Christophe Debruyne. Introducing Context and Context-awareness in Data Integration: Identifying the Problem and a Preliminary Case Study on Informed Consent. In The 22nd International Conference on Information Integration and Web-based Applications & Services (iiWAS '20), November 30-December 2, 2020, Chiang Mai, Thailand, ACM, 2020
(accepted version)

Introducing Context and Context-awareness in Data Integration: Identifying the Problem and a Preliminary Case Study on Informed Consent

Christophe Debruyne
Trinity College Dublin
Dublin 2, Ireland
debruync@tcd.ie

ABSTRACT

Data integration is the process of selecting, preprocessing, and transforming data from heterogeneous sources in data-driven projects. This process also requires the most time, effort, resources. Data integration is such an involved process due to the many informed decisions one has to make. These decisions are influenced by the complex context of a data-driven project. We argue that using said context could facilitate the decision-making processes and even automate some integration steps. However, the problem we identify in this paper is that the context of a data-driven project is tacit and, therefore, not easily accessible by humans and certainly not by software agents. From the SotA, however, we observe that current context models approach context in crude and simplistic terms. Context-aware data integration, proposals are furthermore often built for specific tasks or application domains such as query optimization or a smart home. The current state of affairs is thus is not fit for intelligent data integration. Next to identifying the problem, we postulate that solving this problem requires two steps: formalizing context and using that context for building context-aware agents. We illustrate this notion of "context-aware data integration" with preliminary results obtained with a use case in the domain of GDPR, more specifically the generation of datasets that takes into account informed consent.

CCS CONCEPTS

•Information systems~Data management systems~Information integration•Information systems~World Wide Web~Web data description languages~Semantic web description languages

KEYWORDS

Context-aware Data Integration, Context, Ontology Engineering

WOODSTOCK'18, June, 2018, El Paso, Texas USA
 978-1-4503-0000-0/18/06...$15.00
https://doi.org/10.1145/1234567890

ACM Reference format:

FirstName Surname, FirstName Surname and FirstName Surname. 2018. Insert Your Title Here: Insert Subtitle Here. In *Proceedings of ACM Woodstock conference (WOODSTOCK'18). ACM, New York, NY, USA, 2 pages.* https://doi.org/10.1145/1234567890

1 Introduction

In data-driven projects, *data integration* is the process that requires the most time, effort, and resources in data-driven projects [1]. However, to this day, the integration of data remains mostly reliant on human input and decision making [2]. Practitioners[1] are responsible for making numerous informed decisions while selecting, preprocessing, and transforming the data. The context of a project informs decision making. That context includes, in short, the data-driven project and its characteristics, the disciplines involved in data-driven projects, and the practitioner's (prior) knowledge, experience, and background.

All these aspects constitute the context, are subject to change, and affect how one conducts data integration. Context is critical to make informed decisions –often not technical. The integration of context is the most significant barrier to complex data integration. From the State-of-the-Art, we can observe that two decades of context-aware computing research have led to only limited simple or 'crude' information from sensors being integrated and siloed "contexts", which ignores and underutilizes the vast and complex nature of the context. This profoundly limits the potential of data-driven projects. Context is important.

From a data governance and policy perspective, for example, the use of previously collected personal data for a different purpose will require explicit approval (from an ethics committee or the data subjects). From a data science and statistics perspective, for example, domain expertise is necessary to integrate data meaningfully. These aspects also influence how one implements data integration flows, a challenge from a data engineering perspective.

[1] With "practitioner", we refer to all the stakeholders actively involved in a data-driven project. This to distinguish practitioners from other stakeholders such as clients, users, etc.

In this paper, we will look at the State-of-the-Art on context-awareness in data integration and identify the problem that context is often regarded in too simplistic terms (Section 2). In Section 3, we elaborate on context in data-driven projects and how one should divide the creation of context-aware data integration agents into 1) the creation of context ontologies, and 2) the creation of those agents using those ontologies. In Section 4, we illustrate our approach in the context of GDPR and the generation of GDPR compliant datasets. This is thus an example of policy-compliant data integration. The algorithms used in this illustration were already described in prior work from a technical perspective. Their integration into this paper's narrative as an illustration of context-aware data integration is novel. And finally, in Section 5 we conclude the paper.

