditylang.org/en/v0.8.17/. Accessed: Wednesday 12th April, 2023.

[15] https://trufflesuite.com/ganache/. Accessed: Wednesday 12th April, 2023.

Table 2. Class diagram of the DTindexing smart contract.

DTindexing
private podsCounter: int private resourceCounter: int private dtSubscription: int private podList: Pod[] private resourceList: Resource[]
private searchByType(tp: PodType): Pod[] <<event>> NewPod(idPod: int, obgliationAddress: address) <<event>> NewResource(idResource: int) <<modifier>> validPodId(id: uint, owner: address) public getMedicalPods(idSubscription: uint): Pod[] public getSocialPods(idSubscription: uint): Pod[] public getFinancialPods(idSubscription: uint): Pod[] public registerPod(newReferene: bytes, podType: PodType, podAddress: address): int public registerResource(podId: int, newReferene: bytes, idSubscription: uint): int <<validPodId>> public getPodResources(podId: int, idSubscription: int): Resource[] public deactivateResource(idResource: int): Resource <<validResourceId>>

sending information to the blockchain. The smart contract also offers various search functions that can be useful for consumer nodes. The getPodResources method allows users to obtain a list of Resource structs stored in a specific datastore, identified by its integer identifier. The getResource method accepts an integer identifier as input and returns the Resource struct with that identifier. Referring to our use case scenario, Alice uses getPodResources to read the image's identifier that is given as a parameter to getResource, thanks to which the associated web reference is retrieved.

5.4.2 DTobligations Smart Contract

We use the DTobligations smart contract to model usage policies inside the blockchain environment and execute their monitoring. The architecture of the implementation assumes the deployment of multiple instances of the smart contract, one for each Personal Online Datastore in the network. Each DTobligations smart contract is associated with a specific Personal Online Datastore that is the only entity allowed to establish and manage the rules associated with the stored resources. As we showed in our motivating scenario, the architecture of our implementation assumes the deployment of a dedicated DTobligations instance containing the rules for Bob's Personal Online Datastore. In Table 3, we propose the class diagram of the DTobligations smart contract.

The DTobligations smart contract includes four structs, each of which, models a specific rule: AccessCounterObligation, which restricts the number of resource accesses on a client device; CountryObligation, which imposes restrictions on the countries in which a resource can be used; DomainObligation, which specifies the purposes for which resources can be used; and TemporalObligation, which imposes a maximum duration for resource storage. These are stored in an ObligationRules struct, which can apply to a specific resource or to the entire Personal Online Datastore. The smart contract includes functions that allow nodes to set default rules for their Personal Online Datastore and related resources. For instance, the addDefaultAccessCounterObligation and addDefaultTemporalObligation are used to set rules that are inherited by all the resources of the Personal Online Datastore. Similarly, functions such as addAccessCounterObligation and addTemporalObligation establish rules that are applied to a specific resource of the datastore. Referring to our running example, Bob's Personal Online Datastore invokes the addTemporalObligation giving as input the 'Mesoplodon.jpg' identifier and the integer value that describes the time duration of 20 days. The onlyOwner modifier ensures that certain functions can only be invoked by using the blockchain

Table 3. Class diagram of the DTobligations smart contract.

DTobligations <<extends >> Ownable
dtIndexing: DTindexing defaultPodObligation: ObligationRules resourcesObligation: mapping(int=>ObligationRules)
<<modifier>>hasSpecificRules(resourceId: int) <<modifier>>isValidTemporal(deadline: uint) <<modifier>>isTheResourceCovered(idResource: int) public constructor(dtInd: address, podAddress: address) public getObligationRules(idResource: int): ObligationRules <<isTheResourceCovered>> public getDefaultObligationRules(): ObligationRules public addDefaultAccessCounterObligation(accessCounter: uint) public addDefaultTemporalObligation(temporalObligation: uint) <<isValidTemporal, onlyOwner>> public addDefaultCountryObligation(country: uint) <<onlyOwner>> public addDefaultDomainObligation(domain: DomainType) <<onlyOwner>> public addAccessCounterObligation(idResource: int, accessCounter: uint): ObligationRules <<isTheResourceCovered, onlyOwner>> public addDomainObligation(idResource: int, domain: DomainType): ObligationRules <<onlyOwner, isTheResourceCovered>> public addCountryObligation(idResource: int, country: uint): ObligationRules <<onlyOwner, isTheResourceCovered>> public addTemporalObligation(idResource: int, deadline: uint): ObligationRules <<onlyOwner, isTheResourceCovered, isValidTemporal>> public removeAccessCounterObligation(idResource: int) <<onlyOwner, isTheResourceCovered, hasSpecificRules>> public removeTemporalObligation(idResource: int) <<isTheResourceCovered, onlyOwner, hasSpecificRules>> public removeDomainObligation(idResource: int) <<isTheResourceCovered, onlyOwner, hasSpecificRules>> public removeCountryObligation(idResource: int) <<isTheResourceCovered, onlyOwner, hasSpecificRules>> public removeDefaultTemporalObligation() <<onlyOwner>> public removeDefaultAccessCounterObligation() <<onlyOwner>> public removeDefaultCountryObligation() <<onlyOwner>> public removeDefaultDomainObligation() <<onlyOwner>> public withSpecificRules(idResource: int): bool public monitorCompliance() <<onlyOwner>>

credentials associated with the smart contract's owner. It is applied to the functions for rule modification, which can be invoked only by the owner `Node`. In this way, Bob is sure that modification of the rules can only be executed by his `Personal Online Datastore`.

The main goal of the monitoring procedure is to retrieve evidence from consumer nodes attesting to the utilization of resources, whose policies are represented by the `DTobligations` instance. The smart contract implements the `monitorCompliance` function, solely invocable by the contract owner, to initiate the monitoring procedure. When the function is used, it interacts with a pull-in oracle, that is able to retrieve external information outside the blockchain. Therefore, the `DTobligations` smart contract communicates with the on-chain component of the oracle (i.e. smart contract named `PullInOracle`) by invoking its `initializeMonitoring` function. The oracle generates a new `MonitoringSession` struct instance that contains information about the current state of the session and aggregates the external responses. The same function emits a `NewMonitoring` event. The emission of the event is caught by the off-chain components of the oracle, running in consumer nodes, that forward to the `SGX Intel Trusted Application` the command to provide the usage log of the resources involved. Once the usage log is retrieved, the information contained within it are sent to the on-chain component of the oracle through its `_callback` method. The function aggregates the responses from consumer nodes and updates the involved `MonitoringSession` instance each time it is called. Once all the responses are collected, they are returned to the `DTobligations` smart contract at the end of the process. In our running example, the procedure is started by Bob's `Personal Online Datastore` using the `monitorCompliance` function. Subsequently, Alice's `SGX Trusted Application` is contacted by the pull-in oracle and it is asked to provide the usage log of the 'Mesoplodon.jpg' resource. Alice's response contains information such as the number of local accesses to the image or the time from its retrieval. The evidence provided by Alice's

`SGX Trusted Application` is collected, together with pieces of evidence provided by other nodes in the network, by the pull-in oracle. Finally, the oracle forwards the logs to Bob's instance of `DTindexing`.

6 EVALUATION

We evaluate the implementation of the ReGov framework by taking two distinct approaches. In the first part of this section we revisit the specific requirements usage control requirements that were derived from out motivating scenario. While, in the second part, we examine the security, privacy, and affordability of our implementation.

6.1 Requirement Verification

In this section, we discuss how the previously established requirements are satisfied by our ReGov instantiation, following the methodology described in the study of Terry Bahill and Henderson (2005). Through the discussion of the requirements, we contextualize the use of the trusted execution environment and the blockchain respectively in our architecture. Both requirements are composed of several sub-requirements that express various environmental and technological functions.

6.1.1 (R1) Resource utilization and policy fulfillment must be managed by trusted entities

The first requirement (**R1**) stipulates that **resource utilization and policy fulfillment must be managed by trusted entities**. We employ a trusted execution environment in order to develop a trusted application executable inside our nodes. We implemented it using Intel SGX, as explained in Section 5.3. Our design and implementation choice allows us to satisfy the following sub-requirements:

(R1.1) The trusted entity must be able to store resources obtained from other entities. In the proposed ReGov framework instantiation, all resources retrieved from the data market by the untrusted part of a node are passed to the trusted part of a node in order to store them within the enclave. For storage, we use an Intel SGX function, called Protected File System Library, which allows the management of files containing the resources retrieved within the enclave. We chose to store the data in the enclave because any information stored in it is encrypted and decrypted solely by the enclave.

(R1.2) The trusted entity must support the execution of programmable procedures that enforce constraints associated with resource usage. When a resource stored within the enclave is requested, before retrieving it, the enclave we have implemented executes all the application procedures provided by the resource policy, invoking the necessary enforcement functions. The proposed enclave only allows access to the resource if at the end of the execution of all enforcement procedures, all of them have given a positive result. Otherwise, the resource is not returned and access is denied. It is worth noting that the enforcement mechanism within the trusted application is implemented in a modular way. Although our current implementation is limited to four rule types, this feature allows developers to easily extend our implementation with additional rule types based on their specific needs.

(R1.3) Resources and procedures managed by the trusted entity must be protected against malicious manipulations. In the proposed ReGov implementation, we store resources within the enclave, because it is secure and protected from unauthorized access. The trusted part cannot communicate directly with the outside world and thus avoids interacting with malicious software. In addition, all code included and executed in the trusted part is, in turn, trusted, as it is not possible to use third-party libraries. The data stored within the enclave are encrypted. Therefore, a direct attack on the memory by malicious software would not be able to read the data.

(R1.4) The trusted entity must be able to prove its trusted nature to other entities in a decentralized environment. When it comes to interaction between nodes, in order to prove a node's trustworthiness, we employ the Intel SGX remote attestation within our trusted application. This advanced feature allows a node to gain the trust of a remote node. The provided attestation ensures that the node is interacting with a trusted application using an updated Intel SGX enclave.

6.1.2 (R2) Policy compliance must be monitored via the entities of a governance ecosystem

The second requirement (**R2**) stipulates that **policy compliance must be monitored through entities running in a governance ecosystem**. In our ReGov framework, we propose the adoption of a governance ecosystem that we instantiate on top of blockchain technology. In the following, we show the suitability of blockchain for this role by addressing each sub-requirement.

(R2.1) The governance ecosystem must provide transparency to all the nodes of the decentralized environment. By allowing all nodes to view the complete transaction history of the blockchain technology, we are able to ensure that each participant of the decentralized environment has equal access to information and is able to independently verify the accuracy and integrity of governance data. Additionally, we implement the policy management tasks via smart contracts, the code for which is made publicly available within the blockchain infrastructure. This enables nodes in the decentralized environment to be aware of the governance processes that are being executed.

(R2.2) Data and metadata maintained by the governance ecosystem must be tamper-resistant. Our solution involves the storage of resource metadata and usage policies in data structures that are part of smart contracts. Through smart contracts functions, we implement functionality that can be used to upload and modify stored data. We leverage the asymmetric key encryption mechanism of the blockchain environment to verify that data modifications are performed by authorized users. Once data and metadata of ReGov are validated in a blockchain block, we rely on the cryptographic structure underlying the blockchain to guarantee the integrity of published smart contracts and the information contained therein.

(R2.3) The governance ecosystem and the entities that the form part of the ecosystem must be aligned with the decentralization principles. We fulfill the decentralization principles by proposing a blockchain-based architecture that is inherently decentralized. In our implementation, we publish data and metadata through a network of validators rather than a central authority. This ensures that no single entity has control over shared data and smart contracts that are distributed in the blockchain ecosystem. Through decentralization, we secure the fairness and integrity of policy management and prevent any single authority of the decentralized environment from having too much control or disproportionate decision-making power.

(R2.4) The entities that form part of the governance ecosystem must be able to represent policies and verify their observance. The majority of smart contract technologies are characterized by Turing-complete programming languages. We use the expressive power of smart contracts to implement data structures that can be used to represent usage policies and automate their monitoring. We facilitate the communication between smart contracts and off-chain nodes by integrating oracle technologies that implement the protocols for data-exchange processes.

6.2 Architecture Discussion

In this section, we broaden our discussion on the effectiveness of the proposed decentralized usage control architecture with a particular focus on privacy, security, and affordability. The criteria the discussion is based on have been inspired by the work of Ferrag and Shu (2021).

6.2.1 Security

Several works already show how the decentralized model makes it more difficult for attackers to compromise data, as they would need to gain access to multiple nodes rather than just one central server (Raman et al., 2019; Alabdulwahhab, 2018). As per the vast majority of decentralized web initiatives, our implementation preserves the security of data residing in nodes through the `Personal Online Datastore` component, which performs authentication and rights evaluation procedures to prevent unauthorized access to sensitive information or resources.

Our solution introduces new components into the decentralized model whose security should be discussed. The metadata stored in smart contracts (usage policies and resource indexes) are protected from unauthorized updates through the consensus mechanism of the blockchain platform and its distributed nature, which makes this information immutable. Moreover, the state of distributed applications running in this environment can only be changed by transactions marked by a digital signature. This feature guarantees that usage policy modifications can only be executed by authorized entities.

The `Intel SGX Trusted Execution Environment` provides a separate ecosystem for the execution of a `Trusted Application` that manages resource utilization. It has already shown its effectiveness in terms of preventing the injection of malicious code coming from the operating system of the client's machine (Sabt et al., 2015), which could jeopardize the integrity of the stored resources and the local representation of usage policies. Moreover, we also leverage the security guarantees offered by this technology to establish a protected environment in which the enforcement of the usage policies is ensured, inside the consumer's node.

The monitoring process, thanks to which nodes get evidence of the utilization of their resources, involves the interaction between the `EVM Blockchain` and consumer nodes. The procedure involves the exchange of confidential information, the integrity of which must be secured. Interactions between the involved components are managed via blockchain oracles that are capable of ensuring the legitimacy operations (Al-Breiki et al., 2020b). By definition, oracles establish secure communication protocols that enable on-chain and off-chain computations to send and receive data safely.

Security and verification of data consumption are enforced by the ensemble of smart contracts, trusted execution environments, and remote attestations. Through the latter, data providers are able to remotely verify the integrity of a node's data consumption component and thwart attempts to instantiate malicious consumer nodes in the decentralized environment. Nevertheless, data provision of inappropriate information through published data is a practice that requires automated ex-post checking and whistleblowing (Kirrane and Di Ciccio, 2020).

We remark that ReGov cannot supervise users' actions outside the digital context of the decentralized environment. For example, it is unable to prevent users from taking a picture of a protected image resource using a separate camera, or copying reserved information displayed on the screen. The framework is intended to operate at the digital level. Therefore, ReGov monitors and controls data access, processing, and distribution, ensuring that it is utilized in compliance with the associated policy. Our motivating scenario resorts to a list of approved applications that guarantee fair data elaboration and facilitate misconduct uncovering. Considering the running example, applications like "Socialgram" put in place procedures that counteract OS screen recording actions. In addition, unfair activities that break the enforcement mechanism can be detected by the presented monitoring routines, enabling data owners to indict malicious users.

6.2.2 Privacy

Privacy is key for decentralized web environments trying to take personal data out of the control of single organizations. With usage control, users can benefit from a greater level of privacy, as they have a way to determine how their resources are being used. However, enforcement and monitoring mechanisms that characterize usage control involve the exchange of data and metadata whose confidentiality should constantly be guaranteed.

One of the most critical issues of our solution regarding confidentiality relates to the blockchain metadata, which are publicly exposed in smart contracts. Public blockchains, such as Ethereum, provide public ledgers, thus allowing every node of the decentralized environment to get access to usage policy and resource locations. Despite the possibility of specifying private variables in smart contracts, the method invocations thanks to which those variables are set are recorded in publicly readable transactions. Therefore, blockchain users can freely deduce the state of a private variable by inspecting the public transactions associated with the invocation of the setter methods. In some use cases, it may be desirable to keep this data public. However, there may also be a need to encrypt data stored in the blockchain, so that only authorized parties (those that have access to the decryption key) can read this metadata (Pan et al., 2011; Marangone et al., 2022).

The confidentiality of the shared resources must be regulated after their retrieval inside consumer nodes, in order to apply the constraints associated with their policy rules. Our implementation leverage the `Intel SGX Trusted Execution Environment` that manages retrieved resources through the `SGX Protected File System (PFS)`. One of the key features of SGX-PFS is that it allows for files to be stored in a secure, encrypted format, even when the operating system is not running. This makes it difficult for attackers to access the resources, as they would need to have physical access to the machine and be able to bypass the SGX hardware security features in order to read the contents of the files.

6.2.3 Affordability

The affordability of our solution is strongly related to the costs associated with the smart contracts running in the blockchain ecosystem. `EVM Blockchains` associate the execution of smart contracts with a fee charged to the invoking user, according to the complexity of the code to be executed. This fee is measured in (units of) Gas. In Table 4, we collect the Gas expenses associated with the functions of the `DTobligations` and `DTindexing` smart contracts. The table omits their read functions, for which no transactions need to be sent to the network.

The deployment cost of `DTindexing` is $3{,}255{,}000$ Gas units. The `registerPod` method is the most expensive `DTindexing`'s function ($2{,}082{,}494$ Gas units) as it involves the deployment of a new contract instance, too. The Gas consumption of `registerResource` turns out to be significantly lower, requiring $143{,}004$ Gas units. The least expensive function of the smart contract is `deactivateResource` with an expenditure of $21{,}465$ Gas units.

`DTobligations` is deployed during the registration of a new personal online datastore at the cost of $2{,}057{,}988$ Gas units. `DTobligations` offers methods and functions to modify the obligation rules related to the resources contained in personal online datastore. Among the functions for adding rules, the most expensive one is `addAccessCounterObligation` with a value of $138{,}768$ Gas units. However, the adding of a domain restriction through `addDefaultDomainObligation` costs significantly less with $44{,}219$ Gas units per invocation. Methods for rule deactivation determine a lower expense than the previous ones. The

Table 4. Gas expenditure of the DTobligations and DTindexing smart contracts. Costs are expressed in Gas units.

DTobligations	
Function	**Cost**
deployment	2,057,988
`addDefaultAccessCounterObligation(···)`	62,627
`addDefaultTemporalObligation(···)`	62,638
`addDefaultDomainObligation(···)`	44,219
`addDefaultCountryObligation(···)`	62,561
`addAccessCounterObligation(···)`	138,768
`addTemporalObligation(···)`	97,737
`addCountryObligation(···)`	97,728
`addDomainObligation(···)`	79,452
`removeDefaultAccessCounterObligation(···)`	23,780
`removeDefaultTemporalObligation(···)`	16,079
`removeDefaultDomainObligation(···)`	24,747
`removeDefaultCountryObligation(···)`	23,758
`removeAccessCounterObligation(···)`	28,184
`removeTemporalObligation(···)`	28,151
`removeCountryObligation(···)`	28,173
`removeDomainObligation(···)`	38,111
`monitorCompliance(···)`	42,000

DTindexing	
Function	**Cost**
deployment	3,255,000
`registerPod(···)`	2,082,494
`registerResource(···)`	143,004
`deactivateResource(···)`	21,465

cheapest among them is `removeDomainObligation` (16,079 Gas units). The cost required to initialize a monitoring process through the `monitorCompliance` function is 42,000 units of Gas.

As expected, operations involving new smart contract deployments are the most expensive ones. However, these costs are associated with one-time operations performed at setup time (at the bootstrapping of the platform, or every time a new pod is registered). On the other hand, functions intended for more frequent invocations (e.g., to monitor compliance or update rules) are characterized by significantly lower costs. Costs in fiat money are subject to high variability, as they depend on multiple factors including the network capacity utilization, the price in cryptocurrency per Gas unit, and the market exchange rate of the cryptocurrency. Also, these values change depending on the EVM blockchain in use (e.g., Ethereum[16], Avalanche[17], Polygon[18], and more). At the time of writing, we empirically found variations of four orders of magnitude[19]. However, we remark that our implementation costs align with ERC721 implementations[20]. For example, the deployment fees of the Ethereum Name Service (ENS)[21], a non-fungible token in the

[16] https://ethereum.org/. Accessed: Wednesday 12th April, 2023.

[17] https://www.avax.com/. Accessed: Wednesday 12th April, 2023.

[18] https://polygon.technology/. Accessed: Wednesday 12th April, 2023.

[19]

The amount of gas needed for the deployment of the `DTindexing` smart contract, e.g., is 3,255,000. During our experiments, the price per Gas unit in the Ethereum public network amounted to 36.15 Gwei (one GWei is worth 10^{-9} ETH). The ETH/EUR exchange rate was $1/1590$ EUR. The total gas cost price was thus 187.09 EUR. Other EVM blockchains exhibited lower Gas prices or exchange rates, decreasing the overall cost in fiat money. Considering the Avalanche and Polygon platforms, their Gas price was 42.56 and 168.65 Gwei, respectively. The AVAX/EUR exchange rate was $1/15.67$, and the MATIC/EUR exchange rate was $1/1.19$. As a result, the total expenses amounted to 2.17 and 0.65 EUR, respectively. Data collected: 14 March 2023, 11:30 pm. Our smart contract deployments can be found on the Görli Ethereum test network at https://goerli.etherscan.io/address/0xb0fe7d07947d9dd7cda47825e61ec14b98ef271a, on the Fuji Avalanche test network at https://testnet.snowtrace.io/address/0x0082698263ccc5765c97404af39023daefe20096, and on the Mumbai Polygon test network at https://mumbai.polygonscan.com/address/0x9ee2cb5ef7b1449d615d9fd0f9b167543e0d28eb.

[20] https://eips.ethereum.org/EIPS/eip-721. Accessed: Wednesday 12th April, 2023.

[21] https://etherscan.io/token/0xc18360217d8f7ab5e7c516566761ea12ce7f9d72. Accessed: Wednesday 12th April, 2023.

neighboring area of personal information indexing, amounts to $2{,}443{,}978$ Gas units[22]. The market scenario can support the structural expenses associated with the proposed implementation and provides an incentive system that allows users to earn money by sharing their data. However, cost reduction practices are necessary to increase usability. These include design improvements to the implementation's architecture as well as the adoption of side-chains and layer-2 networks.

7 CONCLUSION

Since its inception, the web has evolved from a read-only medium for information dissemination to a ubiquitous information and communication platform that supports interaction and collaboration globally. Although the web is by design decentralized and thus is not controlled by any single entity or organization, the web as we know it today is dominated by a small number of centralized platforms. Consequently, the decentralized web initiative aims to promote research into tools and technologies that give data owners more control over their data and enable smaller players to gain access to data, thus enabling innovation.

In this paper, we focus specifically on resource governance in a decentralized web setting. We extend the state of the art by proposing a conceptual resource governance framework, entitled ReGov, that facilitates usage control in a decentralized setting, with a particular focus on policy respecting resource utilization and resource indexing and continuous monitoring. In order to demonstrate the potential of our ReGov framework, we propose a concrete instantiation that employs a trusted execution environment to cater for the former, and blockchain technologies to facilitate the latter. The effectiveness of the ReGov framework and our particular instantiation is assessed via a detailed analysis of concrete requirements derived from a data market motivating scenario and an assessment of the security, privacy, and affordability aspects of our proposal.

Future work includes extending our primitive rule syntax to encompass more expressive usage control policies that are based on standard policy languages. Additionally, we plan to explore strategies for reducing the costs associated with the smart contracts running in the blockchain ecosystem. Studying incentivization mechanisms to encourage users to use the platform and possibly gain rewards for sharing information also paves the path for future endeavors.

The community-based categorization of applications interfaced with ReGov is a challenging aspect, the solution to which potentially involves the adoption of dedicated smart contracts for voting and arbitrage mechanisms. Also, erroneous or malicious misuse of ReGov such as the publication and disclosure of otherwise private information is beyond the reach of ReGov and would entail ex-post patrolling of the system. Studying these integrations with our framework is a task we envision for future work. Finally, we aim to conduct case studies with users to evaluate our approach in real-world settings.

Acknowledgments.

The work of D. Basile, C. Di Ciccio, and V. Goretti was partially funded by the Italian Ministry of University and Research under grant "Dipartimenti di eccellenza 2018-2022" of the Department of Computer Science at Sapienza, and by the EU-NGEU NRRP MUR under grant PE00000014 (SERICS), and by the Cyber 4.0 project BRIE. The work of S. Kirrane was funded by the FWF Austrian Science Fund and the Internet Foundation Austria under the FWF Elise Richter and netidee SCIENCE programmes as project number V 759-N.

