(Some of) The Real Bottlenecks to Structured Data Sharing

Zachary G. Ives and Todd J. Green
University of Pennsylvania
{zives,tjgreen}@cis.upenn.edu

The Web and the Internet have become an instrumental part of virtually all fields, particularly those related to technology, engineering, and science. Given the pervasiveness of structured information, as well as the need to access and integrate this data, it seems surprising that the efforts of the data integration community have not gained much traction in "the real world."

Some have argued that this is because information integration is too hard – that, in particular, schema mapping and semantic ambiguities render the problem AI-complete, making widespread data integration an infeasible vision. Perhaps this is true if our goal is to truly replace the entire Web in interconnecting all data – but there are in fact many domains in which large numbers of structured data sources *are* being integrated, often using tools that are more primitive than those proposed in our research community. In particular, fields like the life sciences have become almost completely information-driven as they seek to understand, integrate, and model the basic operating rules of organisms – and the tools are mostly hand-coded import scripts in Perl or Python, built over simple MySQL or Oracle databases (with the occasional flat file).

Thus it seems clear that while integrating data might be hard, progress is being made in the real world. The core question is why our community's products are seldom used. It clearly is not the amount of manual labor required by our tools – after all, writing a Perl script takes at least as long. Nor is it the complexity of our tools – which should be comparable to the tools and languages used in these efforts. The authors have had many discussions with members of the bioinformatics and medical informatics communities, and the main lesson we have learned is that our tools make too many assumptions that are not reflective of the real world – often making them obstacles to getting work done. We discuss the main points below in the hope they can stimulate fresh approaches in our research community. We believe many of these points are not exclusive to the life sciences – they in fact are reflective of a general class of applications that involve loose collaborations.

1. *Existing systems already handle basic capabilities.*

   Our community took a very long time to get to the point where commercial tools became available (and in fact there are still no real data integration solutions available in open source). In the meantime, many life sciences groups needed solutions to their integration needs. They developed suites of import scripts, data warehouses, and Web portals to solve their immediate problems. Now they have operational systems, and while these systems have many limitations, particularly with respect to data freshness or data dependencies, our tools do little to address these limitations. If we do not offer a compelling new capability, there is little incentive for them to change.

2. *Inconsistency and incompleteness are prevalent.*

   Our community's approach is often based on integration by standardization – creating mediated schemas, canonical names, fixed taxonomies. The focus is on top-down design and implementation. Yet integration is a bottom-up process: first two sources develop in isolation, with different user needs and hence different representations. Later the owners of these sources want to collaborate. However, in general they have some very specific wishes:

   (a) Each collaborator wants "all gain and no pain" – sharing data should not require a site to retrain or rewrite applications. It should not prevent updates to data/metadata, or especially corrections to data from elsewhere, even though these might break mappings or data consistency.
   (b) Inevitably, data (not just identifiers and schema elements) from different sources will conflict in a semantic way. This might be because the data is subjective, it might be due to measurement error, or it might be due to

differences in curation/analysis. A "right" answer is not necessarily probabilistic; it may be *relative* to a given site, as there may be differences of opinion. It must be possible to share data even in under these conditions.

(c) Not all data is equally trustworthy, as it comes from different curation processes, sources, and so on. As more collaborators join together, mechanisms for filtering or credibility are essential. Attribution – in the form of data *provenance* – is also key.

In recent years the database community has started to focus on some of these issues, particularly those relating to sharing among peer schemas and to uncertainty – but such efforts have not yet made their way to broadly available data integration tools.

3. *Change is a constant.*

Most integration, exchange, or warehousing systems today assume a very static world: source schemas remain fixed, the mediated schema remains the same, and even (in the case of warehousing or exchange) the data doesn't change frequently. At most, change in the form of new sources (and hence mappings) is supported.

In reality, especially in evolving fields like bioinformatics, everything is constantly changing. At Penn, there is a large staff of programmers whose sole job is to continuously update and expand the Genomics Unified Schema, a standardized schema for bioinformatics data that is used by a number of sites around the world, because the information needs of the community continue to evolve. In turn, the sites maintaining the database *instances* publish weekly or monthly "diffs" describing the changes made to their data. Some of the diffs are simply new data, but others are corrections or addenda to existing entries.

4. *Most scientific data is derived.*

Particularly in the sciences, the data of most value is not raw measurements, but rather the result of significant processing or human curation. Thus, provenance is an important contributor to the understanding and assessment of data: it tracks where data originated, how it was derived, and how different data items relate.

5. *Data exchange is often bidirectional.*

Our community tends to think of data exchange as being unidirectional, as in loading a data warehouse. In fact, many bioinformatics scripts already perform unidirectional import. Yet scientific collaboration is inherently bidirectional. The scripts (and traditional data integration tools) break down when put in bidirectional arrangements, as each source may derive new data from what it imports from its neighbors.

6. *Queries Are topic-revealing.*

A surprising revelation from many of the sciences is that many researchers prefer a self-contained data warehouse because no one can snoop on their queries and "scoop" their results. We seldom think about settings in which queries must be kept confidential.

In our own work on the Orchestra Collaborative Data Sharing System, we have attempted to take a first step towards addressing these criticisms and concerns, in order to take real world data sharing needs into account. However, we believe this is only the beginning, and we invite the reader to consider how data integration tools' shortcomings can be remedied.

Acknowledgements

The authors would like to thank Susan Davidson, Val Tannen, Alon Halevy, Phil Bernstein, Sarah Cohen-Boulakia, Pete White, Chris Stoeckert, Junhyong Kim, Cornelius Rosse, and Lee Hood for many fruitful discussions relating to data integration and its use.