2 Context-awareness and Data Integration

Data integration is traditionally concerned with combining data from various heterogeneous source representations to a target representation. One thus needs to prescribe how one transforms the source into the target with *mappings* [3], [4]. Various (standardized) representations exist for mappings such as SQL for views [3] in a data warehousing setting and R2RML[2], a W3C Recommendation, for declaring mappings from relational data to the Resource Description Framework (RDF). The scientific community has looked into facilitating data integration with ontology matching and aligning [4], rules to link entities in different datasets [5], and even the use of AI [6] and ML [7] to find correspondences in datasets. [8] recently demonstrated how they used NLP to extract knowledge from unstructured text to facilitate data integration. So while the community has made significant strides in representing and executing mappings, and even automating data integration in terms of ontology/schema matching and data interlinking, we have yet to make a big leap forward by using context for driving the whole data integration process. In this section, we will present and assess related work from two perspectives: the context-model and the techniques availing of said model for context-aware data integration.

2.1 Related Work on a Context Model

Context is defined as "any information that can be used to characterize the situation of entities [...] that are considered relevant to the interaction between a user and an application, including the user and the application themselves." [9] The authors of [9] furthermore state that context is typically the location, state, and identity of people and objects. As far as capturing context goes, researchers have tried modeling context for their uses; surveys include [10] for IoT, [11] for mobile environments, and [12] for pervasive computing. In [13] and [14], the authors provide examples of context modeling for personalization and recommender systems, respectively.

The State-of-the-Art often limits itself to a limited number of so-called *sensors*, which are often crude and simple. Those sensors are used to distill information from facts that can be sensed, either via hardware or software, and usually limited to the identity of agents, location, time, and environment [13]. This limitation is still actual after two decades of research in context-aware computing [10]. What we can observe is that one is often concerned with the integration of (specific types of) sensor data and systems are closed – i.e., the applications are in control what (types of) data is processed Few context modeling initiatives take into account *events*, e.g. [15] or *activities*, e.g. [16]. While vague, [17] did introduce the notion of "application context", albeit in a network embedded system. This application context is a group of rules that fire messages to applications. This effort is worth mentioning as it is one of the few considering 'rules' as part of the model next to using rules for using the model.

Context has also been formalized to capture knowledge that is true in different scenario's or so-called "possible worlds" as introduced by [18] and more recently by [19]. Since such efforts are concerned with formalizing "containers" of knowledge and the relationships between these containers, we do not consider those relevant for the related work described later on.

2.2 Beyond the "traditional" notion of context

Looking beyond traditional sensors that could constitute 'context', we can look at initiatives describing datasets. Within the scientific community, however, strides have been made to capture aspects of datasets. A data value vocabulary proposed in [20] aims to assess and quantify a dataset's value for a particular purpose. Their work was informed by an organization's need to assess their assets' value. Not only is that need part of the context, but the so-called data value dimensions are also valuable for subsequent data-integration steps. Luzzu [21] allows one to declare data quality metrics to assess RDF datasets, which can be used to rank datasets in a personalized manner. These vocabularies need to be combined with data description vocabularies such as DCAT [22] and VOiD [23] for future interrogation.

We also consider vocabularies for describing activities beyond traditional sensors. The scientific community has looked into formalizing processes and activities for specific projects. In bioinformatics, [24] proposed an ontology for detailing workflows in that domain. The P-PLAN ontology [25], extending PROV-O[3] was created to prescribe scientific workflows. [26] proposed an ontology for modeling ontology engineering workflows in the Protégé4 ontology development environment. While we can deem the work of [26] as "domain agnostic", since one can apply such projects in any domain, the ontology is created for ontology-engineering projects and used to capture the provenance information of each activity, e.g., to inspect its compliance or reproduce steps. More "generic" initiatives for process models also exist. We have, for instance, sBPMN for business process models [27], but their purpose is to provide semantic interoperability for such models. In [28], the authors

[2] https://www.w3.org/TR/r2rml/, last accessed November 4, 2019

[3] https://www.w3.org/TR/prov-o/, last accessed November 4, 2019. PROV-O a W3C Recommendation for relating activities, entities, agents, and their relationships.

proposed annotating the manipulation and analysis of data in data processing "pipelines" for the creation of meaningful logs– i.e., annotated programs and code. Here, the authors also availed of PROV-O to representing activities, which the authors have used to render data analysis traceable and reproducible. In that sense, they avail of PROV-O to represent and store "sensor data" but do not use it to model the context.

While [17] provided an anecdotal example of grouping knowledge, the State-of-the-Art does not group concepts and relations for specific purposes. This is likely due to the simplicity of the context and the controlled environments of the middleware; there are no two different interpretations of a concept, for instance.

