Assessment of String Tests Strategy for an En Route Air Traffic Control System

Jeff O'Leary
Federal Aviation Administration
800 Independence Avenue, SW
Washington, DC 20591
202-385-8377
Jeff.OLeary@faa.gov

Frederick Woodard
Northrop Grumman
475 School Street SW
Washington DC 20024
202-314-1459
Fred.Woodard@auatac.com

Alok Srivastava
Northrop Grumman
475 School Street SW
Washington DC 20024
202-314-1419
Alok.Srivastava@auatac.com

Denise S. Beidleman
Northrop Grumman
475 School Street SW
Washington DC 20024
202-314-1482
Denise.Beidleman@auatac.com

(This paper was presented at the ACM SIGAda 2007 conference)

ABSTRACT

The Federal Aviation Administration (FAA) depends upon several highly available large and complex computer systems to support air traffic control operations. One of the FAA's major initiatives, the Ada based En-Route Automation Modernization (ERAM) program leverages and significantly expands upon previous developed systems. To determine the effectiveness of the string test implementation before full system level testing, the FAA performed an audit of the software requirements that were verified by string tests.

An industry standard approach was used to determine the number of Shalls to be inspected. Detailed evaluation using multiple software documents and test plans were compared to the string test results provided by the development contractor.

This paper presents details of the audit, its results and reconciliation of the data with the development contractor.

The approach to have selected software Shalls verified through the string test method versus full system test resulted in a cost savings of 50% of the cost of the selected software Shalls' full system test, by avoiding duplication of testing efforts.

The resulting high pass rate of the audit gave confidence in the development contractor's string test results and also demonstrated a high confidence level in the code, thus confirming the high quality of the Ada code effort.

1.0 INTRODUCTION

Current trends show it has become increasingly difficult to deliver complex systems on schedule and within costs. One of the methods the FAA has employed to ensure the delivery of ERAM on time, and within budget, was to perform an audit of the B-Level (lower-level) Shalls, which are being verified through the String Test (ST) Method.

A validation plan, agreed upon by the FAA's ERAM Software Engineering and Quality Control organizations, was first established.

The timing of this audit was critical, as the audit needed to be complete in order to move ahead into the system test portion of the Life Cycle with assurance in the string test results.

This plan documents the string test methodology, the audit process followed, the extent of testing required, how success criteria was determined and the final results of this validation effort.

SIGAda '07 November 4-9, 2007, Fairfax, Virginia, USA.
ACM 978-1-59593-876-3/07/0011.

2.0 BACKGROUND

ERAM is comprised of various sub-systems, directly related to the Computer System Component (CSCs) of the legacy En-Route host system. These sub-systems are further subdivided into Computer System Configuration Items (CSCIs). There are eight operational sub-systems with key requirements that shape the architecture of the operational environment of ERAM.

Two of these sub-systems were selected for the audit; the government reserved the right to do a more thorough inspection, should the success rate fall below the pre-determined pass rate of 95%.

Per the development contractor's process:

A string may be defined as that portion of a system thread performed by a single CSCI (including Commercially Available Software [CAS], Non-Development Items [NDIs], and Common Shared Services [CSS] code) or it may be one or more units within a CSCI that operate to perform a specific function. A string may service many system threads and a particular CSC may appear in multiple strings.

A string test verifies that a collection of units or CSCs work together to satisfy the required functions of the executable address space. String tests should test every required function and as many boundary conditions as feasible in the development environment. String tests verify interfaces as well as required functions and may have associated performance measures. For real time software, test cases for string tests typically consist of external stimuli (messages, timers, callbacks, etc.) that invoke the functions being tested. String test cases may require writing test drivers to simulate components of the system outside the CSCI.

A summary of the string test follows:

Step	Description
Objectives	Integrate CSCs within executable address space, testing that the required function is performed in accordance with the design.
Scope	CSCI threads or set of programs from an operating system address space.
Responsibility	Developer
Approach	Black box testing. Test all functions (main and error paths) using external stimuli or panel entries.
Documentation	Test plans, test cases, and test results are documented including memory analysis.
Environment	Workstation, or other processor as appropriate for units being tested. Target dependent code tested in target environment.
Entrance Criteria	Code, string test plans, string test cases, memory analysis plan and memory analysis procedure complete and inspected.
Exit Criteria	All unit tests completed. All string tests completed. All remaining test failures documented in problem reports with the approval of a software architect before exiting string test. Performance measurements taken for performance-relevant strings. Measurements recorded (unless measurements will be repeated in target environments and recorded then).
Inspections	String test cases formally inspected.

