

Report from Dagstuhl Seminar 12091

Principles of Provenance

Edited by

James Cheney¹, Anthony Finkelstein², Bertram Ludäscher³, and Stijn Vansummeren⁴

1 University of Edinburgh, GB, jcheney@inf.ed.ac.uk

2 University College London, GB

3 University of California, Davis, US, ludaesch@ucdavis.edu

4 Université Libre de Bruxelles, BE, stijn.vansummeren@ulb.ac.be

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 12091 “Principles of Provenance”. The term “provenance” refers to information about the origin, context, derivation, ownership or history of some artifact. In both art and science, provenance information is crucial for establishing the value of a real-world artifact, guaranteeing for example that the artifact is an original work produced by an important artist, or that a stated scientific conclusion is reproducible.

Since it is much easier to copy or alter digital information than it is to copy or alter real-world artifacts, the need for tracking and management of provenance information to testify the value and correctness of digital information has been firmly established in the last few years.

As a result, provenance tracking and management has been studied in many settings, ranging from databases, scientific workflows, business process modeling, and security to social networking and the Semantic Web, but with relatively few interaction between these areas.

This Dagstuhl seminar has focused on bringing together researchers from the above and other areas to identify the commonalities and differences of dealing with provenance; improve the mutual understanding of these communities; and identify main areas for further foundational provenance research.

Seminar 26 February – 2 March, 2012 – www.dagstuhl.de/12091

1998 ACM Subject Classification D.2 Software Engineering, D.3 Programming Languages, H.1 Models and Principles, H.2 Database Management

Keywords and phrases Provenance, Lineage, Metadata, Trust, Repeatability, Accountability


Digital Object Identifier 10.4230/DagRep.2.2.84

1 Executive Summary

James Cheney

Bertram Ludäscher

Stijn Vansummeren

License  Creative Commons BY-NC-ND 3.0 Unported license
© James Cheney, Bertram Ludäscher, and Stijn Vansummeren

The term “provenance” refers to information about the origin, context, derivation, ownership or history of some artifact. In both art and science, provenance information is crucial for establishing the value of a real-world artifact, guaranteeing for example that the artifact is an original work produced by an important artist, or that a stated scientific conclusion is reproducible. Even in everyday situations, we unconsciously use provenance to judge the quality of an artifact or process. For example, we often decide what food to buy based on



Except where otherwise noted, content of this report is licensed under a Creative Commons BY-NC-ND 3.0 Unported license

Principles of Provenance, *Dagstuhl Reports*, Vol. 2, Issue 2, pp. 84–113

Editors: James Cheney, Anthony Finkelstein, Bertram Ludäscher, and Stijn Vansummeren



DAGSTUHL REPORTS

Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

freshness, origin and “organic” labels; and we decide whether or not to believe an online news article based on its source, author, and timeliness.

Maintaining good records of provenance that are sufficient to convince skeptics of the value of an artifact is difficult. It requires reflection or monitoring actions as they are performed. Every step in the chain of ownership of an important work of art needs to be recorded in a secure way, for example, in order to defend against forgery and deter attempts to sell stolen artwork.

Since it is much easier to copy or alter digital information than to alter real-world artifacts, there are even more opportunities for misinformation, forgery and error in the digital world than there are in the traditional physical world. For this reason, the need for provenance is now widely appreciated. Simple and unreliable forms of automatic provenance tracking, such as version numbering, ownership, creation and modification timestamps in file systems, have long been supported as a basic services on which more sophisticated tools can rely. In today’s increasingly networked and decentralized world, however, we anticipate the need for richer provenance recording and management capabilities to be built into a wide variety of systems.

For example, “grid” or “cloud” computing infrastructures are frequently used for scientific computing, as part of a widespread trend towards “eScience”, “cyberinfrastructure” or more recently the data-intensive “fourth paradigm” of science popularized by Jim Gray and others. These systems are complex and opaque. The correctness and repeatability of scientific conclusions (about, for example, climate change) is increasingly being questioned because of the lack of transparency of the complex computer systems used to derive the results. Provenance technology can help to restore transparency and increase the robustness of eScience, countering increasing skepticism of scientific results as evidenced by the so-called “Climategate” controversy in 2009.

This problem is already widely appreciated in scientific settings but is increasingly recognized as a problem in business, industrial and Web settings. Until recently, work on provenance has mostly taken place in relatively isolated parts of existing research communities, such as databases, scientific workflow-based distributed computing, or file systems, or the Semantic Web. However, we believe that to make real progress it will be necessary to form a broader research community focusing on provenance.

In this respect, the aims of Dagstuhl Seminar 12091 “Principles of Provenance” were to:

- bring together researchers from databases, security, scientific workflows, software engineering, programming languages, and other areas to identify the commonalities and differences of provenance in these areas;
- improve the mutual understanding of these communities;
- identify main areas for further foundational provenance research.

The seminar hosted 41 participants in total from the above communities, and included representatives from the W3C Provenance Working group that is in the process of standardizing a common data model for representing and exchanging provenance information.

To improve the mutual understanding of the various communities, the first day of the seminar was devoted to tutorial talks from well-respected members of each community. An overview of these tutorials may be found in the Section “Overview of Tutorials” starting on p. 88.

The rest of the seminar consisted of presentations of recent ongoing provenance research in the various communities, as well as break-out sessions aimed at deepening discussions and identifying open problems. An overview of the talks may be found starting on p. 93. An overview of the breakout sessions may be found starting on p. 102. A list of open problems may be found on p. 105.

2 Table of Contents

Executive Summary

<i>James Cheney, Bertram Ludäscher, and Stijn Vansummeren</i>	84
-------------------------------------------------------------------------	----

Overview of Tutorials

Tutorial: Provenance in Databases <i>Wang-Chiew Tan, Todd J. Green, Chris Ré</i>	88
Tutorial: Provenance in Scientific Workflows <i>Bertram Ludäscher, Shawn Bowers, Paolo Missier</i>	89
Tutorial: Software Engineering, Programming Languages and Security Perspectives <i>Perdita Stevens, Steve Chong, James Cheney</i>	91
Highlights of W3C Provenance Incubator Group and Subsequent WG Activities <i>Luc Moreau, Paul Groth, Simon Miles</i>	92

Overview of Talks

Computation Slices as (Universal) Provenance <i>Umut A. Acar</i>	93
Engineering Options for Better Provenance Capture <i>Adriane Chapman</i>	93
Semantics of the PROV data model <i>James Cheney</i>	93
The Multi-granularity, Multi-Provenance (MMP) Model for Relational Databases <i>Lois Delcambre</i>	94
Using Provenance to enable Reproducible Science <i>Juliana Freire</i>	94
A new Approach for Publishing Workflows: Abstractions, Standards and Linked Data <i>Daniel Garijo</i>	95
The PROV-O Ontology <i>Daniel Garijo</i>	96
On the semantics of SPARQL on annotated RDF <i>Floris Geerts</i>	96
An Overview on W3C PROV-AQ: Provenance Access and Query <i>Olaf Hartig</i>	96
Modelling provenance using Structured Occurrence Networks <i>Paolo Missier</i>	97
The W3C PROV Provenance Data Model <i>Luc Moreau</i>	98
Tracing Where and Who Provenance in Linked Data: A Calculus <i>Vladimiro Sassone</i>	98
Toward Provenance as Cross-cutting Concern <i>Martin Schäler</i>	99

Self-Identifying Sensor Data <i>Christian Skalka</i>	100
When-provenance: Tracing the history and evolution of data <i>Wang-Chiew Tan</i>	100
Temporal semantics for the open provenance model <i>Jan Van den Bussche</i>	101
Cracking the quality jigsaw puzzle using provenance pieces – A speculation, not a solution <i>Jun Zhao</i>	101
Working Groups	
Formal models for provenance <i>Jan Van den Bussche</i>	102
Systems and security perspectives on provenance <i>Nate Foster</i>	103
Social Aspects of Provenance <i>Adriane Chapman</i>	103
Additional discussions	104
Open Problems	
Problems related to formal provenance models	105
Provenance, security, and confidentiality	108
Social Aspects of Provenance	111
Participants	113


3 Overview of Tutorials

3.1 Tutorial: Provenance in Databases

Wang-Chiew Tan (IBM Research & University of California, Santa Cruz, US)

Todd J. Green (University of California, Davis, US)

Chris Ré (University of Wisconsin-Madison, US)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Wang-Chiew Tan, Todd J. Green, Chris Ré

Various kinds of provenance have been defined in the database research community to give a very fine-grained account of the derivation of a piece of data appearing in the output of a database transformation, often a database query [1].

In general we can discern two kinds of approaches: *annotation-based approaches* and *non-annotation-based approaches*. Annotation-based approaches, also called eager approaches, explicitly record information about the derivation of a piece of data in the database itself, typically as an extra attribute in the table. Annotation-based approaches hence require that an annotation representing the provenance of a data item be recorded directly in the database and further require that the annotation be correctly propagated through future database transformations.

Non-annotation-based approaches, also called lazy approaches, in contrast, do not store provenance in the database, but analyze the query answer, the query itself, and the input tables to calculate the provenance of a piece of data. An example of non-annotation-based approach is *why-provenance* (which indicates the source tables that contributed a distinguished output tuple). An example of annotation-based approaches is *where-provenance* (which indicates where in the source database the piece of data was copied from).

How-provenance is an annotation-based approach that goes beyond why-provenance and where-provenance to capture the way in which data items (i.e., tuples) are combined to produce output items (i.e., query result). How-provenance annotations are typically represented using *provenance polynomial expressions* drawn from a semiring, as defined by the work of Green et al. [2]. The tutorial discussed all of these forms of provenance in detail, and illustrated in particular how the provenance polynomial approach to recording how-provenance plays a crucial role in a practical system: the Hazy statistical data processing system [3].

