

Title

PEAR DB: What's New in 1.6.0

[New York PHP](#) Monthly Meeting
February 24, 2004

by Daniel Convissor
[The Analysis and Solutions Company](#)

Introduction to PEAR DB

- PEAR: PHP Extension and Application Repository
- Object oriented API
- Handles 13 of PHP's database extensions

dBase, FrontBase, InterBase, Informix, mSQL, MS SQL Server, MySQL, MySQLi, Oracle, ODBC (tested DB2 & Access), PostgreSQL, SQLite, Sybase

Please Note:

- Sample code has been significantly simplified
- If viewing on the web: set browser to full screen / kiosk mode and change font size / zoom until this page fits comfortably in the whole screen
- Opera 7 Win32 users: our style sheet may cause rendering delays, even though it's valid
- Footer placement works right in Mozilla on multiple OS's, but IE and Opera only on Windows

Improvements in DB 1.6.0

- Really works for more than just MySQL.
- New portability features, making it possible to write applications which can be easily ported between DBMS's.
- Prepare/execute works the same way for all DBMS's and allows escaping of placeholder characters.
- PHP 5 compatibility.
- PHP must be at version 4.2.0 or higher.

Improvements in DB 1.6.0 cont'd

- Introduced *experimental* mysqli driver
- Deployed tableInfo() in more drivers and finalized the move from DB_result to DB_common.
- Added quoteSmart() and quoteSimple(), deprecating the inconsistent quote() and quoteString().
- Improved error reporting.
- Countless documentation corrections.

connect.inc

```
<?php
require_once 'DB.php';

// User, password and db variables defined elsewhere for obvious reasons.

$mysql =& DB::connect("mysql://$muser:$mpw@$mdb");
if (DB::isError($mysql)) {
    die($mysql->toString());
}

$oracle =& DB::connect("oci8://$ouser:$opw@$odb");
if (DB::isError($oracle)) {
    die($oracle->toString());
}

$pgsql =& DB::connect("pgsql://$puser:$ppw@$pdb");
if (DB::isError($pgsql)) {
    die($pgsql->toString());
}

$sqlite =& DB::connect("sqlite:///sdb?mode=0666");
if (DB::isError($sqlite)) {
    die($sqlite->toString());
}
?>
```

Portability: Lowercasing

```
require_once './examples/connect.inc';

$$dbms->query('CREATE TABLE tbl (Cf CHAR(10))');
$$dbms->query("INSERT INTO tbl VALUES ('DOH!')");
$$dbms->setFetchMode(DB_FETCHMODE_ASSOC);
$row =& $$dbms->getRow('SELECT Cf FROM tbl');

if (isset($row['cf'])) {
    echo 'mmm... sweet candy...';
} else {
    print_r($row);
}

$$dbms->query('DROP TABLE tbl');
```

Default Behavior

<code>\$dbms = 'mysql';</code>	<code>\$dbms = 'pgsql';</code>
Array ([Cf] => DOH!)	mmm... sweet candy...

```
$$dbms->setOption('portability', DB_PORTABILITY_LOWERCASE);
```

<code>\$dbms = 'mysql';</code>	<code>\$dbms = 'pgsql';</code>
mmm... sweet candy...	mmm... sweet candy...

Portability: Right Trim

```
require_once './examples/connect.inc';

$$dbms->query('CREATE TABLE tbl (c CHAR(10))');
$$dbms->query("INSERT INTO tbl VALUES ('one')");
$c =& $$dbms->getOne('SELECT c FROM tbl');

if ($c == 'one') {
    echo 'Time for a Scooby Snack!';
} else {
    echo "ZOINKS! \$c='$c' strlen=" . strlen($c);
}

$$dbms->query('DROP TABLE tbl');
```

Default Behavior

<code>\$dbms = 'oracle';</code>	<code>\$dbms = 'mysql';</code>
ZOINKS! \$c='one' strlen=10	Time for a Scooby Snack!

```
$$dbms->setOption('portability', DB_PORTABILITY_RTRIM);
```

<code>\$dbms = 'oracle';</code>	<code>\$dbms = 'mysql';</code>
Time for a Scooby Snack!	Time for a Scooby Snack!

Portability: Null to Empty

```
require_once './examples/connect.inc';

$$dbms->query('CREATE TABLE WMDs (c CHAR(10) NULL)');
$$dbms->query("INSERT INTO WMDs VALUES ('')");
$c =& $$dbms->getOne('SELECT c FROM WMDs');

if (isset($c)) {
    echo 'Vote! And organize others to vote!';
} else {
    echo "W's scruples: " . gettype($c);
}

$$dbms->query('DROP TABLE WMDs');
```

Default Behavior

<code>\$dbms = 'oracle';</code>	<code>\$dbms = 'pgsql';</code>
W's scruples: NULL	Vote! And organize others to vote!

```
$$dbms->setOption('portability', DB_PORTABILITY_NULL_TO_EMPTY);
```

<code>\$dbms = 'oracle';</code>	<code>\$dbms = 'pgsql';</code>
Vote! And organize others to vote!	Vote! And organize others to vote!

