# Quantitative data: learning to share

Monya Baker

Adaptive technologies are helping researchers combine and organize experimental results.



Ten years ago, biologists were content to confine analysis to a single data set generated in their own laboratories, says Olga Troyanskaya, a computational biologist at Princeton University. That mindset is changing. "Now there are so many data sets you can't just stay in your own cave," she says. Data from high-throughput experiments designed to answer one biological question are now routinely used to address others as well, but repurposing data can be difficult. "A lot of bioinformatics is looking at what has happened in other fields and making it work for biology, which is surprisingly non-trivial," says Troyanskaya.

"From a software-engineering perspective, you're used to having a clear definition," explains Hector Rovira at the Institute for Systems Biology (ISB) in Seattle, Washington. Integrated analysis across data sets works best within rigid formats, but as researchers pursue biological questions, experimental designs shift. "The problem with biology is that that domain model is hard to define. It changes. It's too complicated to try to define it a priori." Particularly for results other than genetic sequences, data formats can change faster than software is written: for example, gene-expression chips are replaced with RNA-seq; plate readers analyze ever denser arrays of samples; new types of experiments generate new forms of data.

Researchers in the physical sciences are accustomed to large consortia with very standardized ways of collecting, storing and processing data. Although consortia are becoming more prevalent in the biological sciences, most projects are still run at the level of individual laboratories. That results in a variety of data types and ways of dealing with them, says Dana Robinson, a bioinformatician at the HDF Group, which supports a set of file formats used chiefly by astronomers and physical scientists. "In biology, it's all one-off," he says.

"When we share data in biology, it's in Excel spreadsheets," says Christoph Best, an engineer at Google who has also worked in structural biology and electron microscopy. "Figuring out what is in each column can be difficult." Often, researchers have to write clunky import programs or implement standardizations that can feel cumbersome. These are only partial solutions, and the pace at which data can be kludged together is slower than the pace at which data sets are produced.

Coordinating data sets can be difficult. "Most experimental data have a whole series of different dimensions, and they aren't consistent," says Peter Sorger, a systems biologist at Harvard Medical School. For example, one set of experiments may collect a time series for five cell lines treated with different short interfering RNAs. The next set may look instead at just one time point using several doses. The result? "We increasingly collect quantitative data, but can't effectively transmit it from one person to the next," says Sorger.

Given these challenges, it's not surprising that many efforts now aim to bring together data from disparate, ever-changing experiments. These endeavors have been referred to as "adaptive informatics"[1] and "adaptable data management"[2].

Emerging adaptive technologies include cloud computing, schema-free solutions and unified security systems; domain-specific and process-oriented programming languages; and simpler and more scalable high-performance computing tools. The well-known Galaxy Project, for example, offers a web-based platform allowing researchers to bring together a variety of analytical tools and customize workflows. The GenePattern platform pulls together more than 150 tools for a wide range of studies, including gene expression, SNP analysis, flow cytometry and proteomics, as well as a variety of data-processing tasks.

As biological data sets grow larger, bioinformaticians such as Dana Robinson at the HDF Group are adapting file formats originally developed for the physical sciences.

Just a few examples of these efforts are described here. Two are 'federated systems' that pull together data from various locations. BioMart aims to make data available that are housed at different institutions; Addama is a service architecture built at one institution. In contrast, centralized systems keep data together in one place. LabKey Software provides an open-source platform for far-flung collaborators, whereas the SDCubes data-storage format can be implemented at the level of individual laboratories.

## BioMart: many databases look like one

BioMart, which grew out of efforts to extend the data-mining capacity of the Ensembl genomic databases, now links more than 40 databases hosted in different locations. BioMart was created to ameliorate two common difficulties in interpreting biological data, explains founder Arek Kasprzyk. One problem is that making sense of new data often requires bringing in data from other databases. The other is that biological databases, once established, can be slow to adapt to changing concepts and techniques.

BioMart can access diverse databases from a single interface, says founder Arek Kasprzyk.

Although BioMart reaches into dozens of databases, it provides a single interface, so its users do not need to learn the intricacies of many systems. (Users can select from a handful of interfaces tailored to their expertise.) "We can create a virtual database that appears to the user as one database. Each source can evolve on its own, but the virtual database remains intact," says Kasprzyk. BioMart is open source and freely available. "We wanted to bring the data to the user without requiring big IT departments," says Kasprzyk.