References


- 1 J. Cheney, L. Chiticariu, W. C. Tan Provenance in Databases: Why, How, and Where. *Foundations and Trends in Databases* 1(4), p. 379–474.
- 2 T. J. Green, G. Karvounarakis, V. Tannen. Provenance semirings. *PODS 2007*, p. 31–40.
- 3 C. Ré et al. Hazy: Analyzing Data from More Sources, More Deeply than Ever Before. <http://research.cs.wisc.edu/hazy/>

3.2 Tutorial: Provenance in Scientific Workflows

Bertram Ludäscher (University of California, Davis, US)

Shawn Bowers (Gonzaga University, US)

Paolo Missier (Newcastle University, GB)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Bertram Ludäscher, Shawn Bowers, Paolo Missier

As the natural sciences have become increasingly computational and data-driven,¹ scientific workflows have become popular as a means for scientists to automate computational pipelines, to take advantage of parallel platforms (clusters and clouds), and—last not least—to keep track of data lineage and other provenance information to facilitate *reproducible science*. The tutorial was structured into three parts: (i) Overview: Introduction to scientific workflows and provenance (presented by Bertram Ludäscher); (ii) Technical challenges for managing provenance data from scientific workflows (presented by Shawn Bowers); and (iii) Overview of current research strands in workflow-based provenance (presented by Paolo Missier).

A *scientific workflow* is the description of a process for accomplishing a scientific objective, usually expressed in terms of tasks and their dependencies. Typically, scientific workflow tasks are computational steps for scientific simulations or data analysis steps. Common elements or stages in scientific workflows are acquisition, integration, reduction, visualization, and publication (e.g., in a shared database) of scientific data [5]. Scientific workflows share commonalities with business workflows and business process management approaches, but there are significant differences as well: e.g., the former are data-centric and often use a dataflow execution model, while the latter focus on processes and control-flow; scientific workflows emphasize scalable, automated execution [4], while workflow modeling and analysis are often the focus in business process management [6]. Scientific workflow systems (such as Askalon, Kepler, Pegasus, Taverna, VisTrails, etc.) provide a controlled execution environment for executing computational pipelines and thus offer unique opportunities to capture provenance information [2, 3], which can be used subsequently to explain or “debug” workflow results.

The opportunities to capture detailed provenance information in scientific workflows give rise to a number of technical challenges associated with storing, querying, and presenting (visualizing) scientific workflow provenance information (e.g., see [1, 8]). Some of these issues were presented in the second part of the tutorial, using examples and solutions from the Kepler workflow system.

Standards such as the Open Provenance Model (OPM) [10], which resulted from a community effort starting with the First Provenance Challenge workshop [11], are designed to provide a least common denominator, and thus by design do not include aspects specific to scientific workflow provenance.² As a result, provenance interoperability (e.g., see [7]) remains an important research topic, in particular, when taking into account fine-grained and “precise” provenance in the presence of different execution models, data models, and provenance models of the underlying workflow systems.

In the last part of the tutorial, a high-level taxonomy of research strands in the area of provenance for workflow-based applications was presented [9]. Its main branches are (i)

¹ This is witnessed, e.g., by notions such “e-Science”, the “4th Paradigm” (i.e., data-driven scientific discovery, with the 3rd Paradigm being “simulation/computational science”), and “Big Data”.

² A scientific workflow-centric extension of OPM is under development by the DataONE (dataone.org) Working Group on Provenance in Scientific Workflows.

modelling, (ii) *capturing*, (iii) *exploiting* provenance. Each branch contains a number of bibliographic references (occasionally commented) as its leaves.

The “modelling” branch addresses the topic of the convergence between database and process-based provenance, as well as the emerging research on privacy-preserving provenance, and “human in the loop” provenance. Each of these topics were perceived as increasingly important by the seminar participants. Amongst the main issues in the “capturing” branch are (i) provenance for non-workflow processes, mainly scripting languages for science; (ii) virtual experiments, represented by multiple semi-independent provenance traces; (iii) system-level provenance, and (iii) how to make provenance secure, tamper-evident, and trustworthy. Finally, the “exploitation” branch includes (i) provenance analytics, (ii) Provenance for reproducibility, and (iii) Provenance for improving data engineering. The index [9] is meant to be periodically updated, to form a more comprehensive reference for researchers in this particular area of provenance studies.

References


- 1 M.K. Anand, S. Bowers, and B. Ludäscher. Techniques for efficiently querying scientific workflow provenance graphs. In *Intl. Conf. on Extending Database Technology (EDBT)*, pages 287–298. ACM, 2010.
- 2 S. Davidson, S.C. Boulakia, A. Eyal, B. Ludäscher, T.M. McPhillips, S. Bowers, M.K. Anand, and J. Freire. Provenance in scientific workflow systems. *IEEE Data Eng. Bull.*, 30(4):44–50, 2007.
- 3 S.B. Davidson and J. Freire. Provenance and scientific workflows: challenges and opportunities. In *SIGMOD conference*, pages 1345–1350. ACM, 2008.
- 4 Y. Gil, E. Deelman, M. Ellisman, T. Fahringer, G. Fox, D. Gannon, C. Goble, M. Livny, L. Moreau, and J. Myers. Examining the challenges of scientific workflows. *Computer*, 40(12):24–32, 2007.
- 5 B. Ludäscher, S. Bowers, and T. McPhillips. Scientific workflows. In Ling Liu and M. Tamer Özsu, editors, *Encyclopedia of Database Systems*, pages 2507–2511. Springer, 2009.
- 6 B. Ludäscher, M. Weske, T. McPhillips, and S. Bowers. Scientific workflows: Business as usual? In *Intl. Conf. on Business Process Management (BPM)*, pp. 31–47, 2009. LNCS 5701.
- 7 P. Missier, B. Ludäscher, S. Bowers, S. Dey, A. Sarkar, B. Shrestha, I. Altintas, M.K. Anand, and C. Goble. Linking multiple workflow provenance traces for interoperable collaborative science. In *5th Workshop on Workflows in Support of Large-Scale Science (WORKS)*, New Orleans, 2010.
- 8 P. Missier, N.W. Paton, and K. Belhajjame. Fine-grained and efficient lineage querying of collection-based workflow provenance. In *Intl. Conf. on Extending Database Technology (EDBT)*, pages 299–310. ACM, 2010.
- 9 Paolo Missier. Research strands in workflow-based provenance. homepages.cs.ncl.ac.uk/paolo.missier/doc/Dagstuhl-PoP/Research_strands.html, 2012.
- 10 L. Moreau, B. Clifford, J. Freire, J. Futrelle, Y. Gil, P. Groth, N. Kwasnikowska, S. Miles, P. Missier, J. Myers, et al. The open provenance model core specification (v1. 1). *Future Generation Computer Systems*, 27(6):743–756, 2011.
- 11 L. Moreau, B. Ludäscher, I. Altintas, R.S. Barga, S. Bowers, S. Callahan, G. Chin Jr, B. Clifford, S. Cohen, S. Cohen-Boulakia, et al. Special issue: The first provenance challenge. *Concurrency and Computation: Practice and Experience*, 20(5):409–418, 2008.

3.3 Tutorial: Software Engineering, Programming Languages and Security Perspectives

Perdita Stevens (University of Edinburgh, GB)

Steve Chong (Harvard University, US)

James Cheney (University of Edinburgh, GB)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Perdita Stevens, Steve Chong, James Cheney

This tutorial touched upon three distinct themes: provenance in software engineering (presented by Perdita Stevens); provenance in programming languages (presented by James Cheney); and provenance and security (presented by Steve Chong).

From the earliest days of software engineering, practitioners have been concerned to trace the connections between the requirements that a software system must satisfy and the tests that establish that requirements have been met. This is termed *traceability*, and the same term is then used much more broadly in software engineering could be called provenance. Traceability is typically recorded as a so-called *requirements traceability matrix*, which is formally a binary relation on Requirements and Tests. Even with the best available commercial tool support, maintaining traceability information is a time-consuming partly manual process. It has been repeatedly observed that in practice, this maintenance is not well done. This is not (always) laziness on the part of the developers: the cost/benefit ratio often does not favour doing so. Moreover, the traceability information that is maintained may not be the information that is most needed. It has been reported that most traceability problems require tracing back before the development of the requirements specification which is typically the beginning of the traceability process. If provenance information is to be more widely collected and used it will be important to avoid reproducing these problems. Specifically, it is notable that the above gives, as yet, no common definition of what provenance information, annotation or traces mean, outside the pleasant world of databases.

In programming languages research, a number of sophisticated techniques have been proposed to track and control the flow of information in systems. In this tutorial, these techniques were motivated and explained. Subsequently, it was shown how information flow control techniques could be used to enforce security, and how this links with provenance.

Other concepts related to provenance in programming languages range from simple conveniences such as source code line number information used in compilers, to program slicing (a classical debugging technique widely studied in imperative programming languages), algorithmic debugging, type inference and type error slicing, dependency tracking, and language-based security. This tutorial covered some recent developments in formalizing security properties for provenance, including the properties of disclosure and obfuscation [1]. Additional topics, such as self-adjusting computation, bidirectional programming, and blame and contracts also seem relevant but to date there has been little work relating them and provenance.

References


- 1 J. Cheney. A formal framework for provenance security. In *CSF*, pages 281–293. IEEE, 2011.
- 2 J. Cheney, A. Ahmed, and U. A. Acar. Provenance as dependency analysis. *Mathematical Structures in Computer Science*, 21(6):1301–1337, 2011.

3.4 Highlights of W3C Provenance Incubator Group and Subsequent WG Activities

Luc Moreau (University of Southampton, GB)

Paul Groth (VU University Amsterdam, NL)

Simon Miles (King's College London, GB)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Luc Moreau, Paul Groth, Simon Miles

In 2009, a W3C Provenance Incubator Group³ was charged with the task of providing a state-of-the-art understanding of provenance for Semantic Web technologies, and developing a roadmap for development, and possible standardization of such technologies.

