DATA PROVENANCE AND TRUST

Stratis D Viglas

School of Informatics at the University of Edinburgh, 10 Crichton Street, EH8 9AB, Edinburgh, UK
Email: svglas@inf.ed.ac.uk

1 STATE OF THE ART

The Oxford Dictionary defines *provenance* as "the place of origin, or earliest known history of something." The term, when transferred to its digital counterpart, has morphed into a more general meaning. It is not only used to refer to the origin of a digital artefact but also to its changes over time. By changes in this context we may not only refer to its digital snapshots but also to the processes that caused and materialised the change. As an example, consider a database record r created at point in time t_0; an update u to that record at time t_1 causes it to have a value r'. In terms of provenance, we do not only want to record the snapshots (t_0, r) and (t_1, r') but also the transformation u that when applied to (t_0, r) results in (t_1, r'), that is $u(t_0, r) = (t_1, r')$.

Though the above is a rather mathematical example, provenance is important in a number of contexts and for various reasons. Let us try to instantiate the variables in the previous example to make the need for provenance more succinct. The database record might represent the results of a physical examination of a patient in a hospital, and the transformation might be the application of a certain treatment. In a scientific data management context, the record might be the values of input parameters, and the transformation might be an experiment executed for these parameter values. Provenance, digitally captured, can thus be used to answer questions about the before images and after effects of real-life situations. The most important questions focus on explanation, accountability, and repeatability. Explanation allows us to offer the reasons behind some digital artefact appearing. Accountability allows us to identify those responsible (be they individuals or processes) for the existence of a digital artefact. Repeatability gives us a way to ensure that a digital artefact is fully described, thereby giving us a way to recreate it if necessary.

The provenance of a digital artefact is in itself a difficult and multifaceted problem. Over the past few years, there has been a growing consensus that provenance can be thought of as three—not necessarily disjoint—questions: *why*, *how*, and *where*. *Why-provenance* is the digital recording of the reason why a digital artefact exists. Also termed *lineage*, it is just what the name suggests: why does a specific digital artefact exist? For example, consider the digital recording of the outputs of an experiment: why do these outputs appear? What inputs contributed to their appearance? Assuming the transformation, i.e., the experiment, is fixed and can be executed repeatedly a number of times, then these inputs are the reason *why* this output manifests. Whereas why-provenance addresses the static aspects of the existence of a digital artefact (i.e., what was used to create it), *how-provenance* addresses the more dynamic aspects and gives us a way to reconstruct the process by which the artefact was created. In the previous example of experimental data management, how-provenance captures the operation by which inputs are used and transformed into outputs. Note that this is more general than why-provenance; as such, it is also more difficult to formally model and characterise. Finally, *where-provenance* captures the same information as why-provenance but in the context of locations. Specifically, it is an attempt to model one of the most common ways of acquiring data: copying. For each piece of information, we not only want to record why it is there but also *where* it came from. Such a use-case is typical in large scientific databases where data is routinely copied between sources. For instance, UniProt is an annotated database of protein sequences and contains manually collected information from multiple contributing sources. By answering queries about the where-provenance of data, we can reason about the origin of data and its utility in a dataset. For instance, we might have conflicting information about parts of a certain digital artefact (e.g., different addresses for the same individual). We can then differentiate our workflow based on the different instances of where-provenance, or we can choose to disregard one instance as an incorrect one.

The discussion of where-provenance above hints to an equally important, problem: that of contradiction. In an ideal world, each digital artefact and each of its snapshots can be uniquely identified. But we might have conflicting information about, say, the way an object has been created. Then the question becomes one of *trust*: in the presence of conflicting explanations, can we identify one as better, for some definition of "better"? What should that definition be, and how can we apply it in a generalised setting? Note that we may not only be dealing with conflicting information but with conflicting explanations for the same transformation. For instance, to explain the observation of value y for some quantity at time t_0 and the observation of value $2y$ for the same quantity at time t_1, two different sources might offer two different potential transformations $f(x) = x+y$, or $f(x) = 2x$; both are correct. But which one should we trust? Moreover, can there be cases where placing trust in one authority gives way to using trust maliciously? What is clearly needed is some type of security and trust model that captures not only the transformation and the applier of the transformation but also controls access to data and allows only certain appliers to affect transformations. The situation is aggravated by the need to protect transformations in terms of the data they manipulate (i.e., data is sensitive and so is the transformation by association) or by the need to protect the transformation itself (i.e., the transformation is sensitive). There are numerous examples of such scenarios: personal data is the canonical example when it comes to data; intelligence gathering, uneasy as the idea might be to some, is the canonical example of sensitive transformation provenance.

