Big Spatial Data Integration and Enrichment with Provenance Control

Paulo Pintor[1], José Moreira[1], Rogério Luís de Carvalho Costa[2]

[1]Department of Electronics, Telecommunications and Informatics (DETI)
University of Aveiro – Aveiro, Portugal

[2]CIIC – Polytechnic of Leiria
Leiria, Portugal

{paulopintor,jose.moreira}@ua.pt, rogerio.l.costa@ipleiria.pt

Level: Doctorate Degree (Doctorate in Computer Engineering)

Enrolment date: October 2020 - **Due date:** June 2024

Completed activities: All mandatory courses (09/20-07/21), Bibliographic review (01/21-01/22), Problem Statement (09/21-07/22), Thesis Proposal (01/22-07/22)

Ongoing activities: Thesis proposal defense (10/22), Case studies definition (09/22-01/23), Model formulation (09/22-01/23), Model implementation (01/23-12/24)

Future activities: Thesis writing (09/23-07/24) and defense (07/24), results publication (09/20-07/24)

Abstract. *In the last few years, an increasing number of devices generated vast amounts of data, commonly called Big Data. This phenomenon brought many opportunities - and challenges - in terms of knowledge discovery, as distributed and heterogeneous data may be combined and used to create high-quality models of events and phenomena. Although, the data integration and the transformations over it bring questions about integrity, quality and veracity. Our investigation aims to create a generic model to integrate the data allowing the data enrichment while maintaining provenance information.*

Resumo. *Nos últimos anos temos assistido a um aumento do uso de dispositivos que geram vastas quantidades de dados, que normalmente são conhecidos como Big Data. Este fenómeno trouxe muitas oportunidades e desafios em termos da descoberta de conhecimento, uma vez que a combinação de todos estes dados distribuídos e heterogéneos podem levar a criação de modelos de eventos e fenómenos de alta qualidade. Contudo, a integração destes dados e as transformações sobre os mesmos levantam questões de integridade, qualidade e veracidade. Esta investigação tem como objetivo a integração de dados, com foco em dados espaciais, utilizando um modelo genérico que permita a integração e o enriquecimentos dos dados, mantendo a informação sobre a proveniência dos mesmos.*

1. Introduction

In the last few years, we have witnessed a paradigm shift in the database field. We were used to a centralized environment that is becoming increasingly distributed and heteroge-

neous. Several factors have made data increasingly distributed, including the emergence of the Cloud, smart devices, NoSQL systems, and the Internet of Things (IoT).

Beyond these phenomena, we are generating a vast amount of data daily, commonly called Big data [Abadi et al. 2020, Wang et al. 2019]. Big data, jointly with the rise of open data and data science, attracted experts in several domains who have become interested in data processing, knowledge extraction, and results sharing. The data generated also has another particularity - much of the data is spatial. Spatial data brings more challenges to the integration and manipulation since it is data with specific features.

As stated in [Abadi et al. 2020], one of the main challenges in data science is integrating all the data from all the different data sources and dealing with data wrangling. Data wrangling is essential to ensure that after the data integration, there will be no problems with data duplication, inconsistency or discrepancy, and semantic or syntax problems. Another relevant aspect is providing data scientists and other business experts with access to data. They shouldn't need to understand languages and mechanisms of database systems nor be concerned about data integration and wrangling. They need access to the data to study it and extract knowledge. Thus, it is crucial to have a system that provides high-level transparent integration and information about the data sources and has data and management tools enabling users to deal with data and make their assumptions.

Hence, there is an increasing level of data distribution and the need to access different data sources. In this context, some questions have arisen regarding the data quality and the reliability of data sources. What is the data source used? Can we trust it? What transformations were made to data? Does the query result has the expected quality? In a distributed environment, these questions are essential to infer data quality, avoiding false results in the knowledge discovery.

This thesis intends to develop a model to integrate spatial data with data provenance and enrichment support, contributing to the discussion of data provenance in distributed systems.

