Dataspaces: A New Abstraction for Information Management

Alon Y. Halevy[1], Michael J. Franklin[2], and David Maier[3]

[1] Google Inc.
halevy@google.com
[2] University of California at Berkeley
franklin@cs.berkeley.edu
[3] Portland State University
maier@cs.pdx.edu

Most data management scenarios today rarely have a situation in which all the data that needs to be managed can fit nicely into a conventional relational DBMS, or into any other single data model or system. Instead, we see a set of loosely connected data sources, typically with the following recurring challenges:

- Users want be able to search the entire collection without having knowledge of individual sources, their schemas or interfaces. In some cases, they merely want to know where the information exists as a starting point to further exploration.
- An organization may want to enforce certain rules, integrity constraints, or conventions (e.g., on naming entities) across the entire collection, or track flow and lineage between systems. Furthermore, the organization needs to create a coherent external view of the data.
- The administrators may want to impose a single "support system" in terms of recovery, availability, and redundancy, as well as uniform security and access controls.
- Users and administrators need to manage the evolution of the data, both in terms of content and schemas, in particular as new data sources get added (e.g., as a result of mergers or new partnerships).

The aforementioned data management challenges are ubiquitous – they arise in enterprises (large or small), coordination within and across government agencies, data analysis in large science-related research or development projects, management of libraries (digital or otherwise), information collection and dissemination in the battlefield, search on one's PC desktop or other personal devices, coordination between devices in a "smart" home, and in search for structured objects on the web. In these scenarios, there is some well-understood scope and control across the data and systems within these organizations, and hence one can identify a space of data, which, if managed in a principled way, will offer significant benefits to the organization.

We recently introduced *dataspaces* [1] as a new abstraction for data management for such scenarios, and proposed the development of DataSpace Support Platforms (DSSPs) as an important agenda item for the data management field. In a nutshell, a DSSP offers a suite of interrelated services and guarantees that enables an application developer to focus on the specific challenges of an application, rather than the recurring challenges involved in dealing consistently and efficiently with large amounts of interrelated but disparately managed data.

M.L. Lee, K.L. Tan, and V. Wuwongse (Eds.): DASFAA 2006, LNCS 3882, pp. 1–2, 2006.

Traditionally, data integration and data exchange systems have aimed to offer many of the purported services of dataspace systems. In fact, DSSPs can be viewed as the next step in the evolution of data integration architectures, but are distinct from current data integration systems in the following way. Data integration systems require *semantic integration* before any services can be provided. Hence, although there is not a single schema to which all the data conforms, the system knows the precise relationships between the terms used in each schema. As a result, significant upfront effort is required in order to set up a data integration system.

Dataspace management is not a data integration approach; rather, it is more of a *data co-existence* approach. The goal of DSSPs is to provide base functionality over all data sources, regardless of how integrated they are. For example, a DSSP can provide keyword search over all of the data sources it contains, similar to the way that existing desktop search systems. When more sophisticated operations are required, such as relational-style query processing, data mining, over certain sources, then additional effort can be applied to more closely integrate those sources, in an incremental, "pay-as-you-go" fashion. Furthermore, as we perform more integration tasks, we expect the cost of integration to decrease. Similarly, along the administrative dimension, initially a DSSP can only provide weaker guarantees of consistency and durability. As stronger guarantees are desired, more effort can be put into making agreements among the various owners of data sources, and opening up certain interfaces (e.g., for commit protocols).

To summarize, the distinguishing properties of dataspace systems are the following:

- A DSSP must deal with data and applications in a wide variety of formats accessible through many systems with different interfaces. A DSSP is required to manage *all* the data in the dataspace rather than leaving some out, as with DBMSs.
- Although a DSSP offers an integrated means of searching, querying, updating, and administering the dataspace, often the same data may also be accessible and modifiable through an interface native to the system hosting the data. Thus, unlike a DBMS, a DSSP is not in full control of its data.
- Queries to a DSSP may offer varying levels of service, and in some cases may return *best-effort* or approximate answers. For example, when individual data sources are unavailable, a DSSP may be capable of producing the best results it can, using the data accessible to it at the time of the query.
- A DSSP should offer the tools to create tighter integration of data in the space as necessary.

Reference

1. M. Franklin, A. Halevy and D. Maier. From Databases to Dataspaces: a new abstraction for information management. *SIGMOD Record*, Volume 34(4), pp. 27-33, December, 2005.