Unit Test Frameworks

xUnit

- xUnit is the collective name for several unit testing frameworks that derive their structure and functionality from Smalltalk's SUnit.
- SUnit, designed in 1998, was written in a highly structured object-oriented style, which was modified easily to languages such as Java and C#.

xUnit

- It eventually was used with the majority of current programming languages
- The names of many of these are a variation on "SUnit", usually replacing the "S" with the first letter (or letters) in the name of their intended language ("JUnit" for Java, "RUnit" for R etc.).
- These frameworks and their common architecture are collectively known as "xUnit".

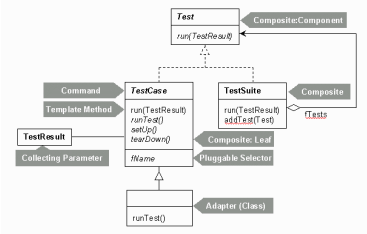
Architecture

- Test runner - an executable program that runs the tests implemented and reports the test results
- Test case - the most elemental class and all unit tests are inherited from here.
- Test fixtures - the set of preconditions or state needed to run a test.
- Test suites - a set of tests that all share the same fixture.
- Each test should be isolated from the others, so it creates all its data before executing and destroys it when it's done (setUp() and tearDown())
- Test Result – the output from the framework including tests run and tests failed

Languages

- RudyUnit, SUnit, CppUnit, Unit++, PerlUnit, VbUnit, NUnit, PyUnit, SQLUnit, XMLUnit, ...

Junit Architecture



See http://junit.sourceforge.net/doc/cookstour/cookstour.htm

Summary - AssertMethods

1. **void assertEquals(boolean expected, boolean actual)**
   Check that two primitives/Objects are equal
2. **void assertTrue(boolean expected, boolean actual)**
   Check that a condition is true
3. **void assertFalse(boolean condition)**
   Check that a condition is false
4. **void assertNotNull(Object object)**
   Check that an object isn't null
5. **void assertNull(Object object)**
   Check that an object is null
6. **void assertSame(boolean condition)**
   The assertSame() methods tests if two object references point to the same object
7. **void assertNotSame(boolean condition)**
   The assertNotSame() methods tests if two object references not point to the same object
8. **void assertArrayEquals(expectedArray, resultArray);**
   The assertArrayEquals() method will test whether two arrays are equal to each other.

assertTrue() and assertFalse()

- The assertTrue() and assertFalse() methods tests a single variable to see if its value is either true, or false.
- If the program under test returns true, the assertTrue() method will return normally. Otherwise an exception will be thrown, and the test will stop.
- If the program under test returns false, the assertFalse() method will return normally. Otherwise an exception will be thrown, and the test will stop.

assertNull() and assertNotNull()

- The assertNull() and assertNotNull() methods test a single variable to see if it is null or not null.
- If the program returns null, the assertNull() method will return normally. If a non-null value is returned, the assertNull() method will throw an exception, and the test will stop.
- The assertNotNull() method works in the opposite way to the assertNull() method,

assertEquals()

- The assertEquals() method compares two objects for equality
- If the two objects are equal, the assertEquals() method will return normally. Otherwise the assertEquals() method will throw an exception.
-
- The assertEquals() method can compare any two objects to each other, it has versions that compare primitive types like int and float to each other.

assertArrayEquals()

- The assertArrayEquals() method will test whether two arrays are equal to each other. In other words, if the two arrays contain the same number of elements, and if all the elements in the array are equal to each other.
- If the arrays are equal, the assertArrayEquals() will proceed without errors. If the arrays are not equal, an exception will be thrown, and the test aborted.

assertSame() and assertNotSame()

- The assertSame() and assertNotSame() methods tests if two object references point to the same object or not. It is not enough that the two objects pointed to are equals according to their equals() methods. It must be exactly the same object pointed to.

TestNG

- TestNG is a testing framework inspired from JUnit and NUnit but introducing some new functionalities that make it more powerful and easier to use, such as:

- Annotations.
- Run tests in arbitrarily big thread pools with various policies available (all methods in their own thread, one thread per test class, etc...).
- Test that code is multithread safe.
- Flexible test configuration.
- Support for data-driven testing (with @DataProvider).
- Support for parameters.

TestNG

- Powerful execution model (no more TestSuite).
- Supported by a variety of tools and plug-ins (Eclipse, IDEA, Maven, etc...).
- Embeds BeanShell for further flexibility.
- Default JDK functions for runtime and logging (no dependencies).
- Dependent methods for application server testing.

- TestNG is designed to cover all categories of tests: unit, functional, end-to-end, integration, etc...