A Comprehensive Model for Provenance

Salmin Sultana and Elisa Bertino

Purdue University
{ssultana,bertino}@purdue.edu

Abstract. In this paper, we propose a provenance model able to represent the provenance of any data object captured at any abstraction layer (workflow/process/OS) and present an abstract schema of the model. The expressive nature of the model makes it potential to be utilized in real world data processing systems.

Keywords: Provenance, Model, Comprehensive, Unified, Generic.

1 Introduction

Existing data provenance systems mostly operate at a single level of abstraction at which they record and store provenance. Provenance systems for scientific data [1][2] record provenance at the semantic level of the application. Other application level provenance systems capture provenance at the granularity of business objects, lines of source code or other units with semantic meaning to the context. Workflow systems record provenance at workflow stages and data/message exchange points. System-call based systems [3][4] operate at the level of system processes and files. While provenance collected at each abstraction layer is useful in its own right, integration across these layers is crucial.

To build a unified provenance infrastructure, defining an expressive provenance model able to represent the provenance of data objects with various semantics and granularity is the first crucial step. Such a model should be able to capture data provenance in a structured way as well as to encapsulate the knowledge of both the application semantics and the system. The model should also support provenance queries that span layers of abstraction, including workflow processes, application objects, and system processes. Despite a large number of research efforts on provenance management, only a few provenance models have been proposed. Most of these models conform only to a particular provenance system's data structure. Although a general provenance model has been proposed by Ni et al. [5], its main focus is on access control for provenance. Also this model is not able to distinguish between application and system level provenance information. In this paper, we propose a comprehensive provenance model that is (i) generic to record the provenance of any data object, (ii) unified to capture and integrate both the application and system level metadata, (iii) focused on interoperability among provenance models and integration of provenance across different systems, (iv) tailored to fine grained access control and originator preferences on provenance, and (v) able to facilitate queries for constructing specialized views of provenance graphs.

P. Groth and J. Frew (Eds.): IPAW 2012, LNCS 7525, pp. 243–245, 2012.

2 Provenance Model

Fig. 1 shows the proposed provenance model consisting of entities and the interactions among them. To characterize our model, we define the provenance as:

The provenance of a data object is the history of the actors, process, operations, inter-process/operation communications, environment, access control and other user preferences related to the creation and modification of the data. The relationships between provenance entities form a data provenance graph (DAG).

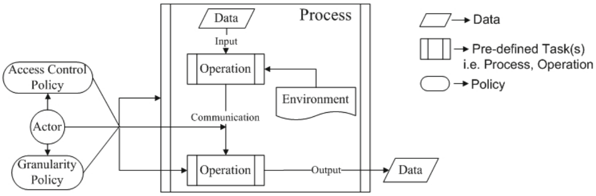

Fig. 1. Proposed Provenance Model

Data creation or manipulation is performed by a sequence of *operation*s initiated by a *process*. A *process*, consisting of a sequence of operations, may be a service/activity in a workflow, a user application, or an OS-level (e.g. UNIX) process. An *operation* executes specific task(s) and causes manipulation to some system or user data. *Communication* represents the interaction (e.g. data flow) between two processes or two operations in a process. Communication between two operations in a process means the completion of an operation following the start of another operation. When the preceding operation results in data, the communication may involve data passing between the operations. The communication may also contain triggers, specific messages, etc. There might be also no explicit message (i.e. communication record) exchange between two operations. An operation may take data as input and output some data. Each data object is associated with a *lineage* record which specifies the immediate data objects that have been used to generate this data. Processes, operations, and communications are operated by *actor*s that can be human users, workflow templates, etc. *Environment* refers to the operational state, parameters, system configurations that also affect the execution of an operation and thus output data.

To address the security and privacy requirements of provenance, we include actor specified *access control policies* that specify whether and how other actors may utilize the provenance records. Since our model can capture the very details of an operation, it might by preferable to allow users to specify the desired level of provenance details. The *granularity policies* allow the users to specify how detailed provenance information they want to be captured and stored.

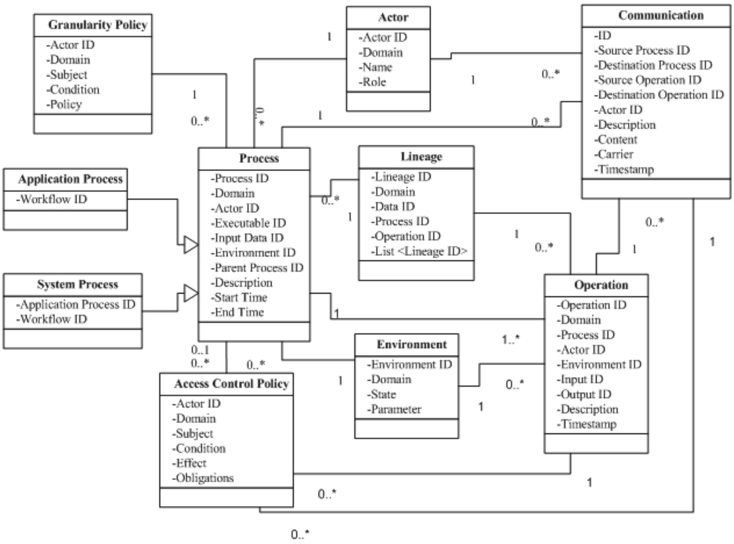

Fig. 2. Class Diagram of Provenance Model

From an implementation perspective, we represent our generic model as relationships among various provenance records as shown in Fig 2. Each provenance record is identified by an ID. Since provenance may be exchanged across different systems, we use *domain* to specify the scope of the records.

3 Conclusion

In this paper, we propose a comprehensive provenance model that can encapsulate the data provenance captured at different stages of a physical/computational process. The model captures the characteristics of standard provenance models which ensures the inter-operability of provenance across different systems.

References

1. Foster, I., Vöckler, J., Wilde, M., Zhao, Y.: Chimera: A virtual data system for representing, querying, and automating data derivation. In: Proc. of the Conference on Scientific and Statistical Database Management (SSDBM), pp. 37–46 (2002)
2. Janée, G., Mathena, J., Frew, J.: A data model and architecture for long-term preservation. In: Proc. of the Conference on Digital Libraries, pp. 134–144 (2008)
3. Frew, J., Metzger, D., Slaughter, P.: Automatic capture and reconstruction of computational provenance. Concurrency and Computation: Practice and Experience 20, 485–496 (2008)
4. Muniswamy-Reddy, K., Holland, D., Braun, U., Seltzer, M.: Provenance-aware storage systems. In: Proc. of the USENIX Annual Technical Conference (2006)
5. Ni, Q., Xu, S., Bertino, E., Sandhu, R., Han, W.: An Access Control Language for a General Provenance Model. In: Jonker, W., Petković, M. (eds.) SDM 2009. LNCS, vol. 5776, pp. 68–88. Springer, Heidelberg (2009)