2. Background

The concern about the provenance field has grown at the same time that we witness more data being generated and more data sources being used. Hence, World Wide Web Consortium (W3C) proposed a standard model and an ontology to describe provenance called PROV [Buneman and Tan 2018, Closa et al. 2017]. PROV aims to describe provenance in Workflow. Thus it has a structure containing three elements: Entity, Activity and Agent. These elements help to describe all the processes that the data have been through from the data source to the final dataset.

Data provenance is another type of provenance. It deals with provenance in the databases, more precisely in database queries. Data provenance has three main types: *why-*, *how-* and *where-provenance*. *Where-* is concerned with the origin of the individual values (instead of tuples) in the query result, *why-* explains the tuples involved, and finally, *how-provenance* intends to explain how to conjugate the tuples to obtain the result.

The research about *how-* and *why-provenance* proposes two different techniques to obtain each provenance type. *Why-provenance* is the set of tuples that contribute to a result, and the technique witness basis is used to collect all the tuples. Based on the

definition in [Buneman et al. 2001, Cheney et al. 2007], given a database *I*, a query *Q* over *I* and a tuple *t* in *Q(I)*, an instance of $I' \subseteq I$ is a witness for *t* if $t \in Q(I')$. This can be denoted as: $Why(Q, I, t) = \{I' \subseteq I | t \in Q(I')\}$.

For *how-provenance* it is used the semiring theory [Buneman et al. 2001, Buneman and Tan 2018, Senellart 2017]. A semiring is defined as $(K, 0, 1, \oplus, \otimes)$ where *K* is a set of data elements that will be annotated using the constants 0 and 1. Given a query *Q* if the tuple *t* contributes to the output result is annotated with 1, otherwise is annotated with 0. The binary operators $\oplus, \otimes$ are used as alternative $\oplus$ and as joint $\otimes$.

Where-provenance is different from the other two types because it is not tuple-based, and as stated in [Senellart 2017], it is not possible to derive it from semiring. [Senellart 2017] proposes to add annotations without algebra terms to create a bipartite graph that shows how the values are connected to build the *where-provenance* information. All these techniques assume the existence of an identifier in all tables or dataset's tuples involved. The identifier is called the provenance token.

The literature shows several solutions to deal with data provenance. ProvSQL [Senellart et al. 2018], Perm [Glavic and Alonso 2009], and GProM [Arab et al. 2018] are some more recently examples of these solutions that aim to give the user information about *where-*, *how-* and *why-provenance* and also solutions for probabilistic query evaluation. The three solutions have different approaches to dealing with the data provenance issue. Perm is an extension to PostgreSQL, and its approach is based on query rewriting. ProvSQL is a lightweight extension for PostgreSQL to support provenance computation and probabilistic query evaluation. It uses semiring theory to compute how-provenance and proposes an extension to semirings called m-semirings to support negation. ProvSQL also supports the capture of where-provenance. GProM is the only solution of these solutions that works with more than one DBMS. It is a middleware solution intended to manage provenance and annotations and supports Oracle, SQLite, and PostgreSQL.

Although data provenance is essential in distributed database environments to help infer the data origins and transformations, dealing with provenance in distributed databases is an open issue. Provenance in non-monotone queries is also an open topic.

3. Research goals and methodology

Designing systems abstracting the underlying complexity of distributed and heterogeneous databases, including spatial databases and GIS, is challenging because it is required to balance conflicting goals such as location transparency, semantic completeness and data provenance. So, we define the following research goals for this thesis:

- To develop a model that deals with data wrangling and data integration in terms of concepts and high-level abstractions, regardless of the type of data model or storage engine, and considering that the organization and the structure of raw data may change over time. Hence, users (e.g., data scientists) will see a high-level — integrated and enriched — representation of raw data, with location transparency;
- To assess the model's capacity to support data provenance and veracity, ensuring its ability to describe data origin, history and dependencies of the data. This is specially challenging because creating high-level abstractions of raw data sources makes it more difficult to describe the origin, the history and the dependencies of the data;

- To assess current query languages, query execution engines and optimizers, proposing required extensions or innovative features to achieve required querying capabilities over the proposed model;
- To enable the representation of big spatial data, evaluating its applicability and benefits in terms of knowledge discovery, including modelling real-world behaviors and forecasting, and maintaining provenance characteristics. Research results should culminate in developing methods and tools to represent distributed and heterogeneous data using high-level representations, letting decision-makers and data scientists focus their work on case studies rather than technology.