Based on the conclusions of the Incubator group, the W3C Provenance Working Group⁴ is currently in the process of defining a family of standards for the representation, exchange, location, and querying of provenance information on the web.

This tutorial gives an overview of the conclusions of the W3C Provenance Incubator group, as well as an overview of the standards that are currently under definition:

- PROV-DM [1] is a data model for provenance that describes the entities, people and activities involved in producing a piece of data or thing in the world. PROV-DM is domain-agnostic, but is equipped with extensibility points allowing further domain-specific and application-specific extensions to be defined.
- PROV-DM is accompanied by PROV-N [2], a technology-independent notation, which allows serializations of PROV-DM instances to be created for human consumption, which facilitates the mapping of PROV-DM to concrete syntax, and which is used as the basis for a formal semantics of PROV-DM that is currently under development.
- PROV-DM is also accompanied by PROV-O [3], a translation of PROV-DM into an OWL ontology for the purpose of expression of provenance in RDF.
- Finally, PROV-AQ [4] specifies how one can use standard Web protocols, including HTTP, to obtain information about the provenance of Web resources. It describes both simple access mechanisms for locating provenance information associated with web pages or resources, as well as provenance query services for more complex deployments.

References

- 1 The Provenance Data Model. L. Moreau, P. Missier (eds.) K. Belhajjame, S. Cresswell, Y. Gil, R. B'Far, P. Groth, G. Klyne, J. McCusker, S. Miles, J. Myers, S. Sahoo. W3C Working Draft, 2012. <http://www.w3.org/TR/prov-dm/>
- 2 The PROV Data Model and Abstract Syntax Notation. L. Moreau, P. Missier (eds.), K. Belhajjame, S. Cresswell, Y. Gil, R. Golden, P. Groth, G. Klyne, J. McCusker, S. Miles, J. Myers, S. Sahoo. W3C Working Draft, 2012. <http://www.w3.org/TR/prov-n/>
- 3 The PROV Ontology: Model and Formal Semantics. S. Sahoo, D. McGuinness (eds.) K. Belhajjame, J. Cheney, D. Garijo, T. Lebo, S. Soiland-Reyes, S. Zednik. W3C Working Draft, 2012. <http://www.w3.org/TR/prov-aq/>
- 4 Provenance Access And Query. L. Moreau, P. Groth (eds.), O. Hartig, Y. Simmhan, J. Myers, T. Lebo, K. Belhajjame, S. Miles. W3C Working Draft, 2012. <http://www.w3.org/TR/prov-o/>

³ http://www.w3.org/2005/Incubator/prov/wiki/W3C_Provenance_Incubator_Group_Wiki

⁴ http://www.w3.org/2011/prov/wiki/Main_Page

4 Overview of Talks

4.1 Computation Slices as (Universal) Provenance

Umut A. Acar (MPI for Software Systems – Kaiserslautern, DE)

License © ⓘ ⊖ Creative Commons BY-NC-ND 3.0 Unported license
© Umut A. Acar

Joint work of Acar, Umut A.; Cheney, James; Levy Paul; Perera, Roly

I present techniques that enable higher-order functional computations to “explain” their work by answering questions about how parts of their output were calculated. As explanations, I consider the traditional notion of program slices, which can be inadequate, and propose a new notion: computation slices. I present techniques for specifying flexible and rich slicing criteria based on partial expressions part of which are replaced by holes and present an “unevaluation” algorithm, for computing least program slices from computations reified as traces. In addition, I define the notion of a computation slices and briefly describe how they minimal computation slices can be computed.

4.2 Engineering Options for Better Provenance Capture

Adriane Chapman (MITRE – McLean, US)

License © ⓘ ⊖ Creative Commons BY-NC-ND 3.0 Unported license
© Adriane Chapman

Joint work of Chapman, Adriane; Allen, David; Blaustein, Barbara; Seligman, Len

Main reference M. D. Allen, A. Chapman, B. Blaustein, L. Seligman, “Provenance Capture in the Wild,” in Proc. of Third Int’l Provenance and Annotation Workshop (IPAW’10), pp. 98–101, LNCS, vol. 6378, 2010.

URL http://dx.doi.org/10.1007/978-3-642-17819-1_12

The research literature contains a fair amount of work about the positive things that can be done with provenance information. All of them though start with the presumption that a system actually has provenance information, which simply is not the case for most systems today. The value of provenance cannot be realized without first capturing it. While most of the literature further assumes central control over the a monolithic system in question (for example, a biomed researcher capturing provenance about their own experimental setup) most systems in the wild are neither centrally controlled nor monolithic in their technology selection. This talk addresses the many options and strategies for capturing provenance in real, large IT systems along with their pros and cons.

4.3 Semantics of the PROV data model

James Cheney (University of Edinburgh)


License © ⓘ ⊖ Creative Commons BY-NC-ND 3.0 Unported license
© James Cheney

The W3C PROV data model is based on an intuition that provenance information records a history of entities, activities, agents and interactions among them. A central and subtle issue is the fact that entities change over time, and the properties we use to describe them may not be fixed. To untangle these issues, the W3C group has been developing a formal

semantics, that is a mathematical model, with respect to which we can assign meanings to PROV statements, thinking of instances of the PROV data model as collections of logical statements describing some past events. In particular, PROV includes relations between different versions of the same entity at different times, or between more and less specific aspects of the same entity. The talk presented the semantics, focusing on these special relations and the underlying mathematical framework that helps explain their properties.

4.4 The Multi-granularity, Multi-Provenance (MMP) Model for Relational Databases

Lois Delcambre (Portland State University)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Lois Delcambre

Joint work of Archer, David; Lois Delcambre


Main reference D. Archer, “Conceptual Modeling of Data with Provenance,” PhD Dissertation, Computer Science Department, Portland State University, 2011, (Lois Delcambre, advisor)

URL <http://www.pdx.edu/sites/www.pdx.edu.computer-science/files/archerthesis2011.pdf>

In a relational database setting, the main interactions with the database are using the insert, update, delete, and query operators. Historically, databases systems with provenance have considered mechanisms to track tuples that produce query answers, e.g., as described by polynomials of various kinds (e.g., based on the work of Todd Green). In this talk, we’ll present a conceptual model for provenance in databases where the database system records all provenance explicitly, at a detailed level for all of the above operators. The database user can easily browse forward and backward through provenance and can issue queries to find current data based on characteristics of the provenance. Features of this model include that we track provenance for values, tuples, attributes, and tables (multi-granularity) and that we allow values in a database to have multiple provenances, e.g., from multiple insertions.

4.5 Using Provenance to enable Reproducible Science

Juliana Freire (Polytechnic Institute of NYU – Brooklyn, US)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Juliana Freire

Joint work of Freire, Juliana; David Koop; Emanuele Santos; Huy T. Vo; Philippe Bonnet; Matthias Troyer; Claudio Silva

URL <http://www.vistrails.org>

Important scientific results give insight and lead to practical progress. The ability to test these results is crucial for science to be self-correcting, and the ability to re-use and extend the results is key for science to move forward. In natural science, long tradition requires that results be reproducible, and in math, formal proofs that can be verified must accompany results. However, the same standard has not been applied for results backed by computational experiments.

Most computational experiments are specified only informally in papers, where experimental results are briefly described in figure captions; the code that produced the results is seldom available; and configuration parameters change results in unforeseen ways. The lack of reproducibility for computational results currently reported in the literature has raised questions about their reliability and has led to a widespread discussion on the importance of

computational reproducibility. However, a major barrier to a wider adoption of reproducibility is the fact that it is hard both for authors to derive a compendium that encapsulates all the components (e.g., data, code, parameter settings, environment) needed to reproduce a result, and for reviewers to verify the results.


As a step towards simplifying the creation and review of reproducible results, and motivated by the needs of computational scientists, we have built an infrastructure that supports the life cycle of computational experiments. This infrastructure makes it easier to generate and share repeatable results by making provenance a central component in scientific exploration, and the conduit for integrating data acquisition, derivation, and analysis as executable components throughout the publication process. Provenance is systematically and transparently captured and it includes all meta-data necessary to reproduce experiments, including the specifications of the computations, input and output data, source code, and library versions. We have also developed a set of solutions to address practical aspects related to reproducibility, including methods to link results to their provenance, explore parameter spaces, wrap command-line tools, interact with results through a Web-based interface, and upgrade the specification of computational experiments to work in different environments and with newer versions of software. This infrastructure has been implemented and released as part of VisTrails (<http://www.vistrails.org>), an open-source workflow-based data exploration and visualization tool, and it is already being used by different groups of scientists. Videos that illustrate the process to create reproducible publications using VisTrails are available at <http://www.vistrails.org/index.php/RepeatabilityCentral>.

References

- 1 D. Koop, E. Santos, P. Mates, H. T. Vo, P. Bonnet, B. Bauer, B. Surer, M. Troyer, D. N. Williams, J. E. Tohline, J. Freire, and C. T. Silva. A provenance-based infrastructure to support the life cycle of executable papers. *Procedia Computer Science*, 4:648–657, 2011. Proceedings of the International Conference on Computational Science, ICCS 2011.

4.6 A new Approach for Publishing Workflows: Abstractions, Standards and Linked Data

Daniel Garijo (Universidad Politécnica de Madrid, ES)

License     Creative Commons BY-NC-ND 3.0 Unported license
© Daniel Garijo

Joint work of Garijo, Daniel; Gil, Yolanda

Main reference D. Garijo, Y. Gil, “A new approach for publishing workflows: abstractions, standards, and linked data,” in Proc. of 6th Workshop on Workflows in Support of Large-Scale Science (WORKS’11), pp. 47–56, ACM, New York, NY, 2011.


