PLUS: Synthesizing Privacy, Lineage, Uncertainty and Security

Barbara Blaustein, Len Seligman, Michael Morse, M. David Allen, Arnon Rosenthal

The MITRE Corporation
7515 Colshire Dr. McLean, VA 22102
{bblaustein,seligman,mdmorse,dmallen,arnie}@mitre.org

***Abstract*—** Privacy, lineage, uncertainty, and security are important to many information integration efforts, and these "PLUS" properties interact in a number of complex ways. This paper presents requirements and use cases for PLUS systems that gracefully handle those interactions. We describe related work, and present the goals of a new research project which is developing a theory and implementation of PLUS systems.

I. Introduction and Vision

When information is combined from diverse sources beyond the confines of a single enterprise, lineage [1] (also called provenance) and uncertainty [2] are essential to help consumers understand if the data should be trusted. In addition, lineage and uncertainty enable derived data to be reconstituted or modified in response to failures or changes in the certainty of base data.

Most prior work has assumed that all users can see all relevant lineage, yet this is often not the case, due to sensitivity of some underlying data resulting from privacy or security concerns. Queries over lineage and uncertainty data must be evaluated subject to privacy and security constraints.

Privacy, lineage, uncertainty, and security interact in a number of complex ways. While lineage and uncertainty interactions are exploited for relational systems [1], interactions among the four properties have not previously been considered. There is also a need to broaden the scope to include non-relational components. In response, we have initiated a project to synthesize related work into a new integrated model, develop a prototype system, and consider the ways in which PLUS properties interact to address realistic use cases.

II. Requirements and Use Cases

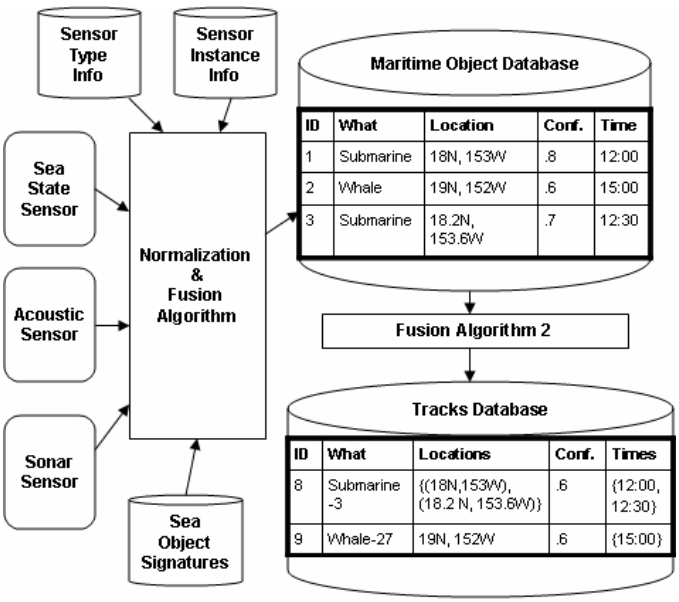
To identify requirements for PLUS systems, we considered a broad range of MITRE's customer applications spanning defense, intelligence, law enforcement, and biomedical information sharing. Common requirements across these domains include:

- *Heterogeneity:* the approach must accommodate relational databases, XML, and monolithic files. In addition, a single data manager cannot be assumed.
- *Lineage with workflows:* derived data often comes from executing complex workflows. Frequently, the PLUS system will have little knowledge of the internals of processes executed as steps in a workflow, in contrast to data derived by executing a database query.
- *Bi-directional lineage traversal:* it is important to reason about both backward ("how was this data derived?") and forward ("which data depends on this?") lineage.
- *Variable granularity*: different component systems will manage data objects and lineage at different levels of granularity (e.g., tuples, tables, or whole databases for relational data, and arbitrary size XML subtrees) .
- *Incomplete disclosure*: PLUS systems must sometimes restrict views of lineage information, due to either privacy or security.
- *Uncertainty*: support is needed for alternatives, confidence, and the possibility that a data object may not exist in the real world.
- *Polyinstantiation*: A database may allow certain users to access only a subset of the data. Updaters may then be unable to see certain values. As a result, they may insert competing values for objects with the same keys [9].

