January 31, 2007

Biotechnology Firm's Growth Mandates Database Migration

Like many growing companies, Luca Technologies, a biotech company located in Golden, Colorado, has outgrown the capabilities of its Microsoft Access database. Databases that start as small team and departmental applications often grow into mission-critical databases that require scalable client-server architecture and more robust security, as is the case with Luca Technologies.

In addition, data compliance regulations have placed pressure on such organizations to move data from end users' computer desktops to a securely managed platform such as Microsoft SQL Server. Because Access and SQL Server are built on quite different technologies, a migration from Access to SQL Server can be challenging.

Luca Technologies is developing a novel, long-term, biotechnology-driven solution to the rising US dependence on foreign energy sources. The company has relied on a critical Access database that their scientists use for proprietary data related to their research projects. Kinna McDavid, Database Analyst, was looking for a solution that would help convert the Access database to SQL Server.

Steps to Migration

In this rapidly evolving science research environment, McDavid sought management approval for a 365 person-day transitional project to convert the company's Access 2003 database to SQL Server 2005. The database had more than 100 tables, more than 600 queries, as well as forms and reports. Changing to an Access ADP front-end with a SQL Server back-end would provide a good interim step en route to the Microsoft .NET platform. The SQL Server back-end would immediately improve security and increase performance for all users. And being able to automate the conversion process would significantly speed up the time and reduce the cost of migrating applications to the SQL platform.

The basic options for converting an Access database to SQL Server involve either linked tables or the creation of an Access Data Project (ADP). Linked tables offers the easier approach because it keeps the current Access database front-end and uses ODBC to connect to upsized SQL tables. However, because this approach continues to use the Jet database engine, performance can be adversely affected. ADP provides a more scalable and secure approach. It offers a true client/server protocol through an OLE DB connection to the upsized SQL tables. However, converting to an ADP is more difficult and involves a significant number of programming changes to make the front-end fully SQL-compliant. It also requires converting Access Queries to SQL-equivalent Views or Stored procedures.

Microsoft provides two tools to assist in the migration of Access databases to SQL Server: the Upsizing Wizard and the second-generation tool SQL Server Migration Assistant for Access (SSMA). The Upsizing Wizard offers useful functions (it can migrate Table Schema, Data, Indexes, and Relationships), but it has limitations. The wizard creates either an ADP or a linked tables solution. If you choose the ADP option, the wizard will upsize Access Queries to Views or Functions, but only the simplest of SELECT Queries will upsize. Conversion of DELETE, UPDATE, INSERT, and CROSSTAB queries is partially supported or not supported at all. The wizard doesn't address conversion of Visual Basic for Applications (VBA) code (Data Access

Object—DAO—references that require conversion to ADO) or the properties and controls of Forms and Reports that contain SQL Statement syntax.

SSMA migrates Table Schema, Data, Indexes, and Relationships reliably and much faster than the wizard. It converts SQL Statements in Access Queries to Views—but with less functionality than the wizard maintains. Like the wizard, SSMA can upsize only the simplest of SELECT Queries. All other types of Access queries aren't supported. SSMA offers only a linked tables solution to SQL Server 2005; it doesn't support creation of an ADP.

Adding the Key Piece
At a SQL PASS conference, McDavid met ConvertU2 professionals who introduced her to 2SQL software, which helps migrate critical Access database applications to SQL Server. Steve Koop developed 2SQL based on his realization that extending the Microsoft tools would make automated ADP conversion possible. The model and process for 2SQL conversion deploys the Microsoft tools, scans Access Tables, Queries, Forms, Reports, and Code Modules for conversion issues, and then applies automated fixes.

As with any conversion tool, 2SQL isn't a 100 percent solution; some manual work is generally required. However, because 2SQL documents all issues detected (not just the one ones it can fix), the time required to address issues is significantly reduced. The full analysis 2SQL provides is available in XML or Access. And 2SQL's library of fixes addresses more than 80 percent of the problems detected.

Accurate Assessment Saves Time and Lowers Cost
Scanning the Luca Technologies database with 2SQL revealed nearly 11,000 conversion issues that 2SQL could fix automatically. Because 2SQL is provided as a software service, each project is priced based on the complexity of the database, which is reflected in the issue count. (Customers pay only for the fixes and the 2SQL analysis.)

McDavid is a do-it-yourself customer who tackled the project with some remote "coaching" from Steve Koop. The project involves taking the database through three phases, beginning with processing by the SQL Server Migration Assistant to move the tables and data to SQL Server 2005, followed by 2SQL processing in "repair mode," and finally by manual repair of the remaining problem areas that 2SQL identified.

Still within the first 30 days of this major project, but with the end in sight, McDavid acknowledges that "with 96.5 percent of the identified issues automatically repaired, this conversion tool is better than any I've used in the past." McDavid expects ConvertU2 to save the company not only time but also $70,000 in direct project consulting costs. "Best yet," McDavid says, "Steve Koop has been an excellent advisor and support resource while we've been learning SQL 2005."

To learn more about ConvertU2 and 2SQL, visit http://www.convertU2.com. To learn more about Luca Technologies, visit http://www.lucatechnologies.com.

Contact Us
About the newsletter—letters@sqlmag.com
About technical questions—http://www.sqlmag.com/forums
About product news—products@sqlmag.com
About your subscription—sqlupdate@sqlmag.com
About sponsoring SQL Server Magazine Update—Richard Resnick.

Subscription Info

SQL Server Magazine Update is brought to you by SQL Server Magazine, the only magazine devoted to helping developers and DBAs master new and emerging SQL Server technologies and issues.

Manage Your Account

You are subscribed as #EmailAddr#
You are receiving this email message because you subscribed to this newsletter on our Web site. To unsubscribe, click the unsubscribe link:Unsubscribe.

View the SQL Server Magazine Privacy Policy.

SQL Server Magazine, a division of Penton Media, Inc.
221 East 29th Street, Loveland, CO 80538
Attention: Customer Service Department