Subject: Semicolon in Strings within SQL Multi Query String
Posted by kong on Sun, 26 Oct 2014 19:10:03 GMT
View Forum Message <> Reply to Message

When calling function executeQuery($query), it passes $query to the function multiQuery ($dbname, $tablename, $query) in dml.mysqli.class.inc, which uses the semicolon as the deliminator to split up $query into sub-queries for execution by MySQL.

However, this will result in incorrect sub-queries when sub-queries contain strings that themselves contain semicolon characters.

To solve this problem, we need to distinguish between semicolons that are part of strings and semicolons that delimit SQL sub-queries. Hence, would like to propose to make below changes in the source code of multiQuery function in dml.mysqli.class.inc.

Change from

```
        if (!is_array($query)) {
            // split string into an array of individual queries
            $array = explode(';', $query);
            $query = '';
            foreach ($array as $value) {
                if (!empty($value) AND substr($value, -1, 1) != ';') {
                    $query[] = $value.';';  // replace query terminator
                } // if
            } // foreach
        } // if
```

To

```
        if (!is_array($query)) {
            // split string into an array of individual queries
            $array = explode(';', $query);
            $query = '';
            $incomplete_query = '';
            foreach ($array as $value) {
            $incomplete_query .= TRIM($value);
             if (substr_count(str_replace("\"","",$incomplete_query),"'")%2) {
              // Odd number single quotes means semicolon was part of a string in one SQL sub-query
              // Must add back the semicolon and loop to restore the remainder of SQL sub-query
              $incomplete_query .=  ';';
             } // if
             elseif (!empty($incomplete_query)) {
                        $query[] = $incomplete_query.';';
              $incomplete_query = '';
             } // elseif
            } // foreach
        } // if
```

Have not checked whether this problem existed for the other database engines.