A sensor fusion example, based on [13], illustrates many of these requirements, which are illustrated in Figure 1. Various ocean sensors relay massive quantities of data about the environment (e.g., water temperature, turbulence, noises being recorded, and sonar readings on the position and speed of submerged objects). A workflow takes XML files containing sensor data, performs geo-temporal normalization on the data, and then runs a fusion algorithm that combines the transformed sensor readings with database information on 1) reliability of the different sensor types, 2) maintenance and performance over time of individual sensor instances, and 3) "signatures" of various kinds of objects such as animals and submarines. The algorithm populates a database of observations of maritime objects, each with an associated type

[1] *Lineage* is "information that helps determine the derivation history of a data product…[It includes] the ancestral data product(s) from which this data product evolved, and the process of transformation of these ancestral data product(s), potentially through workflows [16].

[2] *Uncertainty* describes alternatives and confidence in the accuracy of data values as well as the possibility that some data object may not actually be present as a real-world truth.



ID	What	Location	Conf.	Time
1	Submarine	18N, 153W	.8	12:00
2	Whale	19N, 152W	.6	15:00
3	Submarine	18.2N, 153.6W	.7	12:30

ID	What	Locations	Conf.	Times
8	Submarine -3	{(18N,153W), (18.2 N, 153.6W)}	.6	{12:00, 12:30}
9	Whale-27	19N, 152W	.6	{15:00}

Figure 1: Sensor Fusion Workflow

(e.g., whale, submarine) and confidence. A subsequent fusion algorithm clusters these uncertain observations into "tracks"—i.e., movements over time for the different objects. There is uncertainty about the number of real objects and how observations line up with them. Furthermore, security may create multiple views of the data. For example, the existence and capabilities of certain types of sensors may be a closely guarded secret that not all users have the privileges to see, or an experimenter may share data to be seen only after vetting, and then only with users from collaborating laboratories.

In order to better interpret results, users need to be able to query over the data, its lineage, and its uncertainty. For example:

- Which "whale" readings received >30% of their track lineage from sensors that have low confidence?
- Given that a sonar sensor has been found to be miscalibrated, which tracks and subsequent report products are affected?
- How does confidence in the "submarine" identification (ID 3 in the maritime objects database) differ between users with different privileges (e.g., if one were forbidden to see any data emanating from a key sensor)?
- Which sensors are most frequently involved in high confidence assertions in the tracks database?

Biomedical applications provide another motivating use case. The requirements are similar to sensor fusion, but here the data sensitivity concerns are motivated by patient privacy. In each case, users with different privileges may see different versions of the data.

III. Related Work

Stanford's Trio extends the relational model to support relational queries in the presence of both lineage and uncertainty [14, 18] and demonstrates that including lineage permits a cleaner treatment of uncertainty [1]. We seek to extend Trio results to 1) a heterogeneous environment with XML, relational, and file-based components and 2) derivations via complex workflows.

The Trio software has provided an excellent starting point for exploring lineage and uncertainty. While Trio's current prototype only supports "backward" lineage queries (i.e., showing ancestor tuples in a derivation), we were able to rapidly extend it to support forward lineage (i.e., show a tuple's descendants) and to build a graphical visualizer for lineage in either direction.

Research in probabilistic databases in XML [6, 17] provides a basis for moving toward XML. This work demonstrates advantages of XML over the relational model for representing alternative possible world states.

A large and growing corpus of work investigates lineage in scientific workflows [16]. As a starting point, we adopt the formal model of [2] and also plan to adapt their techniques for simplifying lineage query results with user views of workflows. We also find the distinction in [5] between white, black, and grey box lineage to be useful for the design of a federated lineage capability, where white boxes are well-understood processes such as database queries, black boxes are those where only inputs and outputs are known, and grey boxes are black boxes for which additional constraints on inputs and outputs have been specified.

Privacy and security have been extensively studied in data management. PLUS systems will undoubtedly benefit from the application of anonymization techniques, such as [12]. There has been extensive research on polyinstantiation in the relational model [9, 19]; we believe the modeling will be much easier in a system that uses XML to deal with alternative values. Our contribution will be to study interactions of privacy and security with lineage and uncertainty.

IV. PLUS Project and Prototype

MITRE's newly initiated PLUS project has two primary goals:

- To develop an integrated model as a foundation for PLUS systems. The model will support analysis of interactions and trade-offs.
- To build a prototype system that addresses the range of user needs outlined above.

