



E-Testing: Debugging Your CF Apps

How your HTML, CFML, and web application
looks, works, scales, and more

Charlie Arehart
Systemanage

Our Practice Makes You Perfect™
carehart@systemanage.com



The Challenge

- ◆ Finished coding app or fix to app
 - Ready to turn over to users
 - Have you tested the code?
 - “Running the code” isn’t enough
- ◆ Need to evaluate software quality
 - End-user experience may differ from yours
 - Anticipate load, ongoing maintenance
- ◆ Cheaper to fix problems early
- ◆ Issues apply to lone developer or team
 - Tools exist, though many don’t use them

Topics

- ◆ Seminar covers:
 - HTML-oriented testing
 - CFML Code Testing
 - Web Application Testing
 - CFML Debugging
- ◆ Will look at 30 tests and point to resources to learn more
- ◆ How many are using any such tools?

Part 1: HTML Testing

- ◆ 13 different tests (that I've identified)
 - From simple validation to effects of browser caching, cookie, and javascript support
- ◆ Many CF developers dismiss them
 - But they're free
 - Valuable, if used effectively
 - Challenge: Need to test *output* of CF template
- ◆ Failure to use costs you, company, client
 - in time and debugging effort
- ◆ Simplest test: Is your HTML valid?
 - Will code work in browser? All browsers?

Studio HTML Validation

- ◆ Studio offers HTML validation
 - Tools>Validate Document
 - Provides warnings and errors about HTML problems
 - Problem: need to test output, not source
 - Solution: Copy/paste output of your CF template into Studio blank page, then validate
 - Can even indicate which browser to support
 - Options>Settings>Validation
 - Can set which CF versions to support
 - When validating CF source, warns if tag/function not supported in chosen CF version
 - Useful when developing in R5, deploying in R4.5

HTML Validation Services

- ◆ Web-based services even easier
 - Such as <http://validator.w3.org>
 - Just point it to your site: it sees your HTML output
 - If inside firewall, make copy of output available
 - Others:
 - NetMechanic.com
 - Netscape's "Web Site Garage"
 - http://websitegarage.netscape.com/O=wsg/tuneup_plus/index.html
 - Search net for "html validation" for others



Other HTML Validations

- ◆ Other HTML-oriented tests available at this site (and others like it) include:
 - Link checking
 - Also available within Studio
 - Spell checking
 - Also available within Studio
 - Document weight (download time) testing
 - Also available within Studio
 - Cascading StyleSheet validation
 - Color depth (browser safe palette) testing
 - Image compression testing




Additional Testing

- ◆ See article for further discussion
 - Browser size testing
 - Studio feature supports testing; web services and shareware available
 - JavaScript error testing
 - By default, modern browsers may hide JavaScript errors (from users and YOU!)
 - Turn errors back on during your testing



Accessibility Testing

- ◆ Providing access for disabled visitors
 - Section 508 of US Federal Rehabilitation Act
 - Useful for all developers and users
 - Several sites provide info/support testing, including
 - <http://www.cast.org/bobby>
 - <http://www.microsoft.com/enable/>



Additional Accessibility Testing Tools *

- ◆ Have identified more accessibility testing tools since writing the article
 - Usable.Net (Macromedia Partner)
 - LIFT for DreamWeaver/UltraDev
 - ◆ Tools for accessibility and usability testing
 - ◆ Also available as web service:
http://www.usablenet.com/lift_online_free_trial/index.htm
 - W3 tools for accessibility
 - <http://www.w3.org/WAI/ER/existingtools.html>

Additional Tests *

- ◆ Have identified still more useful tests since writing the article last year
 - These involve making changes to your browser to mimic settings users might make
 - Can't cover details in this given timeslot
 - Will identify tests, enough for you to research
 - Detailed steps provided in available daylong seminar (see last slide for info)
 - Also provides time to show use of tests and tools already discussed and more still to come today

Topic Not Discussed in CFDJ Article

Additional Tests *

- ◆ Testing disabled JavaScript
 - How does your app run if browser does not support JavaScript?
 - Can disable javascript and test
- ◆ Testing disabled cookies
 - How does your app run if browser does not support cookies? Big problem for sessions
 - Can disable cookies and test

Topic Not Discussed in CFDJ Article

Additional Tests *

- ◆ Testing browser caching
 - How does your app run with different settings for browser page caching?
 - Can code to prevent caching (CFHEADER)
 - Should at least test with typical user cache settings

Topic Not Discussed in CFDJ Article

Part 2: CFML Code Testing

- ◆ Feeling cheated with focus on HTML testing?
- ◆ Wondering if CFML code testing is at all possible?
 - It is, but many developers don't know



4 Forms of CFML Testing

- ◆ Syntax checking
- ◆ Code quality testing
- ◆ Included file testing
- ◆ Code timing



CFML Syntax Checking

- ◆ CF includes a syntax checker
 - <http://localhost/CFDOCS/cfmlsyntaxcheck/cfmlsyntaxcheck.cfm>
- ◆ Available as of CF 4
 - Originally intended to ensure old code worked with changes in new release
 - Useful for testing code to at least ensure certain syntax errors have not been introduced before release
 - Doesn't really test all syntax
 - not updated for CF5
 - More details in article



CFML Code Quality Testing

- ◆ Previous tool doesn't test software quality
 - Whether it will perform well
 - Whether you've made any common CF coding mistakes
- ◆ Java and other languages have such tools
 - Why not CF?
 - Is it a pipe dream?



