Ziyuan Best Practices

# Maven Pom

# Test

1. Excluded test during mvn build:

This is what we configure in pom.xml of tzuyu.parent:

<excludes>

<exclude>\*\*/testdata/\*\*/\*.java</exclude>

<exclude>\*\*/test/\*\*/\*.java</exclude>

<exclude>\*\*/\*Testdata.java</exclude>

<exclude>\*\*/\*PackageTest.java</exclude>

</excludes>

* Tests which match these pattern will be excluded during build.
  + \*\*/\*PackageTest.java : testcases to run our tool on external projects which configurations are written in /etc/testdata.csv and loaded automatically by TestPackage.java.

Ex: JavaSlicerPackageTest.java

* + \*\*/testdata/\*\*/\*.java or \*\*/\*Testdata.java: these testcases are actually the testdata for another testcases.

Ex: JunitRunnerTestdata.java is actually not an independent testcase but plays a role as the testdata for JunitRunnerTest.java

* + \*\*/test/\*\*/\*.java: these testcases usually are auto generated by gentest.
* Ex: your JunitRunnerTest needs to have a junit test class called ExampleTest to run, let place ExampleTest in a package like sav.commons.testdata.junitrunner

1. The junit test class should extends sav.commons. AbstractTest for basic settings, especially logging.

# Log

1. Using slf4j: <http://www.slf4j.org/> with log4j binding for test.

Example using slf4j log:

private static Logger log = LoggerFactory.getLogger(SelectiveSampling.class);

# pom.xml

- centralize version of dependencies in parent pom.

- if the dependency is too specific for the module, declare it in the module pom, otherwise, put it in <dependencyManagement> of parent pom.

- for sav.commons, all its dependecies are exclusive by default, a module depends on sav.commons has to declare explicitly. Check sav.commons pom to know which dependencies need to add.

# Code

- Don't use classes in test source in java source (ex: TestConfiguration), otherwise, it will fail when building with mvn.

- Don't use org.junit.Assert in source code, if you want assertion use sav.common.core.utils.Assert instead.