

Evaluating COTS/GOTS Software: Functional Test Criteria

Dr. William H. Dashiell

DoD National Imagery and Mapping Agency

Phil Brashear

EDS Conformance Testing Center

As government agencies move toward commercial off-the-shelf (COTS) and government off-the-shelf (GOTS) software, they realize that vendor descriptions don't always give sufficient information about the products' capabilities. Program Managers (PMs) need to consider not only functionality of software for satisfying user needs, but also compatibility and performance in the program's environment. Unfortunately, PMs responsible for acquiring software, although quite good at developing screening requirements, are often not experienced in writing testable requirements. This paper is addressed to people in that position.

Planned and scheduled software testing—whether unit testing, integration testing, conformance testing, etc.—is a basic element of risk reduction in software acquisition. When software testing is viewed as unsuccessful, it often turns out that fundamental supporting concepts have not been fulfilled. Contributing factors to unsuccessful testing include:

1. Mistaken concepts of software testing definitions, objectives, or applications
2. Disagreements with customers or users regarding methodology, standards, or interpretation of results
3. Budget reductions or insufficient budget to support the required testing effort
4. Conflicts between testing schedule and product acquisition schedule. Too often the product acquisition schedule wins this conflict.

More and more agencies are discovering the need for carefully inspecting functional requirements and formal testing products for conformance to these requirements.

In the following scenario, we meet Pete Anthony, a PM assigned to acquiring a COTS or GOTS Financial Accounting Package. Some problems have arisen, and it has been recommended that he consult with Tom Edward, a contractor Testing Expert (TE) who had completed other software testing for the group. The scenario begins with the arrival of the TE at the PM's office.

Introduction to the Problem

PM: Good morning, Tom. Thanks for coming by.

TE: Glad to be here, Pete. What can I do for you?

PM: You know that my Personnel Management System acquisition went well, so I was tasked with acquiring a Financial Accounting Package. That did not go so well.

TE: What do you mean?

PM: Initially, I talked with prospective users of the package and formed a Product Selection Team (PST) to develop the selection requirement, selection process, and selection schedule. We interviewed users and managers, reviewed literature from various vendors, and wrote our selection requirements. Based on these requirements, the product literature, and user comments we invited five vendors to give demonstrations. The demos indicated that several of the products met our selection requirements. One product appeared to be outstanding in terms of functionality and cost.

Linda, one of the PST members, had recommended that we include an in-house testing process to reduce our risk of wrongly selecting a product. We followed her recommendation and included testing in the selection process and schedule. After the demos, another member raised a serious question. Everything we had done so far had taken a little longer than planned. Would we be able to have the new system running by the planned operational date? To put our problem another way, which was the greater risk, failing to meet the delivery schedule or selecting an inadequate product?

This started us thinking about the testing process again. We discussed it at some length, and finally concluded that the cost/benefit ratio did not justify testing. The vendors' literature and the demos showed that the remaining candidates would meet our needs, so there was no significant risk.

TE: Uh-oh.

PM: Exactly. That was a mistake. What we thought was a small risk, was not.

I directed the Contracting Officer to finalize the contract with the selected vendor, and the vendor arranged for installation. I notified the potential users of the installation, training, and data migration schedule, and told them that the COTS Financial Accounting Package would be operational in 60 days.

The vendor came in to install the software. After five days without success, the vendor pointed out that the contract called for payment of \$150 per person per day to complete the installation. I was upset and demanded to know what was wrong. After all, the demo was perfect! The vendor representative explained that the demo was performed on a different hardware and operating system environment. The product was not really ready to operate on our LAN, which has servers running Linux and clients running Windows NT.

Reducing Risk with Software Testing

TE: What did you do?

PM: I had to cut my losses. I directed the Contracting Officer to cancel the contract, notified the team and my supervisors of the cancellation and the reasons for it. My supervisors were pleased that I had stopped the process before any more damage was done, and asked me to pick up the pieces and restart. The first thing I did was call Linda, who recommended testing in the first place to ask for advice. We discussed how testing should have caught the problems and that neither of us knew enough about testing to do it right. She suggested I call you to see what pointers you could give me.

TE: Sounds like what we need to talk about is conformance testing against your requirements.