A. Developing a PLUS Model

Our first task is to develop a formal model for PLUS systems that meet the requirements described in Section 2. We expect the model to be largely a synthesis of existing work, especially ULDBs [1], scientific workflows [5, 7], probabilistic XML databases [6, 17], and XML security [4, 10].

Based on four of the requirements listed above—uncertainty, polyinstantiation, lineage traversal, and heterogeneity—our initial choice is to extend an XML data model. First, XML provides more natural modelling of alternatives than in relational systems [17]. This is important when there are multiple possible states of the world (e.g.,

Trio's x-tuples and maybe-tuples). This is also useful for handling polyinstantiation, where alternate versions of a fact may be introduced because some users can only see a subset of the data. Second, XML query languages' support for path queries seems to offer an advantage for traversing lineage graphs. Finally, XML offers advantages for meeting the heterogeneity requirement, including the ability to provide PLUS services over heterogeneous sources including XML (both with and without schemas) and relational databases.

The choice of XML raises interesting questions, which the PLUS project is exploring:

- What is necessary to adapt XML query languages to traverse lineage DAGs rather than only trees?
- How do we balance the needs for expressiveness and simplicity in an XML-based PLUS query language? Our starting point will be to define a tractable subset of XQuery, roughly XPath plus closure.
- How can we adapt the prior XML security research to support specification and processing of security and privacy constraints over data that includes both lineage and uncertainty?

An important goal is to create the simplest possible model that meets the majority of our customers' application requirements. We initially focus on augmenting lineage capture and uncertainty modelling for workflow environments.

B. Moving Toward Federation

We observe that Trio's prototype is quite naturally implemented as a richer data model (ULDB) layered on top of a simpler one (relational). This requires mapping queries and results among the two models and augmenting the underlying system with needed additional capabilities (e.g., a richer query language, x-tuples, etc.).

Similarly, we seek to build a PLUS system on top of an XML database which has minimal support for PLUS capabilities. Like Trio, our initial implementation is constructed as a layer over a single database, in this case XML.

However, we note that much of the mapping functionality exists in federated databases, including translating among models and handling functionality that component systems cannot (e.g., uncertainty-aware querying, lineage operators). Given that our customers need distributed, heterogeneous components, we are designing our prototype with an eye toward federation.

In addition to the federation layer, we envision these major components of a PLUS system: a workflow manager, query processor, and wrappers for both data managers and external workflow engines.

When the PLUS system executes a workflow run, possibly with the assistance of external workflow engines, the workflow manager captures lineage information. For workflows run without the PLUS workflow manager, components must be sufficiently "lineage-aware"[3] to provide at least information about immediate parents. Open questions include how to incorporate partial information from runs of legacy workflows (e.g., shell scripts), incorporate fine-granularity lineage hints from provenance-aware services in the workflow (e.g., that when passed the string "New York" it accessed information from www.NewYorkState.gov), and how to index and store such lineage information for efficient querying.

The query processor takes queries in an extended XML query language with support for lineage and uncertainty and translates them into queries that component systems can handle. The query processor returns results in accordance with specified security and privacy constraints.

As noted above, the federation layer needs to map between the PLUS system and components' data models. In some cases, the wrapper will also need to provide globally accessible identifiers to support fine-grained lineage. For example, some relational databases do not expose tuple identifiers to client applications. When tuple-grained lineage is desired, the wrapper would need to provide the needed handle, for example through a mechanism like Life Science Identifiers [20].

External workflow engines also need wrappers to facilitate lineage capture. Some may already capture lineage information but require it to be mapped to the PLUS system's lineage representation. In other cases, the lineage capture would not otherwise be done.

C. Research Issues

We now briefly describe research issues we expect to encounter in our development of a PLUS system:

- *Designing the federation layer*. Much of the complexity belongs here. Its interfaces and algorithms require considerable research.
- *Lineage awareness*. How "lineage aware" do component systems need to be to provide different levels of lineage support? How should such support be described to the federation?
- *Uncertainty awareness*. While some workflow steps may be designed to handle uncertainty (i.e., alternate values, "maybe" tuples, possibly with confidences), most will not. Drawing on data integration frameworks, we will explore use cases and coping strategies that suit PLUS systems.
- *Lineage unavailability due to data sensitivity*. When a lineage query cannot be answered due to security or privacy constraints are there special properties of the query that we can exploit to give a meaningful answer?
- *Cache coherency for derived uncertain information*. We will investigate semantics [8] and algorithms for propagating changes to PLUS information in loosely coupled environments.
- *Flexible granularity lineage*. Typical lineage approaches target either very fine-grained or very coarse-grained lineage. Additional work could address gradual degradation of the granularity of lineage information as performance and storage constraints become an issue.

