Subject: $lock_str FOR UPDATE FOR $tablename with PostgreSQL
Posted by pdv on Wed, 06 Feb 2019 18:55:31 GMT

Hi,

I've run into a problem with the $lock_str "FOR UPDATE OF $tablename" when accessing a PostgreSQL database (on macos). The problem is that if an alias is defined in the FROM-clause, then this alias must be used everywhere. The documentation is clear about this:

Quote:A substitute name for the FROM item containing the alias. An alias is used for brevity or to eliminate ambiguity for self-joins (where the same table is scanned multiple times). When an alias is provided, it completely hides the actual name of the table or function; for example given FROM foo AS f, the remainder of the SELECT must refer to this FROM item as f not foo. If an alias is written, a column alias list can also be written to provide substitute names for one or more columns of the table.

In std.table.class.inc this line (9165)

```
$array = $DML->getData($this->dbname_server, $this->tablename, $where);
```

passes the tablename also if an alias has been used, causing a PostgreSQL-error.

To solve this I've replaced this line by this code:

```
$table_array = extractTableNames($this->sql_from);
if(empty($table_array) || isset($table_array[$this->tablename])) {
$array = $DML->getData($this->dbname_server, $this->tablename, $where);
} else { // the first key is the alias from the FROM clause
$array = $DML->getData($this->dbname_server, array_search($this->tablename, $table_array), $where);
} //if
```

which works for the tests I've done sofar.

Regards,

Patrick

Subject: Re: $lock_str FOR UPDATE FOR $tablename with PostgreSQL
Posted by AJM on Thu, 07 Feb 2019 10:06:37 GMT

Instead of just posting your solution can you please supply sample code that produces the error?

Subject: Re: $lock_str FOR UPDATE FOR $tablename with PostgreSQL
Posted by pdv on Thu, 07 Feb 2019 13:01:26 GMT

Tony,

I have a department(id,...)<-->>employee(id,department_id,...) tables/relationship and when defining the relationship I got this error message (I have made the SQL query more readable):

Fatal Error: PostgreSQL: ERROR: 42P01: relation "dict_column" in FOR UPDATE clause not found in FROM clause LINE 4: ...id, dict_column_jnr.column_seq asc FOR UPDATE OF dict_colum... ^ LOCATION: transformLockingClause, analyze.c:2858 (# 42P01)

SQL query: SELECT column_name AS column_name_jnr
FROM dict_column AS dict_column_jnr
WHERE dict_column_jnr.database_id= 'bxtests'
 AND dict_column_jnr.table_id= 'employee'
 AND dict_column_jnr.column_id= 'department_id'
ORDER BY dict_column_jnr.database_id, dict_column_jnr.table_id, dict_column_jnr.column_seq asc
FOR UPDATE OF dict_column

Error in line 822 of file '/Library/WebServer/Documents/radicore/includes/dml.pgsql.cl ass.inc'.

Host Info: localhost, Server Version: 10.5
Client Encoding: UTF8, Server Encoding: UTF8
Database: DICT, Schema: "DICT", PUBLIC

PHP_SELF: /radicore/dict/related_column(multi4)a.php
CURRENT DIRECTORY: /Library/WebServer/Documents/radicore/dict
SERVER_ADDR: ::1
SERVER_NAME: localhost
HTTP_HOST: localhost
User Id: MGR
Role Id: GLOBAL
REMOTE_ADDR: ::1
REQUEST_URI: /radicore/dict/related_column(multi4)a.php

To get this to work the last clause should have been: FOR UPDATE OF dict_column_jnr.

Regards,

Patrick

Subject: Re: $lock_str FOR UPDATE FOR $tablename with PostgreSQL
Posted by AJM on Fri, 08 Feb 2019 09:57:48 GMT

I will try this out, and if it works I will include the change in the next release.