SIGAda '07 November 4-9, 2007, Fairfax, Virginia, USA.
ACM 978-1-59593-876-3/07/0011.

3.0 APPROACH

3.1 SELECTIONS FOR AUDIT

The two critical sub-systems monitored were Conflict Probe Tools (CPT) and Display Systems (DS), corresponding to CSCIs of Flight Evaluation (EVAL), and En Route Display Management, EDSM. Due to the large size of EDSM, this module was sub-divided into EDSMa and EDSMb.

Sub System	CSCI	# of B-Level Shalls
CPT	EVAL	270
DS	EDSMa	8,317
DS	EDSMb	4,070

Table 1 : CSCI's for Audit

3.2 DETERMINING THE SAMPLE SIZE

Several factors were considered in determining the sample size to be measured. First, the sample size would have to assure a certain confidence in the results. Second, a certain margin for normal process variation and tester error was allowed. A confidence level of 98% was sought, with a 5% margin of error.

A sizing calculator recommended by iSixSigma.com, a quality on-line resource providing Six Sigma information, was selected to determine the sample sizes used; this calculator was supplied by Raosoft Corporation.

The calculator determines the sample size, n, using the following equations.

$$\mathbf{n = N\,x/((N-1)E2 + x)}$$
$$\mathbf{x = Z(c/100)2r(100-r)}$$
$$\mathbf{E = \sqrt{[(N - n)\,x/n\,(N-1)]}}$$

Equations 1-3 : Sample Formulas

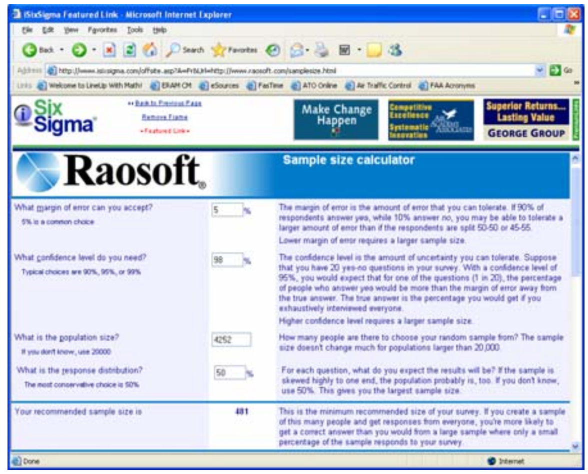

Figure 1 : Raosoft Sample Sizing Calculator

[http://www.isixsigma.com/offsite.asp?A=Fr&Url=http://www.raosoft.com/samplesize.html]

–

For a population size (N) of 4252 Shalls, a margin of error (E) of 5% is assumed, r is the fraction of responses that is of interest, (50% yields the largest sample size) and Z(c/100) is the critical value for the confidence level c (98%). The calculator yields a result of 481 Shalls.

CSCI	# of B-Level Shalls signed off by String Test	Sample Size
EVAL	140	140*
EDSMa	4252	481
EDSMb	438	243

Table 2 : Sample Sizes Calculated

* *The calculator produced a value of 112. For due diligence, the full set was audited.*

3.3 AUDIT PROCESS

A Verification Requirements Traceability Matrix (VRTM) was supplied by the development contractor; this formed the basis of the effort. This VRTM provided key information such as the identifier of the Shall being tested, the success criteria and the pass/fail status of the string test.

SIGAda '07 November 4-9, 2007, Fairfax, Virginia, USA.
ACM 978-1-59593-876-3/07/0011.

The primary focus of the audit was to determine if the success criteria being verified adequately covered the Shall under test – that defined the audit's pass/fail criteria. If the string tests were written in a way to adequately cover the Shall, confidence would be assured in the final string test results.

The following table best describes the steps taken to audit the B-Level Shall test results.