Given the above discussion, provenance and trust are not to be separated. Not only do we need ways of capturing and recording provenance (in terms of origin, snapshots, and transformations), but we also need to be able to distinguish between similar transformations and to be able to reason about which transformation is the correct one or who should have access to apply and record such transformations. There has been a substantial body of research to address provenance and trust over the last few years and, even though a relatively new field, there have been significant surveys on the matter (Buneman, Khanna, & Tan, 2001; Bose & Frew, 2005; Cheney, Chiticariu, & Tan, 2007). As is the case with any evolving field, there is no agreed way of doing things—let alone a standardised one. This is both a curse and a blessing: a blessing since it lifts any limitations to creativity when thinking about the problem, and a curse because there is a pressing need for a solution.

What we do have are characterisations of the problem and solutions to specific problem instances or domains. It makes little sense to enumerate all potential approaches. It makes more sense to enumerate the groundings of these approaches, which are usually one of the following three: formal modelling, semantics, and management rules. The first type of approach, (e.g., Souilah, Francalanza, & Sassone, 2009), attempts to formally model the problem, explicitly allow or disallow certain types of provenance and transformations, and reason about the effects. It treats provenance as either a type of computation—and therefore provenance is a computational model—or as a type of reasoning—and therefore provenance is a logics. An alternative to a formal model is a model based on semantics, (e.g., Chong, 2009). Equally applicable to provenance and trust, this is a (Geerts, Kementsietsidis, & Milano, 2006) more abstract way of tackling the issues. Instead of focusing on the syntax and the properties of data and their transformations, one should focus on the meaning of the transformation, which encompasses how it was actually defined. Using semantics one can reason about security and trust as well: there is a way to express the semantics of sensitivity and to reason about its effects. The final type of grounding is a more practical one and deals with the challenges of managing all this information (e.g., Rosenthal, Seligman, Chapman, & Blaustein, 2009; Archer, Delcambre, & Maier, 2009; Buneman, Chapman, & Cheney, 2006). As mentioned, provenance can be thought of as the recording of information about data and their changes. If the data model, the applicable transformations, and the security controls are fixed, one needs to orchestrate all such operations in a scalable way. That is, the problem now becomes one of managing scale as opposed to one of describing data or meaning.

2 TEN-YEAR VISION

As is the case with any new technological necessity, demand drives supply. It is therefore certainly arguable that provenance and trust will become key players in information management over the next few years. With organisations and individuals generating data at rates and scales never seen before, it is imperative to capture the provenance of the data and address its validity. A ten-year vision for provenance and trust is no easy feat. Given the recent emergence of the field and the exploratory nature of most approaches, one can only describe a list of *desiderata* as opposed to a concrete description of future technology. If there is one axis, however, that most of the efforts will likely focus on in order to make existing approaches interoperable, that axis is *standardisation*.

A Digital Object Identifier is a good starting point towards unique identification of digital resources. While not standardised yet, it serves as a platform for standardising the digital lifecycle. That is, we want to have a way to uniquely identify not only digital artefacts but also digital signatures of physical artefacts (e.g., individuals) and digitally captured workflows associated with the transformations of digital artefacts. We have used the Digital Object Identifier as a starting point and not the seemingly equivalent notion of the Universal Resource Identifier, which is also more or less standardised. The difference between the two is a subtle but important one: the digital-object-identifier standard not only represents a naming and disambiguation scheme, it also encompasses a sustainable way to persist the referenced object. It is therefore our vision that similar identifiers, governed by similar bodies as the International Digital Object Identifier Foundation, will exist for all aspects of the digital lifecycle. In particular, digital artefacts will have a unique identifier that unambiguously identifies them. Moreover, any part of a workflow associated with the creation or transformation of an artefact will also have such an identifier. Finally, each result of a transformation of an artefact will also be uniquely identified. Having such primitives is crucial to both provenance and trust. It deals with the three aforementioned types of provenance: for each type of provenance we will have a single and unambiguous way of capturing its sources and effects, in addition to the imposed transformation.

What will also be standardised are the compositional aspects of identifiers. It is certainly conceivable for parts of a workflow to be composed of other workflows. This implies a clear way of reasoning about what is applicable and what is not. This, in turn, means that there should be enough information recorded at the identifier level to allow for automated reasoning about applicable transformations. For instance, it should be possible to allow only certain types of identifier be used as input to a transformation and to assure that the output of a transformation be of a certain type. These points are also necessary for trust. Users may choose only to trust artefacts and transformations that have identifiers, are of a certain type, or are issued by a certain authority. Another way to look at the problem is through the lens of authentication and repeatability: transformations yielding different results than the expected ones or incompatible results according to their specifications are ill-formed at best, dubious and untrustworthy at worst.