Portability: Error Mapping

- **MySQL unique/primary key violations:**
already exists -> constraint violation
- **MySQL not-null violations:**
constraint violation -> null value violates not-null constraint
- **MS Access ODBC bogus field:**
mismatch -> no such field
- In 1.6.0 this must be set during connect().
- 1.6.1 will allow configuration via setOption().

Portability: Error Mapping Example

```
require_once './examples/connect.inc';

$$dbms->query('CREATE TABLE tbl (c CHAR(10) NOT NULL)');
$result =& $$dbms->query('INSERT INTO tbl VALUES (NULL)');

if ($result->getMessage() ==
    'DB Error: null value violates not-null constraint')
{
    echo 'Phew, the pointy-haired guy went away!';
} else {
    echo 'AAAAGH! ' . $result->getMessage();
}

$$dbms->query('DROP TABLE tbl');
```

Default Behavior

<code>\$dbms = 'mysql';</code>	<code>\$dbms = 'oracle';</code>
AAAAGH! DB Error: constraint violation	Phew, the pointy-haired guy went away!

```
$$dbms->setOption('portability', DB_PORTABILITY_ERRORS);
```

<code>\$dbms = 'mysql';</code>	<code>\$dbms = 'oracle';</code>
Phew, the pointy-haired guy went away!	Phew, the pointy-haired guy went away!

Portability: Delete Count

```
require_once './examples/connect.inc';

$$dbms->query('CREATE TABLE tbl (c CHAR(10))');
$$dbms->query("INSERT INTO tbl VALUES ('one')");
$$dbms->query('DELETE FROM tbl');
$count = $$dbms->affectedRows();

if ($count != 0) {
    echo 'Yes!';
} else {
    echo 'Does everything have to be funny?';
}

$$dbms->query('DROP TABLE tbl');
```

Default Behavior

<code>\$dbms = 'sqlite';</code>	<code>\$dbms = 'pgsql';</code>
Does everything have to be funny?	Yes!

```
$$dbms->setOption('portability', DB_PORTABILITY_DELETE_COUNT);
```

<code>\$dbms = 'sqlite';</code>	<code>\$dbms = 'pgsql';</code>
Yes!	Yes!

Portability: Usage

Portability options can be set when connecting...

```
<?php
$options = array(
    'portability' => DB_PORTABILITY_ALL,
);

$db =& DB::connect('pgsql:///foo', $options);
if (DB::isError($db)) {
    die($db->getMessage());
}
?>
```

... or during runtime...

```
<?php
$db->setOption('portability', DB_PORTABILITY_ALL);
?>
```

Portability: Constants are Bitwised

- Portability mode constants are bitwised.
- Can combine them using `|` and remove them using `^`.
- Turn on all portability options:

```
$db->setOption('portability', DB_PORTABILITY_ALL);
```

- Enable lowercasing and trimming:

```
$db->setOption('portability', DB_PORTABILITY_LOWERCASE |  
DB_PORTABILITY_RTRIM);
```

- Enable all options except trimming:

```
$db->setOption('portability', DB_PORTABILITY_ALL ^  
DB_PORTABILITY_RTRIM);
```

Portability: Backwards Compatibility

- Portability used to be handled by the `optimize` option.
- `optimize` is now deprecated.
- Old scripts work under the new system.
- If `optimize` gets set to `portability` the following drivers go into these modes:
 - `oci8`: `DB_PORTABILITY_LOWERCASE` and `DB_PORTABILITY_DELETE_COUNT`
 - `fbsql`, `mysql`, `mysqli`, `sqlite`: `DB_PORTABILITY_NUMROWS`
- If `optimize` gets set to `performance` new portability system is switched off.

Important tableInfo() Changes

- Finalized move to the DB_common class.
- Added to Sybase and Informix.

Examine a table by passing a table name:

```
<?php
$info = $db->tableInfo('tablename');
print_r($info);
?>
```

Probe a query result by passing a DB_result object:

```
<?php
$result =& $db->query('SELECT * FROM tablename');
$info = $db->tableInfo($result);
print_r($info);
?>
```

Error Code Fixes

- `mssql`: `errorCode()` returns DB's code instead of SQL Server's code. Added `errorNative()` to get Server's code.
- Added mappings in several drivers to provide consistency between `ibase`, `ifx`, `mssql`, `oci8`, `odbc(db2)`, `pgsql`, `sqlite` and `sybase`.
- If error code portability is turned on, that consistency is expanded to `mysql`, `mysqli` and `odbc(access)`.
- Added `DB_ERROR_CONSTRAINT_NOT_NULL` for handling null values in NOT NULL columns.

More Information About PEAR DB

- Home Page: <http://pear.php.net/package/DB>
- [Manual](#)
- [Change Log](#)
- [Installation Instructions](#)
- CVS HTTP: <http://cvs.php.net/cvs.php/pear/DB>
- CVS:

```
cvs -d:pserver:cvsread@cvs.php.net:/repository login  
# password is: phpfi  
cvs -d :pserver:cvsread@cvs.php.net:/repository co pear/DB
```