Issues in Building Practical Provenance Systems

Adriane Chapman and H.V. Jagadish
University of Michigan
Ann Arbor, MI 48109
{apchapma, jag}@umich.edu

Abstract

The importance of maintaining provenance has been widely recognized, particularly with respect to highly-manipulated data. However, there are few deployed databases that provide provenance information with their data. We have constructed a database of protein interactions (MiMI), which is heavily used by biomedical scientists, by manipulating and integrating data from several popular biological sources. The provenance stored provides key information for assisting researchers in understanding and trusting the data. In this paper, we describe several desiderata for a practical provenance system, based on our experience from this system. We discuss the challenges that these requirements present, and outline solutions to several of these challenges that we have implemented. Our list of a dozen or so desiderata includes: efficiently capturing provenance from external applications; managing provenance size; and presenting provenance in a usable way. For example, data is often manipulated via provenance-unaware processes, but the associated provenance must still be tracked and stored. Additionally, provenance information can grow to outrageous proportions if it is either very rich or fine-grained, or both. Finally, when users view provenance data, they can usually understand a SELECT manipulation, but "why did the bcgCoalesce [1] manipulation output that?"

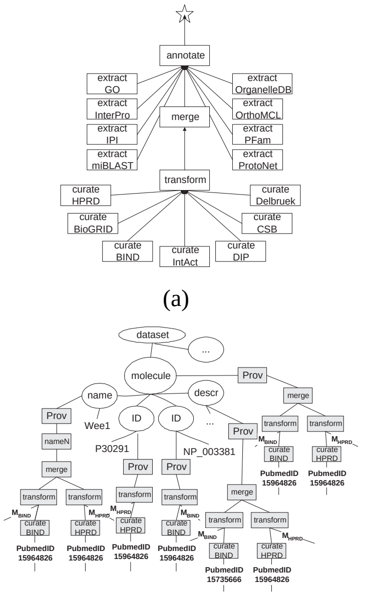
1 Introduction

Once upon a time, there lived a beautiful (and highly intelligent) researcher. She had a sad life chained to her lab bench day and night, slaving for her evil Principal Investigator, collecting data and analyzing numbers. One day a handsome Computer Scientist heard of the researcher's plight and decided to save the damsel in distress. First he built a program that would measure signal intensity in her experiments. Many more programs followed, each designed to reduce the tasks performed by the beautiful (and highly intelligent) researcher. The handsome Computer Scientist dazzled the evil Principal Investigator with the power of his programs and rescued the fair researcher from her lab bench. Just as they were about to ride into the sunset, the evil Principal Investigator popped her warty green face out of the tower and said, "I think you better come back in, I don't understand how or where you got these numbers, but they certainly can't be correct."

The handsome Computer Scientist laughed and cried, "Slaying this dragon will be easy. I maintained provenance!" Unfortunately, the provenance the handsome Computer Scientist kept was coarse-grained and not easy

Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

[0] Supported in part by NSF grant number IIS 0741620 and by NIH grant number U54 DA021519.



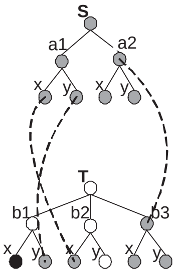
(a)

(b)

Figure 1: 1(a) The workflow used to generate MiMI. 1(b) A data item with provenance from MiMI.

```
Insert x into T/b2
Copy S/a1/y into T/b1/y
Insert y into T/b2
Copy S/a2 into T/b3
Copy S/a1/x into T/b2/x
```

(a)



(b)

Figure 2: 2(a) The user's actions on S and T. 2(b) The provenance links from T to S. Nodes originally in T are white; inserted nodes are black and copied nodes are grey to distinguish user actions.

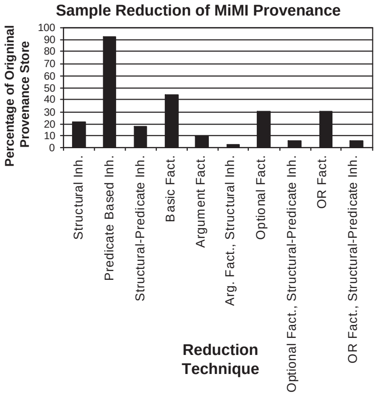


Figure 3: The storage savings for a set of reduction techniques applied to MiMI.

to query with the data itself. The handsome Computer Scientist and beautiful (and highly intelligent) researcher spent the remainder of their lives toiling to understand what happened to the data. *The End.*

The moral of this bedtime story: Don't just maintain provenance, maintain *good* provenance. Knowing that we should store provenance information doesn't mean we actually can, or do, or do it correctly. Even outside of fairy tales, researchers and scientists still have difficulty understanding what happened to their data, particularly when the data is heavily manipulated.