2.3 Related Work on Context-aware Data Integration

We argue that it does not matter what formalism is used to represent the context model, but that formalism needs to be expressive enough to be rendered actionable. The chosen formalism, however, may affect the reasoning capabilities of the overall solution. Rule-based approaches may avail of rule- or inference engines. The HYDRA Middleware Project [17], for instance, relies on the Drools [29] rule engine. Others, such as [15], use the OWL to model context, which allows them to avail of OWL reasoning to infer additional facts. If the chosen formalism has no support for "third-party" reasoning engines, then the "reasoning" needs to be implemented in the solution. It seems, from the aforementioned surveys in different domains (CFR. Section 2.1), that rule languages are more widely adopted. Indeed, OWL reasoners are built for specific reasoning tasks such as classification, inferring subclasses, and satisfiability checking.

One can make a distinction between solutions that integrate context-awareness in the application or the middleware. An example of the former is presented [30] for managing intrusive push notifications on mobile phones. In [17], the authors presented a middleware solution for integrating wireless devices combining a domain ontology with rules. In [31], the authors reported a middleware for a Smart City environment in which stationary sensors are combined with "mobile" sensors (such as a smartphone) to provide more intelligent data services. They achieve this by combining bespoke services with a rule base. As the application domains are specific, the literature does not look into the practice of data integration (and data analysis). Such platforms provide a well-defined and well-scoped environment for data curation and integration activities that limit the complexity of data integration.

In [32], the authors present an approach to optimize query-execution in data integration by examining the network and inspecting metadata of the data sources. Their work was published a decade before any of the open description standards emerging. More recent are [33] and [34], who both recognize that formalizing the UoD of context should be kept separate from the systems. In [33], the authors addressed the problem of defining views over relational databases in a context-aware manner. They proposed to model context as so-called "context dimension trees" (CTDs). CTDs have one root node under which one can find context-dimensions nodes. Context dimensions roughly correspond with classes and relations in an ontology. Under these nodes, one must have one or more context-dimension-value nodes that represent specific instances or values for a context dimension. A context dimension value can, themselves, have context dimensions nodes. This allows one to represent complex paths such as: <CD,Medium> →<CDV.Paper>→<CD,Size>→<CD,A4>. Nodes in the tree are mapped onto partial views, which are then used to compute views based on a user's context. The work in [34] aimed to address context management by separating the systems from the context model and proposed the use for a more expressive ontology language—a combination of OWL and rules. While [33] and [34] recognized the importance of modelling context separately, and proposed formalisms to do so, their contributions focused on the (use of) their formalism and not on the knowledge engineering activities required to formalize context.

In [35], the authors proposed a context knowledge-base for integrating data. More specifically, however, the context knowledgebase contains ontology alignments that are used to create SPARQL CONSTRUCT queries to transform triples in one ontology to another. They thus only focused on the transformation of data, and the alignment had no link with the rest of a data-driven project's context. Finally, [36] introduced various types of context (user, application, environment, and "other") for rewriting queries that users might discover more information. In [37], the paper sheds more light on how context is modeled and stored (using relational tables that are tailored for a particular domain –in this case, "software engineering"). We argue that these are anecdotal examples in which the 'context' captured is specific to one task (e.g., query optimization or query-rewriting).

2.4 Discussion

Context-aware computing (in general), 'Context' in the SotA is mostly concerned with these 'crude' and simple sensors, and these are too simplistic to capture the context of data integration in data-driven projects. Outside initiatives concerned with context, there are efforts that have (tried to) formalize(d) aspects relevant to it.

We observe that context-aware platforms built for data analysis are made for specific domains (e.g., healthcare, and smart cities) and siloed into middleware. While this suggests that context-awareness exists, it is only in a limited, non-transferable scope. Those siloes propose a controlled environment for data integration, thus reducing the challenges. The context is more involved in data integration activities of data-driven projects and will require various context-aware techniques to drive decision-making processes in those. We thus need a more generic approach for the activities of a data integration process, which are not known beforehand. In conclusion, we have not yet truly explored context-awareness in data integration. There have been

[4] https://protege.stanford.edu/, last accessed Aug 6, 2019

anecdotal attempts for specific tasks (e.g., query optimization), and related work in IoT benefited from environments with limited types of data and specific applications/purposes.

3 What is Context in Data Integration?

Data integration is stated to be concerned with "combining data residing at different sources, and providing the user with a unified view of these data" [3]. We refine the definition as "*selecting, preprocessing and transforming* data from different data sources [in order] to create a *unified version* of that data for *data processing*" as to emphasize the various types of decision points of "combining" and leaving both the result of the integration process and whether consumer –human or software open.

