

Coverage for Unit Test

Collecting Coverage Information

You can collect coverage information during unit test execution by running special commands on the following build systems:

- Maven
- Ant
- Gradle

See the Jtest Plugin Manual shipped with Jtest in [INSTALL]/manuals/plugins-manual/index.html for details on how to configure your build system plug-in, as well as how to execute tests and collect coverage during testing.



Associating coverage information collected during unit test execution is supported for JUnit 4 only.

Sending Coverage Information to DTP

In order to collect coverage information and send it to DTP, run the built-in `Unit Tests` configuration during test execution. Maven is used in the following example:

```
mvn clean test-compile jtest:instrument jtest:jtest -Djtest.config="builtin://Unit Tests"
```

See [Unit Testing](#) for information about setting up and executing unit tests.

Merging Coverage Data

In order to properly merge coverage data in DTP, you must specify one or more coverage image tags in the command line or `.properties` settings file. The coverage image(s) is automatically sent to the connected DTP server where it can be associated with a filter.

You can specify a set of up to three tags that can be used to create coverage images in DTP Server with the `report.coverage.images` property:

```
report.coverage.images=[tag1; tag2; tag3]
```

Associate coverage images in DTP in the Report Center administration page (**administration**> **Projects**> **Filters**> **[filter]**).

You can also use the `report.coverage.limit` property to specify a lower coverage threshold:

```
report.coverage.limit=[value]
```

Coverage results lower than this value are highlighted in the report. The default value is 40.