

---

Subject: \$lock\_str FOR UPDATE FOR \$tablename with PostgreSQL

Posted by [pdv](#) on Wed, 06 Feb 2019 18:55:31 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

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

---