Moreover, there shall be a well-defined and robust ownership and security model built on top of the identifier system. Individuals and processes shall be able to access only the artefacts (or workflows or parts of a workflow) the owner of the identifier sees fit. As before, there shall be reasoning aspects so that access control rights are propagated across identifiers, if the owners so desire. For instance, it is conceivable for the owner of a set of artefacts and workflows, who has granted access to these artefacts and workflows, to also implicitly grant access to the resulting artefacts.

Finally, in terms of implementation and deployment, the entire framework for capturing, tracking, and using provenance and trust as a means of processing digital artefacts will be fully decentralised. We do not use the more common term of the framework being distributed. The reason is that a distributed framework is only as good as its weakest link: even though the responsibility is spread across multiple authorities if a substantial number of those authorities stop functioning, so does the entire framework. In a decentralised setting, the failure of a single authority, or a substantial number of authorities, does not affect the liveliness or the operation of the framework.

3 CURRENT CHALLENGES

The previous section presented an idealistic vision for the state of affairs ten years from now. The follow-up question naturally is "how close to that vision are we?" As mentioned, provenance and its manifestations in trust is an emerging research field. As such, there are plenty of incubator approaches, and it is not clear to say which of these approaches is going to be the winner. Consider, for instance, the three groundings of formalisation, semantics, and management that were mentioned before. They are radically different ways of dealing with the problem and touch different aspects. Focusing on formalisation only implies the risk of increasing the expressivity and complexity of any framework beyond what is realistically implementable. Alternatively, dealing only with semantics bears the risk of representing approaches in a more abstract or more ambiguous way than necessary. Note that limiting approaches to a certain type of semantics (e.g., operational or denotational) does not necessarily solve the problem as it limits the implementation alternatives only to frameworks that are capable of preserving this semantics. Finally, to deal with manageability, one should have at least a prototype of a working model of provenance and trust in place so its maintenance and scalability can be addressed.

The first challenge is therefore to treat these alternatives as different dimensions of the same space, rather than as diverging factors (we shall revisit this point later). Historically, this has been easier to accomplish if there is complete control of the entire environment in which the framework operates. Consider, for instance, relational databases: all the important factors of data modelling, programmability, semantics, logical and physical representation, computation, and management are dealt with in their entirety by a single system that has complete control of everything. We are nowhere near such a reality when it comes to provenance and trust.

Another challenge is that provenance and trust will need to be retrofitted to existing infrastructures. This means that there has been a lot of work in building data stores and workflows on top of data stores that are working fine in their isolated environments. Having a fully-fledged decentralised framework that preserves all the locally assigned semantics and processes may well prove utopic unless substantial compromise is made, both in terms of what provenance and trust models we want to support and in terms of what existing provenance and trust practices can be modelled by a common framework. Identifying and reaching this compromise is likely to be one of the greatest challenges.

A further challenge is to deal with sustainable and complete ways of recording and tracking provenance and assigning trust. This process should not be manual but automated. We therefore need potentially new programming languages, or new implementations of existing programming languages that have support for provenance and trust already built in. If the aim is to have provenance and trust as a commodity in ten years' time, then their use and deployment needs to be transparent. Just as we expect certain infrastructures to be ubiquitous, so should provenance and trust recording and tracking mechanisms. This is not confined to the programming level but also appears at the representation level. It is not only a question of what to record but also how to record it and where. The concept of explicitly recording annotations and reasoning about annotations may well become a key player in the area (Geerts, Kementsietsidis, & Milano, 2006). At the same time, any mechanism should be efficient, scalable, and extensible.

There are, finally, potential risks when addressing standardisation that do not come from technical challenges but rather from political ones. Such a danger is—instead of reaching a community-driven and interoperable standard—to have a *de facto* standard in place. This is a common occurrence in computing: individual agendas from companies with vested interest in the standardisation effort are pushed and sometimes implemented and deployed before any standardisation authority decides whether or not they should be part of the standard. As such, it is crucial for the community to have open and accessible provenance and trust models (such as the Open Provenance Model: http://openboth provenance.org) that can serve both as a basis for discussion and as the basis for standardisation.