URL <http://dx.doi.org/10.1145/2110497.2110504>

In recent years, a variety of systems have been developed that export the workflows used to analyze data and make them part of published articles. We argue that the workflows that are published in current approaches are dependent on the specific codes used for execution, the specific workflow system used, and the specific workflow catalogs where they are published. We take a new approach that addresses these shortcomings and makes workflows more reusable through: 1) the use of abstract workflows to complement executable workflows to make them reusable when the execution environment is different, 2) the publication of both abstract and executable workflows using standards such as the Open Provenance Model that can be imported by other workflow systems, 3) the publication of workflows as Linked Data that results in open web accessible workflow repositories. As part of this work, we developed

the OPMW profile for OPM that allows us to publish abstract workflows and link them to the workflow execution provenance. We illustrate this approach using a complex workflow that we re-created from an influential publication that describes the generation of ‘drugomes’.

4.7 The PROV-O Ontology

Daniel Garijo (Universidad Politécnica de Madrid, ES)


License  Creative Commons BY-NC-ND 3.0 Unported license
© Daniel Garijo

Joint work of Garijo, Daniel; Lebo Timothy; Sahoo, Satya; McGuinness, Deborah; Lang, Mike; Belhajjame, Khalid; Cheney, James; Soiland-Reyes, Stian; Zednik, Stephan; Zhao, Jun

In this short talk, I introduce the PROV-O Ontology, an OWL-RL mapping of the PROV Data model. In the presentation I explain briefly the main classes, relationships and the RDF serialization of a complete example.

4.8 On the semantics of SPARQL on annotated RDF

Floris Geerts (University of Edinburgh, GB)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Floris Geerts


Joint work of Cristophides, Vassilis; Fundulaki, Irini; Geerts, Floris; Karvounarakis, Grigoris

We revisit the semantics of SPARQL on RDF in the presence of annotations. It is readily verified that for such a semantics to work correctly, one needs operations on annotations that correspond to the various operators supported by SPARQL, and furthermore, these annotation operations need to adhere to certain algebraic identities. It readily follows that when the positive fragment of SPARQL is considered, a semiring structure on the annotations is required. Semirings, however, do not suffice when dealing with the OPTIONAL construct in SPARQL.

Instead, we identify a new algebraic structure for SPARQL annotations, define a corresponding free object and show how it can be used to evaluate SPARQL on annotated RDF.

4.9 An Overview on W3C PROV-AQ: Provenance Access and Query

Olaf Hartig (Humboldt Universität zu Berlin, DE)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Olaf Hartig

Joint work of Klyne, Graham; Groth, Paul; Moreau, Luc; Hartig, Olaf; Simmhan, Yogesh; Myers, James; Lebo, Timothy; Belhajjame, Khalid; Miles, Simon;

Main reference L. Moreau, O. Hartig, Y. Simmhan, J. Myers, T. Lebo, K. Belhajjame, S. Miles, “PROV-AQ: Provenance Access and Query,” W3C Working Draft, 10 January 2012, edited by Graham Klyne and Paul Groth.

URL <http://www.w3.org/TR/prov-aq/>

This short talk introduces the “Provenance Access and Query” (PAQ) document which is part of the PROV family of documents developed by the W3C Provenance Working Group.






The purpose of PAQ is to describe how to locate, retrieve, and query provenance information on the Web. The talk will briefly introduce the following main contributions of PAQ:

- A simple mechanism for discovery and retrieval of provenance information; and
- More advanced discovery service and query mechanisms.

Finally, we will point out some of the open issues of the current version of PAQ.

4.10 Modelling provenance using Structured Occurrence Networks

Paolo Missier (Newcastle University, GB)

License      Creative Commons BY-NC-ND 3.0 Unported license
© Paolo Missier

Joint work of Missier, Paolo; Randell, Brian; Koutny, Maciej

Main reference P. Missier, B. Randell, M. Koutny, “Modelling Provenance using Structured Occurrence Networks,” in Proc. of 4th Int’l Provenance and Annotation Workshop (IPAW’12), Santa Barbara, CA, June 2012.

URL <http://homepages.cs.ncl.ac.uk/paolo.missier/doc/Dagstuhl-SON-provenance.pdf>


Occurrence Nets (ON) are directed acyclic graphs that represent causality and concurrency information concerning a single execution of a system. Structured Occurrence Nets (SONs) extend ONs by adding new relationships, which provide a means of recording the activities of multiple interacting, and evolving, systems. Although the initial motivations for their development focused on the analysis of system failures, their structure makes them a natural candidate as a model for expressing the execution traces of interacting systems. These traces can then be exhibited as the provenance of the data produced by the systems under observation. In this paper we present a number of patterns that make use of SONs to provide principled modelling of provenance. We discuss some of the benefits of this modelling approach, and briefly compare it with others that have been proposed recently. SON-based modelling of provenance combines simplicity with expressiveness, leading to provenance graphs that capture multiple levels of abstraction in the description of a process execution, are easy to understand and can be analyzed using familiar graph query techniques.

References

- 1 E. Best and R. Devillers. Sequential and concurrent behaviour in Petri net theory. *Theoretical Computer Science*, 55(1):87–136, 1987.
- 2 D. Harel and P. Thiagarajan. Message Sequence Charts. In L. Lavagno, G. Martin, and B. Selic, editors, *UML for Real*, pages 77–105. Springer US, 2004.
- 3 J. Kleijn and M. Koutny. Causality in Structured Occurrence Nets. In C. Jones and J. Lloyd, editors, *Dependable and Historic Computing*, volume 6875 of *Lecture Notes in Computer Science*, pages 283–297. Springer Berlin / Heidelberg, 2011.
- 4 M. Koutny and B. Randell. Structured Occurrence Nets: A Formalism for Aiding System Failure Prevention and Analysis Techniques. *Fundamenta Informaticae*, 97, 2009.
- 5 B. Randell. Occurrence Nets Then and Now: The Path to Structured Occurrence Nets. In L. Kristensen and L. Petrucci, editors, *Applications and Theory of Petri Nets*, volume 6709 of *Lecture Notes in Computer Science*, pages 1–16. Springer Berlin / Heidelberg, 2011.

4.11 The W3C PROV Provenance Data Model

Luc Moreau (University of Southampton, GB)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Luc Moreau

Joint work of Moreau, Luc; Paolo Missier; Belhajjame, Khalid; Cresswell, Stephen; Gil, Yolanda; B'Far, Reza; Groth, Paul; Klyne, Graham; McCusker, Jim; Miles, Simon; Myers, James; Sahoo, Satya

Main reference L. Moreau, P. (eds.) – K. Belhajjame, R. B'Far, S. Cresswell, Y. Gil, P. Groth, G. Klyne, J. McCusker, S. Miles, J. Myers, S. Sahoo, (contributors), “The Provenance Data Model,” W3C Working Draft, 03 February 2012.

URL <http://www.w3.org/TR/prov-dm/>

PROV-DM is a data model for provenance that describes the entities, people and activities involved in producing a piece of data or thing in the world. PROV-DM is domain-agnostic, but is equipped with extensibility points allowing further domain-specific and application-specific extensions to be defined.


PROV-DM is accompanied by PROV-N, a technology-independent notation, which allows serializations of PROV-DM instances to be created for human consumption, which facilitates the mapping of PROV-DM to concrete syntax, and which is used as the basis for a formal semantics of PROV-DM.

References

- 1 The Provenance Data Model. L. Moreau, P. Missier (eds.) K. Belhajjame, S. Cresswell, Y. Gil, R. B'Far, P. Groth, G. Klyne, J. McCusker, S. Miles, J. Myers, S. Sahoo. W3C Working Draft, 02 February 2012. <http://www.w3.org/TR/2012/WD-prov-dm-20120202/>. Latest version: <http://www.w3.org/TR/prov-dm/>

4.12 Tracing Where and Who Provenance in Linked Data: A Calculus

Vladimiro Sassone (University of Southampton, GB)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Vladimiro Sassone

Joint work of Dezani-Ciancaglini, Mariangiola; Horne, Ross; Sassone, Vladimiro

Main reference M. Dezani, R. Horne, V. Sassone, “Tracing where and who provenance in Linked Data: a calculus,” *Theoretical Computer Science*, *in press*. Pre-print available.

URL <http://eprints.soton.ac.uk/335248/>

Linked Data provides some sensible guidelines for publishing and consuming data on the Web. Data published on the Web has no inherent truth, yet its quality can often be assessed based on its provenance.

This work introduces a new approach to provenance for Linked Data. The simplest notion of provenance – viz., a named graph indicating where the data is now – is extended with a richer provenance format. The format reflects the behaviour of processes interacting with Linked Data, tracing where the data has been published and who published it. An executable model is presented based on abstract syntax and operational semantics, providing a proof of concept and the means to statically evaluate provenance driven access control using a type system.

4.13 Toward Provenance as Cross-cutting Concern

Martin Schäler (Universität Magdeburg, DE)

License © © ⊖ Creative Commons BY-NC-ND 3.0 Unported license
© Martin Schäler

Joint work of Schäler, Martin; Schulze, Sandro; Saake, Gunter

Main reference M. Schäler, S. Schulze, G. Saake, “A Hierarchical Framework for Provenance Based on Fragmentation and Uncertainty,” Technical Report FIN-01-2012, School of Computer Science, University of Magdeburg, Germany, 2012.

URL http://www.witi.cs.uni-magdeburg.de/iti_db/publikationen/ps/auto/SSS2012.pdf

Provenance gained much attention in the recent past, especially for explaining and validating origin as well as derivation history of data. Furthermore, this term is used in many communities such as fine-grained annotations in relational databases, domain-specific approaches in scientific workflows, and even to determine source code ownership. Thus, we cannot give a clear definition about provenance sufficient for all these communities. In fact, it is even hard to give a clear definition to one of these communities. As a result, current solutions capturing provenance, are not sufficient in complex systems (e.g., forensics, medical data) where data items cross the borders of multiple systems, having different granularities, or even non computational steps are involved.

