

# Getting Involved

There are many ways you can get involved in CXF:

1. Participate on the mailing lists. Propose ideas. Comment on others ideas.
2. Look at the open JIRA issues
3. Provide feedback on the current code
4. Take a look at some of the ideas below

## CODING IDEAS FOR CXF NEWCOMERS

There are many interesting areas of CXF that you could potentially work on. Some ideas:

- WS-Context & Session support
- An invoker for Ode which uses CXF
- A HTML Form based "tester" for WebServices
- XMPP/Jabber transport
- Increasing unit test coverage. Adding unit tests for areas that are not covered by current test cases is always valuable to the project.
- Support for Web Service Definition Language (WSDL) 2.0
- Castor databinding
- Other WS-\* support; e.g., Quality of Service (WS-Atomic Transactions and WS-Coordination), bootstrapping (WS-MetaDataExchange), WS-BusinessActivity, WS-Eventing and WS-Transfer
- Add visitors for MongoDB and Solr/ ElasticSearch
- See the Roadmap and jump in and help

## HOW TO SUBMIT A PULL REQUEST

- Open a Jira issue
- Fork the github cxf mirror
- In most cases you should base your changes on the master branch. The committers can backport to the maintenance branches then
- Create a new branch named like the JIRA issue you want to edit (e.g CXF-6738)
- Make your changes, test, and build successfully
- Ideally put all changes into one commit. The commit should contain the issue id (e.g [CXF-6738] Replace synchronized blocks ... )
- Push the change to your forked repo into your branch
- If the commit is named like above it will automatically show up in the JIRA issue making it easier to see what changes belong to the issue
- If you need to do changes to your pull request then you should ideally rewrite your commit and do a push -f to your own branch

## APPLYING A PULL REQUEST

- Merge the pull request into the branch it is based on
- Make sure the build works

- Make sure the commits refer to the issue they solve

## **BECOMING A COMMITTER**

- First off, read about How the ASF works. Most importantly, the sections on Meritocracy and Roles. That provides a bit of background.
- The important part is that you need to **earn** the right to be a committer, it's not something we'll give you just because your name is James Gosling. To earn the right, you need to get involved. (see top section above)
- If you become involved, participate in email discussions, submit patches, etc... the current devs may invite you to become a committer through a vote. If the vote passes, that will trigger a bunch of things such as submitting a CLA, creating accounts, etc....

*Hint:* submitting pull requests to Jira issues is the best way. It shows that you are digging into the code, are following best practices, writing tests, etc.... It also annoys the developers to constantly have to review patches and if your patches are all acceptable, they'll start the process to grant committership just to stop having to review patches. 🍌