The use of provenance will assess data sources' quality, reliability and trustworthiness in terms of spatial data (which is different from non-spatial data), where the characteristics may vary over time. One of the main contributions of this thesis is to contribute to the data provenance field in distributed environments, since the work done in this field until now is focused on centralised ones.

The proposed approach has great potential to improve the quality of data integration and thus knowledge discovery, allowing for proper handling of conflicting and missing data and handling semantic, syntactic and representation differences, among others. Potential users include experts in environmental sciences and smart environment planners, such as smart cities and smart farms.

This thesis will contribute with a high-level generic representation model to access distributed databases, also exploring the existing query languages and/or query optimizers to enhance querying capabilities over the proposed model. The model will be validated with spatial data and (preferably) real-world data to evaluate the query and model capabilities. Throughout the thesis plan, we will also contribute with the dissemination and publication of the work done.

We would use the multiple-case-studies strategy to achieve the proposed goals defined in this research [Benbasat et al. 1987]. This strategy is based on the construction of theories and on the application of hypotheses in situations of multiple natures, leading to general results. Therefore, we will initially get deeper learning from state of art and select multiple case studies to create an in-depth description of the problem and the limitations of available solutions. These studies would also support hypothesis and model formulation. Then, identified hypotheses and the proposed general model will be applied and tested over other critical scenarios, confirming its application and generality.

4. Partial results

With the literature review, it was possible to understand that the existing solutions for data provenance are focused on specific databases. Thus after exploring the work done in the ProvSQL [Senellart et al. 2018], we study the possibility of creating a solution capable of dealing with the distributed feature of our problem.

In [Pintor et al. 2022b], we studied a solution for two types of provenance: *how-* and *why-provenance*. This solution is database independent and does not change the query engine. The solution has two modules. The first one does query re-writing, and the second module builds the provenance information.

In our solution, we assume the existence of a function similar to standard SQL

Listagg. This function allows to aggregate/concatenate string values from a group of rows and separate them by a delimiter. This function will be use in query re-write will model, to help add annotations to the query in order to obtain the tokens separated by different delimiters according to the different clauses used in the query. In the case of *unions* or *distinct*, we use a delimiter, for *group by* clauses, another delimiter, and for *joins* we add a new column. The second module contains a specialized algorithm to process the annotations and obtain the provenance information, i.e., and the *How-* and *Why-provenance*.

Table 1. Table orders1

destination	vehicle	token
Lisboa	Train	tk1
Lisboa	Truck	tk2
Porto	Truck	tk3
Aveiro	Truck	tk4

Table 2. Table orders2

destination	vehicle	token
Lisboa	Truck	tk5
Porto	Train	tk6
Porto	Airplane	tk7
Aveiro	Ship	tk8

To exemplify our work with some results, we have tables 1 (orders1) and 2 (orders2), which represent the order's destination and vehicles from a company. For that, we have the column destination, which represents where the order will be delivered, the column vehicle is the vehicle used to deliver it, and "provtoken" is the provenance token needed to apply the data provenance theories. This company has some orders stored in a PostgreSQL database (table 1) and other orders in a Cassandra database (table 2). The tables are horizontally partitioned.