We argue that creating a solution for every application domain covering the versatile characteristics of provenance is inflexible, laborious, or even impossible. In contrast, our vision is to integrate provenance as cross-cutting concern into existing systems efficiently. As first step to realize our vision, we analyzed the literature addressing different parts of provenance to identify commonalities in provenance. Based on the current state of the art, there are three characteristics that seem to hold generally for provenance information [1]. Provenance is unchangeable, fragmentary at different levels of granularity, and contains a certain amount of uncertainty. While the first characteristic is a fundamental prerequisite the latter ones are dimensions of provenance, allowing to build a hierarchical framework covering a broad variety of approaches reaching from coarse grained notations (e.g., Open Provenance Model) to the principles of fine grained formal approaches (e.g., why and where provenance, semiring model). Furthermore, we use this framework to differentiate between provenance and related fields such as causality.

Currently, we started to integrate the cross-cutting provenance concern, based on our framework, into existing systems. Therefore, we analyze the feasibility of applying techniques from modern software engineering allowing a minimal invasive integration and if necessary un-integration of provenance. Furthermore, we evaluate their advantages and drawbacks. As a starting point we have chosen database systems, because there are formal models which can be implemented and recent insights such extensions of the semiring model for aggregate queries and linking provenance to causality seem to be promising to apply parts of the solutions to different data models and programming paradigms. Finally, linking different systems where we capture provenance (in a reliable way) is another important challenge. To this end, we propose the use of invertible watermarking schemes tailored to the requirements of the underlying systems [2].

For the future, we aim at identifying open research issues and present respective solutions, to move the borders hindering to fulfill our vision of provenance as cross-cutting concern.


References

- 1 M. Schäler, S. Schulze, and G. Saake. *A Hierarchical Framework for Provenance Based on Fragmentation and Uncertainty*. Technical Report FIN-01-2012, School of Computer Science, University of Magdeburg, Germany, 2012

- 2 M. Schäler, S. Schulze, R. Merkel, G. Saake, and J. Dittmann. *Reliable Provenance Information for Multimedia Data Using Invertible Fragile Watermarks*. 28th British National Conference on Databases (BNCOD), volume 7051 of LNCS, pages 3–17. Springer, 2011

4.14 Self-Identifying Sensor Data

Christian Skalka (University of Vermont, US)

License  Creative Commons BY-NC-ND 3.0 Unported license

© Christian Skalka

Joint work of Skalka, Christian; Chong, Stephen; Vaughan, Jeffrey

Main reference S. Chong, C. Skalka, J.A. Vaughan, “Self-Identifying Sensor Data,” in Proc. of 9th Int’l Conf. on Information Processing in Sensor Networks (IPSN’10), pp. 82–93, ACM, 2010.

URL <http://dx.doi.org/10.1145/1791212.1791223>

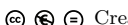
Public-use sensor datasets are a useful scientific resource with the unfortunate feature that their provenance is easily disconnected from their content. To address this we introduce a technique to directly associate provenance information with sensor datasets. Our technique is similar to traditional watermarking but is intended for application to unstructured time-series datasets. Our approach is potentially imperceptible given sufficient margins of error in datasets, and is robust to a number of benign but likely transformations including truncation, rounding, bit-flipping, sampling, and reordering. We provide algorithms for both one-bit and blind mark checking, and show how our system can be adapted to various data representation types. Our algorithms are probabilistic in nature and are characterized by both combinatorial and empirical analyses.

References

- 1 S. Chong, C. Skalka, and J. Vaughan. *Self-Identifying Sensor Data*. In ACM Conference on Information Processing in Sensor Networks (IPSN), 2010.

4.15 When-provenance: Tracing the history and evolution of data

Wang-Chiew Tan (IBM Research & University of California – Santa Cruz, US)

License  Creative Commons BY-NC-ND 3.0 Unported license


© Wang-Chiew Tan

Many scientific, business, and Web datasets produced today are hierarchical and associated with multiple dimensions of time. Archiving such data in a way that preserves the semantics of the different time dimensions can help understand the past and anticipate the future. However, there have been very few systems that can effectively create a semantic archive of such evolving hierarchical data under more than one time dimension.

We have recently developed a system, called Tempura, that supports efficient and compact temporal archiving of evolving hierarchical data under multiple dimensions of time. Tempura creates a multi-dimensional longitudinal record of knowledge about an entity by grouping entities across different snapshots together in the archive. The associated time dimensions are coalesced and independently varied to maintain a consistent view of the entity over time. We call such multidimensional longitudinal knowledge of an entity its *when-provenance*, which intuitively corresponds to when one knows what one knows about the entity. I will describe how the Tempura archive model naturally captures when-provenance, its implementation, and how it can support temporal data visualization.

4.16 Temporal semantics for the open provenance model

Jan Van den Bussche (Hasselt University, BE)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Jan Van den Bussche

Joint work of Kwasnikowska, Natalia; Moreau, Luc; Van den Bussche, Jan


Main reference N. Kwasnikowska, L. Moreau, J. Van den Bussche, “A Formal Account of the Open Provenance Model,” University of Southampton ECS Eprint 21819.

URL <http://eprints.ecs.soton.ac.uk/21819/>

The Open Provenance Model (OPM) is a graph-based data model for the representation of provenance information. Provenance information could be defined roughly as information about “what has happened” during some complex process. OPM is expected to heavily influence a W3C standard for provenance which is in the making. The current OPM specification defines a graph-based syntax, as well as some inference rules. A formal semantics that explains the soundness of these inference rules, and that could be used to prove completeness of the inference rules, was lacking however. In this paper we will propose a temporal semantics for OPM graphs; we will see that the current inference rules are, in fact, incomplete, and we will provide a complete set of inference rules.

4.17 Cracking the quality jigsaw puzzle using provenance pieces – A speculation, not a solution

Jun Zhao (University of Oxford, GB)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Jun Zhao


Joint work of Wf4Ever consortium

Digital science brings sea change to scientific research. A vast number of scientific data is made available in digital format, without their paper counterparts. Digital, or computational, experiments are increasingly used to replace or complement their wet-lab peers. ‘Big’ science becomes possible as scientists start to collaborate using data and methods shared and published on the Web. However, quality of data remains a major concern of scientists. The astronomy scientists we work with explicitly express their concern in trusting and reusing digital data, results and methods published and shared by third parties. To this end, we investigate the role of provenance information in producing a ‘quality stamp’ upon these research resources. We speculate different provenance pieces that can be drawn together. Instead of presenting solutions, we hope to stimulate further discussions regarding this topic.

5 Working Groups

5.1 Formal models for provenance

Led by James Cheney and Jan van den Bussche, and summarized by Jan Van den Bussche (Hasselt University, BE)

License  Creative Commons BY-NC-ND 3.0 Unported license
© Jan Van den Bussche

Joint work of All participants of the formal models for provenance break-out sessions.

There were two break-out sessions concerning formal models for provenance.

In the first session, we reviewed and commented on the current draft of the W3C PROV formal semantics. In particular, we reviewed the notion of *world model*, the notion of *object*, and the connection between objects and “things” in the (real) world. There seemed to be agreement that this setup was a useful approach to modeling provenance. We proceeded by reviewing the notions of *specialization-of* and *alternate-of*. These are relations between objects, but the relations are defined such that they can only hold between objects that refer to the same “thing”. Here we observed that some clauses in the definition, related to attribute values that must agree, were redundant. A general critique that was raised is whether the *thing-of* connection from objects to things should be “cast in stone” or be part of an interpretation that can vary.




In the second session on formal models we discussed another topic, namely, provenance information for database query results in the form of provenance polynomials. More specifically, we looked at the case where queries are not merely positive relational algebra expressions, but full relational algebra expression, involving the difference operator. The discussion on including the difference operator was initiated by Floris Geerts’ talk on recording provenance for the SPARQL language where the semantics of one of the SPARQL operators (namely optional) is expressed by means of a “minus” operation. When a formal “minus” operator is added to the provenance polynomial semiring, extended provenance polynomials can be derived that involve the minus operator. Note that to capture that a tuple is *not* in the query result we assign to it provenance “0”. We worked out an example of a difference operation on relations annotated with tuple ids. For example, suppose we compute the expression $R - (R - S)$ on relations where R contains b with tuple-id x_2 and S contains b with tuple-id x_4 . Then the final result will contain b with tuple-id $x_2 - (x_2 - x_4)$. Now Floris points out that if you provide the semiring with additional axioms that imply that $x_2 - (x_2 - x_4) = x_2x_4$, then we get the same final annotation as we would get when computing $R \cap S$, and indeed $R - (R - S)$ is equivalent to $R \cap S$. So, it seems that the full relational algebra with difference can indeed be handled by an extension of the semiring provenance approach. Unfortunately, there are only two papers on handling difference with provenance semirings [1, 2], and these papers do not seem to make very explicit how this can work. Floris Geerts in the end raised some doubts on the axiom $x_2 - (x_2 - x_4) = x_2x_4$, perhaps this is a reason why it is not explicit in the literature.

References

- 1 Y. Amsterdamer, D. Deutch, V. Tannen: On the Limitations of Provenance for Queries With Difference. CoRR abs/1105.2255: (2011)
- 2 F. Geerts, A. Poggi: On database query languages for K-relations. J. Applied Logic 8(2): 173-185 (2010)

5.2 Systems and security perspectives on provenance

Led by Nate Foster, and summarized by Nate Foster (Cornell University, US)

License    Creative Commons BY-NC-ND 3.0 Unported license
© Nate Foster

Joint work of All participants of the provenance, systems and security break-out sessions.




