How to Integrate SQL Database Content with QuickBooks

MySQL, SQL Server, Other Database Platforms

Garrett Fisher – Senior Controller & Director, Scale Finance LLC

The need to integrate data across platforms speaks for itself. Cost savings, data availability and elimination of human error point to the need for cross-platform data integration on an automated scale. For companies that manage their financial data with QuickBooks, this can become a frequent concern due to its operational limitations. QuickBooks does not provide enterprise resource planning functionality nor the power of an enterprise financial software package. As small companies grow, tracking of operational items graduates from sales orders or invoices in QuickBooks and is generally tracked using software unique to the industry served, with invoicing continuing to be handled in QuickBooks. At times, however, a company may have a custom software package developed (for web-based purposes, for example) that handles day-to-day operational data. The question then becomes, how do you integrate this data seamlessly into your financial software?

For starters, it is necessary to determine the nature of the data that needs to be integrated. Does it already contain final billing information inclusive of dollar amounts? Or, does it only include operational data with the billing function needing to still take place in QuickBooks? In either case, the need can be managed. For data being imported where billing information is already known, an integration where the orders/invoices are imported to QuickBooks as invoices would be the solution. For operational data only, database records from the source location should be imported as a Sales Estimate or a Sales Order, depending on the nature of the data. Note that Sales Orders can be invoiced at the touch of a button inside of QuickBooks, with inventory tracking and "open sales orders" reporting functionality to track unbilled items. Whether or not actual inventory parts are being sold, it is highly suggested to use Sales Orders instead of Estimates as the process is more tightly controlled by QuickBooks and less subject to error.

For those unfamiliar, QuickBooks offers both an API (application programmers interface) and an ODBC (Open DataBase Connectivity) driver to connect to a company file in the background. These interfaces allow for the development of other software programs that can interface with QuickBooks automatically. While some software platforms offer the ability to automatically transfer invoice data into QuickBooks, most do not as an in-house feature. Custom developing an application to accomplish this task is possible, but is most likely not cost effective. If a company has the funds to spearhead the custom development of an application of this nature, then most likely it should be purchasing the next generation of accounting software.

Fortunately, there is a packaged solution that is for sale that can service these needs. AaaTex Corp, of Daytona Beach, FL, sells a software package called Aaatex ShoppingCartIntegrator. Shopping cart? I thought I was integrating from a SQL database? While the primary and

marketed purpose of this application is in fact for the seamless data transfer between e-commerce shopping carts for online stores, the application can be used to handle information from SQL databases of any form (MySQL, SQL Server, etc). The product retails for $299 to $1,299 and will require additional paid support to implement in a custom setting. How does it work?

There is some custom work involved to interface with a proprietary database.

Records to Import: You will have to define the records that will be imported each time the application is run. This is done by providing the SQL query statement that provides the records from the source database. Additionally, all fields that you wish to appear as a sales order or invoice need to be included in the query statement.

Import Tracking: The end-user will have to determine if they are going to manually select ranges of order numbers, or have the system track whether the order was imported for you. If the ShoppingCartIntegrator will track for you, a field will need to be created in the source database table to allow the program to "flag" the item as imported (don't forget to include the criteria in your SQL WHERE clause to only include non-imported items). Also, you will have to define the SQL UPDATE statement that will be used to make the associated change in your source database.

Mapping: Each field coming from the source database must be "mapped" to fields in QuickBooks. Customer, date, items, item descriptions, prices – everything except fields that have a default value within QB have to be manually defined. While this is possible to do on your own, it is not a recommended procedure. AaaTex offers consulting services to handle the implementation of these items.

Printing: Will the invoice or order need to be printed upon import? The appropriate field needs to be marked "To be printed" as necessary in QuickBooks.

Connectivity: The end-user will have to set up the appropriate ODBC connection to the source database and provide all credentials necessary to develop a connection string to connect to the database.

Misc: There are a host of small items to be managed. Will you allow duplicate importing? Would you like the program to stop on "warning" level issues instead of "error" level issues? All of these matters are contingent on the end-users needs and should be addressed on a case-by-case basis.

The program itself requires that QuickBooks be opened the first time to the company file with the Admin login. QuickBooks will verify your permission to allow the ShoppingCartIntegrator to access the company file. This should be granted on an on-going basis, unless you plan on using the Admin login every time you run an import. Additionally, the company file must be open in QuickBooks with any user id for the import to function.

Interestingly, the program can be utilized to import purchasing data to the accounts payable side of the ledger and can be used to import multiple streams of data from different locations – to the same location in QuickBooks or to different locations – depending on your need.

There are some drawbacks to this type of operation. First, the program itself is somewhat difficult to use as the level of options and items to be attended to are rather high for an average user. More importantly, though, there are true operational questions to be asked by a company wishing to go down this road. If a firm wishes to integrate data sources in such a fashion, should they be using QuickBooks? Instead, is there an application on the market that can handle the financial side and the operational side at the same time? If the decision is to move forward anyway, how long will this solution last for the company? If the company is growing, entertaining a truly long-term solution would be highly beneficial.

In any case, automated integration of data is far superior to manual entry – especially when dealing with financial software. Most small companies make the mistake of assuming that since they are small, they have no choice but to live with less technology and less of the benefits of automation. However, with the proper investment of time and funds, significant labor-saving and error-reducing cost savings can be obtained.