

Unit Testing with cfcUnit

#SA3D, Process & Tools

John Mason

Slides & Code – labs.fusionlink.com

Blog - www.codfusion.com



CF.Objective()

Unit Testing with cfcUnit

- Why do Testing?
 - A bad release of code or software will stick in people's minds.
 - Debugging code is twice as hard as writing the code in the first place.
 - Fixing bugs can recreate old bugs.

Unit Testing with cfcUnit

- Speaker Bio
 - Using ColdFusion since 1997
 - Board Member of the Atlanta CF User Group
 - Founder/President of the Atlanta Flex User Group
 - President of FusionLink Inc., a ColdFusion and Flex hosting company based in Atlanta
 - Certified Advance CF
 - MBA from Georgia Southern University, 2001

Several Layers of Testing

- Application Level
 - Acceptance Testing
 - Testing a completed product before transfer of ownership
 - If by the customer..
 - » Beta Testing and User Acceptance testing (UAT)
 - If by the vendor..
 - » Release Acceptance, QA Testing and Blackbox Testing
 - User Interface (UI) Testing
 - Selenium, <http://wiki.openqa.org>

Layers of Testing

- System Level
 - Load Testing
 - The practice of modeling the expected usage of a software program by simulating multiple users accessing the program's services concurrently.
 - Grinder, <http://grinder.sourceforge.net/>
 - Can also do some low level testing with CF Monitor

Layers of Testing

- Code Level
 - Unit Testing
 - Testing the core logic of the software coding

Common Terms in Unit Testing

- Unit Testing
 - Test functionality of a single class or unit
 - So typically in the CF world, we'll be testing our CFCs
- Test Driven development (TDD)
 - Writing the tests first before writing the code

Common Terms in Unit Testing

- Continuous Testing (CT)
 - Runs all the unit tests continuously from the IDE
 - Prevent bugs from getting updating to the source code or to production
 - Let the automated tools to do the heavy lifting

Common Terms in Unit Testing

- Continuous Integration (CI)
 - "A software development practice where members of a team integrate their work frequently, usually each person integrates at least daily - leading to multiple integrations per day. Each integration is verified by an automated build (including test) to detect integration errors as quickly as possible.", Martin Fowler
 - Cruise Control
<http://cruisecontrol.sourceforge.net>

Common complaint about Unit Testing

- Writing Tests takes too much time?
 - "Test automation is key to better quality. Manual checks are more flexible and less expensive to do one time. They are more expensive and less reliable when tests need to be done over and over again.", webtest.canoo.com

Common Unit Testing frameworks

- Java - JUnit
- Databases - dbUnit, SQLUnit, tSQLUnit
- Flash/Flex - ASUnit, FlexUnit and dpUnit
- ColdFusion - CFUnit, cfcUnit, MXUnit
- Just about every language has a Unit Test framework

cfcUnit

- <http://www.cfcunit.org>
- Created by Paul Kenney from Adobe, <http://www.pjk.us/paul/>
- The front end GUI display is a MachII ColdFusion application
- There is also a Flex GUI display that was developed by Laura Arguello, ASFusion
- Can be run from command line with ANT

Common Elements of any UT

- Assertions, functions testing for a particular response, cfcUnit has 23 test assertions
 - Some examples..
 - assertEqualsArray()
 - assertEqualsString()
 - assertTrue()
 - assertRegexMatch()
 - assertObject()

Common Elements of any UT

- Test Fixture
 - Any properties or objects that need to be initialized to run the test
 - Two common methods, setUp() and tearDown()

Common Elements of any UT

- Test Case
 - The smallest unit, tests for a particular response
 - Contains one or more test assertions
- Test Suite
 - A collection of test cases or other test suites

Common Elements of any UT

- Test Runner
 - Calls the tests and displays the results
 - Common method, `startTest()`
- Test Harness
 - Automated scripting to run the tests

Let's look at some code.

The Next Step

- Continuous Integration

The screenshot displays a CI dashboard with the following components:

- Navigation:** Dashboard, Builds, Administration.
- Build Status:** A green banner indicates "cruisecontrol Passed" with a checkmark icon. Details include: Build Time: 30 May 2007 03:29 GMT, Duration: 5 minutes 3 seconds, -07:00, and Build: build.74.
- Latest Builds:** A list of recent builds with their status (green checkmark for success, red exclamation mark for failure):
 - 30 May 2007 03:29 GMT -07:00 build.74 (Success)
 - 30 May 2007 03:22 GMT -07:00 build.73 (Success)
 - 30 May 2007 02:01 GMT -07:00 build.72 (Success)
 - 29 May 2007 14:20 GMT -07:00 build.71 (Success)
 - 29 May 2007 09:13 GMT -07:00 build.70 (Success)
 - 29 May 2007 08:44 GMT -07:00 (Failure)
 - 29 May 2007 08:37 GMT -07:00 build.69 (Success)
- Modifications:** A list of changes made by user 'mli':
 - update widget docs
 - [92] modified /trunk/cruisecontrol/docs/download.html
 - [92] modified /trunk/cruisecontrol/docs/sffeatures.html
 - [92] modified /trunk/cruisecontrol/docs/index.html
 - [92] modified /trunk/cruisecontrol/docs/faq.html
 - [92] modified /trunk/cruisecontrol/docs/developers.html
 - [92] modified /trunk/cruisecontrol/main/docs/config.xml.html
 - [92] modified /trunk/cruisecontrol/reporting/isp/docs/install.html

Conclusion

- Unit Testing is only as good as you make it
- It needs to be integrated into your development process
- Include the Unit Testing classes in your SVN repository
- You actually do have time to do testing!

Recommended Reading

- Pragmatic Programmers Series - <http://www.pragmaticprogrammer.com>
- Code Complete, Steve McConnell
- Writing Effective Test Cases, Alistair Cockburn
- Test-Driven Development: by example, Beck, Kent
- <http://www.extremeprogramming.org>
- <http://www.martinfowler.com/>

Thanks for coming!

- Presentation slides, notes and code examples can be found at..
 - labs.fusionlink.com
- Feel free to email me questions or comments to mason@fusionlink.com

CF

CF.Objective()

Title

- Test Case

CF

CF.Objective()