Step #	Activity to be Performed	Artifact Produced
S – 1.	*Define CSCIs:* Obtain the existing list of CSCIs showing the critical CSCIs	A listing with the critical CSCIs identified
S – 2.	*Determine Completeness of VRTM:* Verify the VRTM displays information in the following fields: Test ID, Test Name, Shall ID, and Verification Method. If complete VRTM is received, continue to step S – 4, otherwise continue to step S – 3	Not Applicable (NA)
S – 3.	*Inform Development Contractor:* Inform Development Contractor of deficiencies in VRTM, await updated VRTM and return to step S – 2	Added record(s) in En Route Test Database
S – 4.	*Analysis:* Using the VRTM provided, obtain the count of B-Level Shalls where 'Verification Method=String Test' (ST), corresponding to the critical CSCIs selected	Test Sample where verification method=ST
S – 5	*Determine Count:* From the count obtained at S –4, for each CSCI, determine the number of B-Level Shalls to be tested to ensure 98% confidence in the representative sample	Count of CSCI Shalls to be tested
S – 6.	*Test Document Generation:* Using the VRTM provided, the counts obtained from S –5, and development contractor documents, create the Test Documents. If multiple STs are used to verify a requirement, record all associated STs	Test Documentation
S – 7.	*Start Actual Verification:* Select ST to be verified	NA
S – 8.	*Compare Shall to Success Criteria:* Compare Shall to Success criteria of ST, obtained from VRTM, documenting results.	Completed Test Logs
S – 9.	*All Shalls Verified?* Determine if all selected STs have been verified. If all STs are verified continue to step S-11, otherwise continue to step S-10	NA
S – 10.	*Advance:* Advance to subsequent ST, return to S –8	NA
S – 11.	*Review Test Logs:* If response to S –9 is yes, Review Test Logs; enter comments and/or discrepancies	Comments and/or Discrepancies entries in En Route Test Database
S – 12.	*Prepare Final Results:* Update Master Test document based on completed test logs, prepare metrics, assess test results.	Metrics for TSR
S – 13.	*Write Test Summary Report (TSR):* Prepare a TSR on completeness and correctness of ERAM B-Level Shall String Test Validation, to include a draft, review time and a final version	Final Version of TSR
S – 14.	Exit	NA

Table 3: Audit Process

SIGAda '07 November 4-9, 2007, Fairfax, Virginia, USA.
ACM 978-1-59593-876-3/07/0011.

4.0 TEST DOCUMENTATION

A test log, recording the following items, was created for each Shall tested:

1. Reviewer Name
2. Date
3. Shall ID
4. Row #
5. String Test (ST) ID
6. CSCI Tested
7. Shall Text
8. ST's Success Criteria
9. Development Contractor's ST Status (Pass, Fail or GFP Blocked)
10. ST covers shall: (Yes or No)
11. Problem Report Created (Yes, No or NA)
12. Problem Report Resolved: (Yes, No or NA)
13. Is ST correctly traced to Shall: (Yes, No)
14. Reviewer's Comments
15. Secondary Reviewer: (Agree, Disagree)
16. Secondary Reviewer's Comments

5.0 INITIAL TEST RESULTS

Test results were compiled and separated into varying categories, depending on the nature and severity of the deficiencies found. A summary of the test results for each CSCI follows. Note: The term 'Completeness of Success Criteria' denotes the success criteria of the string test adequately addressed the Shall under test. The '% Completeness of Success Criteria = 100-(sum of all severe categories)'.

5.1 EVAL FINDINGS

The pass rate of Shalls in the EVAL CSCI was 96.43%.

EVAL Category	Severe (Y/N)	%age.
Team has determined Success Criteria covers Shall	N	58.57%
Typo or Grammatical Error	N	1.43%
Incorrect Tracing of String Test to Shall	N	36.43%
Incomplete Mapping of Shall to String Test	Y	3.57%
COMPLETENESS OF SUCCESS CRITERIA	**NA**	**96.43%**

Table 4: EVAL Test Results

5.2 EDSMA FINDINGS:

The pass rate of Shalls in the EDSMa CSCI was 86.1%.

EDSMa Category	Severe (Y/N)	%age.
Team has determined Success Criteria covers Shall	N	72.1%
Typo or Grammatical Error	N	0.4%
Incorrect Tracing of String Test to Shall	N	13.5%
Incomplete setup conditions	Y	1.5%
Incomplete Mapping of Shall AND Incomplete Setup Conditions	Y	0.4%
Incomplete Mapping of Shall to String Test	Y	9.1%
No Success Criteria found OR success criteria is incorrect	Y	2.9%
COMPLETENESS OF SUCCESS CRITERIA	**NA**	**86.1%**

Table 5: EDSMa Test Results

5.3 EDSMB FINDINGS:

The pass rate of Shalls in the EDSMb CSCI was 81.9%.

