# Contributing to HHVM

We'd love to have your help in making HHVM better. Before jumping into the code, please familiarize yourself with our [coding conventions](hphp/doc/coding-conventions.md). We're also working on a [Hacker's Guide to HHVM](hphp/doc/hackers-guide). It's still very incomplete, but if there's a specific topic you'd like to see addressed sooner rather than later, let us know. For documentation and any other problems, please open an [issue](https://github.com/facebook/hhvm/issues), or better yet, [fork us and send a pull request](https://github.com/facebook/hhvm/pulls). Join us on Freenode in [#hhvm](https://webchat.freenode.net/?channels=hhvm) for general discussion, or [#hhvm-dev](https://webchat.freenode.net/?channels=hhvm-dev) for development-oriented discussion.

If you want to help but don't know where to start, try fixing some of the ["probably easy" issues](https://github.com/facebook/hhvm/issues?q=is%3Aopen+is%3Aissue+label%3A%22probably+easy%22); add a test to hphp/test/slow/something\_appropriate, and run it with hphp/test/run.

All the open issues tagged [PHP5 incompatibility](https://github.com/facebook/hhvm/issues?labels=php5+incompatibility&page=1&state=open) are real issues reported by the community in existing PHP code and [frameworks](https://github.com/facebook/hhvm/wiki/OSS-PHP-Frameworks-Unit-Testing:-General) that could use some attention.

## Code of Conduct

The code of conduct is described in [`CODE\_OF\_CONDUCT.md`](CODE\_OF\_CONDUCT.md)

## Submitting Pull Requests

Before changes can be accepted a [Contributor Licensing Agreement](https://code.facebook.com/cla) must be completed. You will be prompted to accept the CLA when you submit your first pull request. If you prefer a hard copy, you can print the [pdf](https://github.com/facebook/hhvm/raw/master/hphp/doc/FB\_Individual\_CLA.pdf), sign it, scan it, and send it to <cla@fb.com>.

Please add appropriate test cases as you make changes, and make sure that they pass locally before submitting your pull request; see [here](hphp/test/README.md) for more information. All the tests are run via Phabricator, however testing locally greatly speeds up the process of accepting your changes.

### Stable Version Updates

We maintain up to three [stable branches](https://github.com/facebook/hhvm/wiki/Release%20Schedule) at once (the current release plus two [LTS branches](https://github.com/facebook/hhvm/wiki/Long-term-support-%28LTS%29)). To get a fix into one of those branches, first get accepted into master, as described above. Fixes are merged into master and then merged backwards into stable releases as appropriate.

Then, submit another PR against the relevant stable branch(es) cherry-picking your change into that branch, with any changes needed to properly backport. Make sure to explain in the PR summary why the change should be considered for inclusion in the stable branch -- basically, make the case for why the issue the change is fixing is worse than the possible risk of what the change might break (and thus what \*we\* will be responsible for debugging, fixing, and maintaining).

## Quick Links

\* IRC: [#hhvm](https://webchat.freenode.net/?channels=hhvm) and [#hhvm-dev](https://webchat.freenode.net/?channels=hhvm-dev) on Freenode.

\* [Issue tracker](https://github.com/facebook/hhvm/issues)