[22] https://etherscan.io/tx/0xff3ee18523c9ec20e62d31d3d3ce3e8bf25f5ffcdfc4c32cd43ed0a786cc8640. Accessed: Wednesday 12th April, 2023.

REFERENCES

Akaichi, I. and Kirrane, S. (2022a). A semantic policy language for usage control. In *SEMANTiCS (Posters & Demos)* (CEUR-WS.org), 10:1–10:5

Akaichi, I. and Kirrane, S. (2022b). Usage control specification, enforcement, and robustness: A survey. *arXiv preprint arXiv:2203.04800*

Al-Breiki, H., Rehman, M. H. U., Salah, K., and Svetinovic, D. (2020a). Trustworthy blockchain oracles: Review, comparison, and open research challenges. *IEEE Access* 8, 85675–85685

Al-Breiki, H., Rehman, M. H. U., Salah, K., and Svetinovic, D. (2020b). Trustworthy blockchain oracles: Review, comparison, and open research challenges. *IEEE Access* 8, 85675–85685

Alabdulwahhab, F. A. (2018). Web 3.0: The decentralized web blockchain networks and protocol innovation. In *2018 1st International Conference on Computer Applications & Information Security (ICCAIS)*. 1–4. doi:10.1109/CAIS.2018.8441990

Ayoade, G., Karande, V., Khan, L., and Hamlen, K. (2018). Decentralized IoT data management using blockchain and trusted execution environment. In *2018 IEEE International Conference on Information Reuse and Integration (IRI)*. 15–22. doi:10.1109/IRI.2018.00011

Bai, G., Yan, L., Gu, L., Guo, Y., and Chen, X. (2014). Context-aware usage control for web of things. *Security and Communication Networks* 7, 2696–2712

Basile, D., Goretti, V., Di Ciccio, C., and Kirrane, S. (2021). Enhancing blockchain-based processes with decentralized oracles. In *BPM (Blockchain and RPA Forum)*. 102–118

Becker, H., Vu, H., Katzenbach, A., Braun, C. H., and Käfer, T. (2021). Monetising resources on a solid pod using blockchain transactions. In *The Semantic Web: ESWC 2021 Satellite Events*. 49–53

Bonatti, P. A., Kirrane, S., Petrova, I. M., and Sauro, L. (2020). Machine understandable policies and GDPR compliance checking. *KI-Künstliche Intelligenz* 34, 303–315

Buterin, V. et al. (2014). A next-generation smart contract and decentralized application platform. *white paper* 3, 2–1

Cai, T., Yang, Z., Chen, W., Zheng, Z., and Yu, Y. (2020). A blockchain-assisted trust access authentication system for solid. *IEEE Access*

Carroll, E. L., McGowen, M. R., McCarthy, M. L., Marx, F. G., Aguilar, N., Dalebout, M. L., et al. (2021). Speciation in the deep: genomics and morphology reveal a new species of beaked whale mesoplodon eueu. *Proceedings of the Royal Society B* 288, 20211213

Costan, V. and Devadas, S. (2016). Intel sgx explained. *Cryptology ePrint Archive*

Esteves, B. and Rodríguez-Doncel, V. (2022). Analysis of ontologies and policy languages to represent information flows in GDPR. *Semantic Web* , 1–35

Ferrag, M. A. and Shu, L. (2021). The performance evaluation of blockchain-based security and privacy systems for the internet of things: A tutorial. *IEEE Internet of Things Journal* 8, 17236–17260. doi:10.1109/JIOT.2021.3078072

Grünbacher, A. (2003). POSIX access control lists on linux. In *Proceedings of the FREENIX Track: 2003 USENIX Annual Technical Conference*. 259–272

Havur, G., Vander Sande, M., and Kirrane, S. (2020). Greater control and transparency in personal data processing. In *International Conference on Information Systems Security and Privacy (ICSSP)*. 655–662. doi:10.5220/0009143206550662

Hilty, M., Pretschner, A., Basin, D., Schaefer, C., and Walter, T. (2007). A policy language for distributed usage control. In *European Symposium on Research in Computer Security* (Springer), 531–546

Jauernig, P., Sadeghi, A.-R., and Stapf, E. (2020). Trusted execution environments: properties, applications, and challenges. *IEEE Security & Privacy* 18, 56–60

Johnson, D., Menezes, A., and Vanstone, S. (2001). The elliptic curve digital signature algorithm (ecdsa). *International journal of information security* 1, 36–63

Khan, M. Y., Zuhairi, M. F., Syed, T. A., Alghamdi, T. G., and Marmolejo-Saucedo, J. A. (2020). An extended access control model for permissioned blockchain frameworks. *Wirel. Networks* 26, 4943–4954

Kirrane, S. and Di Ciccio, C. (2020). BlockConfess: Towards an architecture for blockchain constraints and forensics. In *AIChain@Blockchain* (IEEE), 539–544. doi:10.1109/Blockchain50366.2020.00078

Koshutanski, H. and Massacci, F. (2003). An access control framework for business processes for web services. In *Proceedings of the 2003 ACM workshop on XML security*. 15–24

Lazouski, A., Martinelli, F., and Mori, P. (2010). Usage control in computer security: A survey. *Computer Science Review* 4, 81–99

Liang, X., Shetty, S., Zhao, J., Bowden, D., Li, D., and Liu, J. (2017). Towards decentralized accountability and self-sovereignty in healthcare systems. In *International conference on information and communications security* (Springer), 387–398

Lind, J., Eyal, I., Kelbert, F., Naor, O., Pietzuch, P., and Sirer, E. G. (2017). Teechain: Scalable blockchain payments using trusted execution environments. *arXiv preprint arXiv:1707.05454*

Mammadzada, K., Iqbal, M., Milani, F., García-Bañuelos, L., and Matulevicius, R. (2020). Blockchain oracles: A framework for blockchain-based applications. In *BPM (Blockchain and RPA Forum)* (Springer), 19–34

Marangone, E., Di Ciccio, C., and Weber, I. (2022). Fine-grained data access control for collaborative process execution on blockchain. *arXiv preprint arXiv:2207.08484*

McGillion, B., Dettenborn, T., Nyman, T., and Asokan, N. (2015). Open-tee–an open virtual trusted execution environment. In *2015 IEEE Trustcom/BigDataSE/ISPA* (IEEE), vol. 1, 400–407

Mohanty, D. (2018). Ethereum for architects and developers. *Apress Media LLC, California* , 14–15

Mühlberger, R., Bachhofner, S., Ferrer, E. C., Di Ciccio, C., Weber, I., Wöhrer, M., et al. (2020). Foundational oracle patterns: Connecting blockchain to the off-chain world. In *BPM (Blockchain and RPA Forum)* (Springer), 35–51

Neisse, R., Pretschner, A., and Di Giacomo, V. (2011). A trustworthy usage control enforcement framework. In *2011 Sixth International Conference on Availability, Reliability and Security*. 230–235. doi:10.1109/ARES.2011.40

Ouaddah, A., Abou Elkalam, A., and Ait Ouahman, A. (2016). Fairaccess: a new blockchain-based access control framework for the internet of things. *Security and communication networks* 9, 5943–5964

Pan, J., Paul, S., and Jain, R. (2011). A survey of the research on future internet architectures. *IEEE Communications Magazine* 49, 26–36

Park, J. and Sandhu, R. (2004). The uconabc usage control model. *ACM transactions on information and system security (TISSEC)* 7, 128–174

Pasdar, A., Lee, Y. C., and Dong, Z. (2022). Connect API with blockchain: A survey on blockchain oracle implementation. *ACM Comput. Surv.* doi:10.1145/3567582

Patel, S., Sahoo, A., Mohanta, B. K., Panda, S. S., and Jena, D. (2019). Dauth: A decentralized web authentication system using ethereum based blockchain. In *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)* (IEEE), 1–5

Quail, C. and Larabie, C. (2010). Net neutrality: Media discourses and public perception. *Global Media Journal* 3, 31

Quintais, J. (2020). The new copyright in the digital single market directive: a critical look. *European Intellectual Property Review*

Ramachandran, M., Chowdhury, N., Third, A., Domingue, J., Quick, K., and Bachler, M. (2020). Towards complete decentralised verification of data with confidentiality: Different ways to connect solid pods and blockchain. In *Companion Proceedings of the Web Conference 2020*. 645–649

Raman, A., Joglekar, S., Cristofaro, E. D., Sastry, N., and Tyson, G. (2019). Challenges in the decentralised web: The mastodon case. In *Proceedings of the Internet Measurement Conference*. 217–229

Rushby, J. M. (1981). Design and verification of secure systems. *ACM SIGOPS Operating Systems Review* 15, 12–21

Sabt, M., Achemlal, M., and Bouabdallah, A. (2015). Trusted execution environment: What it is, and what it is not. In *2015 IEEE TrustCom/BigDataSE/ISPA*. 57–64

Sandhu, R. S. and Samarati, P. (1994). Access control: principle and practice. *IEEE communications magazine* 32, 40–48

Terry Bahill, A. and Henderson, S. J. (2005). Requirements development, verification, and validation exhibited in famous failures. *Systems engineering* 8, 1–14

Toninelli, A., Montanari, R., Kagal, L., and Lassila, O. (2006). A semantic context-aware access control framework for secure collaborations in pervasive computing environments. In *International semantic web conference* (Springer), 473–486

Tran, H., Hitchens, M., Varadharajan, V., and Watters, P. (2005). A trust based access control framework for P2P file-sharing systems. In *Proceedings of the 38th Annual Hawaii International Conference on System Sciences* (IEEE), 302c–302c

Xiao, Y., Zhang, N., Li, J., Lou, W., and Hou, Y. T. (2020). Privacyguard: Enforcing private data usage control with blockchain and attested off-chain contract execution. In *Computer Security – ESORICS 2020*, eds. L. Chen, N. Li, K. Liang, and S. Schneider. 610–629

Xu, X., Pautasso, C., Zhu, L., Gramoli, V., Ponomarev, A., Tran, A. B., et al. (2016). The blockchain as a software connector. In *WICSA* (IEEE Computer Society), 182–191

Xu, X., Weber, I., and Staples, M. (2019). *Architecture for Blockchain Applications* (Springer)

Zhao, C., Saifuding, D., Tian, H., Zhang, Y., and Xing, C. (2016). On the performance of intel sgx. In *2016 13Th web information systems and applications conference (WISA)* (IEEE), 184–187

Zhaofeng, M., Lingyun, W., Xiaochang, W., Zhen, W., and Weizhe, Z. (2020). Blockchain-enabled decentralized trust management and secure usage control of IoT big data. *IEEE Internet of Things Journal* 7, 4000–4015

Zheng, W., Wu, Y., Wu, X., Feng, C., Sui, Y., Luo, X., et al. (2021). A survey of intel sgx and its applications. *Frontiers of Computer Science* 15, 1–15

12. Self-enforcing access control for encrypted RDF

Bibliographic Information

Fernández, J.D., **Kirrane, S.**, Polleres, A. and Steyskal, S., 2017, May. Self-enforcing access control for encrypted RDF. In The Semantic Web: 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28–June 1, 2017, Proceedings, Part I (pp. 607-622). Cham: Springer International Publishing.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Formal analysis, Investigation, Writing - Original Draft, Writing - Review & Editing, Supervision, Project administration, and Funding acquisition.

Copyright Notice

Self-Enforcing Access Control for Encrypted RDF *

Javier D. Fernández[1,2], Sabrina Kirrane[1], Axel Polleres[1,2], and Simon Steyskal[1,3]

[1] Vienna University of Economics and Business, Vienna, Austria
`[firstname.lastname]@wu.ac.at`
[2] Complexity Science Hub Vienna, Vienna, Austria
[3] Siemens AG Österreich, Vienna, Austria

Abstract. The amount of raw data exchanged via web protocols is steadily increasing. Although the Linked Data infrastructure could potentially be used to selectively share RDF data with different individuals or organisations, the primary focus remains on the unrestricted sharing of public data. In order to extend the Linked Data paradigm to cater for closed data, there is a need to augment the existing infrastructure with robust security mechanisms. At the most basic level both access control and encryption mechanisms are required. In this paper, we propose a flexible and dynamic mechanism for securely storing and efficiently querying RDF datasets. By employing an encryption strategy based on Functional Encryption (FE) in which controlled data access does not require a trusted mediator, but is instead enforced by the cryptographic approach itself, we allow for fine-grained access control over encrypted RDF data while at the same time reducing the administrative overhead associated with access control management.

1 Introduction

The Linked Data infrastructure could potentially be used not only to distributedly share public data, but also to selectively share data, perhaps of a sensitive nature (e.g., personal data, health data, financial data, etc.), with specific individuals or organisations (i.e., closed data). In order to realise this vision, we must first extend the existing Linked Data infrastructure with suitable security mechanisms. More specifically, encryption is needed to protect data in case the server is compromised, while access control is needed to ensure that only authorised individuals can access specific data. Apart from the need to protect data, robustness in terms of usability, performance, and scalability is a major consideration.

However, current encryption techniques for RDF are still very limited, especially with respect to the flexible maintenance and querying of encrypted data in light of user access control policies. Initial partial encryption techniques [16, 17] focus on catering for both plain and encrypted data in the same representation

* Supported by the Austrian Science Fund (FWF): M1720-G11, the Austrian Research Promotion Agency (FFG) under grant 845638, and European Union's Horizon 2020 research and innovation programme under grant 731601.

and how to incorporate the metadata necessary for decryption. More recently, [21] proposed the generation of multiple ciphertexts per triple (i.e. each triple is encrypted multiple times depending on whether or not access to the subject, predicate and/or object is restricted) and the distribution of several keys to users. Although finer-grained access control is supported, the maintenance of multiple ciphertexts (i.e. encrypted triples) and keys presents scalability challenges. Additionally, such an approach, or likewise term-based encryption of RDF graphs, means that the *structure* of parts of the graph that should not be accessible could potentially be recovered, thus posing a security risk (cf. for instance [32]).

Beyond RDF, novel cryptography mechanisms have been developed that enable the flexible specification and enforcement of access policies over encrypted data. Predicate-based Encryption (PBE) [22] – which we refer to as Functional Encryption (FE) in order to avoid confusion with *RDF predicates* – enables searching over encrypted data, mainly for keywords or the conjunction of keyword queries, while alleviating the re-encryption burden associated with adding additional data.

Herein, we extend recent findings on FE to RDF, and demonstrate how FE can be used for fine-grained access control based on triples patterns over encrypted RDF datasets. Summarising our contributions, we: (i) adapt functional encryption to RDF such that it is possible to enforce access control over encrypted RDF data in a self-enforcing manner; (ii) demonstrate how encryption keys based on triple patterns can be used to specify flexible access control for Linked Data sources; and (iii) propose and evaluate indexing strategies that enhance query performance and scalability. Experiments show reasonable loading and query performance overheads with respect to traditional, non-encrypted data retrieval. The remainder of the paper is structured as follows: We discuss related work and potential alternatives to our proposal in *Section* 2. The details of our specific approach and optimisations are presented in *Section* 3 and *Section* 4 respectively, and evaluated in *Section* 5. Finally, we conclude and outline directions for future work in *Section* 6.

2 Related Work

When it comes to access control for RDF, broadly speaking researchers have focused on representing existing access control models and standards using semantic technology; proposing new access control models suitable for open, heterogeneous and distributed environments; and devising languages and frameworks that can be used to facilitate access control policy specification and maintenance. Kirrane et al. [23] provide a comprehensive survey of existing access control proposals for RDF. Unlike access control, encryption techniques for RDF has received very little attention to date. Giereth [17] demonstrate how public-key encryption techniques can be used to partially encryption RDF data represented using XML. While, Giereth [17] and Gerbracht [16] propose strategies for combining partially encrypted RDF data with the metadata that is necessary for decryption. Kasten et al. [21] propose a framework that can be used to query encrypted data. In order to support SPARQL queries based on triple patterns each triple is encrypted eight times according to the eight different binding possibilities. Limitations of the approach include the blowup associated with maintaining eight ciphers per triple and the fact that the structure of the graph is still accessible.

Searchable Symmetric Encryption (SSE) [9] has been extensively applied in database-as-a-service and cloud environments. SSE techniques focus on the encryption of outsourced data such that an external user can encrypt their query and subsequently evaluate it against the encrypted data. More specifically, SSE extracts the key features of a query (the data structures that allow for its resolution) and encrypts them such that it can be efficiently evaluated on the encrypted data. Extensive work has been done in basic SSE, which caters for a single keyword [6]. Recent improvements have been proposed to handle conjunctive search over multiple keywords [4], and to optimise the resolution to cater for large scale data in the presence of updates [5, 20, 30]. However, all of these works focus on keyword-based retrieval, whereas structured querying (such as SPARQL) over encrypted RDF datasets would require (at least) an unrestricted set of triple query patterns. In contrast, Fully Homomorphic Encryption (FHE) [15] allows any general circuit/computation over encrypted data, however it is prohibitively slow for most operations [7, 28]. Thus, practical, encryption databases such as CryptDB [28] make use of lighter forms of encryption that still cater for computations (such as sums) over the encrypted data [27], at the cost of different vulnerability/feasibility trade-offs. Recently, predicate encryption [22], whereby predicates correspond to the evaluation of disjunctions, polynomial equations and inner products, enables security in light of unrestricted queries. Predicate encryption has has a proven track record of efficiency in terms of conjunctive equality, range and subset queries.

The solution we propose builds on an existing work that defines access control policies based on RDF patterns that are in turn enforced over RDF datasets [23]. While existing proposals enforce access control over plain RDF data via data filtering (i.e., a query is executed against a dataset which is generated by removing the unauthorised data) or query rewriting (i.e., a query is updated so that unauthorised data will not be returned and subsequently executed over the unmodified dataset), we demonstrate how functional encryption can be used to enforce access control over encrypted RDF data in a self-enforcing manner (i.e., without the need for either data filtering or query rewriting nor a trusted mediator). Unlike previous approaches we store one cipher per triple and employ indexing strategies based on secure hashes (cf. PBKDF2 [19]) that can be used for efficient querying of encrypted RDF. In addition, we propose a mechanism to obfuscate the graph structure with real indexes and dummy ciphers that cannot be decrypted, making the dummy hashes and ciphers indistinguishable from real hashes and ciphers.

3 Secure and Fine-grained Encryption of RDF

Common public-key encryption schemes usually follow an all-or-nothing approach (i.e., given a particular decryption key, a ciphertext can either be decrypted or not) which in turn requires users to manage a large amount of keys, especially if there is a need for more granular data encryption [2]. Recent advances in public-key cryptography, however, have led to a new family of encryption schemes called *Functional Encryption (FE)* which addresses aforementioned issue by making encrypted data self-enforce its access restrictions, hence, allowing for fine-grained access over encrypted information. In a functional encryption scheme, each decryption key is associated with a boolean function and each ciphertext is associated with an element of some attribute space Σ; a decryption key corresponding to a boolean function f is able to decrypt a particular ciphertext associated with $I \in \Sigma$ iff $f(I) = 1$. A functional encryption scheme is defined as a tuple of four distinct algorithms (**Setup**, **Enc**, **KeyGen**, **Dec**) such that:

Setup is used for generating a master public and master secret key pair.

Enc encrypts a plaintext message m given the master public key and an element $I \in \Sigma$. It returns a ciphertext c.

KeyGen takes as input the master secret key and generates a decryption key (i.e., secret key) SK_f for a given boolean function f.

Dec takes as input a secret key SK_f and a ciphertext c. It extracts I from c and computes $f(I)$.

3.1 A Functional Encryption Scheme for RDF

While there exist various different approaches for realising functional encryption schemes, we build upon the work of Katz et al. [22] in which functions correspond to the computation of inner-products over $\mathbb{Z}_N$ (for some large integer N). In their construction, they use $\Sigma = \mathbb{Z}_N^n$ as set of possible ciphertext attributes of length n and $\mathcal{F} = \{f_{\vec{x}} | \vec{x} \in \mathbb{Z}_N^n\}$ as the class of decryption key functions. Each ciphertext is associated with a (secret) attribute vector $\vec{y} \in \Sigma$ and each decryption key corresponds to a vector $\vec{x}$ that is incorporated into its respective boolean function $f_{\vec{x}} \in \mathcal{F}$ where $f_{\vec{x}}(\vec{y}) = 1$ iff $\sum_{i=1}^{n} y_i x_i = 0$.

In the following, we discuss how this encryption scheme can be utilised (i.e., its algorithms adopted[4]) to provide fine-grained access over encrypted RDF triples. Thus, allow for querying encrypted RDF using triple patterns such that a particular decryption key can decrypt all triples that satisfy a particular triple pattern (i.e., one key can open multiple locks). For example, a decryption key generated from a triple pattern `(?,p,?)` should be able to decrypt all triples with `p` in the predicate position.

Encrypting RDF Triples (Enc) To be able to efficiently encrypt large RDF datasets, we adopt a strategy commonly used in public-key infrastructures for securely and efficiently encrypting large amounts of data called *Key Encapsulation* [24]. Key encapsulation allows for secure but slow asymmetric encryption to be combined with simple but fast symmetric encryption by using asymmetric encryption algorithms for deriving a symmetric encryption key (usually in terms of a seed) which is subsequently used by encryption algorithms such as AES [11] for the actual encryption of the data. We illustrate this process in Figure 1.

[4] The **Setup** algorithm remains unchanged.

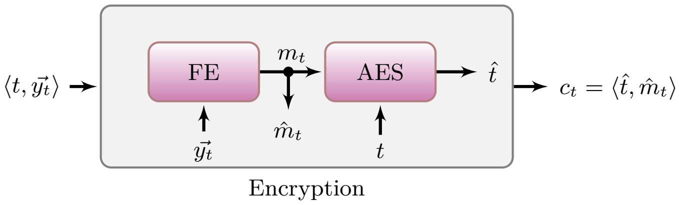

$\langle t, \vec{y_t} \rangle$
FE
m_t
AES
$\hat{t}$
$c_t = \langle \hat{t}, \hat{m}_t \rangle$
$\vec{y_t}$
$\hat{m}_t$
t
Encryption

Fig. 1: Process of encrypting an RDF triple t.

Thus, to encrypt an RDF triple $t = (s, p, o)$, we first compute its respective triple vector (i.e., attribute vector) $\vec{y_t}$ and functionally encrypt (i.e., compute **Enc** as defined in [22]) a randomly generated seed m_t using $\vec{y_t}$ as the associated attribute vector. Triple vector $\vec{y_t}$ where $\vec{y_t} = (y_s, y'_s, y_p, y'_p, y_o, y'_o)$ for triple t is constructed as follows, where σ denotes a mapping function that maps a triple's subject, predicate, and object value to elements in $\mathbb{Z}_N$:

$$y_l := -r \cdot \sigma(l),\ y'_l := r, \text{ with } l \in \{s, p, o\} \text{ and random } r \in \mathbb{Z}_N$$

Table 1 illustrates the construction of a triple vector $\vec{y_t}$ based on RDF triple t.

Triple t	Triple Vector $\vec{y_t}$
t_1 = $(\mathtt{s}_1,\mathtt{p}_1,\mathtt{o}_1)$	$\vec{y}_{t_1} = (-r_1 \cdot \sigma(s_1), r_1, -r_2 \cdot \sigma(p_1), r_2, -r_3 \cdot \sigma(o_1), r_3)$
t_2 = $(\mathtt{s}_2,\mathtt{p}_2,\mathtt{o}_2)$	$\vec{y}_{t_2} = (-r_4 \cdot \sigma(s_2), r_4, -r_5 \cdot \sigma(p_2), r_5, -r_6 \cdot \sigma(o_2), r_6)$
...	...
t_n = $(\mathtt{s}_n,\mathtt{p}_n,\mathtt{o}_n)$	$\vec{y}_{t_n} = (-r_{3n-2} \cdot \sigma(s_n), r_{3n-2}, -r_{3n-1} \cdot \sigma(p_n), r_{3n-1}, -r_{3n} \cdot \sigma(o_n), r_{3n})$

Table 1: Computing the triple vector $\vec{y_t}$ of an RDF triple t.

