

---

Subject: Check for empty \$schema in dml.psql.class.inc

Posted by [pdv](#) on Wed, 06 Feb 2019 17:42:47 GMT

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

---

Hi,

I'm new and trying to access some existing PostgreSQL databases with Radicore and on macos. Sofar I met a few problems which I could only solve by changing the code. I'll describe them in separate messages.

The first problem is similar to the one reported in the thread "SQLSRV Driver", message #4722. I could only connect to the database when hard coding the dbname (which is the postgresql schema).

The problem is in the following code in dml.pgsql.class.inc:

```
if (!empty($this->dbname) AND !empty($schema)) {  
    $result = $this->connect($schema) or trigger_error($this, E_USER_ERROR);  
} else {  
    trigger_error('No value supplied for PGSQL_dbname', E_USER_ERROR);  
} // if
```

This code fails because \$schema is empty, e.g. when querying for the list of databases. \$this->dbname refers here to the PGSQL\_dbname. I removed the check on \$schema. The error message is also misleading in this case since PGSQL\_dbname was not empty.

Regards,

Patrick

---

---

Subject: Re: Check for empty \$schema in dml.psql.class.inc

Posted by [AJM](#) on Thu, 07 Feb 2019 09:49:11 GMT

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

---

PostgreSQL is peculiar because it has the hierarchy server->database->schema->table instead of server->database->table. When the framework code connects to a server it is possible to change the default database (or "schema" in postgres) to whatever you choose, or even to construct a query which accesses tables in more than one database (or "schema" in postgres). For this reason when using postgres the "database" name \*MUST\* be hard coded in the config file, and it is only the "schema" name which is dynamic. Please refer to <http://www.tonymarston.net/php-mysql/postgresql.html#database.connection> for details.

---

---

Subject: Re: Check for empty \$schema in dml.psql.class.inc

Posted by [pdv](#) on Thu, 07 Feb 2019 12:15:56 GMT

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

---

Tony,

I realize the differences between MySQL and PostgreSQL, but it remains confusing.  
I want to access existing postgresql-dbs and I had 2 options: install radicare in each of them in a separate schema, like this:

```
db1
---schema-radicore
-----menu
-----dict
....
---schema1
-----table11
-----table12
---....

db2
---schema-radicore
---.....
```

Or to install radicare in a separate pgsqldb, which I prefer:

```
db-radicore
---schema-public
-----menu
-----dict
....
db1
---schema1
-----table11
-----table12
...
db2
---schema2
-----table21
-----table22
---.....
```

This means I need at least 2 connections, one for a particular db and one for radicare.  
This is my config.inc to access a pgsqldb bxtests with schema public (which I renamed to bxtests):

```
if (preg_match('/^(127.0.0.1|localhost|desktop|laptop)$/i', $_SERVER['SERVER_NAME'])) {
    global $servers;
    // server 0
    $servers[0]['dbhost']    = 'localhost';
    $servers[0]['dbengine']  = 'pgsql';
    $servers[0]['dbusername'] = 'user1';
```

```

$servers[0]['dbuserpass'] = 'password1';
$servers[0]['dbport']     = '';
$servers[0]['dbsocket']   = '';
$servers[0]['dbprefix']   = '';
$servers[0]['dbnames']    = 'bxtests';
$servers[0]['PGSQL_dbname'] = 'bxtests';
$servers[0]['switch_dbnames'] = array('bxtests' => 'public'); // the PostgreSQL schema is 'public'
$servers[0]['ssl_key']     = '';
$servers[0]['ssl_cert']    = '';
$servers[0]['ssl_ca']      = '';
$servers[0]['ssl_cpath']   = '';
$servers[0]['ssl_cipher']  = '';
// server 1
$servers[1]['dbhost']      = 'localhost';
$servers[1]['dbengine']    = 'pgsql';
$servers[1]['dbusername']  = 'user2';
$servers[1]['dbuserpass']  = 'password2';
$servers[1]['dbport']      = '';
$servers[1]['dbsocket']    = '';
$servers[1]['dbprefix']    = '';
$servers[1]['PGSQL_dbname'] = 'radicore';
$servers[1]['dbnames']     = '*';
} // if

```

With this server-configuration, when importing databases (schemas) I found that \$schema was empty; I have not been able to clarify this, since I had not yet installed XDebug. I'll look at this again.