- The **data-driven project and its characteristics**, which include the requirements (technical and non-technical), objectives, budgets, costs, timelines, stakeholders, and policies and regulations. This also includes the history of decisions that have been made.
- The **disciplines involved in data-driven projects**. The community recognizes the transdisciplinary nature of data-driven projects [38] [39]. Disciplines include CS, statistics, ethics and law, and domain expertise, among others.
- The **practitioner's (prior) knowledge, experience, and background** while doing data integration. In [2], the authors provide concrete, anecdotal evidence of the challenges that practitioners face and the importance of practitioners following their intuition or recalling past experiences.

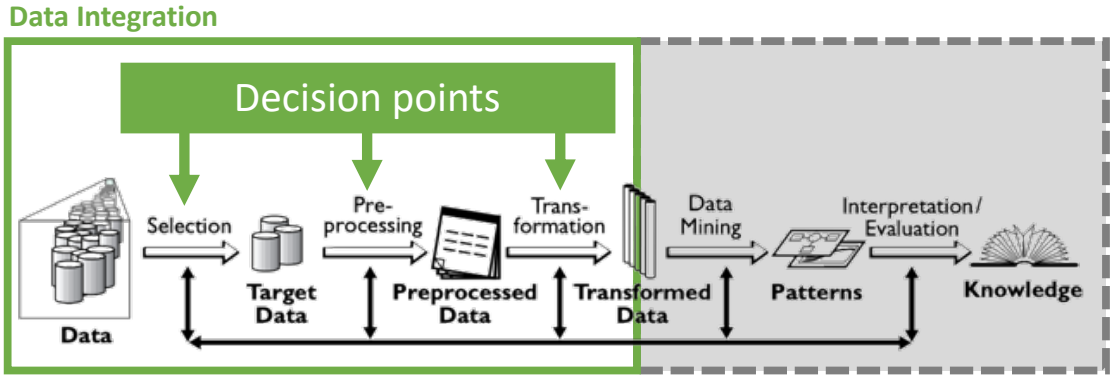

Fig. 1. Informed decisions are made during the selection, preprocessing, and transformation activities of data-integration, which are part of broader data-driven projects. (source original:[40])

In context-aware data integration, one has to consider: 1) conceptualizing and formalizing the model, and 2) propose agents that use that model for facilitating or automating specific data integration processes. The first is considered an ontology engineering activity. The latter is a development activity, as we have to build agents –in some way, shape, or form—that use the model to achieve certain tasks. Granted, depending on the form, the second can be regarded as a knowledge engineering activity as well. Ideally, however, the context-aware techniques should avail of techniques built on semantic technologies to make them as declarative as possible. We illustrate our approach in the next section.

4 Case-study: Consent-aware Dataset Generation

The problem we want to tackle is to generate datasets that took into account the informed consent of users. By ensuring that such datasets are compliant, we can reduce the overhead of post hoc compliance analysis. Data processing, in general, is increasingly the subject of various regulations, such as the General Data Protection Regulation[5] (GDPR). In GDPR, data subjects give their explicit informed consent for their data to be used for specific data processing purposes. Data subjects also have the right to revoke their consent at any given time. One of the challenges here is to identify what is personal or sensitive data, and which (parts) of an organization's data sources contains said data. An example of a purpose is to analyze previous purchases to send a list of recommendations (i.e., a recommender system).

Rather than relying on a team of practitioners to ensure that the data integration steps necessary are compliant, we can adopt semantic technologies to facilitate the context and the generation for datasets for the recommender system. In terms of the project, the context includes the data processing purpose (i.e., the purpose of the service) with the goal being to increase sales, GDPR as the regulation (and the organization's compliance

[5] https://eur-lex.europa.eu/eli/reg/2016/679/oj

with GDPR), and the various data sources used. In terms of disciplines, the context includes data governance and law. Data governance is concerned with data management, roles, and accountability. Important for data governance is knowing where to find information, how it is classified or regarded, and how it should be used. The last touches upon Law and how to conduct activities lawfully. While GDPR is an external "entity" affecting the data-driven project, knowledge of how to interpret, use, and implement that regulation is the discipline. Finally, in terms of the practitioner, the context includes the data engineer or data scientist who needs to compile a dataset for their recommender system.

To solve this problem, we proposed the architecture shown in **Fig. 2**. Parts of the architecture is based on algorithms published in [41] and [42]. The former is concerned with generating data integration mappings from annotated schemas, and the latter is concerned with "filtering" the resulting data for ensuring compliance. We use green to show the various processes and artifacts for the former, and pink for the latter.

