

Re: Res: Res: RE: Problems changing from 1.30 to 1.42

Source: <http://coding.derkeiler.com/Archive/Perl/perl.dbi.users/2004-05/0142.html>

From: Tim Bunce (Tim.Bunce_at_pobox.com)

Date: 05/17/04

Date: Mon, 17 May 2004 20:34:07 +0100
To: Pedro Anisio <pedroanisio@ibestvip.com.br>

On Mon, May 17, 2004 at 11:22:35AM -0300, Pedro Anisio wrote:

> *I used trace() but still cannot figure out what is happening, but still dont*
> *work :-(*
>
> *I'm sending attached the log files and the program.*

working:

```
-> prepare for DBD::mysql::db (DBI::db=HASH(0x8925a28)~0x891f5a0
  'update mensagem set mens_stat = 2 where mens_codigo = 350027984 and mens_stat = 1')
<- prepare= DBI::st=HASH(0x893ea20) at teste_update.pl line 20
-> execute for DBD::mysql::st (DBI::st=HASH(0x893ea20)~0x893ead4)
<- dbd_st_execute 1 rows
```

not working:

```
-> prepare for DBD::mysql::db (DBI::db=HASH(0x8925a5c)~0x891f5e4
  'update mensagem set mens_stat = ? where mens_codigo = ? and mens_stat = ?')
<- prepare= DBI::st=HASH(0x893ea34) at teste_update.pl line 20
-> execute for DBD::mysql::st (DBI::st=HASH(0x893ea34)~0x893eae8 2 350027984 1)
  Binding parameters: update mensagem set mens_stat = '2' where mens_codigo = '350027984' and
mens_stat = '1'
<- dbd_st_execute 0 rows
```

I suspect if you change the working one to put quotes around the values in the WHERE clause it'll break.

This aspect of the change log for 2.9002 applies:

- * Fixed bug where strings that were used in numeric context were not getting quoted on execute(). Now all parameters are bound as varchar by default.

- **NOTE** this is a change in behaviour that MAY cause problems with some SQL statements. If quoted integers, for example, cause any problems, use `bind_param(<column_id>, undef, SQL_INTEGER)` to force a column to be bound as an integer.

perl.dbi.users: Re: Res: Res: RE: Problems changing from 1.30 to 1.42

Out of interest, what's the definition of the mensagem table?

Tim.