Regards,

Patrick

---

Subject: Re: Check for empty \$schema in dml.psql.class.inc

Posted by [AJM](#) on Fri, 08 Feb 2019 10:11:55 GMT

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

If you could find out by using your debugger what is going wrong then that would be a great help.

---

Subject: Re: Check for empty \$schema in dml.psql.class.inc

Posted by [pdv](#) on Sun, 10 Feb 2019 16:16:34 GMT

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

Unfortunately I cannot reproduce the problem without first removing my test-database (bxtests): I

have reactivated the check for !empty(\$schema) and there is no more problem. But I notice that &\_getDBMSengineByHost() is not called anymore, just &\_getDBMSengine() and that might explain the difference.

The problem occurred when I tried to import my bxtests database for the very first time, using the 2-server configuration I've send previously.

Luckily I kept the relevant error-message which shows that \_\_construct is called with an empty \$dbname. And indeed I don't see right away where in &\_getDBMSengineByHost(), \$args['dbname'] has been set before calling the constructor.

2019-01-26 21:59:33

Fatal Error: No value supplied for PGSQL\_dbname (# 256)

Error in line 93 of file '/Library/WebServer/Documents/radicore/includes/dml.pgsql.cl ass.inc'.

PHP\_SELF: /radicore/dict/database(add1)import.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/database(add1)import.php?csrf\_id=5c4cd84538cc  
89.99819281&session\_name=menu1

Page Stack:

[0] main\_menu

[1] dictionary

[2] dict\_database(list1)

[3] dict\_database(add1)import

Backtrace:

0: array =

function: string = errorHandler

args: array =

0: integer = 256

1: string = No value supplied for PGSQL\_dbname

2: string = /Library/WebServer/Documents/radicore/includes/dml.pgsql.cla ss.inc

3: integer = 93

4: array =

args: array =

dbhost: string = localhost

dbusername: string = theusername1

dbuserpass: string = thepassword1

dbprefix: string =

PGSQL\_dbname: string = bxtests

```

    dbname: string = null
    dbport: string = null
    dbsocket: string = null
    schema: string = null
1: array =
file: string = /Library/WebServer/Documents/radicore/includes/dml.pgsql.class.inc
line: integer = 93
function: string = trigger_error
args: array =
    0: string = No value supplied for PGSQL_dbname
    1: integer = 256
2: array =
file: string = /Library/WebServer/Documents/radicore/includes/std.singleton.php5.inc
line: integer = 131
function: string = __construct
class: string = pgsql
type: string = ->
args: array =
    0: array =
        dbhost: string = localhost
        dbusername: string = theusername1
        dbuserpass: string = thepassword1
        dbprefix: string =
        PGSQL_dbname: string = bxtests
        dbname: string = null
        dbport: string = null
        dbsocket: string = null
3: array =
file: string = /Library/WebServer/Documents/radicore/dict/classes/dict_data_base_s01.class.inc
line: integer = 336
function: string = getInstance
class: string = RDCsingleton
type: string = ::
args: array =
    0: string = pgsql
    1: array =
        dbhost: string = localhost
        dbusername: string = theusername1
        dbuserpass: string = thepassword1
        dbprefix: string =
        PGSQL_dbname: string = bxtests
    2: boolean = 1
4: array =
file: string = /Library/WebServer/Documents/radicore/dict/classes/dict_data_base_s01.class.inc
line: integer = 166
function: string = _getDBMSengineByHost
class: string = dict_dataBase_s01
type: string = ->

