

How to build AsmTools

AsmTools Build Instructions

This page describes how to build the AsmTools binary. Some notes before you get started:

- In order to contribute code to the project you must have a contributor role. See OpenJDK Governance for information about the different community roles.
- For a description of how the AsmTools repository is organized, read AsmTools Repository Structure.
- The AsmTools Developer's Guide describes how the source code is organized and provides important information for developers who want to modify the code.
- The AsmTools Source Code Description describes how the AsmTools source code is organized.

Table of Contents

- [Requirements](#)
- [Configure and Run the Build](#)
- [Additional Build Targets](#)

Requirements

The AsmTools build requires the technologies listed in the following table.

Technology	Where to Get It
Ant version 1.8 or later	http://ant.apache.org/
JDK version 8.0	http://www.oracle.com/technetwork/java/javase/downloads/index.html

The Jcov tool repository is located on OpenJDK servers, in the CodeTools Project. It is stored in mercurial vcs, and can be uploaded from mercurial via the following command:

```
% hg clone http://hg.openjdk.java.net/code-tools/asmtools
```

Configure and Run the Build

The following steps describe how to build AsmTools. These instructions assume that your AsmTools local working copy is named `asmtools`.

1. Set up and configure the JDK 8.0.
 - Download and install the JDKs.
 - Set the environment variable `JAVA_HOME` to point to the JDK 8.0.
2. Set up and configure the Ant software.
 - Download and install the Ant software.
 - Set your execution path so that it contains the `ant` command. For example: *ANT-dist-path/bin/ant*
3. Make `asmtools/build` the current directory.
4. Execute the `ant` command.

The output of the build (`asmtools-<VERSION>-build` directory) contains the following files and subdirectories:

- `dist/asmtools-<VERSION>.zip` -- AsmTools distribution file. Contains examples, javadocs, `asmtools.jar` and basic documentation files.
- `release/` -- Unpacked AsmTools distribution.
- `binaries/` -- Snapshot of the built product and intermediate source and binaries which were used. These include unpacked class files, sources, etc.
- `javadoc/` -- Generated copy of the JavaDoc for public API.

Additional Build Targets

The following additional targets are available. You might wish to identify these targets to your IDE (for example, NetBeans).

Build Target	Description
<code>build</code> (default)	Builds all binaries, doc and distributions.
<code>clean</code>	Removes the entire build directory.

clean-build	Runs the clean target, then the build target (convenience target)
help	Prints the usage message.
bin	Produces binaries.
devbuild	Produces binaries and javadocs.
javadoc	Produces javadocs.
test	Performs sanity test. (no content presently)