Steve Nelson's STOMP

- ◆ As in "stomp the bugs"
- ◆ Available from SecretAgents.com
 - Offered as fee-based web service
 - www.secretagents.com/products/index.cfm?fuseaction=product&product_id=1
- ◆ Some will find it worth every penny
 - Available for free trial



About Stomp

- ◆ Process of submitting code
 - Submit zip of source code to web service
- ◆ Report made available in minutes
 - What to fix, why, and how
- ◆ Identifies common (and some easily forgotten) CFML code problems



Sample Errors

- Suggested maxlength for form input field
- Suggested removing Select *
- Recommended `<CFSETTING ENABLECFOUTPUTONLY="yes">` for templates with no text for display (or all the text was within CFOUTPUTs)
- Recommended changing a .cfm file to .htm (had no CF code)
- Recommended case-insensitive versions of find/replace/compare functions
- And more



Worthwhile?

- ◆ I knew not to do (or to do) many of these
 - Still, I had overlooked them, been lazy
- ◆ But I can look on the bright side
 - Only flagged a few out of hundreds of potential errors!
 - Complete list of the tests (which itself may be useful) currently offered at www.secretagents.com/content/index.cfm?fuseaction=testlist



Depth and Flexibility

- ◆ Depth of reporting can be drawback
 - Like grammar tools in word processors
 - Can be overwhelming number of errors found
- ◆ Also, you may not agree that an error should be flagged
 - Or perhaps it should be a suggestion instead
- ◆ Report is presented via Stomp web site
 - each error offers link to send feedback to Stomp administrator about an error



Another Code Testing Tool: ParseDog

- ◆ Focuses on identifying whether custom tag and CFINCLUDE files exist
 - has some extra features that may interest you
- <http://aloha-webdesign.com/dloads/cf/parsedog/releasenotes.htm>



Code Timing

- ◆ How long does it take code to run?
 - No automated “profiling” tools, yet
 - Can “roll your own” with CF’s `getTickCount()` function
 - See CF docs for simple, useful example
- ◆ How long to compile? To run query?
 - CF Admin Debugger info can show several aspects
 - “show detailed processing time breakdown” option shows for all included templates, custom tags



Part 3: Web Application Testing

- ◆ How application as a whole (CF, HTML, JavaScript, etc.) works
 - In both reasonable and unexpected situations
- ◆ Covers several possible tests...



Web App Testing

- ◆ Data validation and "bounds" checking
 - Does your code handle unexpected input well?
- ◆ Functionality testing
 - Does the code work as expected?
- ◆ Security testing
 - Are only authenticated, authorized activities able to take place?
- ◆ 4 more, to be discussed later

How to Test These

- ◆ One approach is Acceptance Testing
 - Simply run the code (or have testers do it)
 - See if it does what's expected (with good/bad input)
- ◆ Another is Regression Testing
 - Run the code after making any changes, to make sure it produces same result as before
- ◆ Useful solution: session record/playback tools
 - Better tools test entire app, not just a page

Like a VCR

- ◆ Most tools work like a VCR
 - Point to starting web page, click "record" and run through site as user would
 - tracks all button clicks, mouse clicks, links taken, even JavaScript or Flash events fired (in better tools)
 - Then playback
 - Can setup automated repeated playback
 - Most allow parameter-driven tests using data in a file or database
 - Allows testing of good, bad data, large volumes



Prominent Testing Tools

- ◆ Empirix (formerly RSW Software)
 - www.empirix.com
- ◆ Mercury-Interactive
 - www.mercuryinteractive.com
- ◆ Segue
 - www.segure.com
- ◆ Rational
 - www.rational.com
- ◆ All mature, relatively expensive



Prominent Testing Tools

- ◆ Newcomer, less expensive: Evalid
 - www.soft.com/eValid/
- ◆ Freebie: well-equipped
 - Microsoft Web App Stress test tool
 - http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnduwon/html/d5wast_2.asp



Demonstration

- ◆ Mercury Interactive Astra QuickTest
 - Will demonstrate record/playback
 - Many more capabilities than can be demoed briefly: again, see seminar
 - Catching regression errors
 - Using parameter-driven testing
 - more



Still More Web App Testing

- ◆ Concurrency testing
- ◆ Performance/load testing
- ◆ Site monitoring
- ◆ Let's look at each...



