Git guide

Git guide

Topics

- Getting started with Git: if you have never played with Git before, this is for you!
- Git IzPack workflow: how one can use Git to work on IzPack.

IzPack Git repositories

The "blessed" IzPack repository is available from https://xircles.codehaus.org/projects/izpack/repos/primary/repo

- Anonymous access:
  - via the Git protocol: git://git.codehaus.org/izpack.git
  - via HTTP: http://git.codehaus.org/gitweb.cgi?p=izpack.git
- Developer access:
  - `ssh://git@git.codehaus.org/izpack.git` (you will need to upload a **SSH DSA2** public key to your account details).

The following is a list of official IzPack developers forks:

- http://github.com/jponge/izpack (Julien Ponge, project leader, project founder and tyrannic despot)
- http://github.com/bonnefoa/izpack-refactoring (Anthonin Bonnefoy, project despot and *hazardous merges master*)
- http://github.com/dduponchel/IzPack (David Duponchel, project developer)

The following is a list of forks from various people around the globe (do not hesitate to add yourself here):

- *add yourself here!*

> **Pull requests on GitHub**
>
> A quick and efficient way to collaborate with an IzPack developer that has a repository on GitHub is to:
>
> 1. fork his/her repository, and
> 2. send **pull requests** to propose your changes.
>
> Pull requests on GitHub are really useful as they make **code reviews** painless!

Funny tip if you are a Subversion fanatic: GitHub repositories can be accessed read-only from Subversion.

Historical considerations

A complete Git conversion of the old Subversion repository can be found at http://github.com/jponge/izpack-full-svn-history-copy (keep in mind that because it is based on the full repository, there are no branches and tags informations, just a linear branch which maps the Subversion revisions).

IzPack first used CVS, then switch in 2004 to Subversion. It switched again, this time to Git, in 2010.