

QwertyManiac BuildingHadoopTrunk

This page documents ways to build Apache Hadoop trunk, and some of the former release branches.

Contents

- [Contents](#)
- [Prerequisites for build](#)
- [Building trunk](#)
- [Building branch-2](#)
- [Building branch-0.23](#)
- [Building branch-0.22](#)
- [Building branch-0.21](#)
- [Building branch-1](#)

Prerequisites for build

1. A Java JDK - To compile and use Apache Hadoop.
 - Most of us use Oracle's JDK or OpenJDK.
 - OpenJDK - <http://openjdk.org>
 - Oracle JDK - <http://java.com>
2. Apache Maven (3+) - To build and manage the Apache Hadoop projects and its dependencies.
 - A latest release of Apache Maven (`mvn`) can be got at <http://maven.apache.org>.
3. Git or Apache Subversion - To fetch Apache Hadoop sources and manage patches.
 - Git is available via <http://git.org>
 - Subversion can be got via <http://subversion.apache.org>
4. Some spirit is always good to have.

Building trunk

The `trunk` branch is where the base of all development happens. All new features and improvements, aside of usual bug fixing, happens on trunk first and then is backported (`copied over`) to a release branch (such as `branch-2` or `branch-1`).

If you are planning to develop a new thing for Apache Hadoop, `trunk` is what you need to familiarize yourself with.

1. Checkout the sources (Use any method below):
 - Using [GitHub](#) mirror: `git clone git@github.com:apache/hadoop.git` `hadoop`
 - Using Apache Git mirror: `git clone git://git.apache.org/hadoop.git` `hadoop`
 - Using the Subversion repo: `svn checkout http://svn.apache.org/repos/asf/hadoop/common/trunk` `hadoop`
2. Download and install Google Protobuf 2.5 (higher may not work) in your OS/Distribution.
 1. On RHEL/CentOS/Fedora, do `yum install protobuf-compiler` 2. On Ubuntu, do `apt-get install protobuf` 3. On OSX, you can get Homebrew and do `brew install protobuf` 4. (The list can go on, but you get the idea, and you have access to a web search engines...) 5. Do ensure the version is right with a `protoc --version`
3. **Optional:** Install all the usual build/development essentials like `gcc`, `'g++/gcc-c++'`, `cmake`, `make`, `zlib`, etc. for various native-code components you may want to hack on.
4. Enter the top level checkout directory (`hadoop`) and issue `mvn install -DskipTests` to kick off the compile.
5. If you want to generate eclipse project files, run: `mvn eclipse:eclipse`. Or alternatively, use the m2e plugin in Eclipse and import these maven projects directly.
6. Post step (4), if you want to run any tests, just head down into the sub-module or the parent-module and issue an `mvn test`. For specific test classes, do `mvn -Dtest=TestClassNameHint test`.

Building branch-2

The `branch-2` branch of Apache Hadoop is where all the 2.x based releases are made, such as the 2.0.0-alpha release, etc.. The 2.x release line is the current future of Apache Hadoop releases.

This is similar to building trunk, but checkout the "**branch-2**" branch before you run the commands.

1. Checkout the sources (Use any method below):

- Using [GitHub](#) mirror: `git clone git@github.com:apache/hadoop.git` `hadoop`
 - Checkout the branch-2 branch once this is done: `cd hadoop; git checkout branch-2`
- Using Apache Git mirror: `git clone git://git.apache.org/hadoop.git` `hadoop`
 - Checkout the branch-2 branch once this is done: `cd hadoop; git checkout branch-2`
- Using the Subversion repo: `svn checkout http://svn.apache.org/repos/asf/hadoop/common/branches/branch-2` `hadoop`

2. If you want to generate eclipse project files, run: `mvn eclipse:eclipse`.

Building branch-0.23

The branch-0.23 is similar to branch-2 but lacks two specific, important changes: Protocol buffer RPCs for services, and High-Availability HDFS. This branch is maintained by Yahoo!'s Hadoop team for their internal distribution of Apache Hadoop deployment and is currently active. All work put in branch-0.23 as part of bug fixing, also makes it to branch-2, thereby making this a subset of branch-2. When branch-2 approaches a stable release, work on this branch is expected to be ended/abandoned by Yahoo! as 2.x gets deployed into their clusters.