No matter the structure of the resulting dataset (XML, CSV, or a graph), the resulting dataset has a schema. The data governance challenge here is to relate these schemas with references to an organization's data. Those annotations are used to generate, on the fly, data integration mappings that fetch information from those databases to populate a schema. The generated dataset needs to consider informed consent that the organization has gathered. This means annotating the schemas with their data processing purposes and formalizing parts of GDPR to achieve this.

GDPR is thus the context and meant we had to conceptualize and formalize (i.e., create an ontology) for the following concepts: "data processing purpose", the "policy" which lists these purposes, and "consent" instances. All these concepts have been described in our consent ontology [42]. The schemas are, in addition to the references to existing sources, annotated with instances of policies and purposes, and instances of the concept Consent contain information on the policy-purpose pairs users have consented to (or withdrawn). The resulting consent information base is then used to filter the generated datasets with the goal to exclude information of those who have not given their consent or whose consent is expired. All processes furthermore generate provenance information, which facilitates the transparency and reproducibility of the pipeline. While we omit the Web Ontology Language (OWL) formalization of the ontology, the important concepts and relations are shown in **Fig. 3**

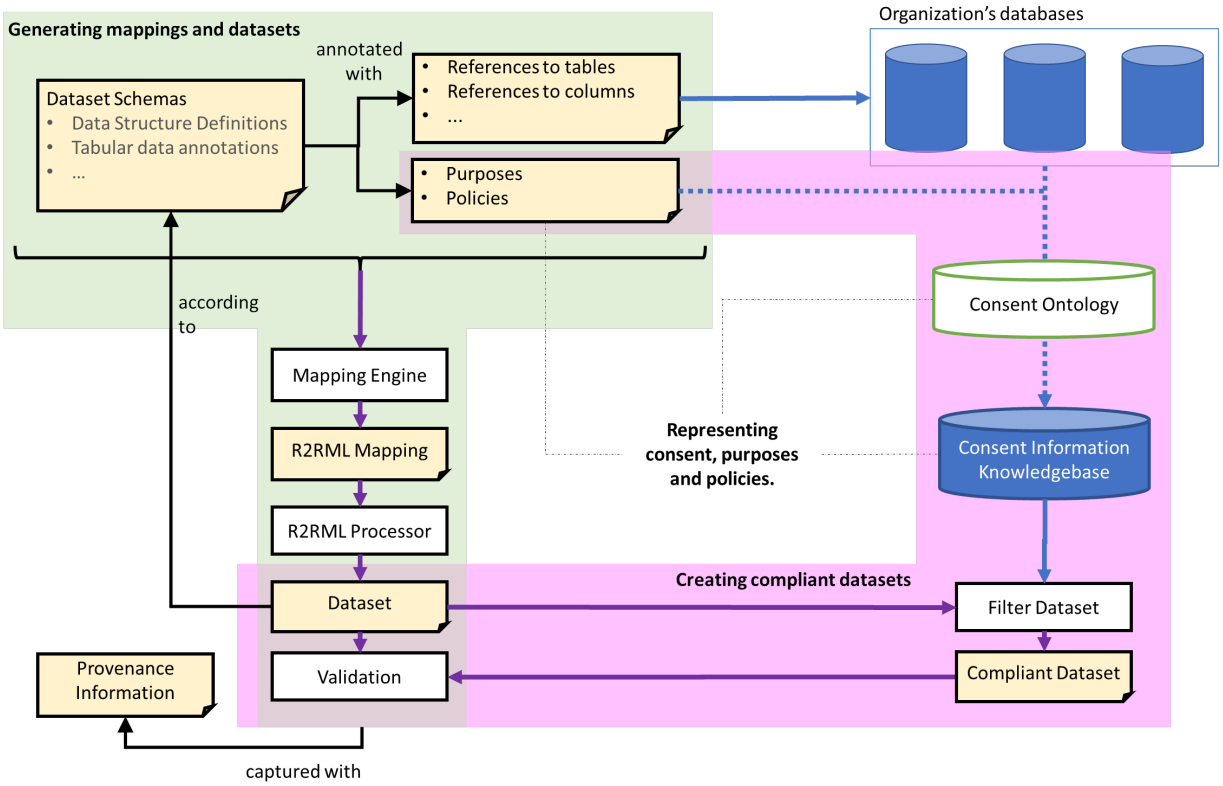

Fig. 2. The various steps involved in generating a policy-compliant dataset taking into account the informed consent of users.