To conclude, we are not very close to having true provenance and trust mechanisms in place any time soon. There are plenty of challenges and limitations, technical and organisational, that simply make it impossible to identify a clear standard or a single way of doing things. However, we are in the favourable situation of having potential solutions, vested interest in the effort, and demand for an all-encompassing standard. To that end, the community should collaborate on realising a standard for provenance and trust.

4 RESEARCH DIRECTIONS PROPOSED

As mentioned, there exist various proposals, techniques, and good solutions to specific problem instances. However, we do not have a reference framework or a list of specifications and/or requirements that a complete solution to the problem should satisfy. At the same time, there is a pressing need for standardisation. Therefore, the priority should be on evaluating existing approaches in two dimensions: (*i*) how well they solve different aspects of provenance and trust and (*ii*) how easy it is to generalise them into a standard. Once dominant approaches have emerged, it is then possible to unify them into a standard.

The previous strategy by no means implies that work on principal aspects of provenance and trust should stop: both problems are far from solved. Fundamental work on the three aspects we mentioned earlier, namely formal modelling, semantics, and management should continue. However, we can no longer afford to let these three strands of work continue to evolve in a disconnected manner. If this happens, there is a clear risk of their never converging into a single framework. Therefore, it is proposed to follow a three-pronged approach with the main parts being:

1. Formal modeling of provenance and trust. It is necessary to have a way to represent all aspects of the digital lifecycle. Fundamental work in formal modeling should consist of three interconnected developments: a data model, a computational model, and a methodology to reason about computation. The data model should be powerful enough to capture not only data and provenance but also metadata. The computational model should encompass not only a sound and complete set of transformations on data conforming to the data model but also have a highly expressive power in order to capture the composition of transformations. At the same time, the computational model should be provenance-aware. It should be clear how provenance is recorded and tracked and how it propagates. Finally, the reasoning principles should make it possible to drill down and isolate any part of a workflow; additionally, they should cater for reasoning to characterise workflow effects.
2. Semantics. A formalisation in itself is a means to an end. For the formalisation to be effective and usable, it needs to have concrete semantics that are verifiable by humans and machines alike. Thus, there is a clear need for the semantics to be developed hand-in-hand with the formalisation. That way, it will be possible to statically analyse the provenance-aware transformations and capture their meaning.
3. Principles for managing provenance and trust. The application domain of the problem at hand is far from an academic exercise: it transcends specific systems and instantiations of the problem. It is imperative to combine formalisation and semantics into a set of principles guiding the building of fully-fledged provenance- and trust-aware systems. The management of provenance and trust should become a first-class citizen in the data management stack; much in the spirit of a database management system being used to manage all aspects of storing data, its stand-alone transformations in the form of transactions and security-based access control should dictate who has access to what.

The characterisation of this approach as a three-pronged one is not merely cosmetic. While research on these three aspects evolves, it is necessary for this evolution to be synergistic: unless there is a common baseline and overarching common goal for all aspects, the objective of standardising practice becomes more difficult if not impossible.

Three further points that were mentioned earlier are identification, decentralization, and retrofitting. In dealing with provenance and trust, we, as researchers, do not have the luxury to start with a clean sheet and/or to operate in isolation. For identification, it is certainly possible to build on existing practices of identifying entities in expressive data models and to reason about them (Buneman, Davidson, Fan, Hara, & Tan, 2003; Berners-Lee, Fielding, & Masinter, 2005). In terms of retrofitting, it is necessary for any research in the area to be applicable on existing data management infrastructure, whether it concerns standard organisational database management systems, scientific data management, or support for experimental sciences. Any developed techniques should be able to be seamlessly integrated with existing infrastructure without having to drastically change practice. If the latter happens, support for provenance- and trust-aware techniques will fail to pick up traction and be adopted. Similarly, there is an inherent need for decentralisation. The world we live in does not have a central management authority: every individual and computational process is capable of producing and transforming data. Striving for independence should be a fundamental aspect of research in the area.

5 RECOMMENDATIONS

The number of recommendations is minimal, but these recommendations effectively capture the main aspects of the previous sections. The first recommendation is the need for the introduction of a standardisation body on provenance and trust. This body should be open in its operation and should start by evaluating existing approaches in order to test their efficacy in terms of applicability and potential for generalisation.

The second recommendation is for the implementation of a canonical provenance- and trust-aware system. This system will act as a reference of what practitioners in the area need (the requirements) and what can be delivered and how. The implementation will capture all necessary use-cases identified by the standardisation bodies.