We use AES to encrypt the actual plaintext triple t with an encryption key derivable from our previously generated seed m_t and return both, the resulting AES ciphertext of t denoted by $\hat{t}$ and the ciphertext of the seed denoted by $\hat{m}_t$ as final ciphertext triple $c_t = \langle \hat{t}, \hat{m}_t \rangle$.

Generating Decryption Keys (KeyGen) As outlined above, decryption keys must be able to decrypt all triples that satisfy their inherent triple pattern (i.e., one query key can open multiple locks). In order to compute a decryption key based on a triple pattern $tp = (s, p, o)$ with s, p, and o either bound or unbound, we define its corresponding vector $\vec{x}$ as $\vec{x}_{tp} = (x_s, x'_s, x_p, x'_p, x_o, x'_o)$ with:

$$\text{if } l \text{ is bound: } x_l := 1, x'_l := \sigma(l), \text{ with } l \in \{s, p, o\}$$
$$\text{if } l \text{ is not bound: } x_l := 0, x'_l := 0, \text{ with } l \in \{s, p, o\}$$

Again, σ denotes a mapping function that maps a triple pattern's subject, predicate, and object value to elements in $\mathbb{Z}_N$. Table 2 illustrates the construction of a query vector $\vec{x}_{tp}$ that corresponds to a triple pattern tp.

Decryption of RDF Triples (Dec) To verify whether an encrypted triple can be decrypted with a given decryption key, we compute the inner-product of their corresponding triple vector $\vec{y_t}$ and query vector $\vec{x}_{tp}$, with $t = (s_t, p_t, o_t)$ and $tp = (s_{tp}, p_{tp}, o_{tp})$:

Triple Pattern tp	Query Vector $\vec{x}_{tp}$
tp_1 = `(?,?,?)`	$\vec{x}_{tp_1} = (0, 0, 0, 0, 0, 0)$
tp_2 = $(\mathtt{s}_2, ?, ?)$	$\vec{x}_{tp_2} = (1, \sigma(s_2), 0, 0, 0, 0)$
tp_3 = $(\mathtt{s}_3, \mathtt{p}_3, ?)$	$\vec{x}_{tp_3} = (1, \sigma(s_3), 1, \sigma(p_3), 0, 0)$
...	...
tp_n = $(\mathtt{s}_n, \mathtt{p}_n, \mathtt{o}_n)$	$\vec{x}_{tp_n} = (1, \sigma(s_n), 1, \sigma(p_n), 1, \sigma(o_n))$

Table 2: Computing the query vector $\vec{x}_{tp}$ that corresponds to a triple pattern tp

$$\vec{y}_t \cdot \vec{x}_{tp} = y_{s_t} x_{s_{tp}} + y'_{s_t} x'_{s_{tp}} + y_{p_t} x_{p_{tp}} + y'_{p_t} x'_{p_{tp}} + y_{o_t} x_{o_{tp}} + y'_{o_t} x'_{o_{tp}}$$

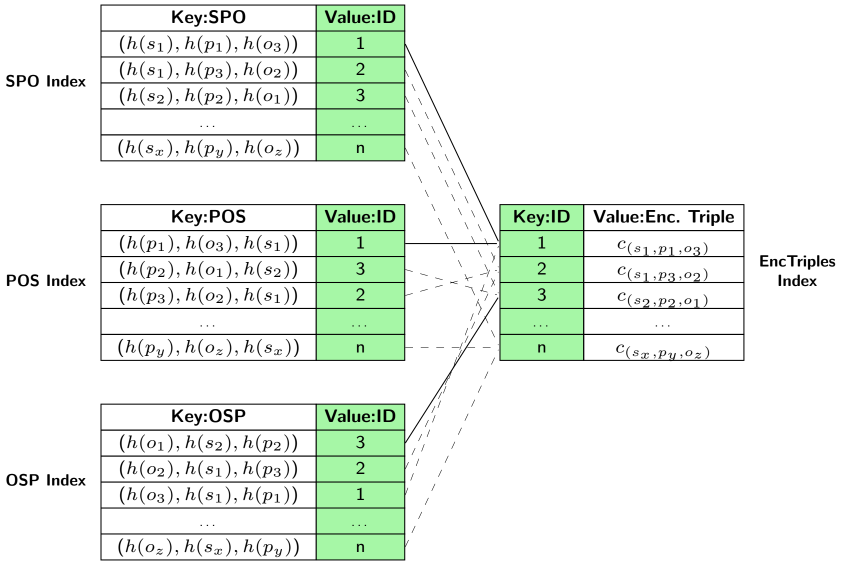
Only when $\vec{y}_t \cdot \vec{x}_{tp} = 0$ is it possible to decrypt the encrypted seed $\hat{m}_t$, hence the corresponding symmetric AES key can be correctly derived and the plaintext triple t be returned. Otherwise (i.e., $\vec{y}_t \cdot \vec{x}_{tp} \neq 0$), an arbitrary seed $m' \neq m_t$ is generated hence encrypted triple c_t cannot be decrypted [26].

4 Optimising Query Execution over Encrypted RDF

The *secure data store* holds all the encrypted triples, i.e. $\{c_{t_1}, c_{t_2}, \cdots, c_{t_n}\}$, being n the total number of triples in the dataset. Besides assuring the confidentiality of the data, the data store is responsible for enabling the querying of encrypted data.

In the most basic scenario, since triples are stored in their encrypted form, a user's query would be resolved by iterating over all triples in the dataset, checking whether any of them can be decrypted with a given decryption key. Obviously, this results in an inefficient process at large scale. As a first improvement one can distribute the set of encrypted triples among different peers such that decryption could run in parallel. In spite of inherent performance improvements, such a solution is still dominated by the available number of peers and the – potentially large – number of encrypted triples each peer would have to process. Current efficient solutions for querying encrypted data are based on (a) using indexes to speed up the decryption process by reducing the set of potential solutions; or (b) making use of specific encryption schemes that support the execution of operations directly over encrypted data [13]. Our solution herein follows the first approach, whereas the use of alternative and directly encryption mechanisms (such as homomorphic encryption [28]) is complementary and left to future work.

In our implementation of such a secure data store, we first encrypt all triples and store them in a key-value structure, referred to as an `EncTriples Index`, where the keys are unique integer IDs and the values hold the encrypted triples (see Figure 2 and Figure 3 (right)). Note that this structure can be implemented with any traditional *Map* structure, as it only requires fast access to the encrypted value associated with a given ID. In the following, we describe two alternative approaches, i.e., one using *three individual indexes* and one based on *Vertical Partitioning (VP)* for finding the range of IDs in the `EncTriples Index` which can satisfy a triple pattern query. In order to maintain simplicity and general applicability of the proposed store, both alternatives consider key-value backends, which are increasingly used to manage RDF data [8], especially in distributed scenarios. It is also worth mentioning that we focus on basic triple



Key:SPO	Value:ID
$(h(s_1), h(p_1), h(o_3))$	1
$(h(s_1), h(p_3), h(o_2))$	2
$(h(s_2), h(p_2), h(o_1))$	3
...	...
$(h(s_x), h(p_y), h(o_z))$	n

Key:POS	Value:ID
$(h(p_1), h(o_3), h(s_1))$	1
$(h(p_2), h(o_1), h(s_2))$	3
$(h(p_3), h(o_2), h(s_1))$	2
...	...
$(h(p_y), h(o_z), h(s_x))$	n

Key:OSP	Value:ID
$(h(o_1), h(s_2), h(p_2))$	3
$(h(o_2), h(s_1), h(p_3))$	2
$(h(o_3), h(s_1), h(p_1))$	1
...	...
$(h(o_z), h(s_x), h(p_y))$	n

Key:ID	Value:Enc. Triple
1	$c_{(s_1,p_1,o_3)}$
2	$c_{(s_1,p_3,o_2)}$
3	$c_{(s_2,p_2,o_1)}$
...	...
n	$c_{(s_x,p_y,o_z)}$

Fig. 2: `3-Index` approach for indexing and retrieval of encrypted triples.

pattern queries as (i) they are the cornerstone that can be used to build more complex SPARQL queries, and (ii) they constitute all the functionality to support the Triple Pattern Fragments [31] interface.

3-Index Approach. Following well-known indexing strategies, such as from CumulusRDF [25], we use three key-value B-Trees in order to cover all triple pattern combinations: `SPO`, `POS` and `OSP Indexes`. Figure 2 illustrates this organisation. As can be seen, each index consists of a *Map* whose keys are the securely hashed (cf. PBKDF2 [19]) subject, predicate, and object of each triple, and values point to IDs storing the respective ciphertext triples in the `EncTriples Index`.

Algorithm 1 shows the resolution of a `(s,p,o)` triple pattern query using the `3-Index` approach. First, we compute the secure hashes `h(s)`, `h(p)` and `h(o)` from the corresponding `s`, `p` and `o` provided by the user (Line 1). Our $hash(s,p,o)$ function does not hash unbounded terms in the triple pattern but treats them as a wildcard '`?`' term (hence all terms will be retrieved in the subsequent range queries). Then, we select the best index to evaluate the query (Line 2). In our case, the `SPO Index` serves `(s,?,?)` and `(s,p,?)` triple patterns, the `POS Index` satisfies `(?,p,?)` and `(?,p,o)`, and the `OSP Index` index serves `(s,?,o)` and `(?,?,o)`. Both `(s,p,o)` and `(?,?,?)` can be solved by any of them. Then, we make use of the selected index to get the range of values where the given `h(s), h(p), h(o)` (or 'anything' if the wildcard '`?`' is present in a term) is stored (Line 3). Note that this search can be implemented by utilising B-Trees [10, 29] for indexing the keys. For each of the candidate ID values in the range (Line 4), we retrieve the encrypted triple for such ID by searching for this ID in the `EncTriples Index` (Line 5). Finally, we proceed with the decryption of

Algorithm 1 `3-Index_Search(s,p,o,key)`

```
1: $(h(s), h(p), h(o)) \leftarrow hash(s, p, o)$;
2: $index \leftarrow selectBestIndex(s, p, o)$;        $\triangleright$ $index = \{SPO|POS|OSP\}$
3: $IDs[\,] \leftarrow index.getRangeValues(h(s), h(p), h(o))$;
4: for each $(id \in IDs)$ do
5:     $encryptedTriple \leftarrow EncTriples.get(id)$;
6:     $< decryptedTriple, status > \leftarrow Decrypt(encryptedTriple, key)$;
7:     if $(status = valid)$ then
8:         output$(decryptedTriple)$;
9:     end if
10: end for
```

the encrypted triple using the `key` provided by the user (Line 6). If the status of such decryption is *valid* (Line 7) then the decryption was successful and we output the decrypted triples (Line 8) that satisfy the query.

Thus, the combination of the three `SPO`, `POS` and `OSP Indexes` reduces the search space of the query requests by applying simple range scans over hashed triples. This efficient retrieval has been traditionally served through tree-based map structures guaranteeing $log(n)$ costs for searches and updates on the data, hence we rely on B-Tree stores for our practical materialisation of the indexes. In contrast, supporting all triple pattern combinations in `3-Index` comes at the expense of additional space overheads, given that each `(h(s),h(p),h(o))` of a triple is stored three times (in each `SPO`, `POS` and `OSP Indexes`). Note, however, that this is a typical scenario for RDF stores and in our case the triples are encrypted and stored just once (in `EncTriples Index`).

Vertical Partitioning Approach. Vertical partitioning [1] is a well-known RDF indexing technique motivated by the fact that usually only a few predicates are used to describe a dataset [14]. Thus, this technique stores one "table" per predicate, indexing `(S,O)` pairs that are related via the predicate. In our case, we propose to use one key-value B-Tree for each `h(p)`, storing `(h(s),h(o))` pairs as keys, and the corresponding ID as the value. Similar to the previous case, the only requirement is to allow for fast range queries on their map index keys. However, in the case of an `SO` index, traditional key-value schemes are not efficient for queries where the first component (the subject) is unbound. Thus, to improve efficiency for triple patterns with unbounded subject (i.e. $(?,p_y,o_z)$ and $(?,?,o_z)$), while remaining in a general key-value scheme, we duplicate the pairs and introduce the inverse `(h(o),h(s))` pairs. The final organisation is shown in Figure 3 (left), where the predicate maps are referred to as `Pred_h(p`$_1$`)`, `Pred_h(p`$_2$`)`,..., `Pred_h(p`$_n$`) Indexes`. As depicted, we add `"so"` and `"os"` keywords to the stored composite keys in order to distinguish the order of the key.

Algorithm 2 shows the resolution of a `(s,p,o)` triple pattern query with the `VP` organisation. In this case, after performing the variable initialisation (Line 1) and the aforementioned secure hash of the terms (Line 2), we inspect the predicate term `h(p)` and select the corresponding predicate index (Line 3), i.e., `Pred_h(p)`. Nonetheless, if the predicate is unbounded, all predicate indexes are selected as we have to iterate through all tables, which penalises the performance

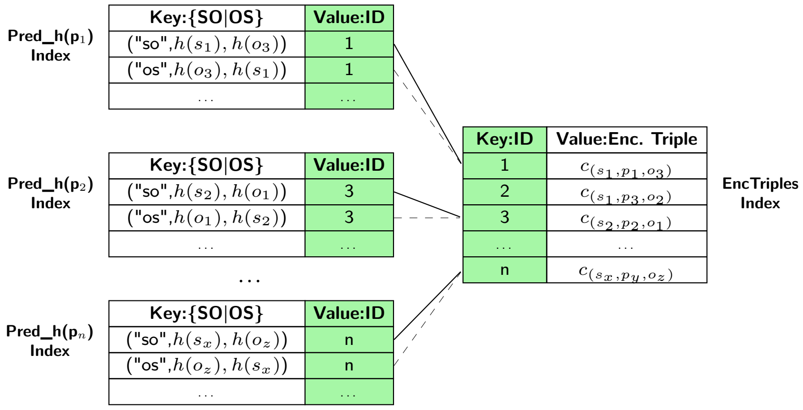


Fig. 3: `Vertical Partitioning (VP)` approach for indexing and retrieval of encrypted triples.

Algorithm 2 `VerticalPartitioning_Search(s,p,o,key)`

```
1: $IDs[\,] \leftarrow ()$;
2: $(h(s), h(p), h(o)) \leftarrow hash(s, p, o)$;
3: $Indexes[] \leftarrow selectPredIndex(h(p))$; ▷ $Indexes \subseteq \{Pred\_h(p_1), \cdots, Pred\_h(p_n)Index\}$
4: for each $(index \in Indexes)$ do
5:     if $(s = ?)$ then
6:         $IDs[\,] \leftarrow index.getRangeValues(\text{"}os\text{"}, h(o), ?)$;
7:     else
8:         $IDs[\,] \leftarrow index.getRangeValues(\text{"}so\text{"}, h(s), h(o))$;
9:     end if
10:    for each $(id \in IDs)$ do
11:        $encryptedTriple \leftarrow EncTriples.get(id)$;
12:        $< decryptedTriple, status > \leftarrow Decrypt(encryptedTriple, key)$;
13:        if $(status = valid)$ then
14:            output$(decryptedTriple)$;
15:        end if
16:    end for
17: end for
```

of such queries. For each predicate index, we then inspect the subject term (Lines 5-9). If the subject is unbounded (Line 5), we will perform a `("os",h(o),?)` range query over the corresponding predicate index (Line 6), otherwise we execute a `("so",h(s),h(o))` range query. Note that in both cases the object could also be unbounded. The algorithm iterates over the candidates IDs (Lines 10-end) in a similar way to the previous cases, i.e., retrieving the encrypted triple from `EncTriples Index` (Line 11) and performing the decryption (Lines 12-14).

Overall, `VP` needs less space than the previous `3-Index` approach, since the predicates are represented implicitly and the subjects and objects are represented only twice. In contrast, it penalises the queries with unbound predicate as it has to iterate through all tables. Nevertheless, studies on SPARQL query logs show that these queries are infrequent in real applications [3].

Protecting the Structure of Encrypted Data. The proposed hash-based indexes are a cornerstone for boosting query resolution performance by reducing

Dataset	Triples	|S|	|P|	|O|	Size (MB)
Census	361,842	51,768	26	6,901	52
Jamendo	1,049,637	335,925	26	440,602	144
AEMET	3,547,154	394,289	23	793,664	726
LUBM	100,000	22,932	18	11,588	15
	200,000	39,244	18	23,749	29
	500,000	87,984	18	60,028	71
	1,000,000	169,783	18	120,464	139
	2,000,000	333,105	18	241,342	277
	5,000,000	820,185	18	604,308	694

Table 3: Statistical dataset description.

the encrypted candidate triples that may satisfy the user queries. The use of secure hashes [19] assures that the terms cannot be revealed but, in contrast, the indexes themselves reproduce the structure of the underlying graph (i.e., the in/out degree of nodes). However, the structure should also be protected as hash-based indexes can represent a security risk if the data server is compromised. State-of-the-art solutions (cf., [13]) propose the inclusion of spurious information, that the query processor must filter out in order to obtain the final query result.

In our particular case, this technique can be adopted by adding dummy triple hashes into the indexes with a corresponding ciphertext (in `EncTriples Index`) that cannot be decrypted by any key, hence will not influence the query results. Such an approach ensures that both the triple hashes and their corresponding ciphertexts are not distinguishable from real data.

5 Evaluation

We develop a prototypical implementation[5] of the proposed encryption and indexing strategies. Our tool is written in Java and it relies on the Java Pairing-Based Cryptography Library (JPBC [12]) to perform all the encryption/decryption operations. While, we use MapDB[6] as the supporting framework for the indexes. We provide an interface that takes as input a triple pattern query and a query key, and outputs the results of the query.

We evaluate our proposal in two related tasks: (i) performance of the data loading (encryption and indexing) and (ii) performance of different user queries (query execution on encrypted data). In both cases, we compare our proposed `3-Index` strategy w.r.t the vertical partitioning (`VP`) approach. Finally, we measure the performance overhead associated with query resolution, introduced by the secure infrastructure, by comparing its results with a counterpart non-secure triplestore. For a fair comparison, we implement the non-secure triplestore with similar `3-Index` and `VP` indexing strategies, storing the RDF data in plain. The approaches are referred to as `3-Index-plain` and `VP-plain` respectively.

Table 3 describes our experimental datasets, reporting the number of triples, different subjects (|S|), predicates (|P|) and objects (|O|), as well as the file size (in NT format). Note that there is no standard RDF corpus that can be used to

[5] Source code and experimental datasets are available at: https://aic.ai.wu.ac.at/comcrypt/sld/.

[6] http://www.mapdb.org/

evaluate RDF encryption approaches, hence we choose a diverse set of datasets that have been previously used to benchmark traditional RDF stores or there is a use case that indicates they could potentially benefit from a secure data store. On the one hand, we use the well-known Lehigh University Benchmark (LUBM [18]) data generator to obtain synthetic datasets of incremental sizes from 100K triples to 5M triples. On the other hand, we choose real-world datasets from different domains: `Census` represents the 2010 Australian census, where sensitive data must be preserved and users could have different partial views on the dataset; `Jamendo` lists music records and artists, where some data can be restricted to certain subscribers; and `AEMET` includes sensor data from weather stations in Spain, which is a real use case where the old data is public but the most recent data is restricted to particular users. Tests were performed on a computer with 2 x Intel Xeon E5-2650v2 @ 2.6 GHz (16 cores), RAM 171 GB, 4 HDDs in RAID 5 config. (2.7 TB netto storage), Ubuntu 14.04.5 LTS running on a VM with QEMU/KVM hypervisor. All of the reported (elapsed) times are the average of three independent executions.

Data loading. Figure 4 shows the dataset load times[7] for the `3-Index` and `VP` strategies. The reported time consists of the time to encrypt the triples using the aforementioned FE scheme, and the time to securely hash the terms and create the different indexes. In contrast, the non-secure triplestores, i.e. the `3-Index-plain` and `VP-plain` counterparts, only require the dataset to be indexed (we also make use of the hash of the terms in order to compare the encryption overhead).

The results show that the time of both the `3-Index` and the `VP` strategy scales linearly with the number of triples, which indicates that the representation can scale in the envisioned Linked Data scenario. It is worth noting that both strategies report similar performance results, where `VP` is slightly faster for loading given that only the subject and object is used to index each triple (the predicate is implicitly given by vertical partitioning). Finally, note that the comparison w.r.t the plain counterparts shows that the encryption overhead can be of one order of magnitude for the smaller datasets. In contrast, the encryption overhead is greatly reduced for larger datasets which is primarily due to the fact that the loading time for large datasets is the predominant factor, as the B-Tree indexes become slower the more triples are added (due to rebalancing).

Query resolution. Figure 5 shows the query resolution time for two selected datasets[8], LUBM with 5M triples and Jamendo, considering all types of triple patterns. To do so, we sample 1,000 queries of each type and report the average resolution time. As expected, the `3-Index` reports a noticeable better performance than `VP` for queries with unbound predicates given that `VP` has to iterate though all predicate tables in this case. In turn, the `3-Index` and the `VP` approaches remain competitive with respect to their non-secure counterparts, if a look-up returns only a small amount of results as it is usually the case for `(s,?,?)`, `(s,?,o)`, `(s,p,o)` queries. However, the more query results that need

[7] We use name abbreviations for LUBM (L), Census (C), Jamendo (J), and AEMET (A).

[8] Results are comparable for all datasets.

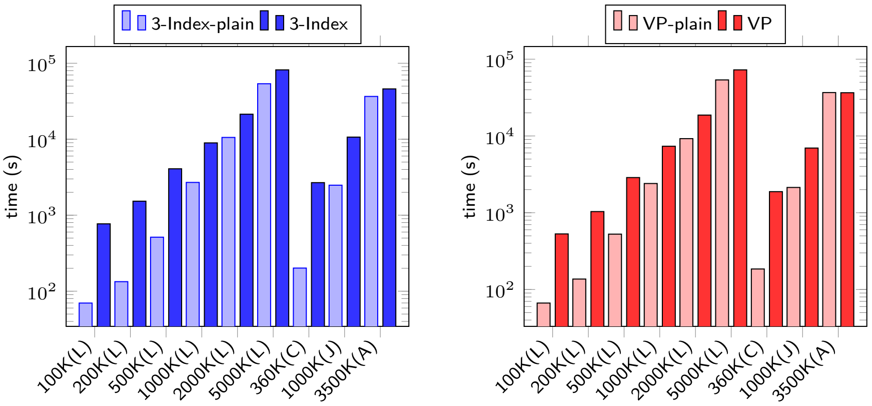


Fig. 4: Time for loading (encrypting+indexing) the entire dataset for `3-Index` and `VP`. We only report indexing time for the non-secure counterparts `3-Index-plain` and `VP-plain`.

to be returned the longer the decryption takes. At this point we also want to stress that due to the nature of our approach, each result triple can be returned as soon as its decryption has finished. This is in line with the incremental nature of the Triple Pattern Fragments [31] approach, which paginates the query results (typically including 100 results per page), allowing users to ask for further pages if required. For example, decrypting Jamendo entirely took about 2256s for `VP` and 2808s for `3-Index`, leading to respective triple decryption rates of 465 triples/s and 374 triples/s in a cold scenario, which already fulfils the performance requirements to feed several Triple Pattern Fragments per second.

Scalability. As mentioned in Section 4, our approach allows for parallel encryption/decryption of triples, thus scales with the system's supported level of parallelisation/number of available cores (e.g., encrypting and indexing (`3-Index`) 10,000 LUBM triples takes about 76s with 16 available cores, 133s with 8, 262s with 4, and 497s with 2 available cores).

Our experiments have shown that (i) the performance of our indexing strategy is not affected by the encryption, hence, is as effective on encrypted data as it is on non-encrypted data, and (ii) the decryption of individual triples is a fast process which can be utilised in our Linked Data scenario, especially under the umbrella of the Linked Data Fragments framework.

