All Aboard the REGEXP

REGEXP – SQL Engine's New Turbo Filter

By Christopher Hardekopf

A *regular expression* is simply a way of matching a string to a pattern of characters. For example, using regular expressions users of BASIS' SQL engine can now locate strings that start with the string "abc" or contain the string "def". In fact, they can even find strings that contain both an "abc" and "def" string separated by zero or more characters such as `'abc%def'`. The beauty of regular expressions (REGEXP) is that developers can use them to match arbitrary patterns of strings as specified by a user, even though limited to patterns expressed in the regular expression syntax the software provides.

The keyword `LIKE` is part of the SQL standard and provides an extremely simple regular expression syntax. In fact, it only has two wildcard characters – an `'_'` that matches any single character and a `'%'` that matches any sequence of zero or more characters. While this syntax is very simple to learn and easy for the SQL engine to optimize (in some cases), it is highly limited since there are numerous patterns that the user cannot match. The keyword `REGEXP` nicely solves this very problem.

While the keyword `REGEXP` gives the user a much more elaborate and useful regular expression syntax, it is not a part of the SQL standard and is therefore not portable to all SQL engines. On the other hand, several SQL engines do provide either REGEXP or some similar syntax for matching more complicated regular expressions. In any case, REGEXP provides a regular expression syntax very similar to PERL regular expressions (for details, refer to Sun's documentation on java.util.regex.Pattern), allowing the user to express much more interesting and useful string pattern matches. For example, a developer can find strings that contain either `abc` or `def` followed by exactly three of any character followed by `xyz` (i.e. `'(abc|def).{3}xyz'`). While the previous example may be somewhat arbitrary, regular expressions make finding data that confirms to commonly-used patterns, such as e-mail addresses, dates, timestamps, etc., a piece of cake. The REGEXP syntax gives much more flexibility to the user; however, the SQL engine can rarely optimize these regular expressions very well. In other words, a query that uses REGEXPs as a primary conditional in joins or on very large tables may run slowly.

Summary

Regular Expressions are a powerful new addition to the SQL engine that provide sophisticated searching and pattern matching that goes far beyond simple wildcards. With their support for matching character classes and boundaries, alternation and grouping, repetition and more, Regular Expressions give you the power to filter your data efficiently.

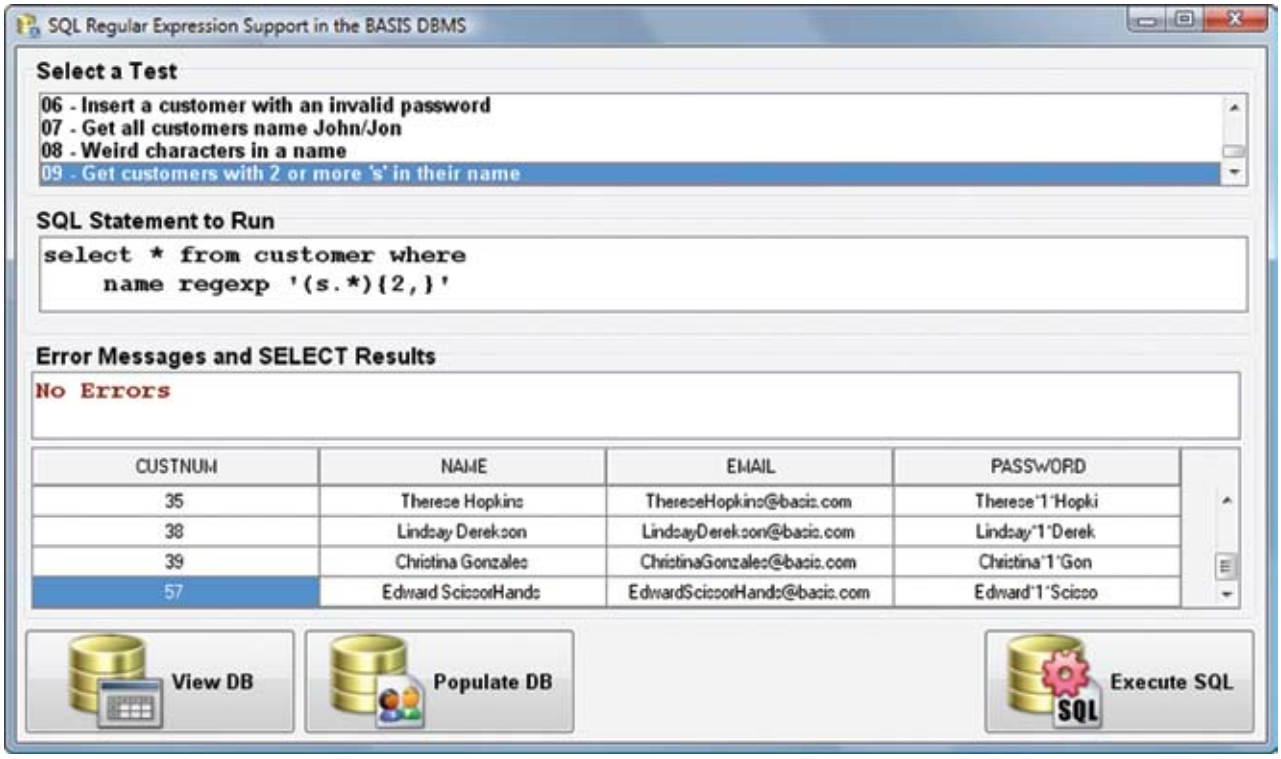

Figure. 1 The SQL Regular Expression Demo program showing the results of a SELECT query utilizing the regexp keyword

Chris Hardekopf
Software Engineer