We have constructed a database of protein interactions, MiMI [16], by manipulating and integrating data from several popular biological sources. Figure 1(a) contains the general workflow used to generate MiMI. As scientists used the data in MiMI, it became apparent that provenance was needed to assist them in understanding and trusting the data presented. A snapshot of provenance information captured in MiMI is shown in Figure 1(b). While watching researchers use provenance information, we realized that their provenance information needs more than just a simple capture-store-fileaway approach. In Section 2, we present both required and recommended features for a database system incorporating provenance information based upon our experience with MiMI. Section 3 describes current provenance systems in light of these desiderata. In Sections 4 and 5 we discuss practical implementation options and conclude.

2 Desiderata

In this section, we will outline a set of features, required and recommended, needed for a database to incorporate provenance.

2.1 How Much Provenance to Capture

I. Granularity Choice *(Required)* Allow provenance to be captured and stored at every granularity. Currently there are two main trends for attaching provenance information: coarse and fine grained. Many workflow systems that generate provenance records attach provenance information at the coarse-grained file level [2, 12, 13, 14, 15]. Other specialized systems attach provenance at the fine grain of attribute [4, 6, 7, 8, 11]. Often, however, systems need a mix of usage. For example, in MiMI, provenance is attached to files, data items and attributes, as shown in Figure 1(b). When attributes, files and data items are broken up or used out of context, provenance is especially important at every granularity.

II. Exact Execution Provenance *(Required)* Record the exact provenance for each specific data item, not just the general provenance for a "class" of items. For example, attributes and data items within files behave differently through a given workflow based on data/attribute type, content, etc. The workflow to generate MiMI is shown in Figure 1(a). If a scientist wishes to know where the Wee1 name attribute came from, pointing to the workflow used is not enlightening, since via the workflow, that attribute could have come from any number of external sources, e.g. BIND, HPRD, etc. Instead, we wish to know that the Wee1 name attribute came from BIND and HPRD, while the P30291 ID attribute came only from HPRD. Moreover, while the MiMI.xml *file* went through a merge process, the P30291 ID attribute never merged with any other information.

III. Provenance Information *(Required)* Permit variation of the form or content of the provenance information. Current provenance systems capture a huge range of information from information about the files used and produced and the scripts run [2, 12, 13, 14, 15] to user annotations [3, 19]. But what exactly is needed to allow individuals to utilize the data? In MiMI, we found storing a mix of provenance information the most successful. For instance, HPRD describes each protein in an XML file, and MiMI's provenance should reference the exact XML file used. On the other hand, user annotations, such as the PubMedID (a unique identifier for biology research articles) used to garner the original information should also be kept. In other words, a provenance system should be flexible enough to store a large range of information as determined by the application.

IV. Capturing Non-automated processes *(Required)* Provide the ability to capture manipulations that are performed outside of automated workflows. While capturing the exact execution for every file, data item and attribute, it is imperative not to miss the actions performed manually by a curator. For instance, in MiMI, because the identity functions that dictate which proteins to merge are generated automatically, an expert user will find a mistake occasionally. The manual correction of this mistake must be reflected in the provenance records. Automatic capture of workflows alone is not enough.

2.2 Systems Issues

V. Source Data Item Identity *(Required)* Keep track of your incoming data. No matter what information is ultimately retained in the data set or provenance store, there must always be a firm, unbending representation for data item identity. Consider the problem in MiMI: 232,680 proteins from seven sources are merged into 117,549 proteins. When the merge process takes place, how do you identify the original components and where they came from? How do you go backwards to look at the original proteins? Even specifying that a protein is from BIND is not enough, since several proteins from BIND can be merged into one. You cannot trace back any further without some notion of source data item identity.

VI. Provenance Storage Size *(Required)* Plan for large provenance store costs. Given the amount of provenance material stored, provenance stores can grow to immense sizes, and easily outstrip the size of the data. MiMI is 270MB; the associated provenance store is 6GB before compression.