6 Conclusion

To date Linked Data publishers have mainly focused on exposing and linking open data, however there is also a need to securely store, exchange, and query also sensitive data alongside (i.e., closed data). Both access control and encryption mechanisms are needed to protect such data from unauthorised access, security breaches, and potentially untrusted service providers. Herein, we presented a

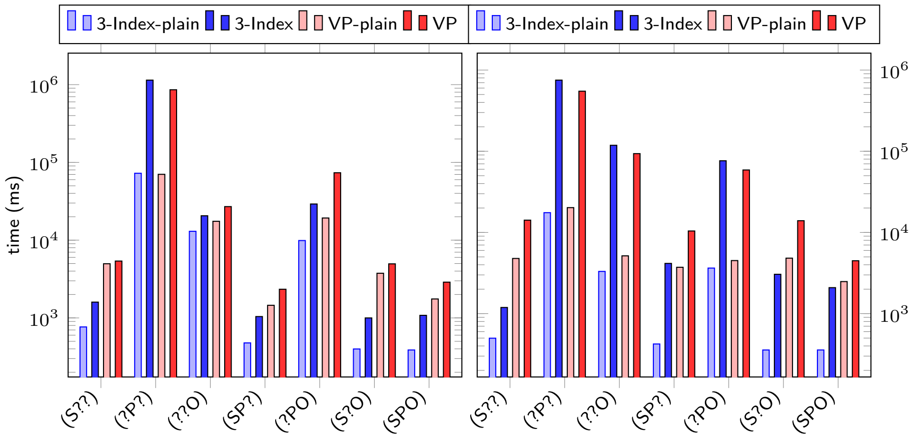


Fig. 5: Cold query times of LUBM with 5M triples (LHS) and Jamendo (RHS) for `3-Index`, `VP`, and their non-secure counterparts in ms (logarithmic y-axis).

mechanism to provide secure and fine-grained encryption of RDF datasets. First, we proposed a practical realisation of a functional encryption scheme, which allows data providers to generate query keys based on (triple-)patterns, whereby one decryption key can decrypt all triples that match its associated triple pattern. As such, our approach operates on a very fine level of granularity (i.e., triple level), which provides a high degree of flexibility and enables controlled access to encrypted RDF data. In existing literature, enforcing access control at the level of single statements or tuples is generally referred to as fine-grained access control (cf. [23]). Then, we presented two indexing strategies (implemented using MapDB) to enhance query performance, the main scalability bottleneck when it comes to serving user requests.

Our empirical evaluation shows that both indexing strategies on encrypted RDF data report reasonable loading and query performance overheads with respect to traditional, non-encrypted data retrieval. Our results also indicate that the approach is relatively slow for batch decryption, but this can be counteracted by the fact that it is suitable for serving incremental results, hence it is particularly suitable for Linked Data Fragments.

In future work, we plan to inspect different indexing strategies in order to optimise the loading time and query performance of large queries. We also consider extending our proposal to cater for named graphs, that is, encrypting quads instead of triples and generating keys based on quad patterns. Finally, we aim to integrate the proposed secure RDF store with a “policy” tier by employing Attribute-based Access Control (ABAC), which will manage the access/revocation to the query keys and serve as fully fledged security framework for Linked Data.

References

[1] D. J. Abadi, A. Marcus, S. R. Madden, and K. Hollenbach. Scalable semantic web data management using vertical partitioning. In *Proc. of Very Large Data Bases*, pages 411–422, 2007.

[2] M. Abdalla, F. Bourse, A. D. Caro, and D. Pointcheval. Simple functional encryption schemes for inner products. In *Proc. of the 18th International Conference on Practice and Theory in Public-Key Cryptography*, pages 733–751, 2015.

[3] M. Arias, J. D. Fernández, M. A. Martínez-Prieto, and P. de la Fuente. An empirical study of real-world sparql queries. *arXiv preprint arXiv:1103.5043*, 2011.

[4] D. Cash, S. Jarecki, C. Jutla, H. Krawczyk, M.-C. Roşu, and M. Steiner. Highly-scalable searchable symmetric encryption with support for boolean queries. In *Proc. of Advances in Cryptology*, pages 353–373. 2013.

[5] D. Cash, J. Jaeger, S. Jarecki, C. S. Jutla, H. Krawczyk, M.-C. Rosu, and M. Steiner. Dynamic searchable encryption in very-large databases: Data structures and implementation. *IACR Cryptology ePrint Archive*, 2014:853, 2014.

[6] Y.-C. Chang and M. Mitzenmacher. Privacy preserving keyword searches on remote encrypted data. In *Proc. of Applied Cryptography and Network Security*, pages 442–455, 2005.

[7] M. Chase and E. Shen. Pattern matching encryption. *IACR Cryptology ePrint Archive*, 2014:638, 2014.

[8] P. Cudré-Mauroux, I. Enchev, S. Fundatureanu, P. Groth, A. Haque, A. Harth, F. L. Keppmann, D. Miranker, J. F. Sequeda, and M. Wylot. NoSQL databases for RDF: an empirical evaluation. In *Proc. of International Semantic Web Conference*, pages 310–325, 2013.

[9] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky. Searchable symmetric encryption: improved definitions and efficient constructions. In *Proc. of Computer and communications security*, pages 79–88, 2006.

[10] P. da Rocha Pinto, T. Dinsdale-Young, M. Dodds, P. Gardner, and M. J. Wheelhouse. A simple abstraction for complex concurrent indexes. In *Proc. of Object-Oriented Programming, Systems, Languages, and Applications*, pages 845–864, 2011.

[11] J. Daemen and V. Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard.* Information Security and Cryptography. Springer, 2002.

[12] A. De Caro and V. Iovino. jpbc: Java pairing based cryptography. In *Proc. of IEEE Symposium on Computers and Communications*, pages 850–855, 2011.

[13] S. D. C. di Vimercati, S. Foresti, G. Livraga, and P. Samarati. Practical techniques building on encryption for protecting and managing data in the cloud. In *The New Codebreakers - Essays Dedicated to David Kahn on the Occasion of His 85th Birthday*, pages 205–239, 2016.

[14] J. D. Fernández, M. A. Martínez-Prieto, C. Gutiérrez, A. Polleres, and M. Arias. Binary RDF representation for publication and exchange (HDT). *J. Web Sem.*, 19:22–41, 2013.

[15] C. Gentry et al. Fully homomorphic encryption using ideal lattices. In *Proc. of ACM Symposium on Theory of Computing*, volume 9, pages 169–178, 2009.

[16] S. Gerbracht. Possibilities to Encrypt an RDF-Graph. In *Proc. of Information and Communication Technologies: From Theory to Applications*, pages 1–6, 2008.

[17] M. Giereth. On Partial Encryption of RDF-Graphs. In *Proc. of International Semantic Web Conference*, volume 3729, pages 308–322, 2005.

[18] Y. Guo, Z. Pan, and J. Heflin. LUBM: A Benchmark for OWL Knowledge Base Systems. *Journal of Web Semantics*, 3(2):158–182, 2005.

[19] B. Kaliski. PKCS #5: Password-Based Cryptography Specification Version 2.0. RFC 2898 (Informational), September 2000.

[20] S. Kamara and C. Papamanthou. Parallel and dynamic searchable symmetric encryption. In *Financial cryptography and data security*, pages 258–274. 2013.

[21] A. Kasten, A. Scherp, F. Armknecht, and M. Krause. Towards search on encrypted graph data. In *Proc. of the International Conference on Society, Privacy and the Semantic Web-Policy and Technology*, pages 46–57, 2013.

[22] J. Katz, A. Sahai, and B. Waters. Predicate Encryption Supporting Disjunctions, Polynomial Equations, and Inner Products. *J. Cryptology*, 26(2): 191–224, 2013.

[23] S. Kirrane, A. Mileo, and S. Decker. Access control and the resource description framework: A survey. *Semantic Web*, 8(2):311–352, 2017. doi: 10.3233/SW-160236. URL http://dx.doi.org/10.3233/SW-160236.

[24] K. Kurosawa and L. T. Phong. Kurosawa-desmedt key encapsulation mechanism, revisited. *IACR Cryptology ePrint Archive*, 2013:765, 2013.

[25] G. Ladwig and A. Harth. CumulusRDF: linked data management on nested key-value stores. In *Proc. of Scalable Semantic Web Knowledge Base Systems*, page 30, 2011.

[26] A. B. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In *Advances in Cryptology*, pages 62–91, 2010.

[27] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology*, pages 223–238, 1999.

[28] R. Popa, N. Zeldovich, and H. Balakrishnan. Cryptdb: A practical encrypted relational dbms. Technical report, MIT-CSAIL-TR-2011-005, 2011.

[29] Y. Sagiv. Concurrent Operations on B*-Trees with Overtaking. *J. Comput. Syst. Sci.*, 33(2):275–296, 1986.

[30] E. Stefanov, C. Papamanthou, and E. Shi. Practical dynamic searchable encryption with small leakage. In *Proc. of Network and Distributed System Security*, volume 14, pages 23–26, 2014.

[31] R. Verborgh, M. Vander Sande, O. Hartig, J. Van Herwegen, L. De Vocht, B. De Meester, G. Haesendonck, and P. Colpaert. Triple Pattern Fragments: a low-cost knowledge graph interface for the Web. *Journal of Web Semantics*, 37–38:184–206, Mar. 2016.

[32] E. Zheleva and L. Getoor. To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles. In *Proc. of World Wide Web*, pages 531–540, 2009.

13. HDTcrypt: Compression and encryption of RDF datasets

Bibliographic Information

Fernández, J.D., **Kirrane, S.**, Polleres, A. and Steyskal, S., 2020. HDTcrypt: Compression and encryption of RDF datasets. Semantic Web, 11(2), pp.337-359.

Habilitation Candidate CRediT roles

Conceptualisation, Methodology, Formal analysis, Investigation, Writing - Original Draft, Writing - Review & Editing, Supervision, Project administration, and Funding acquisition.

Copyright Notice

Semantic Web 0 (0)
IOS Press

HDT$_{crypt}$: Compression and Encryption of RDF Datasets

Javier D. Fernández [a,b,*], Sabrina Kirrane [a], Axel Polleres [a,b] and Simon Steyskal [a,c]

[a] *Institute for Information Business, Vienna University of Economics and Business, Austria*
E-mail: {javier.fernandez,sabrina.kirrane,axel.polleres,simon.steyskal}@wu.ac.at
[b] *Complexity Science Hub Vienna, Austria*
[c] *CT RDA BAM CON-AT, Siemens AG, Austria*
E-mail: simon.steyskal@siemens.com

Abstract. The publication and interchange of RDF datasets online has experienced significant growth in recent years, promoted by different but complementary efforts, such as Linked Open Data, the Web of Things and RDF stream processing systems. However, the current Linked Data infrastructure does not cater for the storage and exchange of sensitive or private data. On the one hand, data publishers need means to limit access to confidential data (e.g. health, financial, personal, or other sensitive data). On the other hand, the infrastructure needs to compress RDF graphs in a manner that minimises the amount of data that is both stored and transferred over the wire. In this paper, we demonstrate how HDT – a compressed serialization format for RDF – can be extended to cater for supporting encryption. We propose a number of different graph partitioning strategies and discuss the benefits and tradeoffs of each approach.

Keywords: RDF, HDT, compression, encryption, linked data protection

1. Introduction

In recent years, we have seen an increase in the amount of structured data published online using the Resource Description Framework (RDF), in a manner that not only lends itself to data integration but also supports data exchange. Although Linked Data publishers focus on exposing and linking *open* data, there are scenarios where individuals and organisations need to store and share sensitive or private data. Additionally, there are number of regulations concerning the financial, medical, personal, or otherwise sensitive data that require companies to employ strong data protection mechanisms, such as encryption and anonymisation. In order to ensure confidentially it is necessary to encrypt the data not only when it is *in transit* but also when it is *at rest*. In such scenarios, where multiple users have different access rights to different parts of the data, users should only be able to access the data they are allowed to access.

When it comes to Linked Data protection, to date research has focused on the encryption of partial RDF graphs using eXtensible Markup Language (XML) encryption techniques [20–22] or proposing strategies for querying encrypted RDF data [31]. One of the primary challenges of existing encryption strategies is that they result in a verbose serialization that prevents their use at scale. RDF compression is an emerging research area that focuses on reducing the space requirements of traditional RDF serializations. One approach to efficient data exchange is a (binary) RDF serialization format known as HDT (Header Dictionary Triples) [15] that can be used to compress large datasets in a manner than can be queried without prior decompression [37]. Together encryption and compression mechanisms could be used to cater for the compact storage and efficient exchange of confidential data.

In this paper, we combine "compression+encryption" functionality for RDF datasets, thus allowing service providers to store and share confidential data while reducing storage and bandwidth usage. In particular, we propose HDT$_{crypt}$, an extension of HDT to

[*] Corresponding author. E-mail: javier.fernandez@wu.ac.at.

represent encrypted datasets for multiple users with different access rights (i.e. users can only access particular subgraphs of the RDF dataset). To do so, we assume a service provider defines the different "access restricted" subgraphs of a dataset, and we investigate different partitioning strategies to better capture and represent the redundancy (i.e. repeated triples and terms) between them in HDT.

The contributions of our paper can be summarised as follows, we: (i) demonstrate how HDT compression can be extended to cater for encrypted RDF data; (ii) examine a number of alternative partitioning strategies that can be used to reduce the number of duplicates in encrypted HDT (referred to as HDT$_{crypt}$); and (iii) compare different partitioning strategies in terms of bandwidth and performance. Experiments show that each of our partitioning strategies is able to achieve space savings over the compression baseline (up to 31%), and are comparable in terms of query performance. We present different space/performance tradeoffs and discuss how partitioning strategies are influenced both by the number of access restricted subgraphs and the distribution of triples across subgraphs.

The rest of the paper is structured as follows: In *Section* 2 we discuss related work on RDF encryption and compression. *Section* 3 provides the necessary background information on HDT and *Section* 4 describes how compression can be combined with encryption. *Section* 5 details the different partitioning strategies that can be used in conjunction with graph based encryption. In *Section* 6 we evaluate using both real-world and synthetic RDF datasets and discusses the trade-off between space and performance. Finally, we conclude and highlight future work in *Section* 7.

2. Related Work

When it comes to encryption and RDF, the focus to date has been on proposing strategies for the partial encryption of RDF graphs [20–22] or the querying of encrypted data [31]. Giereth [21, 22] demonstrate how XML based encryption techniques can be used to encrypt confidential data in an RDF-graph, while all non-confidential data is left as plaintext. Gerbracht [20] built on this work by examining how encryption techniques can be used to encrypt RDF elements and RDF subgraphs, in a manner that reduces the storage overhead. Kasten et al. [31] in turn discuss how data can be encrypted and queried according to SPARQL triple patterns. However this proposal suffers from scalability problems given that each triple is encrypted multiple times depending on whether or not access to the subject, predicate and/or object is restricted. A recent work by Fernández et al. [16] uses Predicate-based Encryption [32] to enable controlled access to encrypted RDF data, i.e., data providers can generate query keys based on (triple-)patterns, whereby one decryption key can decrypt all triples that match its associated triple pattern. In the database and cloud community, Searchable Symmetric Encryption (SSE) [10] has been extensively applied to store and search data in a secure manner. SSE techniques focus on the encryption of outsourced data such that an external user can encrypt their query and subsequently evaluate it against the encrypted data. The more recent Fully Homomorphic Encryption (FHE) [19] technique allows any general circuit/computation over encrypted data, however it is prohibitively slow for most operations [7, 42]. None of these works examine the interplay between encryption and compression, which is the focus of our present paper. In particular, we investigate different HDT compression strategies for RDF datasets, which are organised into different RDF graphs that need to be encrypted with different keys. However, our approach could be adapted to work with partially encrypted graphs.

Following the categorization in [39], an RDF compressor can be classified as either *syntactic* or *semantic*. *Syntactic* compressors try to detect redundancy at the serialisation level, whereas *semantic* compressors try to eliminate logical redundancies. HDT was designed as a binary serialisation format for RDF graphs, but its optimised encodings means that HDT also excels as a syntactic RDF compressor [15, 37]. In HDT RDF data is encoded into two main data-driven components: a *Dictionary* that maps all distinct terms in the dataset to unique identifiers (IDs) (reducing symbolic redundancy), and a triple component that encodes the inner RDF structure as a compact graph of IDs (reducing structural redundancy). This kind of redundancy is also addressed in k^2-triples [1]. However, in the case of k^2-triples the authors perform a predicate-based partition of the dataset into disjoint subsets of (subject, object) pairs. These subsets are highly compressed as (sparse) binary matrices that also allow for efficient data retrieval. RDF compression can also benefit from semantic redundancy. Theoretic foundations of exploiting logical redundancies with respect to rules and grammars have been investigated by [41] and [36], respectively. In particular, the recent compressor gRePair [36] reports the best compression ratios over the structure of RDF graphs (i.e. the graph after ID replacement), to the best of our knowledge.

Likewise, Joshi et al. [29] use rules to discard triples that can be inferred from others, and they only encode these "primitive triples". In doing so they reduce the number of triples and consequently save space. The authors also propose a combination of semantic and syntactic compression, by integrating their approach with syntactic HDT compression techniques. Interestingly the results were similar to those obtained by simply using HDT. Recently, Wu *et al.* [39] have proposed SSP, a hybrid syntactic and semantic compressor. Their evaluation demonstrates that SSP+bzip2 is slightly better than HDT+bzip2. Other approaches, like HDT-FoQ [37] or WaterFowl [9] also enable compressed data to be retrieved without the need for decompression. Both techniques, based on HDT serialization, report competitive performance at the price of using more space than other compressors such as k^2-triples or gRePair.We also use HDT compression, however specifically we examine the syntactic redundancy between RDF graphs that need to be encrypted separately, and propose and evaluate four alternative HDT compression strategies. The exploitation of semantic redundancies within HDT is out of scope and left for future work (for more details on semantic compression and HDT we refer the reader to the work by Hernández-Illera et al. [27]).

3. Preliminaries

Before we present our approach, we need to introduce some concepts and terminology from RDF and HDT. Thereafter, in Section 4, we propose a general mechanism to extend HDT with encryption, termed HDT$_{crypt}$.

As usual, an *RDF Graph G* is a finite set of triples from $I \cup B \times I \times I \cup B \cup L$, where I, B, L denote IRIs, blank nodes and RDF literals, respectively [25]. Figure 1 shows an example of an RDF graph representing two individuals ex:Bob and ex:Alice, and the project ex:pastProject of the latter. In this paper, we discuss different ways to compress and encrypt such datasets, using HDT a particular compression format for RDF graphs.

HDT [15] is a binary, compressed serialization format for optimized RDF storage and transmission, which also allows certain lookups and queries over compressed data. It is therefore very suitable for the efficient exchange and querying of large datasets. HDT encodes an RDF graph G into three components: the *Header* component H holds metadata, including relevant information necessary for discovery and parsing; the *Dictionary* component D is a catalogue that encodes all RDF terms in G and maps each of them to a unique identifier; the *Triple* component T compactly encodes G's graph structure as tuples of three IDs that are used to represent the directed labelled edges in an RDF graph.

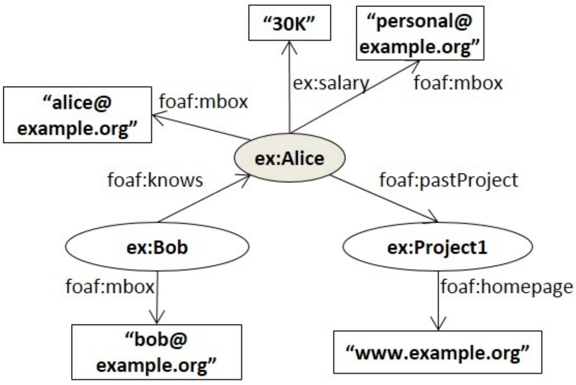


Fig. 1. Example of an RDF graph G.

Figure 2 shows the *Dictionary* component (a), the underlying graph structure (b) and the final *Triple* component (c) for the previous RDF graph G (Figure 1).

3.1. HDT Dictionary Component D

This component organises the terms in a graph G according to their positions in RDF triples, thus we also write $D(G)$ to denote the dictionary component constructed from graph G: the section *SO* manages terms occurring both as subject and object, and maps them to the ID-range `[1, |SO|]`, where `|SO|` is the number of such terms acting as subject and object. Sections S and O comprise terms that only occur as subjects or objects, respectively. Both sections are mapped from `|SO|+1`, ranging up to `|SO|+|S|` and `|SO|+|O|`, respectively. Finally, section P organises all predicate terms, which are mapped to the range `[1, |P|]`. It is worth noting that no ambiguity is possible once we know the role (i.e. the position in a triple, being subject, predicate or object) played by the corresponding ID. For further details, we refer to [38]. For convenience, we write $id(x, D)$ for the particular ID assigned to an RDF term x, whereas we refer to all IDs and RDF terms mapped in a dictionary component D as $ids(D)$ and $terms(D)$, respectively. Note that, for simplicity, we omit the "role" parameter in these functions, which

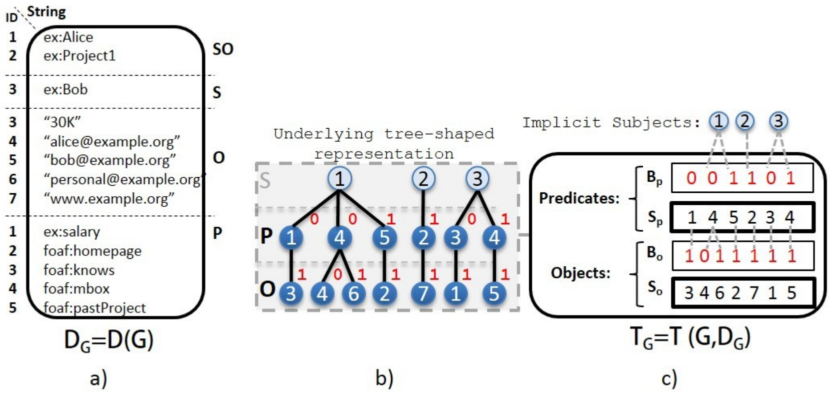


Fig. 2. HDT *Dictionary* and *Triples* for our full graph G.

should be provided in case the terms in subjects (or objects) and predicates are not disjoint [38]. Also, it is worth mentioning that in the original HDT proposal, blank nodes are treated exactly as any other term [15], considering an optional skolemization of blank nodes as a pre-processing step.

3.2. HDT Triple Component T

This component encodes the *structure* of the RDF graph after ID substitution, taking into consideration a particular dictionary D, thus, we write $T(G,D)$ to denote a triple component that was constructed from the triples in G using the IDs in dictionary D. More concretely, RDF triples are encoded as groups of three IDs: $(\mathtt{id}_s\ \mathtt{id}_p\ \mathtt{id}_o)$, where $\mathtt{id}_s$, $\mathtt{id}_p$, and $\mathtt{id}_o$ are the IDs of the corresponding subject, predicate, and object terms in the dictionary. T organises all triples into a forest of trees, one per different subject: the subject is the root; the middle level comprises the ordered list of predicates reachable from the corresponding subject; and the leaves list the object IDs related to each (subject, predicate) pair. This *underlying representation* (illustrated in Figure 2b) is effectively encoded following the *BitmapTriples* approach [15]. It comprises *two sequences*: Sp and So, concatenating all predicate IDs in the middle level and all object IDs in the leaves, respectively; and *two bitsequences*: Bp and Bo, which are aligned with Sp and So respectively, using a 1-bit to mark the end of each list (Figure 2c). In practice, each ID in Sp and So is encoded with a fixed-length encoding, using log(n) bits, where n is the maximum ID in the sequence [15]. Again, we use $ids(T)$ to refer to all IDs used in a triple component T.

3.3. HDT Header Component H

The HDT Header includes (i) the machine-readable metadata that is necessary to process an HDT file (*format metadata*); and (ii) additional human-readable information to describe the dataset (usually in the form of VoID[1] descriptions). The format metadata is mainly focused on characterising the dictionary and triple formats. In general, an HDT file of a graph G consists of a single header H, dictionary D and triples T, $HDT(G) = (H,D,T)$. Nonetheless, the HDT specification [17] is flexible and allows for several dictionaries or triple components to be specified in H as soon as the interpretation of their relationship is provided in the header. It was envisaged that this would be used to split huge RDF graphs into several chunks or streams, where a sequential order of the components is assumed by default [17]. In the following section we exploit and expand this feature to encode a partition of the graph G with several dictionaries and triples.

4. HDT$_{crypt}$: Extending HDT for Encryption

We introduce HDT_{crypt}, an extension of HDT that involves encryption of RDF graphs. We first define the notion of access-restricted RDF datasets and the implications for HDT (Section 4.1). Then, we show an extension of the HDT header component to cope with access-restricted RDF datasets (Section 4.2), which leads to the final HDT_{crypt} encoding. Finally, as HDT_{crypt} can manage several HDT Dictionary components, we describe the required operations to integrate

[1] http://www.w3.org/TR/void/

different Dictionary components within an HDT collection (Section 4.3). These operations will be the basis to represent the shared components between access-restricted datasets efficiently, addressed in Section 5.