The result we want to obtain is the *join* of the global table with itself by the vehicles and use *group by* clause over the destinations. With the horizontal partition, in order to obtain the global table, we need to perform a *union* over the tables. Applying our annotations' method, the *union* query is as follows:

> **SELECT destination, vehicle, listagg(provtoken, ';') WITHIN GROUP (ORDER BY sname) as prov FROM(** SELECT destination, vehicle, **provtoken** FROM postgresql.public.orders1 UNION SELECT destination, vehicle, **provtoken** FROM cassandra.orspace.orders2 **) GROUP BY dest, vehicle)**

The field "token" is the identifier we add to use in the *how-* and *why-provenance* theories. When we perform the *union* those tokens will influence and change the result. To avoid it, we use the function *listagg* and a clause *group by*. The delimiter used in a *union* is a ";". The final query is the following:

> SELECT s1.dest, **listagg(s1.prov, '—') WITHIN GROUP (ORDER BY s1.dest) as prov, listagg(s2.prov, '—') WITHIN GROUP (ORDER BY s1.dest) as prov** FROM (– GLOBAL UNION –) s1, (– GLOBAL UNION –) s2 WHERE s1.vehicles = s2.vehicles GROUP BY s1.destination

We added the two new columns because of the *join*, and since we have a *group by* clause, we also used the *listagg* to aggregate the tokens but now with the delimiter "|". The result obtained from the query is then processed by the algorithm to build the

destination	how
Aveiro	((p.orders1:tk4 ⊗ (p.orders1:tk2 ⊕ c.orders2:tk5)) ⊕ (p.orders1:tk4 ⊗ p.orders1:tk4) ⊕ (c.orders2:tk8 ⊗ c.orders2:tk8))
Lisboa	(((p.orders1:tk2 ⊕ c.orders2:tk5) ⊗ (p.orders1:tk2 ⊕ c.orders2:tk5)) ⊕ ((p.orders1:tk2 ⊕ c.orders2:tk5) ⊗ p.orders1:tk4) ⊕ (p.orders1:tk1 ⊗ (p.orders1:tk3 ⊕ c.orders2:tk6)) ⊕ (p.orders1:tk1 ⊗ p.orders1:tk1))
Porto	(((p.orders1:tk3 ⊕ c.orders2:tk6) ⊗ (p.orders1:tk3 ⊕ c.orders2:tk6)) ⊕ ((p.orders1:tk3 ⊕ c.orders2:tk6) ⊗ p.orders1:tk1) ⊕ (c.orders2:tk7 ⊗ c.orders2:tk7))

Table 3. The *how-provenance* query result

destination	why
Aveiro	{{p.orders1:tk4,p.orders1:tk2},{p.orders1:tk4,c.orders2:tk5} ,{p.orders1:tk4},{c.orders2:tk8}}
Lisboa	{{p.orders1:tk2},{p.orders1:tk2,c.orders2:tk5},{c.orders2:tk5} , {p.orders1:tk2,p.orders1:tk4},{c.orders2:tk5, p.orders1:tk4} ,{p.orders1:tk1,p.orders1:tk3},{p.orders1:tk1,c.orders2:tk6},{p.orders1:tk1}}
Porto	{{p.orders1:tk3}{p.orders1:tk3,c.orders2:tk6},{c.orders2:tk6} ,{p.orders1:tk3,p.orders1:tk1}, {c.orders2:tk6, p.orders1:tk1}, {p.orders1:tk7}}

Table 4. The *why-provenance* query result

provenance information. The algorithm will interpret the delimiters and columns and create the provenance information, as shown in tables 3 and 4.

Table 3 shows the *how-provenance* result and table 4 *why-provenance*. The result in both types has the token with more information. Since we are in a distributed environment, the user needs more information. The first letter is the database - "p" for Postgres and "c" for Cassandra - after the period is the table name, and after the colon is the token.

The *how-provenance* shows how to conjugate the tokens to obtain that tuple. For instance, we can use the "Aveiro" result and change the binary operator for words. We need to join "tk4" with "tk2" or join "tk4" with "tk5" or join "tk4" with itself or "tk8" with itself. The *why-provenance* can be seen as the distribution of the *how-provenance*, and the tokens with binary operator X will create a set. In this type of provenance, we have a set of sets, and by definition, a set does not allow repetitions. Thus if we have (tk4 ⊕ tk4), the set is {tk4}, and if we have two sets like {tk4, tk5} and {tk5, tk4}, one disappears because for *why-provenance* they are the same because we are talking about witnesses and they would be the same witnesses in a different order.