Making BioMart work required decoupling the data-gathering infrastructure from specific biological concepts, an approach called "data-agnostic modeling". Usually, builders expect software to mirror the conceptual domains in whatever field it is used for. Banking software, for example, would have domains for customers, accounts and local branches. But biology is too complex for that, and the conceptual domains are also in constant flux, says Kasprzyk. Conventional software is a bit like hieroglyphics—describing new objects requires making new symbols, he says. BioMart works more like an alphabet: describing new objects does not demand inventing new letters.

One consortium using BioMart to share data is the International Cancer Genome Consortium (ICGC), which is sequencing cancerous and healthy tissues from 500 people for 50 cancers. Each of several affiliated centers manages its own data, but the data are available to all ICGC members. Kasprzyk says that researchers who are not part of a large consortium can still contribute data, however. Similarly, participants can use the database without contributing data. A number of companies are using BioMart to assess their internal, proprietary programs, he says. (The November 2011 issue of the journal *Database* is devoted to several case studies involving BioMart.)

### Addama: gathering information by staying loose

People at the ISB often talk about the need for "informal data management", or software that works with only vague requirements. They want an architecture that can improvise. That typically means that system components are linked in loosely coupled 'layers': an instrument layer for data-providing instruments; a conceptual layer consisting of integrated pipelines, indexing systems and other analytic software; and an organizational layer that works with results from other layers.

Pursuit of this goal resulted in a data-integration architecture called Addama, for 'adaptive data management service architecture'. The goal is to bring tools and data together seamlessly. For example, a common decentralized authentication tool called OpenID allows users to move among resources with just a single username and password. Addama interacts with a range of computing infrastructures, from individual desktop tools to large computing clusters. To keep the architecture flexible, developers avoided committing to any particular programming language. Information is accessed via several established standard protocols rather than one specific data format. New data and additional notations can be added without disrupting formats or data models. Manipulating data in the application layer allows analytic results and data to be moved readily to new applications.

Credit: Hsiao-Ching Chou

Addama is a flexible integration architecture built to suit changing needs at the Institute of Systems Biology, explains software engineer Hector Rovira.

There is a cost to all this flexibility, says Rovira, the ISB software engineer who led design of the architecture. Changes in data formats or analysis systems may change results of other analyses, or even make the system less stable. Nonetheless, Addama consistently supports sophisticated analyses of disparately stored and managed data. For example, the ISB is one of several centers working on a large project known as the Cancer Genome Atlas, which aims to study 20 cancers in 20,000 patients, comparing tumorous and healthy tissues in terms of genome sequence, methylation patterns, and the expression of genes and small RNAs. Addama helps ISB bring together data from several sources, each with its own way of controlling user access and identifying data sets. It can also support analytical tools developed at different centers.

Rovira is a big fan of what's called "unstructured data" or "No-SQL" (SQL stands for 'structured query language'), which allows the structure of data to evolve. The exploratory nature of science chafes against the standard, computer-centric approach of setting data formats before a science project even begins, he says. "For software to be adaptable, you can't have these constraints."

Although Addama is open source, the ISB is not looking to extend the architecture beyond its and its collaborators' needs. Rovira believes that the value of Addama to other groups lies in the approach rather than the actual code. "We don't expect your software to hold the right interfaces; we adapt software into it," he says. "Addama wasn't meant to be one system that could be used to bring in the data. It's one system that can bring in different data technologies."

### LabKey Server: open-source software from commercial developers

Less than two miles away from ISB, engineers at LabKey Software are developing LabKey Server, an open-source platform that helps biologists to share data. "LabKey allows researchers to combine data across multiple experiments, analyze that data online and share both raw data and results of analyses," says company cofounder Mark Igra. But, he says, just explaining what LabKey does can be a challenge. "Scientific data integration doesn't have a well-established set of capabilities that lets everyone know what you're talking about."

In some ways, LabKey's goal—to be a better option than the default collection of Excel spreadsheets—is relatively modest. Labs are notorious for having Excel files spread among laptops, riddled with simple scripts called macros that may make sense only to the person who wrote them. But if data can go into a tabular

Steve Hanson, LabKey

Data-integration company LabKey Software began after professional developers at the Fred Hutchinson Cancer Research Institute decided they wanted to reach a broader community.

format like Excel, says Igra, it can readily go into relational databases that support more sophisticated analyses and allow collaborators to track what experiments have been performed.

LabKey Server relies on SQL, but Igra says that this approach is much more flexible than many people believe. "It's a relational database, but it's not locked into any particular structure," he explains. In fact, one of LabKey's research projects asks scientists to donate spreadsheet data files without adding any annotation. It's part of an effort to develop software that, according to the company website, can automatically "unlock data trapped in Excel".