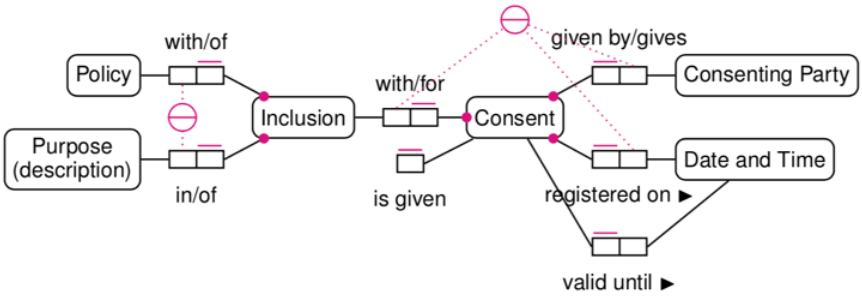

Fig. 3. The concepts and relations in our consent ontology. The concept inclusion captures the ternary relation between Policy, Purpose, and Consent. We considered this to be necessary as an organization may be the same data purposes for different services.

The "implementation" of the context-aware technique for dataset generation has been entirely written in SPARQL. [41] lists the various SPARQL queries for generating executable mappings, and [42] the queries for filtering for compliance. While bespoke code could have been written to build these agents, we believe that the declarative nature of these queries improves transparency for future audits. The listing below illustrates how we use the consent knowledge base to obtain the latest consent information for each user. That information is subsequently used to find all users that have given consent for a particular purpose that has not yet expired. While we cannot provide all queries due to the limited space, we refer the reader to [42].

```
1.  DESCRIBE ?consent WHERE {
2.    ?consent ont:forInclusion ?inclusion .
3.    { # GET THE LATEST INCLUSION FOR A POLICY
4.      SELECT ?inclusion WHERE {
5.        ?inclusion ont:ofPurpose <.../purpose> .
6.        ?inclusion ont:ofPolicy <.../policy> .
7.        <.../policy> dcterms:created ?dt .
8.      } ORDER BY DESC(?dt) LIMIT 1 }
9.    ?consent ont:givenBy ?user .
10.   ?consent ont:registeredOn ?datetime .
11.   # GET THE LATEST CONSENT FOR EACH USER BY
    FILTERING
12.   # THOSE THAT HAVE BEEN SUCCEEDED BY ANOTHER
    CONSENT
13.   FILTER NOT EXISTS {
14.     [ ont:forInclusion ?inclusion ;
15.       ont:givenBy ?user ;
16.       ont:registeredOn ?datetime2 ]
17.     FILTER(?datetime2 > ?datetime) } }
```

Listing 1: Retrieving consent information for a purpose and policy. All consent information is returned by the DESCRIBE query. The query first looks for the latest inclusion of a service. The second part of the query seeks the latest consent information for each user. For brevity, we omitted prefixes, and use <.../purpose> and <.../policy> for IRIs of a purpose and a policy

In this section, we presented a case study on generating datasets that are compliant with the explicit and informed consent given by users. While this section did not contain many technical details, those details have been published and demonstrated in [41] and [42]. The goal of this section was to illustrate how we had to formalize a particular aspect of a data-driven project's context (GDPR and informed consent) to automatically *select and transform* the data for a particular purpose. The various SPARQL queries, executed in a specific sequence, thus constitutes our GDPR-aware dataset generation.

5 Conclusions

In this paper, we presented the problem of context-aware data integration. Context-awareness in Computer Science is all too often focused on simple sensors and approaches context as "things" that can be sensed. Related work on context-aware data integration is sparse and often in specific and siloed environments such as IoT and Smart Homes. The SotA on context-awareness for data integration in data-driven projects is, in our opinion, not sufficient. Achieving this, however, would open opportunities for facilitating data integration practices in organizations.

To achieve context-aware data integration, one has to formalize context into ontologies and build agents that use these ontologies to make them context-aware. Ideally, those agents are –to the extent possible—written in terms of semantic technologies. We illustrated this approach with a preliminary study on generating GDPR-compliant datasets. The case study uses algorithms published and demonstrated in [41]. In this paper, we integrated both into the narrative for context-aware data integration.

ACKNOWLEDGMENTS

This research was conducted with the financial support of Science Foundation Ireland under Grant Agreement No. #13/RC/2106 at the ADAPT SFI Research Centre at Trinity College Dublin. The ADAPT SFI Centre for Digital Media Technology is funded by Science Foundation Ireland through the SFI Research Centres Programme and is co-funded under the European Regional Development Fund (ERDF) through Grant #13/RC/2106.

REFERENCES

[1] R. Wirth and J. Hipp, "CRISP-DM: Towards a Standard Process Model for Data Mining," in *Proceedings of the 4th International Conference on the Practical Application of Knowledge Discovery and Data Mining*, 2000.