We also have developed a work involving spatial data and data provenance to understand if the techniques supported the specificities of this kind of data [Pintor et al. 2022a]. In this study we overview the data provenance and Spatial data subjects and demonstrate how the spatial functions impact data provenance

5. Conclusion

With the work conducted in our investigation with the bibliography review finished, it was possible to create a prototype to deal with data provenance in distributed databases and

study whether data provenance solutions work with spatial data and functions. The next phase is to study several use cases to find real-world distributed scenarios with spatial data and start working on the generic model definition. Another objective is to extend our distributed solution to work with other data provenance types like *Where*-provenance.

References

Abadi, D., Ailamaki, A., Andersen, D., Bailis, P., Balazinska, M., Bernstein, P., Boncz, P., Chaudhuri, S., Cheung, A., Doan, A. H., Dong, L., Franklin, M. J., Freire, J., Halevy, A., Hellerstein, J. M., Idreos, S., Kossmann, D., Kraska, T., Krishnamurthy, S., Markl, V., Melnik, S., Milo, T., Mohan, C., Neumann, T., Ooi, B. C., Ozcan, F., Patel, J., Pavlo, A., Popa, R., Ramakrishnan, R., Ré, C., Stonebraker, M., and Suciu, D. (2020). The Seattle Report on Database Research. *SIGMOD Record*, 48(4):44–53.

Arab, B. S., Feng, S., Glavic, B., Lee, S., Niu, X., and Zeng, Q. (2018). Gprom - A swiss army knife for your provenance needs. *IEEE Data Engineering Bulletin*, 41(1):51–62.

Benbasat, I., Goldstein, D. K., and Mead, M. (1987). The case research strategy in studies of information systems. *MIS Q.*, 11:369–386.

Buneman, P., Khanna, S., Tan, W.-C., and Chiew, W. (2001). Why and where: A characterization of data provenance. *Computer Science*, 1973:316–330.

Buneman, P. and Tan, W. C. (2018). Data provenance: What next? *SIGMOD Record*, 47(3):5–16.

Cheney, J., Chiticariu, L., and Tan, W. C. (2007). Provenance in databases: Why, how, and where. *Foundations and Trends in Databases*, 1:379–474.

Closa, G., Masó, J., Proß, B., and Pons, X. (2017). W3C PROV to describe provenance at the dataset, feature and attribute levels in a distributed environment. *Computers, Environment and Urban Systems*, 64(July):103–117.

Glavic, B. and Alonso, G. (2009). Perm: Processing provenance and data on the same data model through query rewriting. In *Proceedings of the International Conference on Data Engineering*, pages 174–185, Shanghai, China. IEEE.

Pintor, P., Costa, R., and Moreira, J. (2022a). Provenance in spatial queries. In *26th International Database Engineered Applications Symposium - IDEAS 2022 - To Appear*.

Pintor, P., Costa, R. L. d. C., and Moreira, J. (2022b). Why- and how-provenance in distributed environments. In Strauss, C., Cuzzocrea, A., Kotsis, G., Tjoa, A. M., and Khalil, I., editors, *Database and Expert Systems Applications*, pages 103–115, Cham. Springer International Publishing.

Senellart, P. (2017). Provenance and probabilities in relational databases: From theory to practice. *SIGMOD Record*, 46:5–15. 7, 5.

Senellart, P., Jachiet, L., Maniu, S., and Ramusat, Y. (2018). ProvSQL: Provenance and probability management in PostgreSQL. *Proceedings of the VLDB Endowment*, 11(12):2034–2037.

Wang, Y., Dos Reis, J. C., Borggren, K. M., Vaz Salles, M. A., Medeiros, C. B., and Zhou, Y. (2019). Modeling and building IoT data platforms with actor-oriented databases. *Advances in Database Technology - EDBT*, 2019-March(1):512–523.