Towards Certified Distributed Query Processing

Philipp D. Rohde[1,2,3], Maria-Esther Vidal[1,2,3]

[1] TIB Leibniz Information Centre for Science and Technology, Hannover, Germany
[2] L3S Research Center, Hannover, Germany
[3] Leibniz University of Hannover, Hannover, Germany

Abstract
In recent years, knowledge graphs (KGs) have gained more and more importance. As a consequence of that, the number of publicly accessible KGs is increasing. Due to their adoption in many areas, KGs are used in numerous different applications. However, these knowledge graph applications are not developed by the data owners and they might collect data from several linked KGs. It is therefore essential that systems accessing KGs are certified, i.e., each component is certified for a specific use by an entity or agency. In addition, a trace of the performed operations and used data is needed in order to verify that all requirements were met, e.g., some data cannot be transferred from the source to any other component due to privacy restrictions. This work describes the vision of certified distributed querying in the context of an analytics platform. Challenges for such systems are identified and discussed.

Keywords
Certification, Privacy, Access Control, Distributed Query

1. Introduction

The *Resource Description Framework* (RDF) [1] is the W3C standard for publishing data on the Web. Such an RDF data source is commonly referred to as *knowledge graph* (KG). The *Linked Open Data Cloud*[1] represents only a small portion of the publicly accessible KGs that are linked to each other. But despite that fact, the number of datasets in the Linked Open Data Cloud is increasing. KGs are used more and more in the industry, e.g., by large IT companies [2], or for domain-specific tasks, e.g., in biomedicine [3]. With KGs being used in more and more applications, the need for certified approaches rises. Since KGs may contain private data that must be protected against unauthorized access, access control is essential to certified approaches. However, authorized access does not guarantee that the data is used in terms of the owner. Hence, traceability is another important dimension of accessing KGs in order to verify the correct use of the data. In specific scenarios, privacy and traceability may be preserved relatively easy, e.g., in a system where only the own KG is queried, and no external users have access. This, however, does not hold for somewhat uncontrolled – and potentially decentralized – applications on the Web. This paper discusses open challenges in these scenarios.

Proceedings of the ESWC 2023 Workshops and Tutorials
philipp.rohde@tib.eu (P. D. Rohde); maria.vidal@tib.eu (M. Vidal)
0000-0002-9835-4354 (P. D. Rohde); 0000-0003-1160-8727 (M. Vidal)

[1] https://lod-cloud.net/

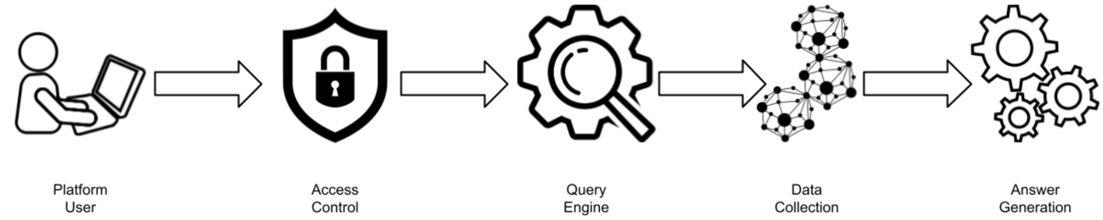

Figure 1: Certified Distributed Querying Scenario

1.1. Certified Distributed Query System

In the context of this paper, a system is called a *distributed query system* if the system answers queries from several data sources. These data sources might be heterogeneous and hosted in different physical locations. An alternative term is *federated querying*. It is assumed that a distributed query system is empowered with the means to control the access to the data sources available to the system. A *distributed* (or federated) query engine is part of the system and responsible for query decomposition, source selection, and planning the execution of the query. Depending on the user request, the necessary data is collected from several sources and combined to form the final answer within the system. While these two steps are usually performed by a query engine, they are explicitly mentioned as the *data collection* and *answer generation* component, respectively. The data sources accessible via the system are also considered to be a component of said system. A distributed query system is called *certified* if it meets the following requirements. *(i)* Each component of the system is certified by a third party, i.e., another entity or agency. These certifications may impose restrictions, e.g., a query engine might only be certified to work on medical data. *(ii)* Each component has to document the performed actions in a way that any third party is able to trace what the component did and verify that the component is working correctly, i.e., the result is sound, and is meeting all further restrictions, e.g., no private data was presented to the user. Throughout this paper, systems of this kind are called *certified distributed query system*.