EDSMb Category	Severe (Y/N)	%age
Team has determined Success Criteria covers Shall	N	77.4%
Typo or Grammatical Error	N	0.0%
Incorrect Tracing of String Test to Shall	N	4.5%

SIGAda '07 November 4-9, 2007, Fairfax, Virginia, USA.
ACM 978-1-59593-876-3/07/0011.

EDSMb Category	Severe (Y/N)	%age
Incomplete Mapping of Shall to String Test	Y	9.9%
No Success Criteria found OR success criteria is incorrect	Y	8.2%
COMPLETENESS OF SUCCESS CRITERIA	**NA**	**81.9%**

Table 6: EDSMb Test Results

6.0 REPORTING FINDINGS AND RESOLUTIONS

Test results were documented via test logs. All problems denoted as 'severe' were addressed by the development contractor. There was large agreement of the results between the FAA and the development contractor. An explanation of the severe category finding and its resolution follows:

a). **Finding:**
Incomplete setup conditions: i.e. the setup condition for the correct testing of the Shall was incomplete.
Resolution:
A system problem report (SPR) was created to update the setup condition and re-run the test.

b). **Finding:**
Incomplete Mapping of Shalls to string test, i.e. the success criteria listed in the string test covers a portion of the Shall.
Resolution 1:
An SPR was created to update the success criteria and have the string test re-run.
Resolution 2:
It was determined the part of the Shall that was not covered by the string test, was covered by another string test. *This new string test was provided and verified by the audit team.*

c). **Findings:**
No success criteria i.e. no success criterion was identified in the string test specified in the VRTM.
Resolution 1:
An SPR was created to update the success criteria and have the string test re-run.
Resolution 2:
It was determined the part of the Shall that was not covered by the string test, was covered by another ST. *This new string test was provided and verified by the audit team.*

d). **Findings:**
The success criteria is incorrect i.e. the success criteria did exist in the string test specified in the VRTM, but was incorrect
Resolution 1:
An SPR was created to update the success criteria and have the string test re-run.
Resolution 2:
It was determined the part of the Shall that was not covered by the string test, was covered by another string test. *This new string test was provided and verified by the audit team.*

6.1 UPDATED TEST RESULTS

Upon review and inspection of the updated string test documentation, the final metrics were recalculated as follows:

CSCI	Success Criteria Pass Rate
EVAL	96.4%
EDSMa	99.9%
EDSMb	96.3%

Table 7: Updated Test Results

SIGAda '07 November 4-9, 2007, Fairfax, Virginia, USA.
ACM 978-1-59593-876-3/07/0011.

7.0 CONCLUSION

For the three CSCI's tested the audit resulted in a pass rate of over 95% and demonstrated confidence in the string test results provided by the development contractor.

For those CSCIs selected, 38% of the Shalls were verified by the string test method, causing a saving of almost 50% of the cost of full system test, given the cost of the string test audit. This shows the string test approach to be very beneficial from a project cost perspective.

From a schedule perspective, resources assigned to the string test effort were re-assigned to the full test effort of the remaining CSCIs, providing a buffer to ensure completion on schedule.

Finally, the test results demonstrated the code to be robust and allowed the government to move to full system testing with assurance in the code verified by the string test method.

8.0 ACKNOWLEDGEMENTS

The development contractor team, especially Mr. Larry Peterson of Lockheed Martin Transportation and Security Solutions, was very helpful in providing data used in the B-Level Shall audit; their assistance is duly noted and appreciated.

9.0 REFERENCES

1. System Specification Document, FAA-ER-2979 (Rev. A), August 16, 2005

2. En Route Automation Modernization (ERAM), System/Segment Specification, Volume II: System Architecture Design Document (SADD), October 14, 2003

3. Standard Sub-Process: Software Design/TSS-E35-S003-V8; January 10, 2007

4. Standard Sub-Process Software Development and Test/TSS-E40-S001-V14; July 14, 2006

SIGAda '07 November 4-9, 2007, Fairfax, Virginia, USA.
ACM 978-1-59593-876-3/07/0011.