```

```
args: array =  
  0: string = localhost  
5: array =  
file: string = /Library/WebServer/Documents/radicore/dict/classes/dict_data base_s01.class.inc  
line: integer = 90  
function: string = _loadDatabases  
class: string = dict_dataBase_s01  
type: string = ->  
args: array =  
  0: string = localhost  
  1: array =  
  2: string =  
  3: array =  
  4: array =  
  public: string = bxtests  
6: array =  
file: string = /Library/WebServer/Documents/radicore/includes/std.table.cla ss.inc  
line: integer = 3276  
function: string = _cm_getInitialData  
class: string = dict_dataBase_s01  
type: string = ->  
args: array =  
  0: array =  
7: array =  
file: string = /Library/WebServer/Documents/radicore/includes/std.add1.inc  
line: integer = 60  
function: string = getInitialData  
class: string = Default_Table  
type: string = ->  
args: array =  
  0: string =  
8: array =  
file: string = /Library/WebServer/Documents/radicore/dict/database(add1)imp ort.php  
line: integer = 9  
args: array =  
  0: string = /Library/WebServer/Documents/radicore/includes/std.add1.inc  
function: string = require_once
```

---

---

Subject: Re: Check for empty \$schema in dml.psql.class.inc

Posted by [AJM](#) on Mon, 11 Feb 2019 09:43:25 GMT

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

---

I have examined the amended code for dict\_database\_s01.class.inc which you sent me, and it actually contains a bug. The first argument in the \_loadDatabases() method you changed from \$dbhost to \$server, yet the first line of the method is still referring to \$dbhost. As it has not been defined the first condition always fails.

---

---

Subject: Re: Check for empty \$schema in dml.psql.class.inc

Posted by [pdv](#) on Tue, 12 Feb 2019 09:44:55 GMT

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

---

Yes, of course, that explains why the problem with the missing \$schema went away. I've replaced \$dbhost by \$server in the check and corrected another bug (see enclosed patch) and now the behaviour is as before: when importing databases, \_loadDatabases() is called and eventually the constructor of dml.psql.class.inc is called with an empty \$args['dbname'].

In \_getDBMSEngineByServer (previously \_getDBMSEngineByHost) \$dbname\_array and \$switch\_dbnames are present but not passed.

But no dbname (or schema) is actually needed in this case.

---

#### File Attachments

1) [patch0002a.diff](#), downloaded 861 times

---

---

Subject: Re: Check for empty \$schema in dml.psql.class.inc

Posted by [AJM](#) on Wed, 13 Feb 2019 09:19:52 GMT

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

---

I cannot work with diff files, so can you send me the amended class file?

---

---

Subject: Re: Check for empty \$schema in dml.psql.class.inc

Posted by [pdv](#) on Wed, 13 Feb 2019 09:44:33 GMT

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

---

Here it comes.

---

#### File Attachments

1) [dict\\_database\\_s01.class.inc](#), downloaded 886 times

---

---

Subject: Re: Check for empty \$schema in dml.psql.class.inc

Posted by [AJM](#) on Thu, 21 Feb 2019 23:30:49 GMT

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

---

I have recreated the situation which causes this problem on my PC and stepped through it with my debugger. I have made some changes which are different from yours, but which fix the problem. See the attached zip file. Check it out and see if it works for you.

---

#### File Attachments

1) [RADICORE-patch-2019-02-21.zip](#), downloaded 792 times

---

Subject: Re: Check for empty \$schema in dml.psql.class.inc  
Posted by [pdv](#) on Mon, 25 Feb 2019 20:21:14 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Seems OK to me.

---

---

Subject: Re: Check for empty \$schema in dml.psql.class.inc  
Posted by [AJM](#) on Tue, 26 Feb 2019 09:39:47 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Great stuff! I will include this in the next release.

---