1.2. Certified Distributed Querying Scenario

Given an online platform for analytics as presented in Figure 1 which retrieves data from several KGs through a SPARQL [4] query engine; the *SPARQL Protocol and Query Language* is the W3C recommendation language to query RDF data. A user of the platform submits the request to visualize the life expectancy and population of Germany for the past ten years. This data is available in the KGs accessible from the analytics platform and requires requesting data from at least two different sources. However, the life expectancy data can only be accessed by people from the same country, i.e., Germany in this example. Additionally, the population data must not be changed, e.g., additions and subtractions are prohibited. Currently, there are no mechanisms that ensure access is only granted to authorized users and trace what happened with the data. In the context of the presented scenario this means that there is no guarantee that the population data is not manipulated. The following section identifies challenges in current systems similar to the one presented in this scenario in order to become a certified distributed query system.

2. Challenges in Certified Distributed Querying

Creating a certified system for distributed querying brings challenges in all the components of such a system. Not only the certification of the components is challenging but also the verification. As described above, for the verification, each component needs to document what it is doing so that another entity or agency can trace back all the performed steps and verify that all restrictions have been met and the result is correct.

2.1. Challenge: Access Control

In terms of access control in a certified distributed query system, a big challenge is the validation of access control policies. The *Open Digital Rights Language* (ODRL) [5] is designed for licensing but also used for defining access policies. Hence, the evaluation of access policies in ODRL is not defined and external data cannot be used. Access control policies can be seen as integrity constraints which is why the *Shapes Constraint Language* (SHACL) [6] seems to be a more natural choice since the semantics of its evaluation is well-defined [7]. However, SHACL requires the data to be in RDF. So in order to use SHACL for access control policies, a knowledge graph with the data to be considered needs to be created. Usually, the relevant data is small, and, therefore, the knowledge graph can be created efficiently *on the fly* [8]. Complementary, some systems like *Solid Pods* [9] implement a per-user access control mechanism directly at the access layer of the RDF graph. Extensions for enabling the use of access control policies are available for some of these systems but they implement different policy languages. Hence, the access control policies need to be re-implemented when used in another system. Since this is time-consuming, it might prevent data owners from keeping their systems up-to-date. While there is some work aiming to solve access control over knowledge graphs, e.g., the fine-grained access control model by Valzelli et al. [10], there is no standard for doing so yet.

2.2. Challenge: Distributed Query Processing

Distributed query processing is an active research area. While the main focus is on performance improvements through new algorithms for source selection, query planing, and optimizing the operators, privacy is considered more and more in recent work [11]. Privacy needs to be considered during all steps of query processing. The source selection has to consider the privacy restrictions of the attributes of the data source. During the query planning phase, the restrictions must be taken into account since certain conditions might impose the use of a specific operator or prohibit the use of it. This might also lead to rearranging the order of the operators due to the restricted access. Operations over attributes that cannot be transferred need to be performed at the source level. This imposes new challenges to query optimization since the performance of an operator implemented at source level is unknown to the query engine. Further, some data can be transferred to the query engine, but it cannot be sent to other sources. In this case, a nested join cannot be used. Privacy in distributed query engines needs to be studied further and existing engines have to implement privacy policies. This opens the research area of *privacy-aware distributed query optimization.*

2.3. Challenge: Data Collection

The data collection in a certified distributed querying system needs to document which data in which data source has been accessed and transferred. While the data collection could be documented by storing the query plan, there is no guarantee that the query engine actually executed that plan. This needs to be certified and verified in order to ensure the correct use of the data. At the point of writing, there is no standard for representing the query plan of a distributed SPARQL query. While the previously discussed issue deals with the correct use of the data, in a certified distributed querying system, also the data sources that are accessed need to be certified so that the retrieved data can be trusted. This is crucial especially when collecting external data for the evaluation of an access control policy since the use of not certified and manipulated data might lead to granted access even though the access should have been denied. The distributed query engine should only consider certified and verified data sources.

2.4. Challenge: Answer Generation

The component that generates the final query result needs to ensure that the answer is correct. Hence, the answer generation component needs to be certified stating that indeed only correct results are generated. Additionally, privacy restrictions have to be ensured since it might be the case that specific data is allowed to be transferred to the answer generation component but cannot be presented to the user. Adding metadata to the query result as a trace of how the answer was generated could help in the verification of the correctness of the result. Such a trace would have to include the information about the source data that was used for generating the answer as well as the operators that produced the output. However, since the answer generation might include private data, not all triples contributing to the answer can be included in the trace. So far, privacy-preserving verifiable query results are an open challenge.

