

# The Plot to Kill PHP MySQL Extension

PHP core developers are planning to kill the PHP original MySQL extension.

If you are using MySQL in your PHP applications for a long time, this may seriously affect you.

## The Plot

- Yes, you read it right. Recently, Phillip Olson sent to the PHP internals mailing list a proposal to kill the original PHP MySQL extension in future PHP versions.
- Well, as you may have read, the idea is not to kill this very popular PHP extension right now in the upcoming PHP 5.4.

# What's Happening

- The idea is to first mark its functions as deprecated in the PHP documentation. The intention is to educate PHP developers to migrate their code to use the mysqli or the PDO extensions instead, or the mysqlnd, why not?

## Need to move

- In future versions, say PHP 5.5 or 5.6 common calls to functions like `mysql_pconnect`, `mysql_query`, etc.. will throw ugly `E_DEPRECATED` notices. And eventually in PHP 6 or later, if it will ever happen, code that uses those functions will be removed from the main PHP distribution permanently.

# Deprecated

## mysql\_db\_query

(PHP 4, PHP 5)

mysql\_db\_query — Selects a database and executes a query on it

### Description

```
resource mysql_db_query ( string $database , string $query [, resource $link_identifier ] )
```

**mysql\_db\_query()** selects a database, and executes a query on it.



### Warning

This function has been **DEPRECATED** as of PHP 5.3.0. Relying on this feature is highly discouraged.

## PDO or mysqli ?

- PHP Data Objects is for people starting from scratch who want an easy way to integrate MySQL into a PHP application.
- Mysqli is an improved version of the old mysql extension and can be used as a replacement without too much work.

- [PDO Drivers](#)

- [CUBRID \(PDO\)](#) — CUBRID Functions (PDO\_CUBRID)
- [MS SQL Server \(PDO\)](#) — Microsoft SQL Server and Sybase Functions (PDO\_DBLIB)
- [Firebird/Interbase \(PDO\)](#) — Firebird/Interbase Functions (PDO\_FIREBIRD)
- [IBM \(PDO\)](#) — IBM Functions (PDO\_IBM)
- [Informix \(PDO\)](#) — Informix Functions (PDO\_INFORMIX)
- [MySQL \(PDO\)](#) — MySQL Functions (PDO\_MYSQL)
- [MS SQL Server \(PDO\)](#) — Microsoft SQL Server Functions (PDO\_SQLSRV)
- [Oracle \(PDO\)](#) — Oracle Functions (PDO\_OCI)
- [ODBC and DB2 \(PDO\)](#) — ODBC and DB2 Functions (PDO\_ODBC)
- [PostgreSQL \(PDO\)](#) — PostgreSQL Functions (PDO\_PGSQL)
- [SQLite \(PDO\)](#) — SQLite Functions (PDO\_SQLITE)
- [4D \(PDO\)](#) — 4D Functions (PDO\_4D)

# PDO Pro's

- Native to PHP as of 5.x
- Supports named parameters as opposed to numerically indexed ?'s
- Same abstraction library supports multiple different RDBM's
- based on its syntax, style, etc.

## Mysqli Con's:

- Has issues with properly storing and retrieving large objects in the database.
- No support for named parameters.
- Otherwise, both libraries are basically different flavors of the same thing. They both have functions to quote parameters and both support parameterized queries.

# Mysqli Pro's

- Almost a direct drop in replacement for the old mysql extension.
- Faster than old mysql extension