[2] M. Muller *et al.*, "How data science workers work with data," in *Conference on Human Factors in Computing Systems - Proceedings*, 2019, pp. 86–94.

[3] M. Lenzerini, "Data integration," in *Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems - PODS '02*, 2002, p. 233.

[4] P. Shvaiko and J. Euzenat, "Ontology Matching: State of the Art and Future Challenges," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 1, pp. 158–176, Jan. 2013.

[5] J. Volz, C. Bizer, M. Gaedke, and G. Kobilarov, "Silk - A Link Discovery Framework for the Web of Data," in *Proceedings of the WWW2009 Workshop on Linked Data on the Web, Madrid, Spain, April 20, 2009.*, 2009.

[6] R. Isele and C. Bizer, "Learning expressive linkage rules using genetic programming," *Proc. VLDB Endow.*, vol. 5, no. 11, pp. 1638–1649, Jul. 2012.

[7] M. A. Sherif, A. C. Ngonga Ngomo, and J. Lehmann, "WOMBAT – A generalization approach for automatic link discovery," in *European Semantic Web Conference ESWC 2017: The Semantic Web*, 2017, pp. 103–119.

[8] M.-E. Vidal and S. Jozashoori, "Semantic Data Integration Techniques for Transforming Big Biomedical Data into Actionable Knowledge," 2019, pp. 563–566.

[9] A. K. Dey, G. D. Abowd, and D. Salber, "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications," *Human–Computer Interact.*, vol. 16, no. 2–4, pp. 97–166, Dec. 2001.

[10] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context Aware Computing for The Internet of Things: A Survey," *IEEE Commun. Surv. Tutorials*, vol. 16, no. 1, pp. 414–454, 2014.

[11] M. Poveda-Villalón, M. C. Suárez-Figueroa, R. García-Castro, and A. Gómez-Pérez, "A context ontology for mobile environments," in *Proceedings of the Second Workshop on Context, Information and Ontologies, CIAO@EKAW 2010, Lisbon, Portugal, October 11, 2010*, 2010.

[12] C. Bettini, O. Brdiczka, K. Henricksen, A. Ranganathan, and D. Riboni, "A survey of context modelling and reasoning techniques," *Pervasive Mob. Comput.*, vol. 6, no. 2, pp. 161–180, Apr. 2010.

[13] A. Zimmermann, M. Specht, and A. Lorenz, "Personalization and Context Management," *User Model. User-adapt. Interact.*, vol. 15, no. 3–4, pp. 275–302, Aug. 2005.

[14] G. Adomavicius and A. Tuzhilin, "Context-aware recommender systems," in *Recommender Systems Handbook, Second Edition*, 2015, pp. 217–253.

[15] H. Chen, T. Finin, and A. Joshi, "An ontology for context-aware pervasive computing environments," *Knowl. Eng. Rev.*, vol. 18, no. 3, pp. 197–207, Sep. 2003.

[16] Xiao Hang Wang, Da Qing Zhang, Tao Gu, and Hung Keng Pung, "Ontology based context modeling and reasoning using OWL," in *IEEE Annual Conference on Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second*, 2004, pp. 18–22.

[17] A. Badii, M. Crouch, and C. Lallah, "A Context-Awareness Framework for Intelligent Networked Embedded Systems," in *2010 Third International Conference on Advances in Human-Oriented and Personalized Mechanisms, Technologies and Services*, 2010, pp. 105–110.

[18] J. McCarthy, "Notes on formalizing context," in *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (I)*, 1993, pp. 555–560.

[19] S. Klarman and V. Gutiérrez-Basulto, "Description logics of context," *Journal of Logic and Computation*, vol. 26, no. 3. Oxford University Press, pp. 817–854, 06-Jun-2016.

[20] J. Attard and R. Brennan, "A Semantic Data Value Vocabulary Supporting Data Value Assessment and Measurement Integration," in *Proceedings of the 20th International Conference on Enterprise Information Systems - Volume 2: ICEIS*, 2018, pp. 133–144.

[21] J. Debattista, Sö. Auer, and C. Lange, "Luzzu—A Methodology and Framework for Linked Data Quality Assessment," *J. Data Inf. Qual.*, vol. 8, no. 1, pp. 1–32, Oct. 2016.

[22] F. Maali and J. Erickson, *Data Catalog Vocabulary (DCAT).* 2014.

[23] K. Alexander, R. Cyganiak, M. Hausenblas, and Z. Jun, "Describing Linked Datasets with the VoID Vocabulary," *W3C Interes. Gr. Note 03 March 2011*, 2011.