2.5. Ethical & Legal Challenges

While the previous sections describe technical challenges for certified distributed query systems, there are also ethical and legal aspects that need to be considered. Obviously, the systems need to adhere to the GDPR. While technical solutions for preserving privacy might be in place, it is not clear how the compliance with the GDPR can be enforced. Another open question regarding the GDPR is about who is responsible in the case of a violation of the GDPR; is it the entity who certified the violating component, the entity who deployed the component, or maybe the entity who developed the component. The certification process also must be regulated in order to prevent entities from issuing wrong certifications. Assume there are two entities A and B that are tightly coupled. Entity A developed an answer generation component that respects the privacy in the context of the generated answer but stores all the private data in a database owned by entity A. Since entity B profits from the malicious acts of entity A, entity B issues a certification for the answer generation component stating that is respects all regulations. In case like this, there needs to be a regulation so that entities A and B can be sued and sentenced. The presented ethical and legal challenges are only two examples to raise awareness for this non-technical aspect of certified distributed query systems, e.g., usage control and certification.

3. Conclusion

This paper describes the vision of certified distributed querying and analyses the challenges of such systems as access control, operators for distributed query processing, data collection, and answer generation. Ethical and legal challenges are also discussed briefly using usage control and certification as two examples for these aspects of certified systems. A well-defined standard for expressing and validating access control policies is needed in terms of access control. Distributed query engines need to implement privacy policies, and privacy-aware distributed query optimization has to deal with the impact of privacy on the query execution performance. Data collection presents the challenge of documenting the data access and use as well as certifying and verifying the data sources in order to eliminate the risk of retrieving manipulated data. When it comes to query answer generation, privacy-preserving metadata about how the query result was generated is mandatory to achieve the vision.

References

[1] F. Manola, E. Miller, RDF Primer, W3C Recommendation, 2004. URL: https://www.w3.org/TR/2004/REC-rdf-primer-20040210/.

[2] N. F. Noy, Y. Gao, A. Jain, A. Narayanan, A. Patterson, J. Taylor, Industry-scale knowledge graphs: lessons and challenges, Commun. ACM 62 (2019) 36–43. doi:10.1145/3331166.

[3] D. N. Nicholson, C. S. Greene, Constructing knowledge graphs and their biomedical applications, Comput Struct Biotechnol J. 18 (2020) 1414–1428. doi:10.1016/j.csbj.2020.05.017.

[4] E. Prud'hommeaux, A. Seaborne, SPARQL Query Language for RDF, W3C Recommendation, 2008. URL: https://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/.

[5] R. Iannella, S. Villata, ODRL Information Model 2.2, W3C Recommendation, 2018. URL: https://www.w3.org/TR/2018/REC-odrl-model-20180215/.

[6] H. Knublauch, D. Kontokostas, Shapes Constraint Language (SHACL), W3C Recommendation, 2017. URL: https://www.w3.org/TR/2017/REC-shacl-20170720/.

[7] J. Corman, J. L. Reutter, O. Savković, Semantics and validation of recursive shacl, in: The Semantic Web – ISWC 2018, 2018, pp. 318–336. doi:10.1007/978-3-030-00671-6_19.

[8] E. Iglesias, S. Jozashoori, M.-E. Vidal, Scaling up knowledge graph creation to large and heterogeneous data sources, J. Web Semant. 75 (2023). doi:10.1016/j.websem.2022.100755.

[9] S. Capadisli, T. Berners-Lee, R. Verborgh, K. Kjernsmo, Solid Protocol, Solid Community Group Submission, 2021. URL: https://solidproject.org/TR/2021/protocol-20211217.

[10] M. Valzelli, A. Maurino, M. Palmonari, A Fine-grained Access Control Model for Knowledge Graphs, in: Proceedings of the 17th International Joint Conference on e-Business and Telecommunications (ICETE 2020), 2020, pp. 595–601. doi:10.5220/0009833505950601.

[11] M. Goncalves, M.-E. Vidal, K. M. Endris, PURE: A Privacy Aware Rule-Based Framework over Knowledge Graphs, in: Database and Expert Systems Applications, 2019, pp. 205–214. doi:10.1007/978-3-030-27615-7_15.