VII. Manipulation Information *(Recommended)* Maintain detailed manipulation information. Most provenance systems keep track of the scripts or manipulations that have been applied to the data. Some, such as [14], allow users to modify process order, and change applications to achieve the desired results. However, this requires an innate knowledge of each process, such as *SELECT* or *bcgCoalese* [1]. An alternative approach would be to maintain information to generate result explanations. Thus, when a user asks, “Why did P30291 from HPRD merge with NP_003381 from BIND?” no innate knowledge of the merge process is required for the answer; it can be automatically derived based on information in the provenance store.

VIII. Inter-system Provenance *(Recommended)* Build toward inter-operability of provenance systems. As systems grow and become interconnected, provenance should be interchangeable. As MiMI has grown in popularity, it has become a reference to other applications such as PubViz [22]. These systems also attempt to maintain some notion of provenance. However, they should not be required to store provenance information found in MiMI. Instead, they should store the provenance associated with their actions, then point to MiMI for the provenance beyond their borders.

2.3 Usability

IX. User Interactions *(Required)* Allow users to actively utilize provenance information at many levels. As discussed previously, there can be a huge amount of provenance information to trawl through. This information should not be stored away out of sight in case there is major problem, it should be available to end users. How can an end user navigate this deluge of information? In MiMI, we have noticed that user’s needs fall into several categories: dataset generation overview, data item overview and particular manipulation overview. Users should be able to see provenance information at many levels.

X. Provenance Queriability with the Data *(Required)* Provide support for querying provenance and data together. Provenance is an essential component in assisting end users in trusting and using the data. To this end, the provenance information should be queriable with the data itself. In MiMI, queries often consist of intersections of data and provenance. For instance, “Return all molecules located in the mitochondria *(data)* that were reported by HPRD or IntAct *(provenance)*.” Making the provenance records available, but forcing users to do a processing step to join them with the data is an undue burden.

XI. Error Finding and Fixing *(Recommended)* Enable easy provenance store maintenance. Consider the following scenario: a user queries MiMI, and notices two molecules have been merged that should not have been. The user reports it. What happens? Hopefully the error will be corrected and the two mis-merged proteins will be separated. But what about the provenance store? Mechanisms must be in place to incrementally update the provenance store to allow for error finding and fixing.

3 Current Provenance Systems

There are several provenance systems that have been applied to real scientific data [5, 21], and espouse many of the desiderata discussed above. The PASOA project [15, 17] has been applied to several real-world scientific endeavors. It is concerned with the origins of a result or determining when results are invalid and has paid specific attention to desiderata VII, VIII and X. Chimera [12], is concerned with data derivation and shines in desiderata V, VI and VIII. Additionally, myGRID [13] is a collaborative environment for scientists with provenance handling; myGRID handles desiderata VIII and IX. Other workflow systems have integrated provenance information such as VisTrails [14], Redux [2], and those participating in the International Provenance and Annotation Workshop

Challenge [18]. In general these systems are thoughtful of VI and IX. However, workflow based systems so far fail in desiderata II, III, and IV.

Outside of workflow-based systems, very few database provenance systems have been applied to real-world scientific problems. However, we would like to highlight several systems that satisfy various desiderata. First, Trio [19] fulfills the notions for desiderata II, V and X very well. In [3], problems with desiderata III, V and X are explored. [20] are working on desideratum IV. Also, [7, 8] take an interesting look at desiderata II and V. Finally, [11] and [10] are both tackling desideratum IX.

4 Finding Practical Solutions

Each desideratum discussed above is a challenge to satisfy. In this section, we suggest how two of these challenges can be met.

4.1 Capturing Non-automated Processes

Human curators are often responsible for the content of specialized databases, or for “tweaks” in existing automated systems. The Uniprot consortium employs more than seventy scientists for curation. In some curated databases, the database designer augments the schema with provenance fields for the curator to populate; in “tweaked” systems, the actions often go unrecorded.

Using an appropriate architecture, and a language to express user actions, it is possible to capture these non-automated processes. By forcing a user to manipulate the database through a program that can track his movements, the user’s actions can be distilled into Copy, Insert and Delete. Once we know (1) what action is occurring, (2) where in the current database the action is occurring, and (3) where any incoming data is from, we can effectively store provenance on the user’s edits. Figures 2(a)–2(b) show an example series of user edits and their record in the provenance store. For further details, please refer to [6].

4.2 Reducing the Provenance Store

As stated in Desideratum VI, provenance information can balloon to gargantuan sizes. Utilizing features of the provenance store, it is possible to perform some reduction. A family of Factorization algorithms and two distinct Inheritance algorithms can reduce the provenance store by up to a factor of 20. Using Factorization, by breaking provenance records down into smaller pieces, it is possible to decrease repeated information. Using Inheritance, properties of the dataset are used to reduce the storage needed for the provenance. Figure 3 shows the ability of these algorithms to compress the provenance space needed for MiMI. Details of the algorithms and experiments can be found in [9].

