PHP extensions per engine

Which PHP extension each database needs (sqlite3, pgsql, mongodb with its Composer library, mysqli, oci8), where it comes from on the common Linux families, how the setup wizard checks it, and how to confirm that the...

Source file: `doc/eng-us/help/database-administrators/php-extensions-per-engine.md` Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/database-administrators/php-extensions-per-engine.md

This page is for database administrators who prepare a server, or who review a host's PHP build, before an Exponential install. It answers: which PHP extension each engine needs, where to get it, how the installers check for it, and how to make sure every PHP that touches the database has it. Source: Database support, PHP extensions and `settings/setup.ini`.

Contents

1. One extension per engine
2. How the setup wizard checks
3. Several PHPs on one server
4. MongoDB: extension and library
5. Oracle: oci8 and the client
6. Checklist

One extension per engine

Engine	Extension	Where it comes from
SQLite 3	`sqlite3`	Bundled with PHP; RHEL family `php-pdo` (it also carries `pdo_sqlite`), Debian and Ubuntu `php<version>-sqlite3`. `pdo_sqlite` is not needed by the kernel.
PostgreSQL	`pgsql`	Bundled with PHP; `php-pgsql`, `php<version>-pgsql`
MongoDB	`mongodb` (PECL) and the Composer library `mongodb/mongodb`	`pecl install mongodb` or `php-pecl-mongodb`; the library with `composer require mongodb/mongodb`
MariaDB, MySQL	`mysqli`	Bundled with PHP (mysqlnd); `php-mysqlnd`, `php<version>-mysql`
Oracle	`oci8` (PECL)	PECL oci8 3.4 for PHP 8.4 and 8.5, 3.2 for PHP 8.1 to 8.3, built against an Oracle client; EPEL, Remi or your vendor

Install only the one your site uses. Each driver stops early when its extension is missing; the SQLite driver, for example, logs "SQLite3 extension was not found, the DB handler will not be initialized." (9.2).

How the setup wizard checks

`settings/setup.ini` defines a critical test:

INI

```ini
[database_extensions]
Extensions=sqlite3;mysqli;pgsql;mongodb;oci8
Require=one
```

The test passes when at least one of them is loaded. The optional test `database_all_extensions` lists the ones that are missing; it is information only. If `sqlite3` is missing, the wizard chooses the first database whose extension is loaded and says so. The wizard offers Oracle only when `oci8` is loaded and `extension/ezoracle` is present.

Several PHPs on one server

The PHP that serves the web site and the PHP on the command line are configured separately, and Velocity may run yet another binary. Check each one:

Shell

```shell
php -m | grep -iE 'sqlite3|pgsql|mysqli|mongodb|oci8'      # the command line PHP
php --ri sqlite3 | grep -i library                           # the SQLite library version (3.33 o
r later)
```

For the web server's PHP, open **Setup > System information**: the PHP card names the SAPI and the Database card shows the driver class that answered (System information). The same page opened through Velocity's port reports Velocity's PHP. On a Plesk or cPanel server each PHP version has its own `php.ini` and its own list of modules; see PHP selection and FPM pools.

Cronjobs and console commands use the command line PHP. A site whose pages work but whose cronjobs fail with a missing driver usually has the extension in PHP-FPM only.

MongoDB: extension and library

The adapter `expMongoDB` uses the PECL extension `mongodb` and the library's `MongoDB\Client`. `composer.json` only suggests the library, so a plain install does not have it. Without it the error is `Class "MongoDB\Client" not found` (12.10). Requirements:

- PHP 8.2 or later (the only driver with a higher floor than PHP 8.0).
- Extension `mongodb` 1.18 or later.
- Library `mongodb/mongodb`; CI installs `^1.21 || ^2.0`.

Oracle: oci8 and the client

`oci8` must be built against an Oracle client. The Instant Client is enough, and `ORACLE_HOME` is not needed with it. Set `LD_LIBRARY_PATH` (and `TNS_ADMIN` for TNS aliases) only when the client is outside the system library path, and then in the PHP-FPM pool (`env[...]`) and in Velocity's environment as well, because each process reads them once (9.6). The `php.ini` settings `oci8.persistent_timeout`, `oci8.max_persistent` and `oci8.ping_interval` govern persistent connections and cannot be changed at run time (Oracle: persistent connections, DRCP and tuning).

Checklist

- ☐ The extension of the chosen engine is loaded in the web server's PHP.
- ☐ The same extension is loaded in the command line PHP (cronjobs, console).
- ☐ Under Velocity, the same extension is loaded in Velocity's PHP.
- ☐ SQLite: the library in PHP is 3.33 or later.
- ☐ MongoDB: PHP 8.2+, extension 1.18+, library `mongodb/mongodb` installed.
- ☐ Oracle: `oci8` matches the PHP version; the client libraries are found by every process.
- ☐ Extensions for engines the site does not use are not installed.

Related pages

- Supported engines and versions
- PHP versions and extensions (for hosting companies)
- Chapter 2, Requirements
- Free documentation: this page is free to copy, translate and improve under the GNU FDL 1.3.

Become an Exponential Partner

Partners share their information, provide funding, contribute back and develop the products with us, and that is how Exponential grows. A partner website and dashboard are planned for mid-2027: please become a partner today to speed up our growth rate and land new features faster.

Partners: start here | How to apply

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.