Perhaps the best way to tackle the problem is to follow the last two rules of Jim Gray for building data-intensive management systems: (*i*) start the design with “twenty queries”, and (*ii*) go from “working to working” (Hey, Tansley, & Tolle, 2009). The first of these rules is a heuristic by which—in building a new data management infrastructure—one should identify the twenty most important questions that need to be answered; these may well be provenance and/or trust use cases that a system that is fully provenance- and trust-aware should be capable of addressing. The second rule means that we cannot address all twenty questions in one go; rather, it is an incremental and iterative process. What needs to happen is to build from the ground up and address the questions in sequence. Start with a design that tackles the first question or use-case; then, minimally enrich that design so that the next question in the sequence is addressed; then iterate using this methodology until all questions are addressed.

To conclude, the recommendations are to take a pragmatic approach to solving the problem, rather than a more idealistic one. It would certainly be nice if we could address the problem by assuming zero knowledge and starting from scratch. However, not treating existing infrastructure and practice as fundamental building blocks to any approach we aim to standardise almost completely eliminates the potential for adapting the standard. It is therefore necessary to take informed and practically grounded steps towards identifying a solution. Only in that way can the ten-year vision of the area be realised.

6 REFERENCES

Archer, D., Delcambre, L., & Maier, D. (2009) A framework for fine-grained data integration and curation, with provenance, in a dataspace. *First Workshop on the Theory and Practice of Provenance.* USENIX Association.

Benjelloun, O., Das Sarma, A., Halevy, A., Theobald, M., & Jennifer, W. (2008) Databases with Uncertainty and Lineage. *The VLDB Journal 17* (2), pp 243-264.

Berners-Lee, T., Fielding, R. T., & Masinter, L. (2005) *RFC 3986 - Uniform Resource Identifier (URL): Generic Syntax.* Retrieved from the World Wide Web, June 28, 2013: http://tools.ietf.org/html/rfc3986

Bhagwat, D., Chiticariu, L., Tan, W.-C., & Vijayvargiya, G. (2004) An annotation management system for relational databases. *VLDB*, pp 900-911.

Bose, R., & Frew, J. (2005) Lineage Retrieval for Scientific Data Processing: A Survey. *ACM Computing Surveys, 37* (1), 1-28.

Buneman, P., Chapman, A., & Cheney, J. (2006) Provenance Management in Curated Databases. *SIGMOD.* ACM.

Buneman, P., Davidson, S., Fan, W., Hara, C., & Tan, W.-C. (2003) Reasoning about Keys for XML *28* (8), pp 1037-1063.

Buneman, P., Khanna, S., & Tan, W.-C. (2001) Why and Where: A Characterization of Data Provenance. *ICDT,* pp 316-330. Springer.

Chapman, A., Jagadish, H. V., & Ramanan, P. (2008) Efficient provenance storage. *SIGMOD*, pp 993-1006.

Cheney, J., Chiticariu, L., & Tan, W.-C. (2007) Provenance in Databases: Why, How, and Where. *Foundations and Trends in Databases, 1* (4), pp 379-474.

Chong, S. (2009) Towards semantics for provenance security. *First Workshop on the Theory and Practice of Provenance.* USENIX Association.

Cong, G., Fan, W., & Geerts, F. (2006) Annotation propagation revisited for key preserving views. *CIKM*, pp 632-641.

Cui, Y., Widom, J., & Wiener, J. L. (2000) Tracing the lineage of view data in a warehousing environment. *ACM Trans. Database Syst. 25* (2), pp 179-227.

Geerts, F., Kementsietsidis, A., & Milano, D. (2006) Mondrian: Annotating and querying databases through colors and blocks. *ICDE.* IEEE Computer Society.

Hey, T., Tansley, S., & Tolle, K. (2009) *The Fourth Paradigm: Data-Intensive Scientific Discovery.* Microsoft Research.

Lynch, C. A. (2001) When documents deceive: trust and provenance as new factors for information retrieval in a tangled web. *Journal of the American Society for Information Science and Technology 52*, pp 12-17.

Rosenthal, A., Seligman, L., Chapman, A., & Blaustein, B. (2009) Scalable Access Controls for Lineage. *First Workshop on the Theory and Practice of Provenance.* USENIX Association.

Souilah, I., Francalanza, A., & Sassone, V. (2009) A formal model of provenance in distributed systems. *First Workshop on the Theory and Practice of Provenance.* USENIX Association.

Wang, Y. R., & Madnick, S. E. (1990) A Polygen Model for Heterogeneous Database Systems: The Source Tagging Perspective. *VLDB*, pp 519-538.

(Article history: Available online 30 July 2013)