Note that this is the last remaining active branch that does not follow the new 1, 2, 3... style of version numbering and chooses to use the older style of 0.2 x. Do not let that confuse you - its just that they couldn't get a proper first-class numbering by the time they chose to continue work on this subset.

This is similar to building trunk, but checkout the "**branch-0.23**" branch before you run the commands.

1. Checkout the sources (Use any method below):
 - Using [GitHub](#) mirror: `git clone git@github.com:apache/hadoop.git` `hadoop`
 - Checkout the branch-0.23 branch once this is done: `cd hadoop; git checkout branch-0.23`
 - Using Apache Git mirror: `git clone git://git.apache.org/hadoop.git` `hadoop`
 - Checkout the branch-0.23 branch once this is done: `cd hadoop; git checkout branch-0.23`
 - Using the Subversion repo: `svn checkout http://svn.apache.org/repos/asf/hadoop/common/branches/branch-0.23` `hadoop`

2. If you want to generate eclipse project files, run: `mvn eclipse:eclipse`.

Building branch-0.22

Although not officially stated anywhere yet, the branch-0.22 is apparently abandoned or unmaintained currently. As of the current wiki update here, the last commit to have hit the branch-0.22 was half a year ago, and the last release the branch made was even further back.

Officially the last branch and higher release of Apache Hadoop to include MR1 (JobTracker/TaskTrackers), when YARN arrived, this branch was stopped full-time development upon and work moved onto branch-2.

0.22 and below used [Apache Ant](#) as the build tool. You need the latest **Apache Ant** installed and the 'ant' executable available on your PATH before continuing.

1. Checkout the sources (Use any method below):
 - Using [GitHub](#) mirror: `git clone git@github.com:apache/hadoop.git` `hadoop`.
 - Check out the branch-0.22 branch once this is done: `cd hadoop; git checkout branch-0.22`
 - Using Apache Git mirror: `git clone git://git.apache.org/hadoop.git` `hadoop`
 - Check out the branch-0.22 branch once this is done: `cd hadoop; git checkout branch-0.22`
 - Using the Subversion repo: `svn checkout http://svn.apache.org/repos/asf/hadoop/common/branches/branch-0.22` `hadoop`

2. **Optional:** Install all the usual build/development essentials like **gcc**, **autoconf**, **automake**, **make**, **zlib**, etc. for various native-code components you may want to hack on.

3. There are three projects subdirectories that lie under the root hadoop directory: **common/**, **hdfs/**, and **mapred/**. You will need to build each one individually, or build the ones you are interested in.

1. For instance, to build the "mapred" project, you need to begin by entering its directory: `cd hadoop/hdfs`. 2. To then compile the whole project, run: `ant compile`. 3. The above instructions can be repeated for `hadoop/common` and `hadoop/hdfs` project directories.

3. If you want to generate eclipse project files, under each project's root directory, run: `ant eclipse`.

Building branch-0.21

This branch and its releases has officially been abandoned and is tagged unsupported. We advise users to move onto 2.x based releases, or at least go back to 1.x.

Building branch-1

Formerly known as branch-0.20, and branch-0.20-security, and instruction also applies for branch-0.20-append. The merger of all three formed branch-1 recently, and is a stable Apache Hadoop branch.

1.0, being an older build branch, still uses [Apache Ant](#) as the build tool. You need the latest **Apache Ant** installed and the 'ant' executable available on your PATH before continuing.

This is almost similar as building branch-0.22, but there is just one project directory to worry about.

1. Checkout the sources (Use any method below):

- Using [GitHub](#) mirror: `git clone git@github.com:apache/hadoop.git hadoop`.
 - Check out the branch-1 branch once this is done: `cd hadoop; git checkout branch-1`
- Using Apache Git mirror: `git clone git://git.apache.org/hadoop.git hadoop`
 - Check out the branch-1 branch once this is done: `cd hadoop; git checkout branch-1`
- Using the Subversion repo: `svn checkout http://svn.apache.org/repos/asf/hadoop/common/branches/branch-1 hadoop`

2. **Optional:** Install all the usual build/development essentials like **gcc**, **autoconf**, **automake**, **make**, **zlib**, etc. for various native-code components you may want to hack on.

3. The source code all lies under the same project directory, so you just need to issue an Ant build: `cd hadoop; ant compile`

4. If you want to generate eclipse project files, under each project's root directory, run: `ant eclipse`.