4.1. Representing access-restricted RDF datasets

We consider hereinafter that users wishing to publish *access-restricted RDF datasets* divide their complete graph of RDF triples G into (named) graphs, that are accessible to other users, i.e. we assume that access rights are already materialised per user group in the form of a set (cover) of separate, possibly overlapping, RDF graphs, each of which are accessible to different sets of users.

Borrowing terminology from [26], an *access restricted RDF dataset* (or just "dataset" in the following) is a set $DS = \{G, (g_1, G_1), \dots, (g_n, G_n)\}$ consisting of a (non-named) default graph G and named graphs s.t. $g_i \in I$ are graph names, where in our setting we require that $\{G_1, \dots, G_n\}$ is a cover[2] of G. We further call DS a *partition* of G if $G_i \cap G_j = \emptyset$ for any $i \neq j$; $1 \leqslant i, j \leqslant n$. Note that from any dataset DS, a *canonical partition* DS' can be trivially constructed (but may be exponential in size) consisting of all non-empty (at most $2^n - 1$) subsets G'_S of triples $t \in G$ corresponding to an index set $S \in 2^{1,\dots,i}$ such that $G'_S = \{t \mid t \in \bigcap_{i \in S} G_i \wedge \neg\exists S' : (S' \supset S \wedge t \in \bigcap_{j \in S'} G_j)\}$.

Figure 3 shows an example of such a dataset composed of three access-restricted subgraphs (or just "subgraphs" in the following) G_1, G_2, G_3 for the previous full graph G (Figure 2a). Intuitively, this corresponds to a scenario with three access rights: users who can access general information about projects in an organisation (graph G_1); users who have access to public email accounts and relations between members in the organisation (graph G_2); and finally, users who can view personal information of members, such as the salary and personal email accounts (graph G_3). As can be seen, the triple (ex:Alice foaf:mbox "alice@example.org") is repeated in subgraphs G_2 and G_3, showing a redundancy which can produce significant overheads in realistic scenarios with large-scale datasets and highly overlapping graphs. Canonical partitioning groups these triples into disjoint sets so that no repetitions are present. In our example in Figure 3, the set $G'_{\{2,3\}}$, which can simply be written as G'_{23}, holds this single triple, (ex:Alice foaf:mbox "alice@example.org"), hence this triple is not present in G'_2 and G'_3. In this simple scenario, G'_1 is equivalent to G_1 as it does not share triples with other graphs.

Thus, we consider hereinafter an *HDT collection* corresponding to a dataset DS denoted by $HDT(DS) = (H, \mathcal{D}, \mathcal{T})$ as a single H, plus sets $\mathcal{D} = \{D_1, \dots, D_n\}$, $\mathcal{T} = \{T_1, \dots, T_m\}$ of dictionary and triple components, respectively, such that the union of triple components encodes a cover of G, i.e. the overall graph of all triples in the dataset DS. We do not assume that there is a one-to-one correspondence between individual triple components in $\mathcal{T}$ and graphs in DS; different options of mapping subgraphs to HDT components will be discussed in Section 5 below. The relation between the dictionaries and the triple components (in other words, which dictionaries are used to codify which triple components) is also flexible and must be specified through metadata properties. In our case, we assume $H = \{R, M\}$ to contain a relation $R \subseteq \mathcal{D} \times \mathcal{T}$, which we call the *dictionary-triples map* with the implicit meaning that dictionary components encode terms used in the corresponding triple components, and M is comprised of additional header metadata (as mentioned above, the header contains a variety of further (meta-)information in standard HDT [17], which we skip for the considerations herein). It is worth noting that we do not prescribe that either D or T do not overlap. However, it is clear that one should find an unambiguous correspondence to decode the terms under $ids(T)$.

Thus, we define the following admissibility condition for R. An HDT collection is called *admissible* if:

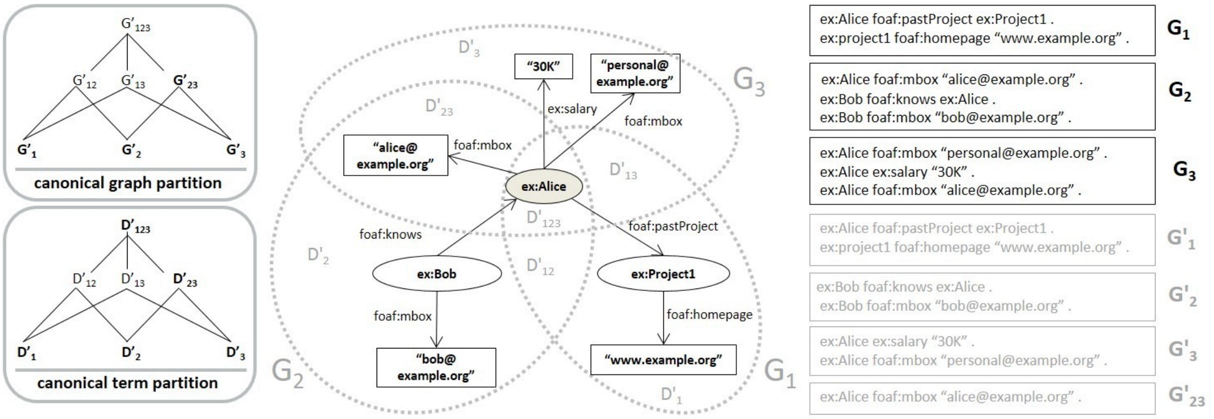
- $\forall D_i, D_j \in \mathcal{D} : (D_i, T), (D_j, T) \in R \wedge i \neq j \implies terms(D_i) \cap terms(D_j) = \emptyset$
- $\forall T \in \mathcal{T} : i \in ids(T) \implies \exists (D, T) \in R \wedge i \in ids(D)$

For any admissible HDT collection HDT we define the T*-restricted collection* HDT^T as the collection obtained from removing: (i) all triple components $T' \neq T$ from HDT; (ii) the corresponding D' such that (D', T') is in R and (D', T) is not in R; and (iii) the relations (D', T') from R. Thus allowing an HDT collection to be filtered by erasing all dictionary and triple components that are not required for T.

4.2. HDT_{crypt} encoding

We now introduce the final encoding of the HDT_{crypt} extension. HDT_{crypt} uses AES (Advanced Encryption

[2] In the set-theoretic sense.



ex:Alice foaf:pastProject ex:Project1 .
ex:project1 foaf:homepage "www.example.org" .

ex:Alice foaf:mbox "alice@example.org" .
ex:Bob foaf:knows ex:Alice .
ex:Bob foaf:mbox "bob@example.org" .

ex:Alice foaf:mbox "personal@example.org" .
ex:Alice ex:salary "30K" .
ex:Alice foaf:mbox "alice@example.org" .

ex:Alice foaf:pastProject ex:Project1 .
ex:project1 foaf:homepage "www.example.org" .

ex:Bob foaf:knows ex:Alice .
ex:Bob foaf:mbox "bob@example.org" .

ex:Alice ex:salary "30K" .
ex:Alice foaf:mbox "personal@example.org" .

ex:Alice foaf:mbox "alice@example.org" .

Fig. 3. An access-restricted RDF dataset such that G comprises three separate access-restricted subgraphs G_1, G_2, G_3; the graph's canonical partition is comprised of four non-empty subgraphs $G'_1, G'_2, G'_3, G'_{23}$, whereas the *terms* in these graphs can be partitioned into five non-empty subsets corresponding to the dictionaries $D'_1, D'_2, D'_3, D'_{23}, D'_{123}$.

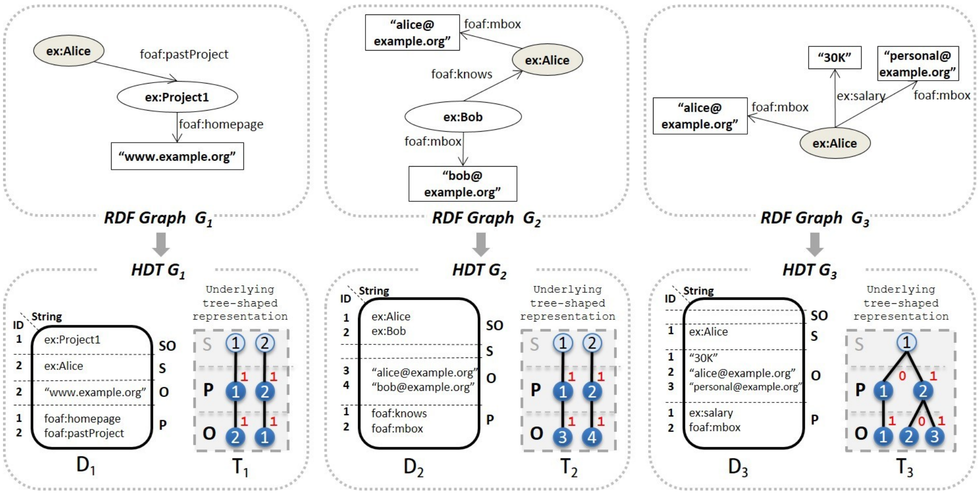


Fig. 4. HDT$_{crypt-A}$, create and encrypt one HDT per partition.

Standard) [11] to encrypt the D and triple components of an HDT collection and extends the header H with a keymap $kmap : \mathcal{D}_{crypt} \cup \mathcal{T}_{crypt} \mapsto I$ that maps *encrypted components* to identifiers (IRIs), which denote AES keys that can be used to decrypt these components.

Thus, $HDT_{crypt} = (H, \mathcal{D}_{crypt}, \mathcal{T}_{crypt})$ where $H = \{R, kmap, M\}$, $R \subseteq \mathcal{D}_{crypt} \times \mathcal{T}_{crypt}$, and the components in $\mathcal{D}_{crypt}$ and $\mathcal{T}_{crypt}$ are encrypted with keys identified in *kmap*.

The operations to *encrypt* and *decrypt* the dictionary and triples are described as follows. First, the operation *encrypt* takes one or more dictionary and triples and encrypts the components with a given key. Formally, we write $encrypt(c, key_{crypt}) = c_{crypt}$, where $c \in \mathcal{D} \cup \mathcal{T}$ to denote the component $c_{crypt} \in \mathcal{D}_{crypt} \cup \mathcal{T}_{crypt}$ obtained by encrypting c with the key key_{crypt}. While, we add an identifier of the components to the header metadata. In other words, $id(c_{crypt}) \mapsto IRI(key_{crypt})$ is added to the kmap, where id denotes the ID of the component in $\mathcal{D}_{crypt}$ and $\mathcal{T}_{crypt}$ and IRI a unique identifier for the symmetric key.

For the decryption, it is assumed that an authorized user u has partial knowledge about these keys, i.e. they have access to a partial function $key_u : I_u \mapsto K$ that maps a finite set of "user-owned" key IDs $I_u \subseteq I$ to the set of AES (symmetric) keys K. The decryption simply takes the given compressed component(s) and performs the decryption with the given symmetric key. Formally, we write $decrypt(c_{crypt}, key_{crypt}) = c$, where $c_{crypt} \in \mathcal{D}_{crypt} \cup \mathcal{T}_{crypt}$ to denote the component $c \in \mathcal{D} \cup \mathcal{T}$ obtained from decrypting c_{crypt} with the key $key_{crypt} = key(kmap(c_{crypt}))$. Further we write $decrypt(HDT_{crypt}, I_u)$ to denote the non-encrypted HDT collection consisting of all decrypted dictionary and triple components of HDT_{crypt} which can be decrypted with the keys in $\{key_u(i) \mid i \in I_u\}$. In other words, the T-restriction of HDT_{crypt} is defined analogously to the above-said.

4.3. Integration operations

Finally, we define two different ways of integrating dictionaries $D_1, \ldots, D_k \in \mathcal{D}$ within an HDT collection: *D-union* and *D-merge*. In the former, we replace dictionaries with a new dictionary that includes the union of all terms. In the latter, we establish one of the dictionaries as the dictionary baseline and rename the IDs of the other dictionaries.

4.3.1. D-union

The *D-union* is only defined for $D_1, \ldots, D_k \subseteq \mathcal{D}$ if the following condition holds on R: $\forall (D_i, T) \in R : (\neg \exists D_j \notin D_1, \ldots, D_k$ such that $(D_j, T) \in R)$. In other words, we can perform a D-union if all T-components depending on dictionaries in the set $D_1, \ldots, D_k$ only depend on these dictionaries. Then, we can define a trivial D-union of HDT wrt. $D_1, \ldots, D_k$, written $HDT_{D_1 \cup \ldots \cup D_k}$, as follows:

- replace $\{D_1, \ldots, D_k\}$ dictionaries with a single dictionary $D_{1 \ldots k} = D_1 \cup \ldots \cup D_k$, such that $\forall x \in terms(D_1) \cup \ldots \cup terms(D_k)$
  - $x \in terms(D_{1 \ldots k})$
  - $id(x, D_{1 \ldots k})$ is obtained by sequentially numbering the terms in $terms(D_1) \cup \ldots \cup terms(D_k)$ upon an (arbitrary) total order, e.g., lexicographically ordering the terms (as it is done in HDT dictionaries by default).
- replace all $(D_i, T) \in R$, $i \in \{1, \ldots, k\}$, with new $(D_{1 \ldots k}, T')$ relations, where T' is obtained from T by replacing the original IDs from D_i with their corresponding new IDs in $D_{1 \ldots k}$.

4.3.2. D-merge

In the more general case where the condition for D-unions does not hold on $D_1, \ldots, D_k \subseteq \mathcal{D}$, we can define another operation, *D-merge*, written $HDT_{D_1 \triangleright \ldots \triangleright D_k}$. We start with the binary case, where only two dictionaries D_1 and D_2 are involved; $HDT_{D_1 \triangleright D_2}$ is obtain as follows:

- replace D_1 and D_2 with a single $D_{12} = D_1 \triangleright D_2$,[3] such that
  - $\forall x \in terms(D_1) : id(x, D_{12}) = id(x, D_1)$
  - $\forall x \in terms(D_2) \setminus terms(D_1) : id(x, D_{12}) = id(x, D_2) + max(ids(D_1))$
- replace all $(D_1, T_1) \in R$ with (D_{12}, T_1)
- replace all $(D_2, T_2) \in R$ with (D_{12}, T_2'), where T_2' is obtained from T_2 by analogous ID changes.

D-merge can then be trivially generalized to a sequence of dictionaries assuming left-associativity of $\triangleright$ operator. That is, $HDT_{D_1 \triangleright D_2 \triangleright \ldots \triangleright D_k} = HDT_{((D_1 \triangleright D_2) \triangleright \ldots) \triangleright D_k}$.

For convenience, we extend the notation of $T(G, D)$ from Section 3.2 to D-unions and D-merges: let $(D_1, \ldots, D_k)$ be a sequence of dictionaries and G an RDF graph such that $terms(G) = \bigcup_{D_i \in (D_1, \ldots, D_k)} terms(D_i)$. Then we will write $T(G, (D_1 \cup \ldots \cup D_k))$ and $T(G, (D_1 \triangleright \ldots \triangleright D_k))$ for the triples part generated from G according to the combined dictionary $((D_1 \cup D_2) \cup \ldots) \cup D_k$ and $((D_1 \triangleright D_2) \triangleright \ldots) \triangleright D_k$ respectively. Finally, we note that for any admissible HDT collection, both D-union and D-merge preserve admissibility.

5. Efficient Partitioning HDT$_{crypt}$

After having introduced the general idea of HDT$_{crypt}$ and the two different ways of integrating dictionaries within an HDT collection, we now discuss four alternatives strategies that can be used for distributing a dataset DS across dictionary and triple components in an HDT$_{crypt}$ collection. These alternatives, referred to as HDT$_{crypt-A}$, HDT$_{crypt-B}$, HDT$_{crypt-C}$ and HDT$_{crypt-D}$, provide different space/performance tradeoffs that will be evaluated in Section 6. We note that HDT behaves differently than the normal RDF merge regarding blank nodes in different "partitions" as, by default, HDT does not rename the blank nodes to avoid shared labels [28]: the original blank nodes are skolemized to constants (unique per RDF graph) and preserved across partitions, so that we do not need to consider blank node (re-)naming separately.

5.1. HDT$_{crypt-A}$: A Dictionary and Triples per Named Graph in DS

The baseline approach is straightforward, we construct separate HDT components $D_i = D(G_i)$ and

[3] We use the directed operator $\triangleright$ instead of $\cup$ here, since this operation is not commutative.

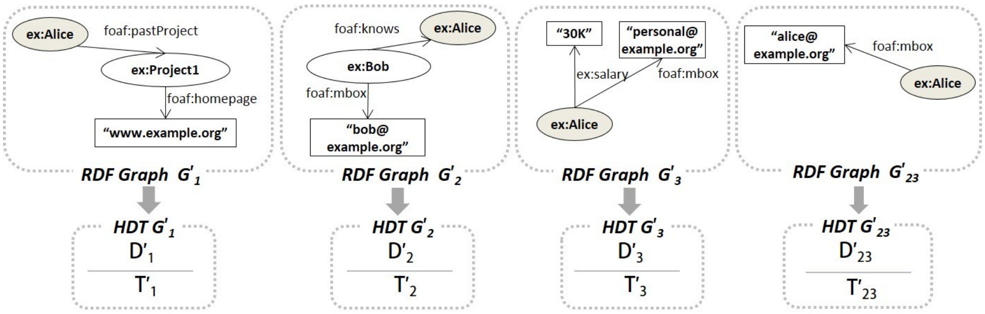


Fig. 5. HDT$_{crypt-B}$, extracting non-overlapping triples.

$T_i = T(G_i, D_i)$ per graph G_i in the dataset, see Figure 4, thereafter each of these components is encrypted with a respective, separate key, identified by a unique IRI $id_i \in I$, i.e., $kmap(D_i) = kmap(T_i) = id_i$ and $R = \{(D_i, T_i) \mid G_i \in DS\}$. For re-obtaining graph G_i a user must only have access to the key corresponding to id_i, and can thereby decrypt D_i and T_i and extract the restricted collection HDT^{T_i}, which corresponds to G_i. Obviously, this approach encodes a lot of overlaps in both dictionary and triples parts: that is, for our running example from Figure 4, the IRI for ex:alice is encoded in each individual D component and the overlapping triples in graphs G_2 and G_3 appear in both T_2 and T_3 respectively (cf., Figure 4).

5.2. HDT$_{crypt-B}$: Extracting non-overlapping Triples in DS'

In order to avoid the overlaps in the triple components, a more efficient approach could be to split the graphs in the dataset DS according to their canonical partition DS' and again construct separate (D,T)-pairs for each subset $G'_S \in DS'$, see Figure 5. That is, we create $D'_S = D(G'_S)$ and $T'_S = T(G'_S, D'_S)$ per graph $G'_S \in DS'$, where $S \in 2^{1,\dots,i}$ denotes the index set corresponding to a (non-empty) subset of DS'. R in turn contains pairs (D'_S, T'_S) and *kmap* entries for keys identified by I'_S per G'_S used for the encryption/decryption of the relevant D'_S and T'_S. The difference for decryption now is that any user who is allowed access to G_i must have all keys corresponding to any I'_S such that $i \in S$ in order to re-obtain the original graph G_i.

First, the user will decrypt all the components for which they have keys, i.e. obtaining a non-encrypted collection HDT' consisting of components $\mathcal{D}' = \{D'_1, \dots, D'_k\}, \mathcal{T}' = \{T'_1, \dots, T'_k\}$ consisting of the components corresponding to a partition of G_i. Then, for decompressing the original graph G_i, we create separate T'_S-restricted HDTs, which are decompressed separately, with G_S being the union of the resulting subgraphs.

5.3. HDT$_{crypt-C}$: Extracting non-overlapping Dictionaries in DS'

Note that in the previous approach, we have duplicates in the dictionary components. An alternative strategy would be to create a *canonical partition* of *terms* instead of triples, and create separate dictionaries $D'_S \in \mathcal{D}'$ for each non-empty term-subset,[4] respectively. Figure 6 shows the canonical partition of terms in our running example: as can be seen, the original dictionary is split into five non-empty terms-subsets corresponding to the dictionaries D'_{123} (terms shared in all three graphs), D'_{23} (terms shared in graphs G_2 and G_3 that are not in D'_{123}) and D'_1, D'_2, D'_3 (terms in either G_1, G_2 or G_3 resp. and are not shared between graphs). This partition can be computed efficiently, thanks to the HDT dictionary D of the full graph G, which we assume to be available[5]. To do so, we keep[6] an auxiliary bitsequence per graph G_i (see Figure 6, top left), each of size $terms(D)$. Then, we iterate through triples in each graph G_i and, for each term, we search its ID in D, marking such position with a 1-bit in the bitsequence of G_i. Finally, the dictionaries of the subsets can be created by inspecting the combinations of 1-bits in the bitsequences: terms in $D'_{xy\cdots z}$ will be those with a 1-bit in the bitsequences of graphs $xy \cdots z$ and 0-bits in other graphs. For instance, in Figure 6, D'_{123} is constituted only by ex:alice, because it is the only term with

[4] Again, here $S \in 2^{1,\dots,n}$ represents an index set.

[5] All HDT$_{crypt}$ strategies are evaluated from an existing full graph G. Our evaluation in Section 6 also reports the time to create the HDT representation of the full graph G

[6] This auxiliary structure is maintained just at compression time and it is not shipped with the encrypted information.

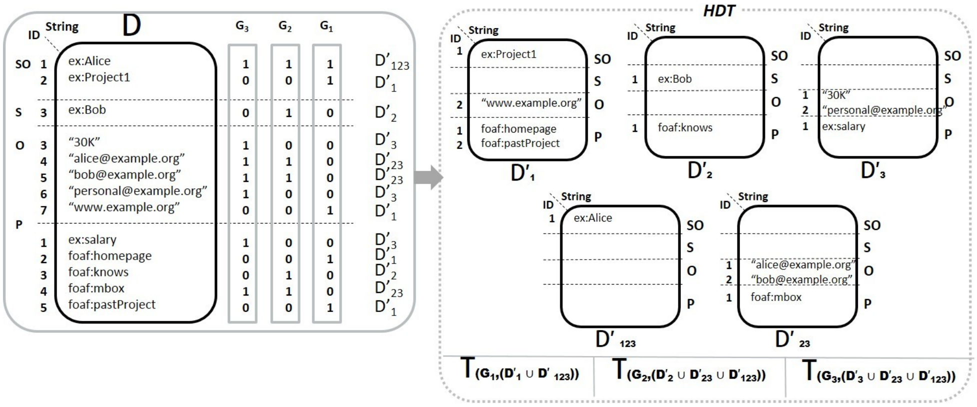


Fig. 6. HDT$_{crypt-C}$, extracting non-overlapping dictionaries.

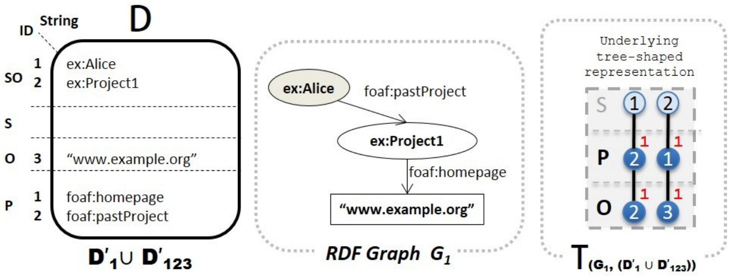


Fig. 7. Union of dictionaries (in HDT$_{crypt-C}$) to codify the non-overlapping dictionaries of a partition.

three 1-bits in the bitsequences of G_1, G_2 and G_3. In contrast, ex:Project1 will be part of D'_1 as it has a 1-bit only in the bitsequence of G_1.

The number of triple components in this approach are as in HDT$_{crypt-A}$, one per graph G_i. However, they are constructed slightly differently as, in this case, we have a canonical partition of terms and one user will just receive the dictionaries corresponding to subsets that correspond to terms in the graph G_i that they have been granted access to. In other words, the IDs used in each T_i should unambiguously correspond to terms, but these terms may be distributed across several dictionaries.[7] Thus, we encode triples with a D-union (see Section 4.3) of the D'_S such that $i \in S$. That is, for each G_i we construct $T_i = T(G_i, (\bigcup_{i\in S} D'_S))$, and add the respective pairs (D'_S, T_i) in R.