5 Conclusions

The benefits of maintaining provenance are already apparent. Based on our experience with MiMI, we outline several desiderata that the next generation of provenance systems should meet. We outline some of the challenges in meeting these desiderata and suggest some directions to find a solution.

References

[1] J. Annis, Y. Zhao, J. Voeckler, M. Wilde, S. Kent, and I. Foster. Applying chimera virtual data concepts to cluster finding in the Sloan Sky Survey. *IEEE*, 2002.

[2] Roger S. Barga and Luciano A. Digiampietri. Automatic capture and efficient storage of escience experiment provenance. In *Concurrency and Computation: Practice and Experience*, 2007.

[3] Deepavali Bhagwat et al. An annotation management system for relational databases. In *Proc. of the Intl. Conf. on Very Large Data Bases (VLDB)*, pages 900–911, 2004.

[4] R. Bose and J. Frew. Composing lineage metadata with XML for custom satellite-derived data products. In *SSDBM*, pages 275–284, 2004.

[5] Rajendra Bose and James Frew. Lineage retrieval for scientific data processing: a survey. *ACM Comput. Surv.*, 37(1):1–28, 2005.

[6] Peter Buneman, Adriane Chapman, and James Cheney. Provenance management in curated databases. In *ACM SIGMOD*, pages 539–550, June 2006.

[7] Peter Buneman, James Cheney, and Stijn Vansummeren. On the expressiveness of implicit provenance in query and update languages. In *ICDT*, pages 209–223, 2007.

[8] Peter Buneman, Sanjeev Khanna, and Wang-Chiew Tan. Why and Where: A characterization of data provenance. In *ICDT*, pages 316–330, 2001.

[9] Adriane Chapman, H.V. Jagadish, and Prakash Ramanan. Efficient provenance storage. in submission, 2008.

[10] Kwok Cheung and Jane Hunter. Provenance Explorer - customized provenance views using semantic inferencing. In *International Semantic Web Conference*, pages 215–227, 2006.

[11] Shirley Cohen, Sarah Cohen Boulakia, and Susan Davidson. Towards a model of scientific workflows and user views. In *DILS*, pages 264–279, 2006.

[12] Ian Foster, Jens Vockler, Michael Eilde, and Yong Zhao. Chimera: A virtual data system for representing, querying, and automating data derivation. In *International Conference on Scientific and Statistical Database Management*, pages 37–46, July 2002.

[13] Ian Foster, Jens Vockler, M Wilde, and Yong Zhao. The virtual data grid: a new model and architecture for data-intensive collaboration. In *CIDR*, 2003.

[14] Juliana Freire, Claudio T. Silva, et al. Managing rapidly-evolving scientific workflows. In *IPAW*, 2006.

[15] Paul Groth, Simon Miles, and Luc Moreau. Preserv: Provenance recording for services. In *Proceedings of the UK OST e-Science second All Hands Meeting 2005 (AHM'05)*, 2005.

[16] Magesh Jayapandian, Adriane Chapman, et al. Michigan Molecular Interactions (MiMI): Putting the jigsaw puzzle together. *Nucleic Acids Research*, pages D566–D571, Jan 2007.

[17] Simon Miles, Paul Groth, Miguel Branco, and Luc Moreau. The requirements of recording and using provenance in e-science experiments. *Journal of Grid Computing*, 5(1):1–25, 2007.

[18] Luc Moreau, Bertram Ludäscher, et al. The First Provenance Challenge. *Concurrency and Computation: Practice and Experience*, 2007.

[19] Michi Mutsuzaki, Martin Theobald, et al. Trio-One: Layering uncertainty and lineage on a conventional DBMS. In *CIDR*, pages 269–274, 2007.

[20] M Seltzer, J Ledlie, C Ng, D Holland, K Muniswamy-Reddy, and U Braun. Provenance aware sensor data storage. In *NetDB*, 2005.

[21] Yogesh Simmhan, Beth Plale, and Dennis Gannon. A survey of data provenance in e-science. *SIGMOD Record*, 34(3):31–36, 2005.

[22] Weijian Xuan, Pinglang Wang, Stanley J. Watson, and Fan Meng. Medline search engine for finding genetic markers with biological significance. *Bioinformatics*, 23:2477–2484, 2007.