Concurrency Testing

- ◆ Are there problems when multiple users access the code, or when a single user opens multiple browsers?
 - Not even really about “load” testing, just testing more than a single user!
 - Many sites brought to their knees with just a couple extra users
- ◆ Simple freeware stress testing tools may help with this
 - What happens running even a single page



Free Stress Testing Tools

- ◆ WebHammer
 - <http://www.serverobjects.com/products.htm>
 - Eval version limited to 2 request threads
- ◆ Paessler Webserver Stress Tool
 - <http://www.paessler.com/tools/WebStress/webstress.htm>
 - Free for 5 users, modest cost to add more
- ◆ Simply simulate http requests to a given page on server
 - Problem: often need to test more than just a given page, and with lots of different data



Performance/load testing

- ◆ Also, beyond simple concurrency testing
 - Can the code (and environment) sustain a large load of hundreds, thousands of users?
- ◆ Recorded, parameterized tests of previous record/playback testing tools can be re-used
 - Is where these tools really shine



Load Testing Process

- ◆ Extensions to previous testing tools
 - Rerun the same recorded scripts
 - Some load tools use special scripting instead
 - Parameterize for varying data (using file or database)
 - Use tool's feature to create "virtual users"
 - Can mimic many, perhaps thousands, hitting the site and running through paths of pages
 - Not just banging away on a single page
 - Measure/report performance



Site monitoring

- ◆ Similar to previous tests, but monitoring automatically, over long run
 - Is the site up? Is anyone aware when it goes down? Did changes break the site?
- ◆ Can also leverage previous tools' recorded scripts
 - Each vendor offers monitoring tools to extend functional and load testing for ongoing monitoring



More Site Monitoring

- ◆ Smaller players exist as well
 - freshwater.com, tracert.com, others
 - Some just ping site to report if it's up or not
- ◆ CF5 offers probe mechanism to do this as well
 - See Admin>Tools>System Probes



Other Web App Testing

- ◆ Still other tests can be considered
 - COM/EJB/CORBA/Java component testing
 - Web service testing
 - WAP/WML testing
 - Search engine result optimization testing
- ◆ Will leave it to you to investigate



Other E-Testing

- ◆ May consider other testing as well
 - usability
 - documentation
 - installation
 - configuration
 - reliability/recovery
 - end-to-end
 - unit
 - white/gray/black-box



Don't Forget Acceptance Testing

- ◆ Is the code really what the user wants?
- ◆ More about prototyping than testing
 - During design, development, before rollout
- ◆ Wireframing
 - Create simple but adequate representations of the look and feel (and flow) of a site
 - Embraced by FuseBox community
 - But all can benefit
 - Discussed in CFDJ article by Hal Helms
 - "Exploring the Development Process," Vol 3, Issue 3



Integration Testing

- ◆ Your CF App and server interact with other servers and services
- ◆ Test also availability/performance of
 - Web Server, Cluster (if any)
 - Application Server (CF)
 - Database, DBMS, database server
 - Mail server, ftp server, so on

CFML Debugging *

- ◆ Though not really “testing”, remember:
 - Administrator-enabled Debugging
 - what variables are being passed to this template, and how long is it taking to run?
 - CFDUMP
 - what are the current variables and their values?
 - Log analysis
 - what errors are occurring, to whom, and why?
 - Error Handling
 - Trapping and sending notification of errors`

Topic Not Discussed in CFDJ Article

Studio Debugger *

- ◆ Studio integrated debugger can be powerful
 - what lines of code are executing and why
 - Set breakpoints, monitor/change variables
 - Stop when given condition is reached
 - Problem: hard to setup, sometimes flaky
 - It can work. Daylong seminar will cover use
 - Also available in JRun Studio for debugging JSP

Topic Not Discussed in CFDJ Article



Conclusion

- ◆ Lots of testing possibilities
 - May be thinking
 - How have I gotten by without it?
 - Or, how will I ever remember to do them all?
 - Or, we don't need no stinking testing!
 - Just consider the possibilities, try them out
 - Remember that many tools are free, some even built into Studio or CF Server



Summary

- ◆ We covered:
 - HTML-oriented testing
 - CFML Code Testing
 - Web Application Testing
 - CFML Debugging
- ◆ Review these slides, accompanying article every few months as a checkup
 - Consider my one-day seminar in future
 - Several books, articles available



Resources

- ◆ Article for this seminar
 - E-Testing: Debugging Your Projects, CFDJ Vol 3 Issue 10 (Oct 2001)
 - <http://www.sys-con.com/coldfusion/article.cfm?id=349>
- ◆ Many books, articles, URLs listed in article. Key ones:
 - Testing Applications on the Web: Test Planning for Internet-Based Systems
 - by Hung Quoc Nguyen; Wiley, 2000
 - The Web Testing Handbook
 - By Steven Splaine, et al; Software Quality Engineering Pub, 2001



One Day Seminar

- ◆ Many more details and demonstrations
 - Additional forms of testing, and tools
 - Demonstrations of each tool
 - More detail on topics skipped here
 - ◆ Browser setting tests, debugger, and more
 - Offered privately to organizations, perhaps publicly at scheduled locations on limited basis



One Day Seminar

- ◆ For questions on topics covered
 - Or information on the 1 day seminar, or other in-house consulting/training
 - Or presenting at another CF user group
- ◆ Contact: Charlie Arehart,
Systemanage
 - carehart@systemanage.com
 - 301 604 8399