## Do you want to contribute?

This is a README for developers wanting to contribute to the MongoDB PHP

driver.

## Current branches

\* v1.6 is the current stable branch, critical fixes only allowed here

\* master is the development branch for new features

## Github

The official repository of this driver is on

[Github](http://www.github.com/mongodb/mongo-php-driver/). When fixing a bug

in the current release branch, please ensure you branch from it, and submit the

Pull Request against it again. If there has been a while since you update your

local fork, please ensure you rebase it properly :)

Bugfixes and Pull Requests should be filed against the \*lowest release branch\*

All \*new\* development happens in master (i.e. new features, BC changes, ..).

## Running the tests

To run the test, you'll have to configure the tests/utils/cfg.inc file (copy

the [tests/utils/cfg.inc.template](tests/utils/cfg.inc.template) to

tests/mongo-test-cfg.inc and edit it).

The testing framework bootstraps mongod environments (standalone, replicaset,

sharding, authentication...) with the help of the mongo shell utility, on

non-standard ports, so there is no need to worry about overwriting your local

environment.

To boot up the environment run:

$ make servers

And to tear it down again, after running the tests, run:

$ make stop-servers

If you'd only like to bootup (and run) specific set of setups you can enable

individual setups by defining an environment variable called

MONGO\_SERVER\_[STANDALONE|REPLICASET|MONGOS]=yes before executing 'make servers'.

And finally, to execute the tests run:

$ make test

## Writing tests

Test templates are available in [tests/templates/\*.phpt](tests/templates/).

These target dedicated MongoDB environments, such as a ReplicaSet or a

standalone MongoDB with authentication, or over mongos.

## Code coverage

Code coverage is generated by running the test suite after having enabled

coverage before compiling the driver

$ phpize

$ ./configure --enable-coverage

$ make clean all test

Then you should have a coverage/ folder with the juicy details.