LabKey Server also provides flexible data visualization. "In software, the traditional approach is 'one tool fits all'," explains Elizabeth Nelson, LabKey's outreach director. "But every lab has a different way that it wants to see its data, so it's really important for them to be able to customize how they analyze and visualize their data."

For example, LabKey is used by the Center for HIV-AIDS Vaccine Immunology, a consortium of universities and academic medical centers that is conducting long-term observational studies of individuals at risk of contracting HIV. The center, sponsored by the US National Institute of Allergy and Infectious Diseases, uses LabKey to track results and analysis of specimens from participating institutions. Unlike many open-source products, LabKey was designed with data confidentiality requirements in mind and can easily manage what information various users see. In some instances, certain columns are hidden to protect patient identities; in others, users can see only the rows containing data that were analyzed at a specific institution.

## SDCubes: flexible and machine-readable spreadsheets

Harvard's Sorger and his colleagues have a simple approach that essentially transforms spreadsheets for easier data sharing, even among scientists who are not already collaborators. Spreadsheets are often convenient for storing raw data, says Sorger, but "what's missing is the metadata; we get big spreadsheets of data, but no one knows what they are."

Consistent with the spirit of reusing and repurposing, semantically typed data cubes, or SDCubes, combine two well-established technologies: hierarchical data format (HDF) and extensible markup language (XML). HDF offers a way to store very large numerical data sets; many individual users create files that are tens of terabytes ($10^{12}$ bytes) in size. HDF is used extensively in earth science and astronomy, fields in which data collected are highly standardized and based on observations rather than perturbations. HDF is less suited to the textual data often used to describe molecular and cellular experiments: response to gene knockdown agents or lineage studies using genetic reporters in pulse-chase experiments, for example. XML, however, can incorporate exactly this kind of information—the metadata—so that the experimental design itself is captured in a machine-readable format.

Because HDF files can be much larger than Excel files, results of many experiments can be easily stored and readily accessed in one file, from which data can be accessed in smaller 'windows' that demand less memory. Sorger and his colleagues are currently using SDCubes for data from high-throughput microscopy experiments analyzing the responses of individual cells to varying levels of small molecules.

The problem is that, for biologists looking for a way to house their latest data, an Excel-type spreadsheet is still probably the most convenient short-term solution. But scientists need to think beyond their current projects or the length of a graduate student's tenure, Sorger says. "If you store experimental results in little tiny pieces, you can't ever put it together. SDCubes can let data build up over months and years."

Peter Sorger developed SDCubes to keep large data files manageable and to retain metadata about experimental design.

Software engineers at the nonprofit HDF Group, which supports HDF users, were not involved in the development of SDCubes, but they expect that similar biology projects will be forthcoming, particularly as the sizes of biological data sets begin to approach those seen in the physical sciences. HDF can handle the volume and complexity of biologists' data, says Mike Folk, executive director of the HDF Group. "It can operate in any computing environment. It is extensible; as your data needs change and grow, it can adapt." The main problem for biologists is that there is currently no query engine built into the system, says Dana Robinson, a bioinformatician at HDF. "It's like a great file cabinet that has a very complicated lock." Plans to build a biology-friendly front end are in the works, he says.

## Human solution

Whatever resources are used to bring data together, there is a growing assumption among researchers that data and computing resources should be readily reused, repurposed and extended by other scientists. "There are a lot of programs that come out every week in bioinformatics," says Avi Ma'ayan, a systems biologist at Mount Sinai Hospital. "I can't think of one that is actually taking over—but a common theme is that you can attach pieces to it." In fact, the fastest way to make data more useful for more purposes may lie in "*ad hoc* development," in which applicable tools are identified and strung together as needed, says Sarah Killcoyne, who is project manager for research informatics at the ISB. "No one can build one system and hope it works," she says. "In the life sciences, we have gotten a lot better at sharing."

That makes sense to Princeton's Troyanskaya. "If you develop everything from scratch, you'll never keep up," she says.

1. Millard, B.L., Niepel, M., Menden, M.P., Muhlich, J.L. & Sorger, P.K. *Nat. Methods* **8**, 487–492 (2011).
2. Boyle, J. *et al. BMC Bioinformatics* **10**, 79 (2009).

**Monya Baker is technology editor for *Nature* and *Nature Methods* (m.baker@us.nature.com).**