We discussed general issues related to provenance and security, as well as some specific security mechanisms provided in systems being developed by two of the participants. Overall, there was broad agreement that security issues are critically important, and that failing to deal with them could hinder the broader adoption of provenance. One clear set of issues concerns the confidentiality and integrity of provenance metadata itself. For example, mechanisms for ensuring that unauthorized users do not access or modify provenance metadata are obviously needed. The group discussed using cryptography as a means for obtaining secure and tamper-proof storage of provenance, but also noted that because provenance tends to be stored for a very long time, current cryptography may not provide sufficient protection. Peter Buneman proposed time-limited archiving systems as a potentially interesting idea for future work. Another set of issues concerns evaluating queries over provenance. It is well known that queries can be used to indirectly obtain information about the underlying data – cf. the case involving the Netflix Prize data [1]. This is exacerbated in systems with provenance, since knowing how a query result was computed can provide useful information to an attacker. The group discussed several scenarios including employee reviews (where provenance might identify the co-workers involved in producing the reviews) and elections (where provenance might reveal an individual’s vote). Although existing work on database privacy seems to provide the basic framework for reasoning about privacy-preserving queries, no systems we know of handle the complicated graph structures often used to represent provenance or adequately captures the “entanglement” between provenance and the underlying data. Lastly, the group discussed whether security mechanisms should be built into the systems that collect provenance or imposed after the fact. Adriane Chapman and Ashish Gehani described the treatment of provenance in PLUS and SPADE. Both systems provide mechanisms for restricting the information incorporated into provenance artifacts.

References

- 1 A. Narayanan, V. Shmatikov. Robust De-anonymization of Large Sparse Datasets. In IEEE Symposium on Security and Privacy (Oakland) 2008, p. 111–125.

5.3 Social Aspects of Provenance

Led by Carole Goble, and summarized by Adriane Chapman (MITRE – McLean, US)

License    Creative Commons BY-NC-ND 3.0 Unported license
© Adriane Chapman

Joint work of All participants of the social aspects of provenance break-out sessions.

We discussed the social needs, benefits, risks, obstacles, incentives and challenges of provenance capture and usage. It was noted that there are rewards and incentives for using provenance, which are often reaped by different individuals than the ones who have the burden of reporting provenance. Three key use cases were presented to facilitate discussion:

1. Employee Feedback [1]. Consider three employees give private feedback on a co-worker’s performance. They are willing to do so because their responses are kept private. The

employer’s provenance record could divulge sensitive information about the reviews, e.g. All of the reviews were negative. If the provenance record contains 3 reviews, and there are only 3 other co-workers, the employee knows that all co-workers shared negative feedback.

2. Corporate Structure [2]. Consider an organization with a specific task. Division of labor means that different individuals within the organization have very different jobs. As a reduced example, Alice reads newspapers and synthesizes a report. Bob builds a program to fuse all of Alice’s reports on a given topic. Cathy takes these fused reports, needs to understand the sources originally used (were they trustworthy, is there duplication) and makes a decision (e.g., to invest or not). Doug, the manager, needs to understand how well Alice, Bob and Cathy are performing. Cathy and Doug are obvious users of provenance, but the burden of creation lies more heavily on Alice and Bob.
3. Scientific Usage [3]. A scientific user has the incentive to wish to track provenance for very positive reasons: to enable understanding of scientific results; to receive due credit; etc. However, divulging provenance also has potential negative consequences: someone stealing the secret sauce; someone seeing all of the ugly dead-ends explored; etc.

Using these use cases as a basis, the group explored trade-offs of trust, levels of friendliness, and cost in terms of capturing and exposing all, some or no provenance.

References

- 1 U. Braun, A. Shinnar, and M. Seltzer. Securing Provenance. In USENIX HotSec, 2008.
- 2 A. Chapman and A. Rosenthal. Provenance Needs Incentives for Everyone. In TaPP, 2011.
- 3 C. Goble, D. De Roure, and S. Bechhofer. Accelerating scientists’ knowledge turns. In IC3K, 2012.

5.4 Additional discussions

The participants also held a number of informal research discussions as is normal for a Dagstuhl seminar. Of particular note:

- discussion of semantics and other features of the W3C PROV standard among WG members (Moreau, Groth, Missier, Cheney, Garijo, Eckert, Hartig, Zhao)
- development of a “best practice” mapping from Dublin Core to PROV by Garijo and Eckert.
- discussion of the provenance semiring model among researchers who had not previously been exposed to it, leading to an accessible, informal “nano-tutorial” due to Lois Decambre (lightly edited)

In order to interpret the most informative version of T.J. Green’s provenance polynomials with relational queries that involve only select, project, join/cross product, and union, just imagine that every tuple in a relational database is identified by a unique symbol. You can think of it like a label that is assigned to each tuple. And imagine that these labels are: a, b, c , etc.

Then a provenance polynomial such as $a^2 + 2ab + c^2$ associated with a given tuple (call it x) in the query answer, tells us that x is in the query answer because: the appearance of the tuple a – twice – resulted in x . (That is, some table was joined with itself and the tuple labeled a joined with itself and produced x .)

Or (because the $+$ symbol means “or” ... or “UNION”) the presence of tuple a and b (together) – twice – produced x . So, the tuple a joined with tuple b – in two

different situations – and they both produced x . We know that it happened in two different situations because of the multiplier of 2. We can think of it as a joined with b and then elsewhere in the query processing, a joined with b another time: $2ab = ab + ab$.

Or: the presence of tuple b – twice – results in x . Once again – this represents a self-join where the tuple b joining with itself – produced x .

In the polynomial, multiplication means that both input tuples needed to be present (typically through a join) and addition means that either of the two combinations would be sufficient to produce the output (typically through a union).

So, the provenance polynomial simply tells us, precisely, all of the ways that input tuples combined to produce this output tuple (x , in my example). It's a complete recording of the provenance (or at least, as complete as one can have using semiring annotations); there's no other combination of tuples in the input that could lead to x . I also mentioned that the polynomials (like $a^2 + ab$) are actually combining tuple labels; they are **not** necessarily numerical variables in the classical sense – like one would see in a polynomial in an algebra class in middle school.

I may have also mentioned that one of the ways you can use the polynomial is to figure out if x (in this example) still belongs in the query answer if one or more of the tuples in the input database disappear (or is not trusted or whatever). If the tuple symbol is replaced with '1' if it exists and '0' if it doesn't, then you can find out whether x still belongs in the query answer by evaluating the polynomial.

6 Open Problems

Over the course of their discussions, the seminar participants have identified the following core set of open problems that require investigation.

6.1 Problems related to formal provenance models

Reported by Umut Acar, Sarah Cohen-Boulakia, Paul Groth, Lois Delcambre, Bertram Ludäscher, Simon Miles, Paolo Missier, and Stijn Vansummeren, summarizing discussion by other participants

6.1.1 The semiring based approach towards provenance

For query languages with limited expressive power, like the positive fragment of the relational algebra, recent research has shown it possible to define the formal mathematical structure that provenance annotations should take in order to be able to interpret these annotations in a way that corresponds to the execution of the query. In particular, for the positive fragment of the relational algebra, this mathematical structure is the semiring [10]. Extending this approach to more powerful query languages, such as query languages with aggregation [3] or non-monotonic operators [11, 12, 4] is challenging. Indeed, when considering the difference operator there are various reasonable but non-equivalent mathematical structures that can describe its execution and semantics, all depending on the context in which the provenance annotations are to be used [11, 4].

More work on extending the semiring approach towards provenance with set difference and other non-monotonic operators seems necessary to get a full understanding of the issues involved. In particular:

- Semiring-based approaches do two things: they extend the data model in order to cope with set, bag, probabilistic, etc. kind of data; and they allow for the modeling of annotations. What if we only stick to one single semantics (say set, or bags)? Does difference then still cause a problem?
- How can one redefine the semantics of non-monotonic operators in query languages that operate under an open-world assumption, such as SPARQL, in order to allow for a simple characterization of the structure of provenance annotations?
- What is the limit of the provenance polynomials approach towards provenance? For example, can it be reconciled with approaches such as where-provenance that do not necessarily respect all query equivalences?

6.1.2 Program Traces

In contrast to database query languages with limited expressiveness, it is more difficult to give a detailed provenance account of the execution of a program written in a fully-expressive programming language.

There is a large space of different possible forms of execution traces aimed at different applications:

- profiling, debugging, slicing [14]
- dynamic information flow security [13]
- incremental recomputation (self-adjusting computation [2])
- possibly others (for example bidirectional computation [5] or blame/contracts [9])

Models of provenance used in databases address a special class of computations (and changes that “subtract information”) which make it possible to obtain nice properties, such as the homomorphism commutation property in the semiring model. However, such techniques are relatively fragile with respect to extensions: for example to handle negation, we need to generalize the semiring model in one direction, to handle aggregation, we need to generalize it in another direction, etc. This is analogous to the problems of denotational semantics in classical programming language theory, while modern language researchers often use operational techniques that are easier to combine but arguably more ad hoc. Thus, models of provenance can be developed based on an operational notion of trace which simply records everything that (seemingly) makes sense to record during execution, in a form that can be processed later.

The main challenges for using detailed traces as provenance are:

- Recording control-flow and both control and data dependence relationships linking parts of the input to parts of the program or output in a clean way.
- Defining principled forms of slicing or transformations on traces.
- Identifying good tradeoffs between performance and precision, for example through abstraction or slicing on traces.
- Developing provenance models suitable for high-level explanation for non-technical users, for example users of scientific programming languages such as R.