[3] Note that this requirement is intended to be weaker than the notion of "provenance-aware" in [11].

V. Conclusions

We have presented application requirements, described relevant work, and introduced the MITRE PLUS project, which seeks to develop a model for systems that gracefully handle interactions among privacy, lineage, uncertainty, and security in a heterogeneous environment. In addition, we are developing a prototype implementation and will apply it to customer problems, including a more complex version of the sensor fusion problem in Section 2.

References

[1] O. Benjelloun, A. Das Sarma, A. Halevy, and J. Widom, ULDBS: Databases with Uncertainty and Lineage. In *VLDB*, 2006.

[2] O. Biton, S. Cohen-Boulakia, and S. Davidson, Querying and Managing Provenance through User Views in Scientific Workflows. To appear in *ICDE*, 2008.

[3] S. Cohen-Boulakia, O. Biton, S. Cohen, and S. Davidson. Addressing the Provenance Challenge Using ZOOM. Concurrency and Computation: Practice and Experience, 2007.

[4] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati. A Fine-Grained Access Control System for XML Documents. *ACM TISSEC*, May 2002.

[5] S. Davidson, B. Ludaescher, T. McPhillips, S. Bowers, M. Kumar Anand, J. Freire, Provenance in Scientific Workflow Systems. To appear in *Data Engineering*, 2008.

[6] A. de Keijzer and M. van Keulen. *Scalable Uncertainty Management*, volume 4772, chapter Quality Measures in Uncertain Data Management, pp. 104-115. Springer Berlin/Heidelberg, 2007.

[7] I. Foster, J. Vöckler, M. Wilde, and Y. Zhao. Chimera: A Virtual Data System for Representing, Querying, and Automating Data Derivation. In *Scientific and Statistical Database Management*, 2002.

[8] T. Green, G. Karvounarakis, Z. Ives, and V. Tannen, Update Exchange with Mappings and Provenance, *VLDB* 2007.

[9] S. Jajodia, R. Sandhu, B. Blaustein. Solutions to the Polyinstantiation Problem. In *Information Security: An Integrated Collection of Essays*, M. Abrams et al., eds., IEEE Computer Society Press, 1995.

[10] G. Kuper, F. Massacci, and N. Rassadko. Generalized xml security views. *ACM symposium on Access control models and technologies (SACMAT)*, 2005

[11] J. Ledlie, C. Ng, and D. A. Holland. Provanance-Aware Sensor Data Storage. In *ICDE Workshops (ICDEW'05)*, 2005.

[12] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitsubramaniam. l-Diversity: Privacy Beyond k-Anonymity. In *ICDE*, 2006.

[13] C. J. Matheus, D. Tribble, M. M. Kokar, M. G. Ceruti, and S. C. McGirr. Towards a Formal Pedigree Ontology for Level-One Sensor Fusion. In *10th International Command & Control Research and Technology Symposium*, 2005.

[14] M. Mutusuzaki, M. Theobald, A. De Keijzer, J. Widom, P. Agarawal, O. Benjelloun, A. Das Sarma, R. Murthy, and T. Sugihara. Trio-One: Layering Uncertainty and Lineage on a Conventional DBMS. In *CIDR*, 2007.

[15] M. Seltzer, K.-K. Muniswamy-Reddy, D. A. Holland, U. Braun, and J. Ledlie. Provenance-Aware Storage Systems. Harvard University Computer Science Technical Report TR-18-05, Harvard Univ., Cambridge, MA, 2005.

[16] Y. L. Simmhan, B. Plale, and D. Gannon. A Survey of Data Provenance in e-Science. *SIGMOD Record*, vol. 34, pp. 31-36, 2005.

[17] M. van Keulen, A de Keijzer, and W. Alink. A Probabilistic XML Approach to Data Integration. In *ICDE*, 2005.

[18] J. Widom. Trio: A System for Integrated Management of Data, Accuracy, and Lineage. In *CIDR*, 2005.

[19] M. Winslett, K. Smith, X. Qian. Formal query languages for secure relational databases. In *ACM Transactions on Database Systems*, 1994.

[20] http:// lsids.sourceforge.net