[24] J. Ison *et al.*, "EDAM: An ontology of bioinformatics operations, types of data and identifiers, topics and formats," *Bioinformatics*, vol. 29, no. 10, pp. 1325–1332, 2013.

[25] D. Garijo and Y. Gil, "The P-PLAN Ontology," 2014. [Online]. Available: http://vocab.linkeddata.es/p-plan/.

[26] A. Sebastian, N. F. Noy, T. Tudorache, and M. A. Musen, "A generic ontology for collaborative ontology-development workflows," in *International Conference on Knowledge Engineering and Knowledge Management*, 2008, pp. 318–328.

[27] A. Witold, A. Filipowska, M. Kaczmarek, and T. Kaczmarek, "Semantically Enhanced Business Process Modeling Notation," in *Semantic Technologies for Business and Information Systems Engineering: Concepts and Applications*, S. Smolnik, F. Teuteberg, and O. Thomas, Eds. Hershey, PA: IGI Global, 2012, pp. 259–275.

[28] M.-Á. Sicilia, E. García-Barriocanal, S. Sánchez-Alonso, M. Mora-Cantallops, and J.-J. Cuadrado, "Ontologies for Data Science: On Its Application to Data Pipelines," in *Metadata and Semantic Research*, 2019, pp. 169–180.

[29] M. Proctor and Mark, "Drools: A Rule Engine for Complex Event Processing," in *Proceedings of the 4th international conference on Applications of Graph Transformations with Industrial Relevance*, Springer-Verlag, 2012, pp. 2–2.

[30] K. Fraser, B. Yousuf, and O. Conlan, "A context-aware, info-bead and fuzzy inference approach to notification management," in *2016 IEEE 7th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, 2016, pp. 1–7.

[31] C. Dobre and F. Xhafa, "Intelligent services for Big data science," *Futur. Gener. Comput. Syst.*, vol. 37, pp. 267–281, 2014.

[32] Z. G. Ives *et al.*, "An adaptive query execution system for data integration," in *Proceedings of the 1999 ACM SIGMOD international conference on Management of data - SIGMOD '99*, 1999, vol. 28, no. 2, pp. 299–310.

[33] C. Bolchini, E. Quintarelli, and L. Tanca, "CARVE: Context-aware automatic view definition over relational databases," *Inf. Syst.*, vol. 38, no. 1, pp. 45–67, Mar. 2013.

[34] V. Vieira, P. Brezillon, A. C. Salgado, and P. Tedesco, "Towards a Generic Contextual Elements Model to Support Context Management," in *Proceedings of the 4th International Workshop on Modeling and Reasoning in Context (MRC 2007) with Special Session on the Role of Contextualization in Human Tasks (CHUT)*, 2007, pp. 49–60.

[35] D. Braines, Y. Kalfoglou, P. Smart, N. Shadbolt, and J. Bao, "A data-intensive lightweight semantic wrapper approach to aid information integration.," in *Proceedings of the Fourth International Workshop on Contexts and Ontologies (C&O) Collocated with the 18th European Conference on Artificial Intelligence (ECAI-2008)*, 2008.

[36] G. Karabatis, "Using Context in Semantic Data Integration," *Int. J. Interoperability Bus. Inf. Syst.*, vol. 3, pp. 9–21, 2006.

[37] G. Karabatis *et al.*, "Using semantic networks and context in search for relevant software engineering artifacts," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 14, no. 74–104, 2009.

[38] L. Cao, "Data science: Challenges and directions," *Commun. ACM*, vol. 60, no. 8, pp. 59–68, Aug. 2017.

[39] C. Bizer, P. Boncz, M. L. Brodie, and O. Erling, "The meaningful use of big data," *ACM SIGMOD Rec.*, vol. 40, no. 4, p. 56, Jan. 2012.

[40] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "The KDD process for extracting useful knowledge from volumes of data," *Commun. ACM*, vol. 39, no. 11, pp. 27–34, Nov. 1996.

[41] C. Debruyne, D. Lewis, and D. O'Sullivan, "Generating Executable Mappings from RDF Data Cube Data Structure Definitions," in *On the Move to Meaningful Internet Systems: OTM 2018 Conferences - Confederated International Conferences: CoopIS, CT&C, and ODBASE 2018, Valletta, Malta, October 22-26, 2018. Proceedings*, 2018, vol. 11230, pp. 333–350.

[42] C. Debruyne, H. J. J. Pandit, D. Lewis, and D. O'Sullivan, "Towards Generating Policy-Compliant Datasets," in *13th IEEE International Conference on Semantic Computing, ICSC 2019, Newport Beach, CA, USA, January 30 - February 1, 2019*, 2019, pp. 199–203.