6.1.3 Provenance in Scientific Workflows

In database queries or when using program slicing, computations may be statically analyzed (at least partially) and thus can be seen as “white box” operations (i.e., one can “see” inside

of them and analyze the operations). Scientific workflows often correspond to “grey boxes”, having some parts that can be seen and analyzed (e.g., the workflow structure or “wiring” itself), but also many other parts that are “black boxes”, i.e., existing third-party services or applications whose code and inner workings are often unknown. On the other hand, scientific workflow systems provide a controlled execution environment with various opportunities to capture detailed provenance information at runtime. The workflow execution models of systems differ widely however, leading to challenges when trying to interpret or interoperate workflow provenance information. Some related questions include:

- Is there a common core that underlies different models of computation across workflow systems and scripting languages?
- Can we enhance runtime provenance recorded by workflow systems with compile-time knowledge about the given model of computation and provenance (cf. [6])?
- In what sense does one model give “more detailed” provenance than another model? And can we find meaningful, formal mappings between different models?
- Is it possible to discern, given a provenance trace, whether the trace could have been produced by one variant of a workflow, but not another (e.g., see [8]), or by one workflow system (e.g., Kepler) but not another (e.g., Taverna)?
- More generally, can we formalize the (provenance) semantics of different workflow systems, define key properties such as “reproducibility” or “replayability”, and prove that given systems do or do not enjoy them?
- Can traces or semantics for concurrent languages be adapted to support modeling and reasoning about provenance?

References

- 1 U. A. Acar, A. Ahmed, J. Cheney and R. Perera. A core calculus for provenance. In *POST*, pages 410–429. Springer-Verlag, 2012.
- 2 U. A. Acar, G. E. Blelloch, and R. Harper. Adaptive functional programming. *ACM Trans. Program. Lang. Syst.*, 28(6):990–1034, 2006.
- 3 Y. Amsterdamer, D. Deutch, V. Tannen: Provenance for aggregate queries. *PODS 2011*: p. 153–164.
- 4 Y. Amsterdamer, D. Deutch, V. Tannen: On the Limitations of Provenance for Queries With Difference. *TAPP 2011*.
- 5 A. Bohannon, J. N. Foster, B. C. Pierce, A. Pilkiewicz, and A. Schmitt. Boomerang: resourceful lenses for string data. In *POPL*, pages 407–419. ACM, 2008.
- 6 S. Bowers, T. McPhillips, and B. Ludäscher. Declarative rules for inferring fine-grained data provenance from scientific workflow execution traces. In *Intl. Provenance and Annotation Workshop (IPAW)*, Santa Barbara, 2012.
- 7 J. Cheney, S. Chong, N. Foster, M. Seltzer, and S. Vansummeren. Provenance: A future history. In *OOPSLA Companion (Onward! 2009)*, pages 957–964, 2009.
- 8 S. Dey, S. Köhler, S. Bowers, and B. Ludäscher. Datalog as a lingua franca for provenance querying and reasoning. In *Workshop on the Theory and Practice of Provenance (TaPP)*, Boston, 2012.
- 9 C. Dimoulas, R. B. Findler, C. Flanagan, and M. Felleisen. Correct blame for contracts: no more scapegoating. In *POPL*, pages 215–226, New York, NY, USA, 2011. ACM.
- 10 T. J. Green, G. Karvounarakis, V. Tannen: Provenance semirings. *PODS 2007*: p. 31–40.
- 11 F. Geerts, A. Poggi: On database query languages for K-relations. *J. Applied Logic* 8(2): p. 173–185 (2010)
- 12 T. J. Green, Z. G. Ives, V. Tannen: Reconcilable Differences. *Theory Comput. Syst.* 49(2): p. 460–488 (2011).

- 13 P. Shroff, S. F. Smith, and M. Thober, “Dynamic dependency monitoring to secure information flow,” in *CSF*. IEEE, 2007.
- 14 M. Weiser. Program slicing. In *ICSE*, pages 439–449, 1981.

6.2 Provenance, security, and confidentiality

Reported by Adriane Chapman, Ashish Gehani, Andrew Martin, and Steve Zdancewic, summarizing discussion by other participants

The discussions of provenance and security covered a number of sub-topics, each with open problems, including confidentiality, integrity, completeness, threat models, and regulation.

6.2.1 Threat models and formalization

Many researchers (including several workshop participants) are developing mechanisms for securing provenance in different systems (e.g., [2, 6, 10]). Sometimes, these mechanisms are straightforward adaptations of standard protection mechanisms (cryptography, digital signatures) to provenance viewed as data. Often, however, the nature of the provenance information makes additional attacks possible — which we may call *provenance-specific attacks* or *provenance failures* [7]. For example, knowing that a particular graph is provenance generated by a known workflow may enable inferences that allow an attacker to guess parts of the graph that were redacted. Definitions of key security properties such as disclosure and obfuscation [1] or privacy for workflow provenance [5] provide a foundation for understanding provenance-specific attacks, on which we can build provably correct policies or mechanisms for securing provenance in different settings (for example for general-purpose programming languages [1]). However, there is currently little recognition of these problems in the formal security world ([1], the first paper on formal foundations for provenance security, appeared only last year) and thus there is little interaction between theory and practice of provenance security.

Moreover, there is currently little work on threat models for provenance, that is, identifying what we believe an attacker can or cannot do and what we want to prevent them from doing. Again, a key issue is identifying how provenance-specific attacks differ from generic attacks on systems or protocols that may happen to involve provenance. Specifically, work on information flow, auditing, integrity, and tracing is relevant, as is work on provenance in concurrency models.

Open problems:

- connecting the practical provenance security mechanisms being deployed in systems with the foundational notions of correctness or security for provenance,
- developing threat models for provenance,
- identifying aspects that make provenance security different from simply securing the underlying data.

6.2.2 Confidentiality

Consider the problem of protecting patient data including its provenance. Naturally, the raw data and provenance can be protected using standard access control or privacy/anonymity techniques (the latter is, of course, already a very hard problem). However, when provenance is also involved, we need to ask why provenance protection is different from the standard problems of protecting confidential data.

For the common case where the provenance is represented as a graph, access control policies on nodes and edges can be established, that limits access to the base information [3, 14]. However, knowing constraints on the structure of the underlying graph (for example knowing that a graph was generated by a known workflow) can make it possible for attackers to infer more information. Similarly, anonymization techniques for graph data in social networks suggests that knowledge of the graph structure can weaken security [1, 4, 11, 12, 16, 17]. However, we cannot assume that the constraints on provenance graphs are secret. Thus, there is a basic tradeoff between confidentiality and utility of provenance.

Open problems:

- Adapting notions of disclosure and obfuscation to provenance graphs
- Understanding the common constraints on provenance graphs and developing policy languages that express common confidentiality requirements
- Identifying limits on safe release of provenance information

6.2.3 Integrity and Completeness

The more value perceived about the data (and its provenance), the greater the motivation for attack. It is not worth protecting a 99 cent piece of data with a \$99 protection strategy. Broadly, integrity has several facets: protecting information from alteration by unauthorized users, being able to prove that information is valid (e.g. has not been changed since creation by an authorized user), and being confident that the information is complete (or at least, that you know how complete it is), for example to detect when changes to source data invalidate other data.

For the first problem, existing techniques such as digital signatures or trusted hardware modules (TPM) may help, as with protecting the integrity of ordinary data. For provenance, it may be more important to provide verifiable links between versions of the same data [10]. In some settings, write-once, read-many (WORM) storage offers a capability to record data that is provably unchanged over time (available as a commodity product).

For the second, being able strongly to tie a (certain version of) the software (and contextual libraries, etc.) to a particular data item apparently generated by that software, is desirable. Techniques of watermarking achieve this well in certain contexts; an approach using the “chain of trust” associated with the TPM is also an active research area [13]. In addition, watermarking has been applied to provenance for video data [7] and sensor data [1].

For the third, the issue of completeness of provenance, a motivating example is a researcher who is discovered to have falsified some results (e.g. the South Korean cloning researcher case a few years ago). Other researchers may have used these results or raw data and now this work needs to be revisited as well. This issue crosses over to the social aspects of provenance surrounding what are you providing, what are the risks, and the benefits. It is also related to the discussion of formal models of provenance (e.g. completeness of traces, reproducibility).

Open problems:

- How can digital signatures, TPMs, WORM storage or other basic mechanisms be combined to ensure provenance is protected from unauthorized alteration? Is it just a matter of protecting the provenance “as data” or is further work needed for different forms of provenance?
- How can we provably certify (or audit) provenance? Can standard watermarking or steganography techniques be used or are new techniques needed? How do the incentives and capabilities to falsify provenance differ from those of ordinary data?
- For the purposes of security, what are appropriate definitions of completeness for provenance?

6.2.4 Regulation

There are many, and often conflicting, laws and regulations regarding provenance. In some cases, the law is specifically concerned with protection of citizens/patients, such as HIPAA and the European Data Protection Directive. These regulations encourage not keeping any, or very little, provenance information because it increases the risk of exposure and attack. On the other hand, some laws, such as Sarbanes-Oxley, facilitate attacks because they require organizations to keep everything.

Open questions:

- How can we ensure that provenance security models and mechanisms are appropriate fits for legal regulations?
- Can provenance techniques provide legally admissible evidence that regulations have or have not been met?

