

Contributor Workflow

New to the Project?

[Here](#) are some issues that newbies can try.

Getting the Source

Before starting, ensure that a recent version of Maven is installed.

```
git clone https://git-wip-us.apache.org/repos/asf/helix.git
cd helix
# if you need to work on a specific branch, like helix-0.6.x:
# git checkout helix-0.6.x
./build
```

If you use Eclipse, be sure to run the following:

```
mvn eclipse:eclipse
```

Opening an Issue

All code changes in Helix must have an open issue attached to them prior to code review. To create an issue:

- Visit <https://issues.apache.org/jira/browse/HELIX>
- Click "Create Issue"
- Create an issue with a minimum of a summary and description
- If you plan on making the associated code change, assign it to yourself

If you plan on taking on an existing issue, visit the corresponding issue page and assign it to yourself.

Coding

Style

Please follow the [Coding Style](#) guidelines. The easiest method is to simply apply helix-style.xml to your IDE to allow for automatic formatting.

Submitting code for review

- Ensure that you have an account on <http://reviews.apache.org>
- Commit your local changes, and prefix the commit with the issue tag:

```
git commit -a -m "[HELIX-NNN] Description of the code change"
```

- Ensure that Review Board utilities are installed by following directions here: <http://www.reviewboard.org/docs/manual/dev/users/tools/post-review/>
- Run the review submission script:

```
./hpost-review.sh HEAD^..HEAD NNN
```

Replace the commit range with the appropriate range and NNN with the actual issue number.

- On Review Board, if there are specific project members who should take a look at the issue, add them to the reviewer list. Also, be sure to add a description of any testing that was done
- Submit the review

If there are changes to be made from the review, do the following:

- Unstage the commit:

```
git reset --soft HEAD^
```

- Make the changes
- Commit everything again locally:

```
git commit -a -m "[HELIX-NNN] Description of the code change"
```

- Create a patch:

```
git format-patch HEAD^
```

- In Review Board, click Update --> Update Diff and attach the new patch

Submit the patch

Once you receive a "Ship It" from one of the project members, you can do the following to submit the patch:

- Create a patch:

```
git format-patch HEAD^
```

- Upload the patch to the issue page, i.e. <https://issues.apache.org/jira/browse/HELIX-NNN>
- Comment on the issue detailing the fix and linking back to the code review