WikiLIMS — Next-Gen Data Management

MICHAEL CARIASO

The typical science lab generates data that are never recorded electronically. Some of these data end up in old notebooks, but a lot of it is abandoned because recording takes more time and effort than the perceived value. Even valuable data are abandoned when there is no place to record it where it might ever be found.

Relational databases (DBs) such as Oracle or Access are the most common way of storing and querying large datasets, but not the only way. Exotic multidimensional systems allow WalMart to visualize details of every purchase in near real time, while open source Hadoop simplifies ad hoc massively parallel searches.

Laboratory bioinformatics does most of its storage and query with the humble file system due the current emphasis on sequencing. Relational DBs play a smaller but important role, indexing data and cataloging the discoveries mined from them. I believe relational DBs would be more common in bioinformatics if they didn't impose the upfront cost of schema design. However without a schema a relational DB isn't much better than a simple text file.

Wikis such as Wikipedia.org and OpenWetWare.org are databases, but they are largely text written by humans for humans. A wiki can also serve as a database to be read and written by computer programs. They fit somewhere between simple text files and a full relational DB. You may be surprised at how well a wiki can substitute for a relational DB. Equally impressive are the things a wiki can easily do that a relational DB never could.

Using a Wiki Well

There are numerous wiki tools available but my experience is mainly as a MediaWiki user. I've been pleased with the developer community and comforted by the proven scalability. Wikipedia serves approximately 2.5 billion page views per day and stores nearly 10 million pages across more than 300 servers. Serving 100 million page views costs $40. A wiki can scale.

Libraries such as Perlwikipedia and Pywikipedia allow developers to read and write to the wiki as easily as a file or a relational DB. At BioTeam, we've built programs which add functionality to the devices in our labs, such as the Roche 454 FLX and the Illumina Genome Analyzer. These machines now automatically record the details of each run as a new page in our client's wikis. The raw data are stored on a web accessible RAID, while the wiki page stores metadata (who ran what when) in an easily parsed format. Templates stored in the wiki show the metadata with hyperlinks to the raw data and related wiki pages. As our needs grow, the templates can also call arbitrary functions in PHP or the language of your choice.

We're now capturing data electronically with no human intervention required. A single device doing this isn't particularly compelling, but two or more can be. The wiki becomes an electronic lab notebook written entirely by your machines. Because it can be viewed and edited with a Web browser, scientists can begin by using it as a web portal to recently completed analyses. Each discovery can be recorded into the wiki, one click away from the supporting raw data. Simple programs can build reports from or read and write into the wiki. An automatic full history of every page provides a safety net, and a way to view how our understanding of the data has evolved. This enables cycles in which humans and software each complement the other's strengths.

This is the perfect time for a wiki. And once you have one, you will find it just keeps growing to accommodate more data you've previously ignored.

Relational DBs spend up-front effort to design a schema that enables optimized runtime performance. There are times when that is essential, and in these cases a wiki is inappropriate. But more often, I'm reassured in knowing that the DB will be plenty fast enough, worrying instead whether I've designed a schema to handle every case. This is a perfect time for a wiki. And once you have one, you will find it just keeps growing to accommodate more data you've previously ignored.

Visit DBpedia.org to explore what a wiki can do that a relational can't. There you'll be able to ask SQL-style queries against Wikipedia. This is possible because they've built RDFs that describe the structured data within Wikipedia. This is the Semantic Web in action. By defining RDFs for your wiki, it too can join the semantic web. This allows programs to better understand your data, opening up opportunities for more powerful queries.

Michael Cariaso is the senior scientific consultant for the BioTeam. He can be reached at cariaso@bioteam.net.