Figure 7 illustrates this merge of dictionaries for the graph G_1 and the respective construction of $T(G_1, (D'_1 \cup D'_{123}))$. The decompression process after decryption is the exact opposite. For decompressing the graph G_i, the decrypted dictionaries $\bigcup_{i\in S} D'_S$ are used to create a D-union D_i which can be used to decompress the triples T_i in one go. Finally, as a performance improvement at compression time, note that, although the canonical partition of terms has to be built to be shipped in the compressed output, we can actually skip the creation of the D-union dictionaries to encode the IDs in the triples. To do so, we make use of the bitsequences to get the final IDs that are used in the triples: One should note that the ID of a term in a D-union of a graph G_i is the number of previous 1-bits in the bitsequence of G_i (for each SO, S, O, and P section). For instance, in our example in Figure 7, ex:Project1 is encoded with the ID=2. Instead of creating D_1, we can see that in the bitsequence of G_1 (see Figure 6, top right) we have two 1-bits in the predicate section up to the position where ex:Project1 is stored in the original dictionary, hence its ID=2.

[7] Given the partition definition, it is nonetheless true that a term appears in one and only one term-subset.

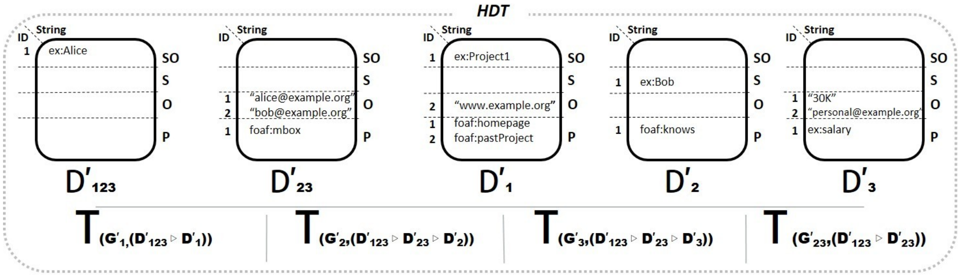


Fig. 8. HDT$_{crypt-D}$, extracting non-overlapping dictionaries and triples.

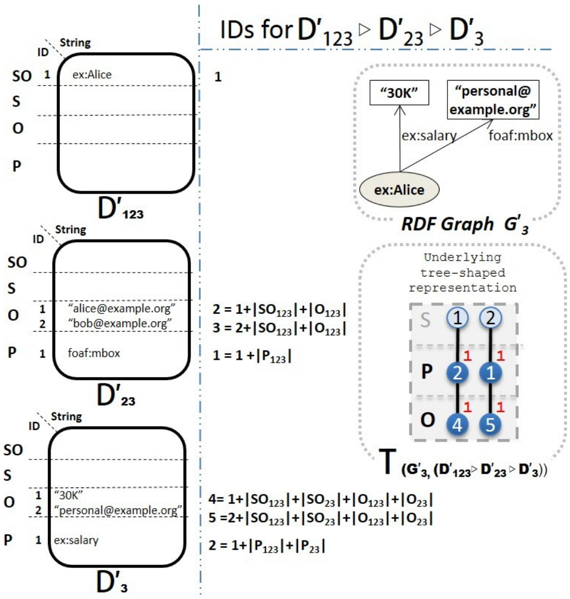


Fig. 9. Merge of dictionaries (in HDT$_{crypt-D}$) to codify the non-overlapping dictionaries and triples of a partition.

5.4. HDT$_{crypt-D}$: Extracting non-overlapping Dictionaries and Triples in DS'

In HDT$_{crypt-D}$, we combine the methods of both HDT$_{crypt-B}$ and HDT$_{crypt-C}$. That is, we first create a *canonical partition of terms* as in HDT$_{crypt-C}$, and a *canonical partition of triples* DS' as in HDT$_{crypt-B}$. Then, we codify the IDs in the subsets of DS' with the IDs from the dictionaries. Note, however, that in this case there is – potentially – an n:m between the resulting dictionary and triple components. In other words, triples in T'_S can include terms that are not only in D'_S as they may be distributed across several term-subsets. For instance, in our running example, T'_1 in HDT$_{crypt-B}$ includes ex:Alice (see Figure 5) which is stored in D'_{123} in HDT$_{crypt-C}$ (see Figure 6). One alternative could be to create a D-union of each graph G'_S and codify triples in T'_S with the corresponding IDs. However, it is trivial to see that this would lead to an exponential number of D-union dictionaries, one per T'_S component. In addition, we would need to physically recreate all these dictionaries at compression time, and also at decompression time in order to decompress each single graph G'_S. Thus, we perform a D-merge approach (see the definition in Section 4.3), which fits perfectly with n:m-relations. This is illustrated in Figure 8. As can be seen, triples in each G'_S of the canonical partition are encoded with an appropriate D-merge of term-subsets. A practical example is shown in Figure 9, representing the encoding of graph G'_3. As defined in D-merge, IDs are assigned in order, that is for a merge $D'_1 \triangleright \ldots \triangleright D'_k$, the IDs in D'_k are shifted $max(ids(D'_1)) + \ldots + max(ids(D'_{k-1}))$. For instance, in our example, the predicate ex:salary will be encoded in G'_3 with the ID=2, because its local ID in D'_3 is 1, and it has to be shifted $max(ids(D'_{123})) + max(ids(D'_{23})) = 1$, hence its final ID= $1{+}max(ids(D'_{123})){+}max(ids(D'_{23})) = 2$. Note that here we restrict the dictionaries D' per section (SO, S, O and P). Given the special numbering of IDs in HDT, where S and O IDs follow from SO as explained in Section 3.1. This is illustrated in our example, e.g. the object "30K" with local ID=1 in D'_3 is mapped in the D-merge dictionary with 4, as it sums up all the previous objects and subjects IDs in D'_{123} and D'_{23}.

It is worth mentioning that no ambiguity is present in the order of the D-merge as it is implicitly given by the partition DS' as per the canonical term partition. Thus, the decompression follows the opposite process: for each graph T'_S in the partition of the graph G_i, the user processes each ID and, depending of the value, they get the associated term in an appropriate term subset. For instance, if the user is accessing the predicate ID=2 in our example, one can easily see that $2 > |P_{123}| + |P_{23}|$, so dictionary D'_3 has to be used[8]. The local ID to look at is then $2 - |P_{123}| - |P_{23}| = 1$, hence the predicate ID=1 in D'_3 is inspected and then foaf:pastProject is retrieved. Finally, note that not all terms in a D-merge are necessarily used when encoding a particular T'_S. For instance, in our example in Figure 9, the object "bob@example.org" with ID=2 in D'_{23} (and ID=3 in the D-merge) is not used in T'_3. However, this ID is "blocked": it cannot be used by a different object in T'_3 as this ID is taken into account when encoding the present objects ("30K" and "personal@example.org"), once we sum the $max(ids(D'_{23}))$ as explained above. The same consequence applies to subjects, so that subject IDs are not necessarily correlative in T'_S. This constitutes a problem for the HDT Bitmap Triples encoding (presented in Section 3.2), given that it represents subjects implicitly assuming that they are correlative. Thus, HDT$_{crypt-D}$ has to explicitly state the ID of each subject, which constitutes a space overhead and a drawback of this approach, despite the fact that duplicate terms and triples are avoided. Technically, instead of a forest of trees, triples are codified as tuples of three IDs, using an existing HDT triples representation called *Plain Triples* [17].

6. Evaluation

This section evaluates the performance of HDT$_{crypt}$ by comparing each of the aforementioned partitioning strategies with respect to the performance of the algorithms and the size of the compressed encrypted dataset. We first describe our experimental setup in detail. Then, we present our evaluation results in terms of three distinct yet related tasks: (i) performance of compression and encryption algorithms and size of resulting datasets; (ii) performance of decryption and decompression algorithms; and (iii) performance of triple pattern queries[9] over the compressed datasets, which constitute the basis for SPARQL's graph pattern matching [26].

Finally, we provide a summary and discussion of the results in Section 6.5. Additional experiments can be found in Appendix A.

6.1. Experimental Setup

The proof-of-concept HDT$_{crypt}$ prototype[10] uses the existing HDT-C++ library[11] for compression and decompression, and standard Java libraries for AES encryption/decryption[12].

The evaluation is performed on three different datasets, described in Table 1.

First, we selected DBpedia, the well-known RDF knowledge base extracted from Wikipedia, which was chosen due to the volume and variety of the data and large number of dictionary terms therein. We used two different versions, DBpedia 3.8[13] and the latest version 2016-10[14], which is double the size of the previous one. Hereinafter, we will use the term *DBpedia* to refer to both versions, as the results are comparable. Then, we chose a realistic scenario using the configuration used in SAFE [33], a query federation engine with access control. The SAFE dataset includes public statistical data (referred to as external) and anonymised clinical data (internal).

Additionally, in order to test the scalability of the various partitioning strategies we use the Lehigh University Benchmark (LUBM) [24] data generator to obtain synthetic datasets of incremental sizes from 1,000 universities (LUBM1K, including 0.13 billion triples) to 4,000 universities (LUBM4K, 0.53 billion triples). Table 1 shows the original dataset sizes in plain N-Triples (NT). In addition, we provide details of the size of the datasets compressed with *gzip*, `HDT` and `HDT+gz` (gzip compression over the HDT file). This shows that our HDT compression ratios are in line with the original proposal [15]. Finally, the last column of the table shows the time (in minutes) to compute the HDT representation of each dataset. In turn, the HDT creation time for LUBM grows linearly with the number of triples. This result is also in accordance with the HDT technique, which reports linear scalability regarding the input size and the terms in the dictionary (cf. [15]). The two versions of DBpedia also show a similar behaviour: DBpedia 2016-10 doubles the number of triples of DBpedia 3.8 and its dictionary triples

[8] We abuse notation to denote the cardinality of a set, e.g. $|P_{123}|$, as the maximum id represented in such dictionary set.

[9] Matching RDF triples in which each component may be a variable

[10] Source code and all experiment data are available at the HDT$_{crypt}$ homepage: https://aic.ai.wu.ac.at/ComCrypt/HDTcrypt/

[11] https://github.com/rdfhdt/hdt-cpp

[12] http://docs.oracle.com/javase/8/docs/technotes/guides/security/crypto/CryptoSpec.html

[13] http://wiki.dbpedia.org/data-set-38

[14] http://wiki.dbpedia.org/develop/datasets/dbpedia-version-2016-10

Table 1

Statistical dataset description

						Size (GB)				
DATASET	TRIPLES	|SO|	|S|	|O|	|P|	NT	NT+gz	HDT	HDT+gz	HDT creation time (m)
DBpedia 3.8	0.43BN	22.0M	2.8M	86.9M	58.3K	61.6	4.9	6.4	2.7	96
DBpedia 2016-10	0.84BN	44.5M	55.9M	225.6M	63.8K	122.0	9.6	12.1	5.0	249
SAFE	0.07BN	171.5K	7.4M	3.6M	346	12.4	0.3	0.6	0.07	10
LUBM1K	0.13BN	5.0M	16.7M	11.2M	18	18.0	0.6	0.7	0.2	18
LUBM2K	0.27BN	10.0M	33.5M	22.3M	18	36.2	1.3	1.5	0.5	36
LUBM3K	0.40BN	14.9M	50.2M	33.5M	18	54.4	1.9	2.3	0.8	57
LUBM4K	0.53BN	19.9M	67.0M	44.7M	18	72.7	2.5	3.1	1.0	78

Table 2

% Duplicates and size of subgraphs.

			Size of subgraphs (GB)											
SUBGRAPHS	DATASET	DUP %	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}	G_{11}	G_{12}
6	DBpedia 3.8	11.62%	11.6	11.7	11.5	11.7	11.6	11.5						
	DBpedia 2016-10	11.62%	23.2	23.2	23.0	23.1	23.0	22.5						
9	DBpedia 3.8	22.32%	8.9	8.9	8.9	8.8	8.9	8.8	8.8	8.9	8.7			
	DBpedia 2016-10	22.32%	17.6	17.5	17.5	17.4	17.5	17.5	17.5	17.1	17.4			
12	DBpedia 3.8	32.54%	7.6	7.6	7.7	7.6	7.6	7.6	7.7	7.6	7.6	7.6	7.6	7.5
	DBpedia 2016-10	32.54%	15.2	15.0	15.2	15.1	15.1	15.1	15.1	15.1	15.1	15.1	14.7	15.2
8	SAFE	0.00%	7.0	3.2	1.9	0.1	0.1	0.1	0.01	0.01				
6	LUBM1K	37.89%	14	5.2	5	4.5	1.6	0.6						
	LUBM2K	37.89%	27	10.7	10.7	8.9	3.1	1.3						
	LUBM3K	37.89%	39.5	16.1	15.2	13.4	4.6	1.9						
	LUBM4K	37.89%	52.8	21.4	20.3	17.9	6.1	2.4						
9	LUBM1K	38.26%	14	5.2	4.5	3.1	1.6	1.3	0.9	0.6	0.2			
	LUBM2K	38.26%	27	10.7	8.9	6.2	3.1	2.3	1.9	1.3	0.4			
	LUBM3K	38.26%	39.5	16.1	13.4	9.2	4.6	3.4	2.8	1.9	0.6			
	LUBM4K	38.26%	52.8	21.4	17.9	13.0	6.1	4.6	3.7	2.4	0.8			
12	LUBM1K	38.10%	8.8	5.1	4.5	4.3	1.5	1.3	1.1	1.1	0.9	0.7	0.6	0.2
	LUBM2K	38.10%	17.7	10.1	8.9	8.6	3.1	2.5	2.3	2.1	1.9	1.5	1.3	0.4
	LUBM3K	38.10%	26.6	15.2	13.4	12.9	4.6	3.8	3.4	3.2	2.8	2.2	1.9	0.6
	LUBM4K	38.10%	35.6	20.3	17.9	17.2	6.1	5.1	4.6	4.2	3.7	3.0	2.4	0.8

the number of terms. As a result, the HDT creating time increases 2.6 times.

For the LUBM dataset we group data based on the rdf:type of resources and use these groupings to generate three different subgraph datasets (the size of each subgraph is shown in Table 2):

- 12 subgraphs, composed of *UnderGraduateStudent* (G_1), *Courses* (G_2), *Publication* (G_3), *GraduateStudent* (G_4), *Department* (G_5), *ResearchAssistant* (G_6), *AssociateProfessor* (G_7), *TeachingAssistant* (G_8), *FullProfessor* (G_9), *AssistantProfessor* (G_{10}), *University* (G_{11}) and *Lecturer* (G_{12}).
- 9 subgraphs, composed of the union of *UnderGraduateStudent* and *GraduateStudent* (G_1), *Courses* (G_2), *Publication* (G_3), the union of *AssistantProfessor, ResearchAssistant*, and *TeachingAssistant* (G_4), *Department* (G_5), *AssociateProfessor* (G_6), *FullProfessor* (G_7), *University* (G_8) and *Lecturer* (G_9).
- 6 subgraphs, composed of *UnderGraduateStudent* and *GraduateStudent* (G_1), the union of *AssistantProfessor, ResearchAssistant*, *TeachingAssistant*, *Lecturer*, *AssociateProfessor*, *FullProfessor* (G_2), *Courses* (G_3), *Publication* (G_4), *Department* (G_5) and *University* (G_6).

When triples represent relations between resources of different types all incoming/outgoing relations are replicated in both subgraphs.

For DBpedia (in the case of both versions), we generate 6, 9 and 12 subgraphs, each containing randomly selected triples amounting to 10% of the entire corpus (thus ensuring overlaps among subgraphs). Triples that do not appear in any subgraph are subsequently distributed evenly among the subgraphs.

In the case of SAFE, the dataset is already organised in 8 subgraphs, composed of 5 *external* graphs, including statistical data from well-known organisation such as Eurostat and FAO, and 3 *internal* graphs including aggregated clinical data represented as RDF data cubes [33].

Given that the complexity of the partitioning is directly related to the number of duplicates across subgraphs, the size of each of the subgraphs and the overall duplicate ratio, as $\frac{(totalTriples - UniqTriples)}{totalTriples}$, is presented in column DUP % of Table 2. Note that the type-based selection of subgraphs in LUBM generates a skewed distribution of subgraph sizes but similar duplicate ratio (of approximately 38%) at increasing sizes (LUBM1K to LUBM4K). Thus, the comparison between techniques focuses on the effect of the 6/9/12 subgraphs and the efficiency at large scale. In contrast, the even distribution of DBpedia is reflected in the similar size of its subgraphs. Given that the number of duplicates increase with the number of subgraphs (12%, 22% and 33% for 6/9/12 respectively), the effect of duplicates is also evaluated. In SAFE, the already given 8 subgraphs contains few repeated triples (less than 0.01%). Note that the *internal* subgraphs corresponds to graphs G_4, G_5 and G_7 in Table 2, i.e. the public *external* information corresponds to the biggest partitions.

In the following we show the performance results of each of the algorithms (compression and encryption, decryption and decompression, integration and querying). Experiments were performed in a –commodity server– (Intel Xeon E5-2650v2 @ 2.6 GHz, 16 cores, RAM 180 GB, Debian 7.9.). All of the reported (elapsed) times are the average of three independent executions in a cold cache scenario (caches are empty at the start of each process).

6.2. Compression and Encryption

Table 3 shows the compression and encryption times as well as corresponding resulting file sizes[15] of the datasets for different partitioning strategies, whereas Table 4 shows the respective number of resulting dictionary and triple components.

The results show that $\text{HDT}_{crypt-C}$ is both the fastest and also produces the most compact representation (only marginally outperformed in space by $\text{HDT}_{crypt-D}$ in particular LUBM cases). $\text{HDT}_{crypt-C}$ is 37% faster than the baseline approach $\text{HDT}_{crypt-A}$ in DBpedia (we refer to the average in both DBpedia versions hereafter), and 40% faster in LUBM. In SAFE, with few duplicates, $\text{HDT}_{crypt-C}$ is still 18% faster.

In contrast, $\text{HDT}_{crypt-B}$ is the slowest approach with a mean of 68% over the baseline, because it needs to create many dictionaries (e.g. 3904 in DBpedia 2016-10 as shown in Table 4) with overlapping terms. In turn, $\text{HDT}_{crypt-D}$ is highly influenced by the number of dictionary components, due to the additional complexity of creating the resp. triple components from the D-merge. Thus, $\text{HDT}_{crypt-D}$ is faster than the baseline in LUBM with 6 or 9 subgraphs, with few components as shown in Table 4, but it shows a worse performance in LUBM 12 subgraphs, as well as in all DBpedia and SAFE datasets.

Note that, as stated in Section 5, the creation of $\text{HDT}_{crypt-B}$, $\text{HDT}_{crypt-C}$ and $\text{HDT}_{crypt-D}$ assumes that the HDT representation of the full graph G is already computed[16]. Otherwise, the HDT creation time (reported in Table 1) should be considered as a once-off overhead. In the worst case (i.e. the conversion is done for the sole purpose of encrypting a single dataset with a particular number of subgraphs), adding this time would make the $\text{HDT}_{crypt-C}$ perform similarly to the baseline in LUBM. In DBpedia, with a richer dictionary of terms, $\text{HDT}_{crypt-C}$ would be 35-50% slower than the baseline.

Additionally, when compared with the baseline approach $\text{HDT}_{crypt-A}$, $\text{HDT}_{crypt-C}$ achieves a mean of 33% space saving in DBpedia and 26% space saving in LUBM. In general, $\text{HDT}_{crypt-B}$, $\text{HDT}_{crypt-C}$ and $\text{HDT}_{crypt-D}$ benefit from having an increasing number of overlapping dictionaries/triples, hence the DBpedia even distribution produces more space savings. For the same reason, an increasing number of subgraphs leads to more duplicates and space savings w.r.t the baseline, e.g. $\text{HDT}_{crypt-C}$ in LUBM achieves 24%, 26% and 27% savings with 6, 9 and 12 subgraphs respectively. It is worth mentioning that despite the fact that $\text{HDT}_{crypt-D}$ isolates the non-overlapping dictionaries and triples, there is an overhead in the representation as we do not use Bitmap Triples but Plain Triples (as stated in Section 5.4). This is more noticeable in DBpedia with long predicate and object lists. It is worth highlighting that, in SAFE, with almost no duplicates,

[15] Note that encryption produces negligible size overheads on the compressed files.

[16] In fact, HDT is becoming popular to store and serve large datasets by publishers and third parties, and a large portion of datasets in the Linked Open Data cloud is already available in HDT thanks to the project LOD Laundromat [3], crawling and serving the HDT conversion of datasets (http://lodlaundromat.org/wardrobe/).

Table 3

Performance of compression and encryption algorithms.

		Compression Time (minutes)				Encryption Time (seconds)				Size (GB)			
SUBGRAPHS	DATASET	*crypt-A*	*crypt-B*	*crypt-C*	*crypt-D*	*crypt-A*	*crypt-B*	*crypt-C*	*crypt-D*	*crypt-A*	*crypt-B*	*crypt-C*	*crypt-D*
6	`DBpedia 3.8`	102	197	**62**	121	98.17	108.84	**80.85**	91.07	9.64	9.33	**7.43**	8.59
	`DBpedia 2016-10`	306	565	**197**	378	144.54	143.06	**109.95**	123.91	18.91	18.38	**14.04**	16.25
9	`DBpedia 3.8`	117	225	**72**	142	124.65	140.40	**98.24**	125.94	11.64	10.91	**7.92**	8.76
	`DBpedia 2016-10`	267	520	**175**	315	182.61	172.50	**113.18**	128.33	23.12	21.78	**15.00**	16.58
12	`DBpedia 3.8`	131	214	**81**	152	**156.52**	245.16	187.90	221.89	13.87	12.49	**8.49**	8.91
	`DBpedia 2016-10`	300	485	**202**	326	228.92	201.26	**128.29**	138.70	27.79	25.11	**16.14**	16.88
8	`SAFE`	9	18	**8**	14	**4.07**	4.91	4.27	5.20	**0.53**	0.61	**0.53**	0.65
6	`LUBM1K`	34	41	**21**	33	12.25	11.21	**9.85**	10.94	1.40	1.08	**1.05**	**1.05**
	`LUBM2K`	78	94	**47**	73	21.88	18.74	**17.95**	18.24	2.86	2.19	**2.15**	2.16
	`LUBM3K`	125	143	**72**	112	32.24	26.82	**25.72**	26.00	4.35	3.31	**3.28**	3.30
	`LUBM4K`	169	191	**97**	151	54.56	33.83	**33.17**	33.90	5.65	4.45	**4.41**	4.45
9	`LUBM1K`	37	42	**21**	36	12.96	11.93	**11.33**	11.89	1.44	1.09	1.06	**1.04**
	`LUBM2K`	78	88	**45**	73	22.80	19.56	**18.97**	19.98	2.93	2.21	2.17	**2.14**
	`LUBM3K`	126	144	**71**	114	33.79	28.02	27.61	**27.58**	4.45	3.34	3.31	**3.26**
	`LUBM4K`	174	194	**98**	158	60.11	35.66	35.53	**35.36**	5.97	4.49	4.44	**4.42**
12	`LUBM1K`	36	44	**23**	39	12.78	13.48	**12.21**	12.98	1.45	1.11	1.06	**1.05**
	`LUBM2K`	75	94	**49**	82	23.32	21.62	**20.23**	21.38	2.96	2.25	2.17	**2.15**
	`LUBM3K`	116	142	**73**	126	33.92	29.03	**28.35**	29.50	4.50	3.41	3.31	**3.26**
	`LUBM4K`	158	190	**99**	175	60.80	37.85	38.20	**37.36**	6.03	4.56	**4.44**	**4.44**

Table 4

Number of dictionaries/triples in each approach.