References

- 1 L. Backstrom, C. Dwork, and J. Kleinberg: Wherefore Art Thou R3579X? Anonymized Social Networks, Hidden Patterns, and Structural Steganography, WWW, 2007.
- 2 B. Blaustein, A. Chapman, L. Seligman, M. D. Allen, and A. Rosenthal: Surrogate Parenthood: Protected and Informative Graphs, PVLDB, 2010.
- 3 A. Chebotko, S. Chang, S. Lu, F. Fotouhi, and P. Yang: Scientific Workflow Provenance Querying with Security Views, WAIM, 2008.
- 4 G. Cormode, D. Srivastava, S. Bhagat, and B. Krishnamurthy: Class-based graph anonymization for social network data, PVLDB, vol. 2, 2009.
- 5 S. Davidson, S. Khanna, T. Milo, D. Panigrahi, and S. Roy: Provenance Views for Module Privacy, PODS, 2011.
- 6 A. Gehani and U. Lindqvist, Bonsai: Balanced Lineage Authentication, 23rd Annual Computer Security Applications Conference (ACSAC), IEEE Computer Society, 2007.
- 7 A. Gehani and U. Lindqvist, VEIL: A System for Certifying Video Provenance, 9th IEEE International Symposium on Multimedia (ISM), 2007.
- 8 A. Gehani, B. Baig, S. Mahmood, D. Tariq, and F. Zaffar: Fine-Grained Tracking of Grid Infections, 11th ACM/IEEE International Conference on Grid Computing (GRID), 2010.
- 9 A. Gehani and M. Kim, Mendel: Efficiently Verifying the Lineage of Data Modified in Multiple Trust Domains, 19th ACM International Symposium on High Performance Distributed Computing (HPDC), 2010.
- 10 R. Hasan, R. Sion, M. Winslett, The Case of the Fake Picasso: Preventing History Forgery with Secure Provenance, 7th USENIX Conf. on File and Storage Technologies (FAST), 2009.
- 11 M. Hay, G. Miklau, D. Jensen, D. Towsley, and P. Weis: Resisting Structural Identification in Anonymized Social Networks, VLDB, 2008.
- 12 K. Liu and E. Terzi, Towards Identity Anonymization on Graphs, SIGMOD, 2008.
- 13 J. Lyle and A. Martin, Trusted Computing and Provenance: Better Together, in proceedings TaPP'10, USENIX.
- 14 A. Rosenthal, L. Seligman, A. Chapman, and B. Blaustein: Scalable Access Controls for Lineage, in Theory and Practice of Provenance, 2009.
- 15 J. Zhang, A. Chapman, and K. LeFevre: Fine-Grained Tamper-Evident Data Pedigree, Secure Data Management, 2009.
- 16 E. Zheleva and L. Getoor: Preserving the Privacy of Sensitive Relationships in Graph Data, PinKDD, 2007.
- 17 B. Zhou and J. Pei: Preserving Privacy in Social Networks Against Neighborhood Attacks, ICDE, 2008.

6.3 Social Aspects of Provenance

Reported by Carole Goble and Jim Frew, summarizing discussions involving Shawn Bowers, Kai Eckert, Paul Groth, Luc Moreau, Perdita Stevens, Jun Zhao, and other participants

Provenance discussions have typically been couched in terms of benefit to the consumer. Anecdotally, users are enthusiastic about provenance becoming available to them but less obliging about supplying provenance on their data to others. At the seminar, a discussion group covered a wide range of issues concerning the rewards, risks, burdens, and benefits of provenance; how these relate to technical requirements or proposals; and how to evaluate whether current or future solutions address these needs (and are worth the costs).

The example of traceability in software engineering gives cause for concern: despite a large amount of research on the subject, experience in the field suggests that the benefits of adopting traceability techniques may not outweigh their costs.

The discussion group produced a substantial outline which (together with other materials in this report and on the seminar wiki) may form the basis for a longer “manifesto” paper by participants in the seminar. The following discussion of open problems is distilled from that outline.

6.3.1 Rewards, risks, burden, benefit of provenance

Part of the problem of identifying the rewards, risks, burdens, and benefits of provenance is terminological: people disagree on what provenance is, and whether it “is” metadata, trust, quality or identity information, or just a record of this information. The working group identified the different needs and goals of provenance consumers and producers.

- How can we untangle confusion among provenance, metadata, trust, quality, and identity?
- How can we develop infrastructure that provides “stealth/ninja provenance” – merging into existing information infrastructure.
- How can we design appropriate provenance capture mechanisms based on (clear understanding of) what we, or users, will eventually want to use it for?

6.3.2 Technical requirements and capabilities

The group also identified a lifecycle for provenance production: capture, preparation, sharing, and using, and identified benefits and risks for producers and consumers of provenance.

- How can we design mechanisms that take into account the motivations (and demotivations) on provenance producers (voluntary, peer pressure, mandatory) and different classes of consumers (self, friends, family/colleagues, public)?
- Likewise, how can we develop systems that take into account the different stages of production (raw data, preliminary results, polished results, publication)?
- How do we reconstruct or complete provenance when it was not originally captured?

6.3.3 Evaluation

Finally, the group produced a draft checklist for projects or tool providers to characterize what aspects of provenance they do or do not handle. This could serve as a basis for comparison of different techniques, offsetting economic costs considerations.

- How can we evaluate compliance with a collection of requirements on provenance systems?

- How would the costs/benefits of provenance be affected by developing standards or infrastructure that provides it pervasively, rather than in heterogeneous ways in different systems? Is it worth it?

6.3.4 Pointers to the literature

Much of the discussion can be framed by the literature in data sharing and collaboration behaviours in knowledge enterprises and scientific communities [9, 11, 8, 3, 6, 1, 5]. There is also a useful literature (partially covered by the above) examining the incentives, behaviour patterns, models and quality of voluntary information. Additional references include [2, 10]. The whole area of motivations to contribute wikis is a useful area to look at (e.g. [7]). Jane Hunter had previously highlighted sensitivities around the publishing of provenance and a desire to “provenance spring clean” [4].

References

- 1 Borgman, C.L. The Conundrum of Sharing Research. *Journal of the American Society for Information Science and Technology*, 1–40 (2011)
- 2 Andrew J. Flanagan and Miriam J. Metzger, The credibility of volunteered geographic information. *GeoJournal* (2008) 72:137–148
- 3 Howison, J., Herbsleb, J.D. Scientific software production: incentives and collaboration. *Proc ACM 2011 Conf Computer Supported Cooperative Work*, 513–522 (2011)
- 4 Hunter, J. Scientific Publication Packages – A Selective Approach to the Communication and Archival of Scientific Output., *Intl J of Digital Curation* 1 (1) (2006)
- 5 Liebowitz, J., Ayyavoo, N., Nguyen, H., Carran, D., Simien, J. Cross-generational knowledge flows in edge organizations. *Industrial Management & Data Systems*, 107(8) 1123–1153 (2007)
- 6 Nielson, M. *Reinventing Discovery: The New Era of Networked Science*. Princeton University Press (2011)
- 7 Stacey Kuznetsov. 2006. Motivations of contributors to Wikipedia. *SIGCAS Comput. Soc.* 36, 2, Article 1 (June 2006). See also: <http://www.staceyk.org/personal/WikipediaMotivations.pdf>
- 8 Stodden, V. The Scientific Method in Practice: Reproducibility in the Computational Sciences. MIT Sloan Research Paper No. 4773-10. doi:10.2139/ssrn.1550193 (2010)
- 9 Tenopir C, Allard S, Douglass K, Aydinoglu AU, Wu L, et al. (2011) Data Sharing by Scientists: Practices and Perceptions. *PLoS ONE* 6(6): e21101. doi:10.1371/journal.pone.0021101
- 10 Wikipatterns.com: a toolbox of patterns & anti-patterns, and a guide to the stages of wiki adoption. <http://www.wikipatterns.com/display/wikipatterns/Wikipatterns>
- 11 Yakowitz, J. Tragedy of the Data Commons. *Harvard J of Law and Tech*, Vol. 25 (2011)

Participants

- Umut A. Acar
MPI for Software Systems –
Kaiserslautern, DE
- Shawn Bowers
Gonzaga Univ. – Spokane, US
- Peter Buneman
University of Edinburgh, GB
- Adriane Chapman
MITRE – McLean, US
- James Cheney
University of Edinburgh, GB
- Stephen Chong
Harvard University, US
- Sarah Cohen-Boulakia
Université Paris Sud, FR
- Victor Cuevas-Vicenttin
St. Martin-d’Heres, FR
- Lois Delcambre
Portland State University, US
- Kai Eckert
Universität Mannheim, DE
- Nate Foster
Cornell University, US
- Juliana Freire
Polytechnic Institute of NYU –
Brooklyn, US
- James Frew
University of California – Santa
Barbara, US
- Irimi Fundulaki
FORTH – Heraklion, GR
- Daniel Garijo
Universidad Politécnica de
Madrid, ES
- Floris Geerts
University of Edinburgh, GB
- Ashish Gehani
SRI – Menlo Park, US
- Carole Goble
University of Manchester, GB
- Todd J. Green
University of California – Davis
and LogicBlox, US
- Paul Groth
Free Univ. – Amsterdam, NL
- Torsten Grust
Universität Tübingen, DE
- Olaf Hartig
Humboldt Univ. zu Berlin, DE
- Melanie Herschel
Université Paris Sud, FR
- Bertram Ludaescher
Univ. of California – Davis, US
- Andrew Martin
University of Oxford, GB
- Simon Miles
King’s College – London, GB
- Paolo Missier
Newcastle University, GB
- Luc Moreau
University of Southampton, GB
- Leon J. Osterweil
University of Massachusetts –
Amherst, US
- Christopher Re
University of Wisconsin –
Madison, US
- Vladimiro Sassone
University of Southampton, GB
- Martin Schäler
Universität Magdeburg, DE
- Margo Seltzer
Harvard University, US
- Christian Skalka
University of Vermont, US
- Perdita Stevens
University of Edinburgh, GB
- Wang-Chiew Tan
IBM Research & University of
California – Santa Cruz, US
- Jan Van den Bussche
Hasselt University, BE
- Stijn Vansummeren
Université Libre de Bruxelles, BE
- Marianne Winslett
Univ. of Illinois – Urbana, US
- Steve Zdancewic
University of Pennsylvania, US
- Jun Zhao
University of Oxford, GB

