Bazaar in five minutes

Release 2.8.0dev1

Bazaar Developers

August 11, 2021

Contents

1 Introduction

Bazaar is a distributed version control system that makes it easier for people to work together on software projects.

Over the next five minutes, you'll learn how to put your files under version control, how to record changes to them, examine your work, publish it and send your work for merger into a project's trunk.

2 Installation

This guide doesn't describe how to install Bazaar but it's usually very easy. You can find installation instructions at:

- **GNU/Linux:** Bazaar is probably in your GNU/Linux distribution already.
- **Windows:** installation instructions for Windows.
- **Mac OS X:** installation instructions for Mac OS X.

For other platforms and to install from source code, see the Download and Installation pages.

3 Introducing yourself

Bazaar records changes to source code, and it records who made the change. The person is identified by their name and email address. (If you're concerned about spam, you don't need to use a real address that you actually read, but the convention is that it looks like an email address.)

Before you start working, let's tell Bazaar who you are. Using your name and email address, instead of John Doe's, type:

```
$ bzr whoami "John Doe <john.doe@gmail.com>"
```

You can check what identity is stored in Bazaar's configuration:

```
$ bzr whoami
John Doe <john.doe@gmail.com>
```

4 Starting a new project

Let's suppose we want to store a new project under Bazaar. First, we'll make a *repository directory* to hold all our work related to this project, where developers can create branches to test development of specific features or, more generally, modifications to the working file set.

After creating the repository, change to that directory, and create the project's main trunk branch.

```
$ bzr init-repo sample
Shared repository with trees (format: 2a)
Location:
  shared repository: sample
$ bzr init sample/trunk
$ cd sample/trunk
Created a repository tree (format: 2a)
Using shared repository: /home/john/sample/
```

5 Adding files

Now that we have the trunk, we need to move to that directory and create some example files for the first version of that project. Create a file test1.txt using a text editor (like emacs, nano, or notepad), and save it. Then we'll "add"

the file, which tells bzr we want it to track changes:

```
bzr add test1.txt
```

and then commit, which saves a snapshot of all versioned files:

```
bzr commit -m "Added first line of text"
```

6 Making changes to your files

Let's change a file and commit that change to your branch.

Edit test1.txt in your favourite editor, then use bzr add to tell bzr to track changes to this file

```
$ echo test test test > test1.txt
$ bzr add test1.txt
adding test1.txt
```

bzr diff shows the changes between the last revision in this branch, and your current tree (or, with the −r option, between any two trees).

```
$ bzr diff
=== modified file 'test1.txt'
--- test1.txt   2007-10-08 17:56:14 +0000
+++ test1.txt   2007-10-08 17:46:22 +0000
@@ -0,0 +1,1 @@
+test test test
```

Commit your work to the Bazaar branch:

```
$ bzr commit -m "Added first line of text"
Committing to: /home/john/sample/trunk/
added test1.txt
Committed revision 1.
```

7 Viewing the revision log

You can see the history of your branch by browsing its log:

```
$ bzr log
revno: 1
committer: John Doe <john.doe@gmail.com>
branch nick: trunk
timestamp: Mon 2006-10-08 17:46:22 +0000
message:
  Initial import
```

8 Publishing your branch on Launchpad

Launchpad is a suite of development and hosting tools for software projects. You can use it to publish your branch. (You can also publish branches onto your own server or other hosting services.)

The steps to publishing branches on Launchpad are:

1. Create a Launchpad account: visit the Launchpad login page and choose to create a new account.

2. Bazaar uses the SSH encryption and authentication protocol to connect to Launchpad. You need to first create an SSH key on your own computer, by running the command:

   ```
   $ ssh-keygen
   ```

3. Upload your SSH public key to Launchpad.

4. Make a team for your project. Even if you're starting as the only developer on this project, creating a new one now will let you more easily add other people later.

5. Create a project.

6. Tell Bazaar your Launchpad account name. If your account is john.doe, type

   ```
   $ bzr launchpad-login john.doe
   ```

7. Push the branch for your project. Once you've committed your changes locally, you can publish them as the trunk of your new project by saying

   ```
   $ bzr push lp:~sample-developers/sample/trunk
   ```

   (Of course, using the team and project names you just chose.)

9 Creating your own copy of another branch

To work with someone else's code, you can make your own copy of their branch. Let's take a real-world example, Bazaar's GTK interface:

```
$ bzr init-repo ~/bzr-gtk
$ bzr branch lp:~bzr/bzr-gtk/trunk ~/bzr-gtk/john
Branched 292 revision(s).
```

Bazaar will download all the files and complete revision history from the bzr-gtk project's trunk branch and create a copy called `john`.

Now, you have your own copy of the branch and can commit changes with or without a net connection. You can share your branch at any time by publishing it and, if the bzr-gtk team want to use your work, Bazaar makes it easy for them to merge your branch back into their trunk branch.

10 Updating your branch from the main branch

While you commit changes to your branch, it's likely that other people will also continue to commit code to the parent branch.

To make sure your branch stays up to date, you should merge changes from the parent into your personal branch:

```
$ bzr merge
Merging from saved parent location: http://bazaar.launchpad.net/~bzr/bzr-gtk/trunk
All changes applied successfully.
```

Check what has changed:

```
$ bzr diff
```

If different branches have made changes to the same areas of the same files, then merging them may generate conflicts. When this happens, Bazaar puts text markers like <<<<<<< into the files, and records them in a list of conflicted files. You should edit the files to reflect the way you want to resolve the conflicts, use `bzr diff` to check the changes, and then `bzr resolve` to mark them as resolved.

If you're happy with the changes, you can commit them to your personal branch:

```
$ bzr commit -m "Merge from main branch"
Committed revision 295.
```

11 Learning more

You can find out more about Bazaar in the Bazaar User Guide.

To learn about Bazaar on the command-line:

```
$ bzr help
```

To learn about the ''foo" topic or command:

```
$ bzr help foo
```

12 Licence

Copyright 2007-2011 Canonical Ltd. Bazaar is free software, and you may use, modify and redistribute both Bazaar and this document under the terms of the GNU General Public License version 2 or later. See <http://www.gnu.org/licenses/>.