		Dictionaries			Triples	
				crypt-C	*crypt-A*	*crypt-B*
SUBGRAPHS	DATASET	*crypt-A*	*crypt-B*	*crypt-D*	*crypt-C*	*crypt-D*
6	`DBpedia 3.8`	6	63	63	6	63
9	`DBpedia 3.8`	9	510	511	9	510
12	`DBpedia 3.8`	12	3836	4095	12	3836
6	`DBpedia 2016-10`	6	63	63	6	63
9	`DBpedia 2016-10`	9	511	511	9	511
12	`DBpedia 2016-10`	12	3904	4095	12	3904
8	`SAFE`	8	32	48	8	32
6	`LUBM`	6	20	23	6	20
9	`LUBM`	9	39	64	9	39
12	`LUBM`	12	55	122	12	55

only $\text{HDT}_{crypt-C}$ is competitive in space with the baseline, while $\text{HDT}_{crypt-B}$ and $\text{HDT}_{crypt-D}$ have to pay a slight overhead for keeping the different structures, which cannot leverage the minimal duplication across subgraphs.

Encryption times are only a small portion of the publication process, where $\text{HDT}_{crypt-C}$ is generally the fastest approach except for DBpedia 3.8 with 12 subgraphs and SAFE, for which $\text{HDT}_{crypt-A}$ is the fastest, and for LUBM3K/LUBM4K with 9 subgraphs as well as LUBM4K with 12 subgraphs where $\text{HDT}_{crypt-D}$ is marginally faster. Thus we can conclude that both the number of files that need to be encrypted as well as their respective file sizes influence the overall encryption time. Finally, it is worth noting that – as expected – the performance time of the compression and encryption, as well as the result file sizes show linear growth with increasing LUBM datasets.

6.3. Decryption and Decompression

According to our use case scenario we assume that a user has been granted access to more than one named graph, but not the whole dataset. For a fair comparison, given the skewed size distribution of subgraphs in LUBM (see Table 2), we set up a scenario where the user has been granted access to half of the total subgraphs, including the smallest, average and largest subgraphs. This configuration corresponds to decrypting and decompressing the subgraphs referred to as $M^6 = \{G_1, G_3, G_6\}$, $M^9 = \{G_1, G_2, G_5, G_8, G_9\}$ and $M^{12} = \{G_1, G_2, G_6, G_7, G_{11}, G_{12}\}$ in the case of 6, 9 and 12

Table 5

Performance of decryption and decompression algorithms for M^6, M^9 and M^{12}, i.e., half of the 6/9/12 subgraphs including the smallest/average/largest subgraphs.

		Decryption Time (seconds)				Decompression Time (minutes)			
SUBGRAPHS	DATASET	*crypt-A*	*crypt-B*	*crypt-C*	*crypt-D*	*crypt-A*	*crypt-B*	*crypt-C*	*crypt-D*
M^6	`DBpedia 3.8`	**61.56**	79.08	64.92	79.80	22	18	**14**	18
	`DBpedia 2016-10`	**108.64**	125.36	108.69	127.51	51	46	**39**	53
M^9	`DBpedia 3.8`	**88.64**	148.52	111.84	129.31	26	22	**17**	25
	`DBpedia 2016-10`	**146.93**	200.97	151.41	171.56	49	45	**36**	50
M^{12}	`DBpedia 3.8`	**93.10**	220.46	195.11	242.85	22	22	**17**	26
	`DBpedia 2016-10`	**160.88**	256.05	179.65	206.75	37	34	**27**	37
M^6	`LUBM1K`	10.82	11.37	**9.80**	13.74	8	7	**5**	7
	`LUBM2K`	19.24	22.83	**17.15**	27.62	16	14	**11**	15
	`LUBM3K`	28.35	31.65	**24.78**	45.14	24	20	**16**	22
	`LUBM4K`	48.56	43.03	**33.70**	59.46	32	27	**21**	29
M^9	`LUBM1K`	12.84	13.36	**11.86**	17.52	8	10	**6**	8
	`LUBM2K`	22.77	24.47	**20.63**	33.15	17	21	**12**	16
	`LUBM3K`	32.94	37.32	**30.30**	48.95	26	32	**18**	23
	`LUBM4K`	**48.00**	52.35	51.36	70.12	34	41	**24**	32
M^{12}	`LUBM1K`	**10.75**	11.54	11.73	15.84	7	6	**5**	7
	`LUBM2K`	**18.50**	20.30	18.99	30.40	14	13	**10**	14
	`LUBM3K`	**26.60**	31.08	27.00	45.35	21	19	**15**	20
	`LUBM4K`	**36.62**	39.48	39.09	66.57	29	25	**19**	27

Table 6

Performance of decryption and decompression algorithms for M_L^8 and M_S^8 in the SAFE dataset.

		Decryption Time (seconds)				Decompression Time (seconds)			
SUBGRAPHS	DATASET	*crypt-A*	*crypt-B*	*crypt-C*	*crypt-D*	*crypt-A*	*crypt-B*	*crypt-C*	*crypt-D*
M_L^8	`SAFE`	**3.98**	4.45	4.01	4.70	182	169	**118**	174
M_S^8	`SAFE`	**1.01**	2.75	1.05	2.14	6	74	**4**	56

subgraphs respectively. As for the SAFE dataset, we consider a scenario where a subset of the *external* and *internal* datasets are accessed. In particular, we also took half of the datasets, $M_L^8 = \{G_1, G_4, G_5\}$, including the largest *external* dataset G_1, and $M_S^8 = \{G_4, G_5, G_6\}$, of smaller size.

Table 5 shows the time to decrypt and decompress each of the respective subgraphs in the case of DBpedia and LUBM, while Table 6 shows the results for SAFE.

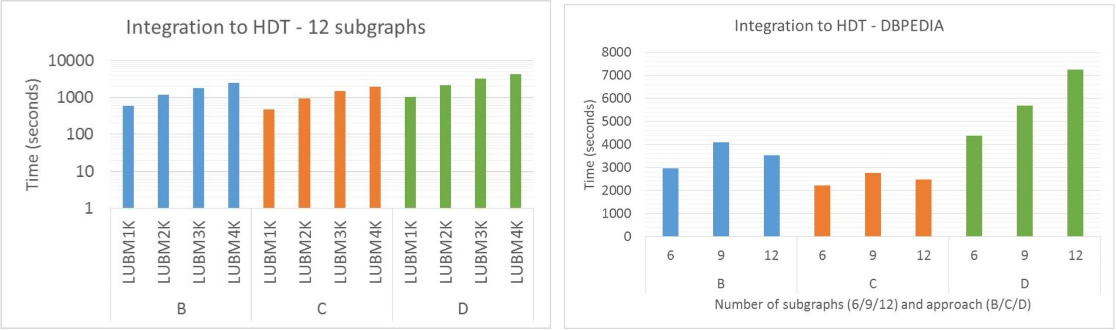
Decryption times are almost negligible compared to the decompression time – similar to encryption vs. compression time. Again, the number of files is the dominating factor, hence HDT$_{crypt-A}$ is the fastest approach regarding decryption.

Regarding decompression, (as per compression) HDT$_{crypt-C}$ is the fastest approach, achieving a mean of 30% time savings in DBpedia and LUBM w.r.t the baseline HDT$_{crypt-A}$. In DBpedia, given the even distribution, having 6 subgraphs is always slightly faster than 9 and 12 subgraphs as the latter generates more duplicates. Regarding the number of graphs in LUBM, 6 and 12 subgraphs behave similarly, while the decompression of 9 subgraphs is slightly slower. Nonetheless, we could verify that the difference between 9 and 12 subgraphs is due to the slightly bigger total file size produced by M^9 in comparison to M^{12}. In turn, the difference between 9 and 6 subgraphs is a consequence of the larger number of generated dictionary/triples between 9 and 6 subgraphs (as shown in Table 4). As per compression, there is a linear increase in performance times with increasing dataset sizes.

Finally, although the results for the SAFE dataset (shown in Table 6) follow a similar behaviour, it is worth mentioning that HDT$_{crypt-B}$ and HDT$_{crypt-D}$ have to pay the price of loading additional structures (even in the presence of minimal duplication). Results show that, while this pays off in the case of the larger subset like M_L^8, for a small subset like M_S^8, HDT$_{crypt-A}$ and HDT$_{crypt-C}$ are clearly faster than HDT$_{crypt-B}$ and HDT$_{crypt-D}$.

6.4. *Querying Compressed Data*

One of the main advantages of HDT compression is that it is possible to perform SPARQL triple pattern queries directly on the compressed data [37]. Whereas this also holds for approach HDT$_{crypt-A}$, as it already



(a) 12 subgraphs (M^{12}) for DBpedia

(b) Comparison of 6/9/12 subgraphs (M^6, M^9 and M^{12}) for DBpedia

Fig. 10. Integration of the dictionary and triple components of M^6, M^9 and M^{12} into one HDT per subgraph in DBpedia (average of the performance in both DBpedia versions).

consists of one file per subgraph, the other approaches presented, HDT$_{crypt-B}$, HDT$_{crypt-C}$ and HDT$_{crypt-D}$, split a subgraph in different dictionary (D) and triple (T) components. For these latter approaches, query resolution can be done by two strategies:

1. *Querying an integrated HDT*: This strategy integrates all the dictionary and triple components of a subgraph into a new HDT (i.e. converting to the baseline HDT$_{crypt-A}$) which can be then queried.
2. *Local query on each dictionary and triple component*: In this case, the query is performed locally in each dictionary and triple component and the results are then integrated. Note that HDT$_{crypt-C}$ is not viable for this strategy as it would require to perform the D-union of all the dictionaries in order to search the triples IDs, which is then equivalent to integrating HDT$_{crypt-C}$ into a new HDT to be queried.

The following evaluation first inspects the performance overhead of the integration required by the former strategy. Then, we evaluate the query performance of the latter. For exemplary purposes, we present the average results of the DBpedia datasets, while the performance for LUBM and SAFE can be found in Appendix A.

Note that, although there are a number of strategies for querying *encrypted* data *directly* (see e.g., [4]), we consider these orthogonal and leave combining them with our partitioning for future work.

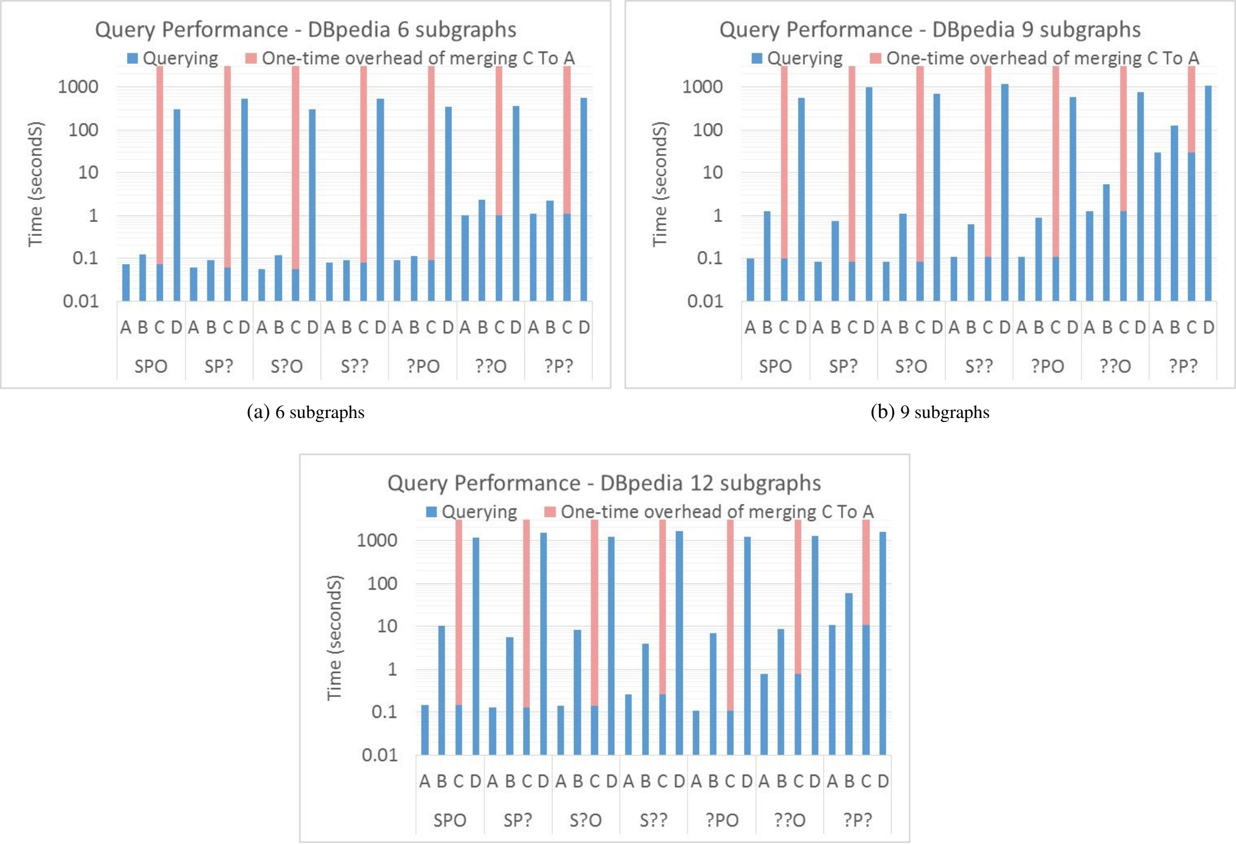
6.4.1. Integrating dictionary and triple components into a new HDT

Following our use case scenario, we assume that a user has decrypted half of the subgraphs, the i.e. M^6, M^9 and M^{12} subgraphs. Figure 10 shows the time required by each strategy (i.e. HDT$_{crypt-B}$, HDT$_{crypt-C}$ and HDT$_{crypt-D}$) to integrate their dictionary and triple components into one HDT per subgraph (e.g. G_1, G_2, G_6, G_7, G_{11} and G_{12} for M^{12}), *similarly to* the baseline HDT$_{crypt-A}$. This integration is performed as follows. First, all dictionary components are fed into a new dictionary, reorganizing the mapping between all terms and their corresponding IDs (as defined in Section 3.1). This first process is similar to the first step of the D-union (see Section 4.3.1). Then, we read the triple components and use the new dictionary to convert the triples to the new IDs, integrating all of them in a single new triple component per subgraph[17].

We present the time to integrate the dictionary and triple components of M^{12} into the corresponding subgraphs (Figure 10 a), for DBpedia. Yet again we see that HDT$_{crypt-C}$ is the fastest approach, 29% and 56% faster than B and D in DBpedia. In general, all approaches show a linear increase over dataset sizes, as shown in Appendix A.

A comparison in terms of number of subgraphs is shown in Figure 10 b, reporting the times of merging M^6, M^9 and M^{12} for DBpedia (the trends are similar for all datasets). As expected, given that the integration process yields to a partial decompression of the dictionary and triple components, the integration performance follows the same pattern as the decompression. That is, the even distribution of DBpedia results in a faster performance for 6 subgraphs, whereas the excessive duplicates of 12 penalises its performance.

[17] Note that this process slightly differs from the D-union as the latter only replaces the new IDs in each of the input triple component.



(a) 6 subgraphs

(b) 9 subgraphs

(c) 12 subgraphs

Fig. 11. Performance of Triple Patterns over DBpedia (average of the performance in both DBpedia versions).

6.4.2. *Query Performance*

We evaluate the query performance of all partitioning strategies in our use case scenario. Thus, for each subgraph in M^6, M^9 and M^{12} (and M^8_S and M^8_L in SAFE) we first generate 30 random queries for each triple pattern type[18], assuring an even presence of different predicates. Figure 11 shows the average execution time of the selected queries for both DBpedia versions (the results for LUBM4K and SAFE are presented in Appendix A). Note that, as shown in the previous section, the integration into a new HDT results in a non-negligible time to perform the process. Thus, for HDT$_{crypt-A}$, HDT$_{crypt-B}$ and HDT$_{crypt-D}$ we follow the strategy where queries are executed locally in each dictionary and triple component. In contrast, query execution in HDT$_{crypt-C}$ would require the D-union of all the dictionaries to be created, which is then equivalent to integrating HDT$_{crypt-C}$ into a new HDT to be queried. As such, the performance time for HDT$_{crypt-C}$ is presented as the sum of the time taken to create one integrated HDT (performed once), as previously explained in Section 6.4.1, and to subsequently query the integrated HDT (note again that this latter is equivalent to querying HDT$_{crypt-A}$).

Regarding the comparison between our strategies for partitioning, results show that HDT$_{crypt-A}$ and HDT$_{crypt-B}$ have the best performance for all patterns. This can be attributed to the fact that they benefit from efficient Bitmap Triples indexes, while HDT$_{crypt-D}$ must use Plain Triples (as stated in Section 5.4) that perform sequential scans to resolve queries. Note that HDT$_{crypt-D}$ is only competitive in the case of `(?p?)` queries (i.e. retrieving all subjects and objects related to a given predicate), given that most of the triples are returned and the total time is similar to a full sequential scan. In addition, Bitmap Triples indexes are less efficient for such query types [37]. As previously stated, HDT$_{crypt-C}$ behaves as HDT$_{crypt-A}$ but there is

[18] All queries are available at the HDTcrypt repository.

Table 7

Summary of performance of different HDT$_{crypt}$ strategies, where ⋆ ⋆ ⋆ stands for the best performance.

Strategy	Comp. & Encryp.			Decryp. & Decomp.	Querying	
	`Time`	`Size`	`Preconditions`	`Time`	`Time`	`Preconditions`
crypt-A	⋆ ⋆	⋆	*None*	⋆	⋆ ⋆ ⋆	*None*
crypt-B	⋆	⋆ ⋆	HDT of full graph G	⋆ ⋆	⋆ ⋆	*None*
crypt-C	⋆ ⋆ ⋆	⋆ ⋆ ⋆	HDT of full graph G	⋆ ⋆ ⋆	⋆ ⋆	Once-off integration to a new HDT
crypt-D	⋆	⋆ ⋆ ⋆	HDT of full graph G	⋆ ⋆	⋆	*None*

Table 8

Influence of the increasing number of subgraphs and duplicates in the performance of different HDT$_{crypt}$ strategies, where $+++$ stands for very positive and $---$ for very negative.

Strategy	Comp. & Encryp.		Decryp. & Decomp.	Querying
	`Time`	`Size`	`Time`	`Time`
crypt-A	$-$	$-$	$-$	$-$
crypt-B	$-$	$+++$	$-$	$-$
crypt-C	$-$	$+++$	$-$	$-$
crypt-D	$---$	$++$	$-$	$--$

a once-off overhead associated with merging all dictionary and triple components into one HDT (represented in red in Figure 14).

In turn, it is also worth mentioning that HDT$_{crypt-B}$ query performance is closer to the baseline HDT$_{crypt-A}$ in the scenario with 6 subgraphs. This is mainly due to the larger number of dictionaries/triples to be queried in a scenario with a higher number of subgraphs (as shown in Table 4), which penalises the HDT$_{crypt-B}$ and HDT$_{crypt-D}$ methods. In this scenario, HDT$_{crypt-A}$ is the most efficient approach for query execution. The noticeable performance difference against the rest of the partitioning approaches suggests that the once-off merging that is required for HDT$_{crypt-C}$ can be amortised if the dataset is meant for intensive querying after decryption.

6.5. *Discussion of the results*

Overall, our empirical evaluation showed interesting results and allows us to draw conclusions on the applicability of each strategy. We summarize a ranking of results for each scenario in Table 7, and we outline the influence of the increasing number of subgraphs and duplicates in the data in Table 8, detailed as follows:

– HDT$_{crypt-C}$ is the most effective technique in terms of compression and decompression times, as well as compression sizes. In particular, it achieves additional 26-33% space saving over the –already compressed– baseline (HDT$_{crypt-A}$), and it is 37-40% faster to compress, and 30% faster to decompress. Note that the impact of these space and time savings are even more noticeable when dealing with big data. As we noticed, if the original HDT of the full graph is not available beforehand, then the creation of HDT$_{crypt-C}$ can take more time than the baseline (it results in approx. the same time in LUBM and 35-50% slower in DBpedia, with a rich dictionary of terms), but it keeps the aforementioned noticeable space savings. In the extreme case of isolated subgraphs with few duplicates, as in SAFE, HDT$_{crypt-C}$ takes the same space as the baseline and is still 18% faster to encrypt.

– In contrast, HDT$_{crypt-C}$ does not allow the user to directly query the exchanged information. If such a scenario is required, this can be solved with a once-off conversion from HDT$_{crypt-C}$ to HDT$_{crypt-A}$. This conversion can be done for any strategy, but it is indeed faster for HDT$_{crypt-C}$.

– HDT$_{crypt-B}$ and HDT$_{crypt-D}$ also reduce the size of the baseline (HDT$_{crypt-A}$), and can be directly queried. Results show that HDT$_{crypt-B}$ and HDT$_{crypt-D}$ gain 6-24% and 24-26% space over the baseline respectively, at the cost of an extra 68% and 23% time for compression (performed only once by the data publisher). In turn, the decompression time outperforms the baseline by 7% and 9% for HDT$_{crypt-B}$ and HDT$_{crypt-D}$ respectively. In the extreme case of isolated subgraphs with few duplicates, as in SAFE, HDT$_{crypt-B}$ and HDT$_{crypt-D}$ suffer from a slight space overhead (15-23%) over the baseline, and non negligible additional decompressing times.

– The performance of directly querying several subgraphs in HDT$_{crypt-B}$ is close to the baseline HDT$_{crypt-A}$. Nonetheless, it is penalised at larger

number of partitions (such as 12 in our experiments) and larger number of duplicates (such as our even distribution in DBpedia). HDT$_{crypt-D}$ suffers from the additional problem of performing sequential scans, and is not competitive but for queries that retrieve large number of results.
- Encryption and decryption times are almost negligible compared to the compression/decompression counterparts.
- Compression sizes, compression and decompression times show linear growth with increasing dataset size.
- In general, an increasing number of subgraphs leads to more duplicates and more space savings of our novel HDT$_{crypt-B}$, HDT$_{crypt-C}$ and HDT$_{crypt-D}$ partitioning approaches over the baseline HDT$_{crypt-A}$. In turn, less data file sizes result in faster decompression of our novel approaches. In contrast, the compression time is penalised given that more components have to be generated. Our experiments also showed that the number of subgraphs does not have a strong influence on the query performance, but the skewed distribution of sizes and the large number of components (such as in DBpedia) can result in slight differences between scenarios.

7. Conclusions and Future Work

To date Linked Data publishers have focused on exposing and linking *open* data, however the Linked Data infrastructure could be extended to cater for the storage and exchange of confidential data. In this paper, we discussed how HDT compression can be extended to cater for RDF datasets which needs to be encrypted. Specifically, we proposed a number of different compression strategies that are compatible and demonstrated the need for careful integration when it comes to compressed encrypted RDF data. From our evaluation we can see that our proposal HDT$_{crypt-C}$ outperforms the other partitioning strategies both in terms of compression and decompression time, and it also produces the most compact representation, resulting in 26-31% space savings over the –already compressed– baseline. HDT$_{crypt-B}$ and HDT$_{crypt-D}$ also reduce the size of the baseline significantly. Whereas, when it comes to querying HDT$_{crypt-A}$ and HDT$_{crypt-B}$ outperform HDT$_{crypt-C}$, which incurs additional overhead as the dictionaries and triples need to be integrated in order to support querying. Additionally, we note that compression, decompression and query performance is influenced both by the number of access restricted subgraphs and the distribution of triples across subgraphs, especially in HDT$_{crypt-D}$. In future work, we plan to extend our existing work to cater for querying over encrypted compressed data without the need for decryption. Our current work considers basic triple pattern resolution, while the HDT approach can be used as the basic engine to resolve full SPARQL queries. Our plan is to support this possibility on the compressed and encrypted data in future work.

Acknowledgements

Supported by the European Union's Horizon 2020 research and innovation programme under grant 731601, Austrian Science Fund (FWF): M1720-G11, the Austrian Research Promotion Agency (FFG) under grant 845638 (SHAPE) and grant 861213 (CitySpin), and MINECO-AEI/FEDER-UE ETOME-RDFD3:TIN2015-69951-R. Axel Polleres was supported by the "Distinguished Visiting Austrian Chair" program as a visiting professor hosted at The Europe Center and the Center for Biomedical Research (BMIR) at Stanford University. Special thanks to Ratnesh Sahay for providing the SAFE dataset.

References

[1] S. Álvarez-García, N.R. Brisaboa, J.D. Fernández, M.A. Martínez-Prieto and G. Navarro, Compressed vertical partitioning for efficient RDF management, *Knowl. Inf. Syst.* **44**(2) (2015), 439–474. doi:10.1007/s10115-014-0770-y. https://doi.org/10.1007/s10115-014-0770-y.

[2] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak and Z. Ives, Dbpedia: A nucleus for a web of open data, in: *Proc. of ISWC*, 2007. doi:10.1007/978-3-540-76298-0_52.

[3] W. Beek, L. Rietveld, H.R. Bazoobandi, J. Wielemaker and S. Schlobach, LOD laundromat: a uniform way of publishing other people's dirty data, in: *Proc. of ISWC*, LNCS, Vol. 8796, Springer, 2014, pp. 213–228. doi:10.1007/978-3-319-11964-9_14.

[4] M. Bellare, A. Boldyreva and A. O'Neill, Deterministic and Efficiently Searchable Encryption, in: *Advances in Cryptology - CRYPTO 2007, 27th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2007, Proceedings*, 2007, pp. 535–552. doi:10.1007/978-3-540-74143-5_30. https://doi.org/10.1007/978-3-540-74143-5_30.

[5] D. Brickley, R.V. Guha and (eds.), RDF Vocabulary Description Language 1.0: RDF Schema, W3C Recomm., W3C, 2004. http://www.w3.org/TR/rdf-schema/.

[6] N. Brisaboa, R. Cánovas, F. Claude, M.A. Martínez-Prieto and G. Navarro, Compressed String Dictionaries, in: *Proc. of SEA*, 2011, pp. 136–147. doi:10.1007/978-3-642-20662-7_12.

[7] M. Chase and E. Shen, Pattern Matching Encryption, *IACR Cryptology ePrint Archive* **2014** (2014), 638. http://eprint.iacr.org/2014/638.

[8] L. Costabello, S. Villata, N. Delaforge and F. Gandon, Linked Data Access Goes Mobile: Context-Aware Authorization for Graph Stores, in: *WWW2012 Workshop on Linked Data on the Web, Lyon, France, 16 April, 2012*, 2012. http://ceur-ws.org/Vol-937/ldow2012-paper-05.pdf.

[9] O. Curé, G. Blin, D. Revuz and D.C. Faye, WaterFowl: A Compact, Self-indexed and Inference-Enabled Immutable RDF Store, in: *Proc. of ESWC*, LNCS, Vol. 8465, 2014, pp. 302–316. doi:10.1007/978-3-319-07443-6_21.

[10] R. Curtmola, J. Garay, S. Kamara and R. Ostrovsky, Searchable symmetric encryption: improved definitions and efficient constructions, in: *Proc. of CSS*, ACM, 2006, pp. 79–88. doi:10.1145/1180405.1180417.

[11] J. Daemen and V. Rijmen, *The Design of Rijndael: AES - The Advanced Encryption Standard*, Information Security and Cryptography, Springer, 2002. ISBN 3-540-42580-2. doi:10.1007/978-3-662-04722-4. https://doi.org/10.1007/978-3-662-04722-4.

[12] H.V. de Sompel, R. Sanderson, M.L. Nelson, L. Balakireva, H. Shankar and S. Ainsworth, An HTTP-Based Versioning Mechanism for Linked Data, in: *Proceedings of the WWW2010 Workshop on Linked Data on the Web, LDOW 2010, Raleigh, USA, April 27, 2010*, 2010. http://ceur-ws.org/Vol-628/ldow2010_paper13.pdf.

[13] J.D. Fernández, Binary RDF for scalable publishing, exchanging and consumption in the web of data, PhD thesis, 2012. doi:10.1145/2187980.2187997. http://doi.acm.org/10.1145/2187980.2187997.

[14] J.D. Fernández, A. Polleres and J. Umbrich, Towards Efficient Archiving of Dynamic Linked Open Data, in: *Proceedings of the First DIACHRON Workshop on Managing the Evolution and Preservation of the Data Web co-located with 12th European Semantic Web Conference (ESWC 2015), Portorož, Slovenia, May 31, 2015.*, 2015, pp. 34–49. http://ceur-ws.org/Vol-1377/paper6.pdf.

[15] J.D. Fernández, M.A. Martínez-Prieto, C. Gutiérrez, A. Polleres and M. Arias, Binary RDF representation for publication and exchange (HDT), *J. Web Sem.* **19** (2013), 22–41. doi:10.1016/j.websem.2013.01.002. https://doi.org/10.1016/j.websem.2013.01.002.

[16] J.D. Fernández, S. Kirrane, A. Polleres and S. Steyskal, Self-Enforcing Access Control for Encrypted RDF, in: *Proc. of ESWC*, 2017. doi:10.1007/978-3-319-58068-5_37.

[17] J.D. Fernández, M.A. Martínez-Prieto, C. Gutiérrez and A. Polleres, *Binary RDF Representation for Publication and Exchange (HDT)*, W3C Member Submission, 2011. doi:10.1016/j.websem.2013.01.002. https://www.w3.org/Submission/HDT/.

[18] J.D. Fernández, M. Arias, M.A. Martínez-Prieto and C. Gutiérrez, Management of Big Semantic Data, in: *Big Data Computing*, Taylor and Francis/CRC, 2013, Chap. 4. doi:10.1201/b16014-7.

[19] C. Gentry, Fully homomorphic encryption using ideal lattices., in: *Proc. of STOC*, Vol. 9, ACM, 2009, pp. 169–178. doi:10.1145/1536414.1536440.

[20] S. Gerbracht, Possibilities to Encrypt an RDF-Graph, in: *Proc. of ICTTA*, IEEE, 2008, pp. 1–6. doi:10.1109/ictta.2008.4530288.

[21] M. Giereth, On Partial Encryption of RDF-Graphs., in: *Proc. of ISWC*, LNCS, Vol. 3729, Springer, 2005, pp. 308–322. ISBN 3-540-29754-5. doi:10.1007/11574620_24. http://dblp.uni-trier.de/db/conf/semweb/iswc2005.html#Giereth05.

[22] M. Giereth, PRE4J - A Partial RDF Encryption API for Jena, *ACAD MED* **70**(3) (2006), 216–223. http://jena.hpl.hp.com/juc2006/proceedings/giereth/paper.pdf.

[23] J.M. Giménez-García, J.D. Fernández and M.A. Martínez-Prieto, HDT-MR: A Scalable Solution for RDF Compression with HDT and MapReduce, in: *Proc. of ESWC*, 2015, pp. 253–268. doi:10.1007/978-3-319-18818-8_16.

[24] Y. Guo, Z. Pan and J. Heflin, LUBM: A Benchmark for OWL Knowledge Base Systems, *JWS* **3**(2) (2005), 158–182. doi:10.2139/ssrn.3199255.

[25] C. Gutiérrez, C. Hurtado, A.O. Mendelzon and J. Perez, Foundations of Semantic Web Databases, *JCSS* **77** (2011), 520–541. doi:10.1016/j.jcss.2010.04.009.

[26] S. Harris and A. Seaborne, SPARQL 1.1 Query Language, W3C Recomm., W3C, 2013. http://www.w3.org/TR/sparql11-query/.

[27] A. Hernández-Illera, M.A. Martínez-Prieto and J.D. Fernández, Serializing RDF in compressed space, in: *Data Compression Conference (DCC), 2015*, IEEE, 2015, pp. 363–372. doi:10.1109/dcc.2015.16.

[28] A. Hogan, M. Arenas, A. Mallea and A. Polleres, Everything You Always Wanted to Know About Blank Nodes, *Journal of Web Semantics (JWS)* (2014), 42–69. doi:10.2139/ssrn.3199109.

[29] A. Joshi, P. Hitzler and G. Dong, Logical Linked Data Compression, in: *Proc. of ESWC*, LNCS, Vol. 7882, Springer, 2013, pp. 170–184. doi:10.1007/978-3-642-38288-8_12.

[30] L. Kagal, T. Finin and A. Joshi, A Policy Based Approach to Security for the Semantic Web, in: *Proc. of ISWC*, 2003. doi:10.1007/978-3-540-39718-2_26.

[31] A. Kasten, A. Scherp, F. Armknecht and M. Krause, Towards Search on Encrypted Graph Data, in: *Proceedings of the Workshop on Society, Privacy and the Semantic Web - Policy and Technology (PrivOn2013) co-located with the 12th International Semantic Web Conference (ISWC 2013), Sydney, Australia, October 22, 2013.*, 2013. http://ceur-ws.org/Vol-1121/privon2013_paper5.pdf.

[32] J. Katz, A. Sahai and B. Waters, Predicate Encryption Supporting Disjunctions, Polynomial Equations, and Inner Products, *J. Cryptology* **26**(2) (2013), 191–224. doi:10.1007/s00145-012-9119-4. https://doi.org/10.1007/s00145-012-9119-4.

[33] Y. Khan, M. Saleem, M. Mehdi, A. Hogan, Q. Mehmood, D. Rebholz-Schuhmann and R. Sahay, SAFE: SPARQL Federation over RDF Data Cubes with Access Control, *Journal of biomedical semantics* **8**(1) (2017), 5. doi:10.1186/s13326-017-0112-6.

[34] P. Kolari, L. Ding, S. Ganjugunte, A. Joshi, T.W. Finin and L. Kagal, Enhancing Web Privacy Protection through Declarative Policies, in: *6th IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY 2005), 6-8 June 2005, Stockholm, Sweden*, 2005, pp. 57–66. doi:10.1109/POLICY.2005.15. https://doi.org/10.1109/POLICY.2005.15.

[35] V. Kolovski, J. Hendler and B. Parsia, Analyzing Web Access Control Policies, in: *Proc. of WWW*, 2007. doi:10.1145/1242572.1242664.

[36] S. Maneth and F. Peternek, Grammar-based graph compression, *Information Systems* **76** (2018), 19–45, ISSN 0306-4379. doi:10.1016/j.is.2018.03.002.

[37] M.A. Martínez-Prieto, M. Arias and J.D. Fernández, Exchange and Consumption of Huge RDF Data, in: *Proc. of ESWC*, LNCS, Vol. 7295, Springer, 2012, pp. 437–452. doi:10.1007/978-3-642-30284-8_36.

[38] M.A. Martínez-Prieto, J.D. Fernández and R. Cánovas, Querying RDF Dictionaries in Compressed Space, *SIGAPP Appl. Comput. Rev.* **12**(2) (2012), 64–77. doi:10.1145/2340416.2340422.

[39] J.Z. Pan, J.M. Gómez-Pérez, Y. Ren, H. Wu and M. Zhu, SSP: Compressing RDF data by Summarisation, Serialisation and Predictive Encoding, Technical Report, K-Drive, 2014. http://www.kdrive-project.eu/wp-content/uploads/2014/06/WP3-TR2-2014_SSP.pdf.

[40] S. Pearson and M.C. Mont, Sticky policies: an approach for managing privacy across multiple parties, *Computer* **44**(9) (2011), 60–68. doi:10.1109/mc.2011.225.

[41] R. Pichler, A. Polleres, S. Skritek and S. Woltran, Complexity of redundancy detection on RDF graphs in the presence of rules, constraints, and queries, *Semantic Web* **4**(4) (2013), 351–393. doi:10.3233/SW-2012-0076. https://doi.org/10.3233/SW-2012-0076.

[42] R. Popa, N. Zeldovich and H. Balakrishnan, Cryptdb: A Practical Encrypted Relational dbms., Technical Report, MIT-CSAIL-TR-2011-005, 2011. http://hdl.handle.net/1721.1/60876.

[43] E. Rissanen, Extensible access control markup language (xacml) version 3.0, OASIS Standard, available at http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-en.html, OASIS Committee Specification, 2013.

[44] O. Sacco, A. Passant and S. Decker, An Access Control Framework for the Web of Data, in: *Proc. TrustCom*, 2011, pp. 456–463. doi:10.1109/trustcom.2011.59.

[45] G. Schreiber and Y. Raimond, RDF 1.1 Primer W3C Working Group Note, 2014. https://www.w3.org/TR/rdf11-primer/.

[46] S. Steyskal and S. Kirrane, If you can't enforce it, contract it: Enforceability in Policy-Driven (Linked) Data Markets, in: *Joint Proceedings of the Posters and Demos Track of 11th International Conference on Semantic Systems - SEMANTiCS 2015 and 1st Workshop on Data Science: Methods, Technology and Applications (DSci15) 11th International Conference on Semantic Systems - SEMANTiCS 2015, Vienna, Austria, September 15-17, 2015.*, 2015, pp. 63–66. http://ceur-ws.org/Vol-1481/paper21.pdf.

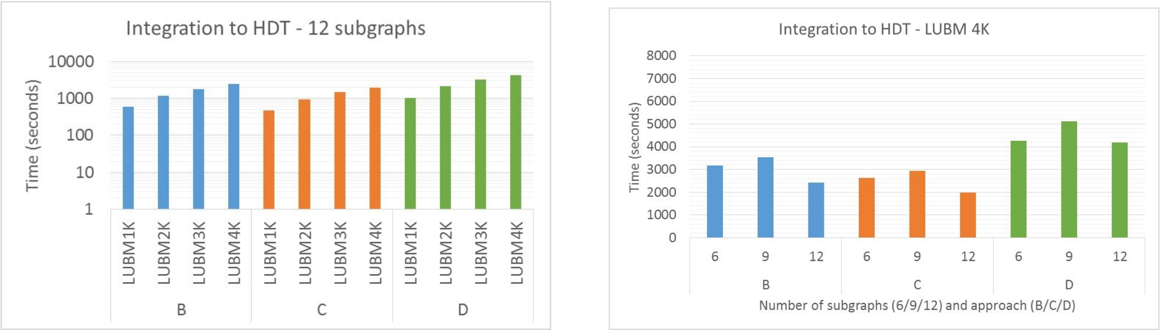
[47] Q. Tang, On Using Encryption Techniques to Enhance Sticky Policies Enforcement, Technical Report, CTIT University of Twente, 2008. http://eprints.eemcs.utwente.nl/14262/.

Appendix A. Additional Performance Results

This appendix comprises the performance results for all datasets. See Section 6 for a description of the corpus and the complete discussion of results.

A.1. Integrating dictionary and triple components into a new HDT

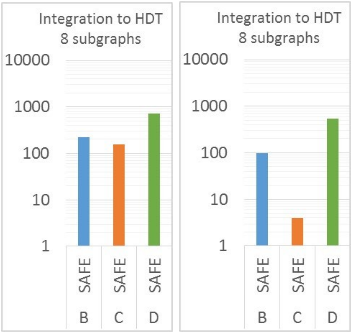
Figure 12 shows the time (in seconds) to integrate the dictionary and triples components of half of the partitions (M^6, M^9 and M^{12} as explained in Section 6) of LUBM into a single HDT per subgraph. We present the time to integrate the dictionary and triple components of M^{12} into the corresponding subgraphs (Figure 12 a), and a comparison in terms of number of subgraphs (Figure 10 b). Figure 13 shows the integration of the SAFE dataset for the two scenarios, M^8_L (left) and M^8_S (right).



(a) 12 subgraphs (M^{12}) for LUBM

(b) Comparison of 6/9/12 subgraphs (M^6, M^9 and M^{12}) for LUBM4K

Fig. 12. Integration of the dictionary and triple components of M^6, M^9 and M^{12} into one HDT per subgraph.

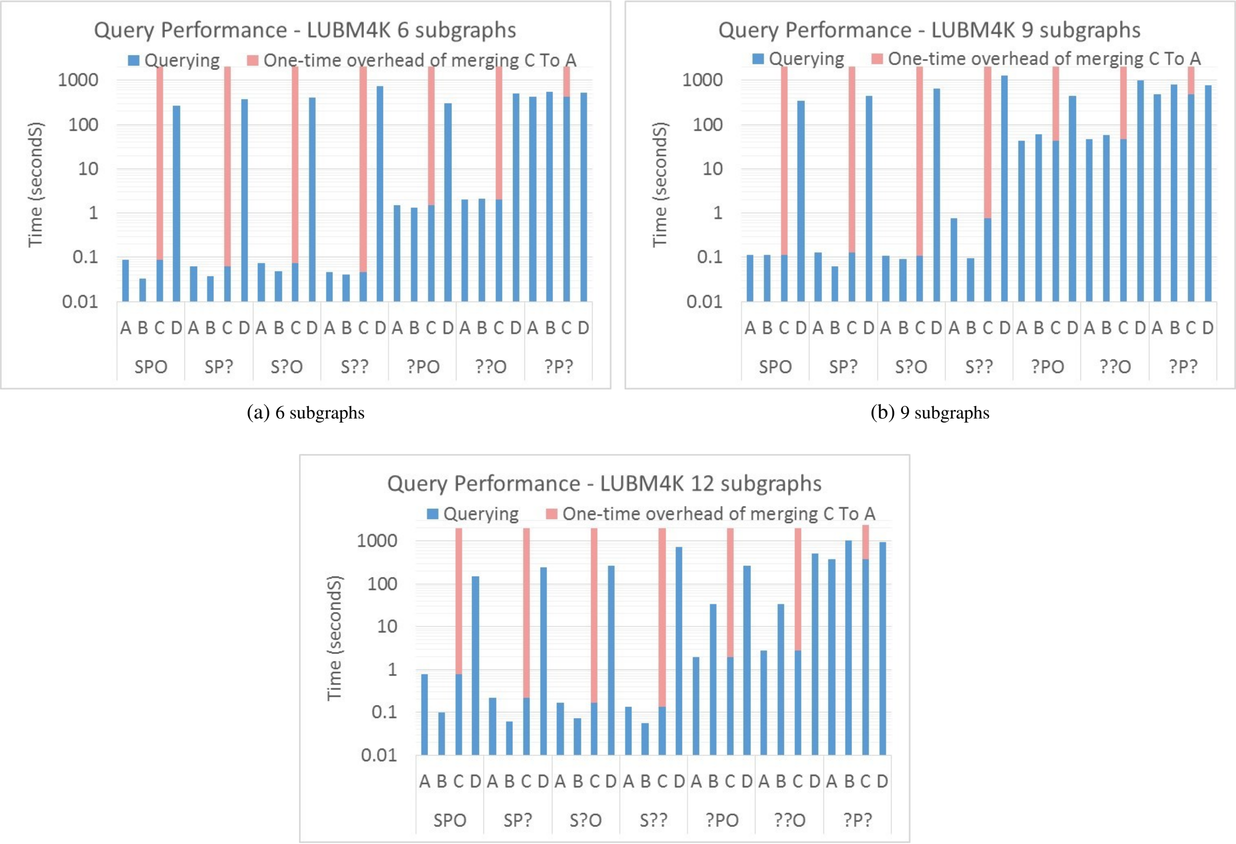


(a) M^8_L (left) and M^8_S for SAFE

Fig. 13. Integration of the dictionary and triple components of M^8_S and M^8_L into one HDT per subgraph.

A.2. Querying Compressed Data

Figures 14 and 15 show the performance of the selected Triple Patterns over LUBM4K and SAFE, respectively. Results for smaller datasets of LUBM4K follow the same trends. As in the case of DBpedia, presented in Section 6, results show that HDT$_{crypt-A}$ and HDT$_{crypt-B}$ have the best performance for all patterns, outperforming its results when few triples are returned, such as (spo) and (sp?) queries. Note that, although HDT$_{crypt-B}$ has to query more dictionaries and triple components than HDT$_{crypt-A}$, the number of total components is very limited in LUBM (the number of components is shown in Table 4) and each component is smaller in HDT$_{crypt-B}$ than in HDT$_{crypt-A}$. For instance, the resolution of a (sp?) pattern using HDT$_{crypt-A}$ for M^{12} in LUBM4K (see performance results in Figure 14 a) has to query 6 large triple components (one per subgraph), where duplicated triples can be present. In contrast, for HDT$_{crypt-B}$ we could verify that there are 37 triple components in M^{12}, but they are smaller and triples do not overlap. As for SAFE, note that the dataset is particularly small and has few overlapping triples, hence the techniques performance similarly, except for the aforementioned additional overheads in HDT$_{crypt-D}$.



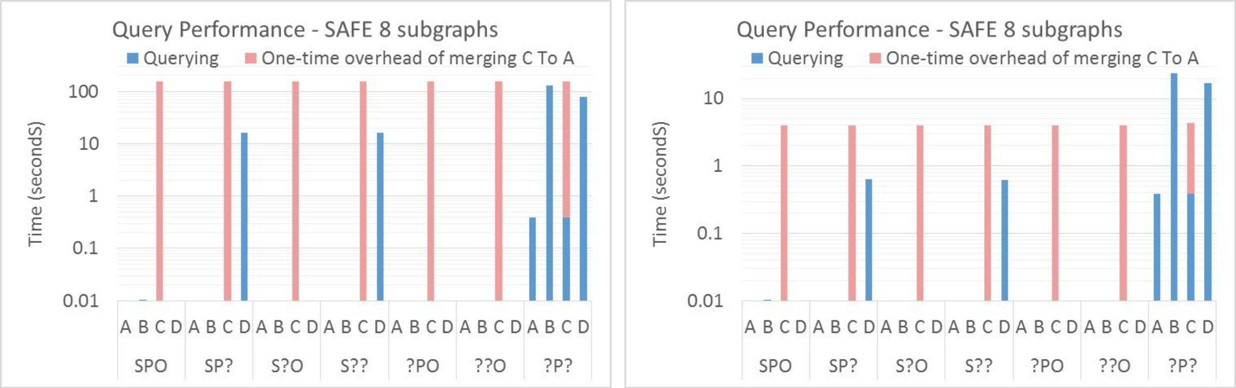
(a) 6 subgraphs

(b) 9 subgraphs

(c) 12 subgraphs

Fig. 14. Performance of Triple Patterns over LUBM4K.

Finally, Figure 16 presents the results of a particular scenario designed to evaluate the potential influence of the number of graphs in a fair manner. Note that, in the previous use case, the number of results could differ in each subgraph as M^6, M^9 and M^{12} include different subgraphs (e.g. *ResearchAssistant* is included as G_6 in M^{12} but it is present neither in M^9 nor M^6). This fact hampers a fair comparison of the query performance, given that the number of results could differ. This situation is even worse in DBpedia, where each subgraph contains randomly selected triples. Thus, for this particular comparison, we select the *University* subgraph in LUBM, which is present in M^{12} (as G_{11} in Table 2), M^9 (as G_8) and M^6 (as G_6). We then generate 30 random triple pattern queries of each type (similarly to the previous scenario) and perform such queries on the aforementioned *University* subgraph. Figure 16 reports the total performance of all queries for LUBM4K (results are similar for smaller sizes). Note that HDT$_{crypt-A}$ reports the same time in all cases and they compress the same subgraph. In general, results are in line with the previous observations regarding the influence of subgraphs for decompression. That is, in general, 12 subgraphs is the fastest approach, whereas the larger size of the files and their duplication ratio place also a burden on the query performance of 9 subgraphs. Nonetheless, we can find a minor difference in HDT$_{crypt-D}$, where the



(a) 8 subgraphs, M_L^8

(b) 8 subgraphs, M_S^8

Fig. 15. Performance of Triple Patterns over SAFE.

case of 6 subgraphs reports the worst performance. A closer look at the generated dictionary and triple components for the particular *University* subgraph allows us to conclude that this particular case produced a skewed distribution of sizes in 6 subgraphs. For example, the largest dictionary component takes 75MB, whereas it is only 27MB and 12MB for 9 and 12 subgraphs respectively. Note that although this skewed distribution is also present in DBpedia, in practice, HDT$_{crypt-D}$ can be slower with 12 subgraphs than with 6 subgraphs, given that the much larger number of dictionary and triple components in 12 subgraphs (due to the duplication ratio) are the predominant factor.

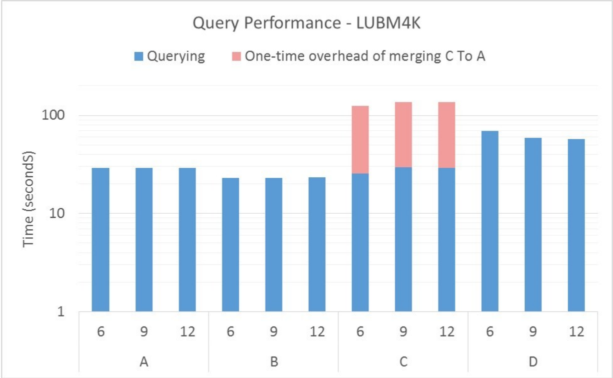


Fig. 16. Performance of all Triple Patterns over LUBM4K in the *University* subgraph.