PM: I am not sure that I know exactly what you mean by conformance testing. Could you be more specific?

TE: Conformance testing is simply formal testing of a product against a set of conditions to which it must conform. For example, my organization tests compilers for programming languages, display software for Computer Graphics Meta-file (CGM), and other products. The conformance requirements are provided by international standards, and our testing is highly formalized, using extensive suites of tests that have been validated by experts in the various fields.

PM: Why would anyone need that level of formality? Is it not enough that the supplier implemented ISO standard C++ in his compiler or that his CGM interpreter is widely used in industry? If more assurance is required, can we just verify that the compiler processes our own programs correctly, or

that the CGM interpreter properly displays the pictures that we care about?

TE: In this information age, we have come to depend on software to create, process, transmit and store critical information. Our missions live and die on the correctness and reliability of such software. Just consider the rapid growth of electronic commerce in both the government and commercial worlds. For instance, the Department of Defense (DoD) has hundreds of sites all over the world that are processing contract data. They rely on the correct behavior of the software to keep that data correct, secure, and synchronized. Airframe manufacturers depend on CGM interpreters to display design diagrams in hard copy and on screens; if those diagrams are not correctly rendered, costly errors can result. Weapons systems depend on software for targeting, for vehicle control, and for stores management. If a compiler does not translate the programmer's intent correctly, warfighter and civilian lives can be threatened.

PM: OK, I can understand the importance for high-risk situations, but is conformance testing really needed for my Financial Accounting Package? When I buy a television set, I do not subject it to all sorts of tests.

TE: That is because first of all, you know that if it does not work correctly, you can get satisfaction from your warranty. You might miss a football game or episode of JAG, but no long-term damage is done. However, can you afford the loss or corruption of your Financial Accounting Package data? Can you afford the down time when payroll checks are due? If the information is important enough to be managed, then it is too important to risk. The consequences of nonconformance for your Financial Accounting Package may not be life-threatening, but the inconvenience could be pretty severe. You apparently have already experienced the problem when you could not install the selected software package.

PM: That is true. I agree that conformance testing is what I need. We already have our selection requirements, so all we need to do is develop tests.

Reducing Risk—Defining Test Requirements

TE: What are some of your requirements?

PM: Here are a few examples: One of our data requirements is that the product must provide a flexible and common tracking numbering system. Under security requirements, it must support multiple levels of administrative access to the data. Finally, one of our interface requirements is that the product must be able to export data in a format readable by a spreadsheet program. What do you think?

TE: To be polite about it, I think they need work.

PM: Why? Aren't they sufficient to separate sheep from goats?

TE: Exactly, but I could not build a test for any of them. They are screening requirements, not testable requirements. They are at a pretty high level of abstraction. The granularity of your requirements is too large to allow me to detect the differences we need to detect if we are to see the risks for each choice we may take. You will need to develop testable requirements based on your screening requirements.

PM: Screening requirements, testable requirements. What is the difference?

TE: Screening requirements help to identify software that offers functionality similar to what you need; that is, separate the sheep from the goats. You have that. What you need now is assurance that this functionality performs as advertised and that your needs are really met. That requires more detail. I would expect each screening requirement to lead to several testable requirements, each addressing a specific facet of the screening requirement.

Look at these one at a time. Your first requirement calls for a "flexible and common" numbering system. The first thing that I notice is ambiguity. Do you mean that the numbering system must be common to multiple features, or that the product must allow the use of a numbering system that is commonly used in your organization's tracking? If the latter is what you want, then what are the candidates? How can we define *flexibility*? What kind of flexibility do you need, and to what degree? Finally, what kinds of entities need to be numbered? The test team has to know exactly what you want in order to produce satisfactory tests.

PM: I think I see what you want. We need to specify this requirement more carefully. I could start with something like this: The product must provide a system of numbering data items that allows association of expenses and revenues with projects, bids and proposals, training efforts, and other activities. The numbering system must be multilevel and must be consistent across all uses; for example, the first portion of the identifier might be used to identify the category (service project, bid/proposal, training, etc.) while the second might be used to distinguish expenses and revenues. The product must allow the agency to define the number of levels and the meaning and format of each.

TE: That is a good start. I still cannot write tests based on this requirement, but you have removed a good deal of the ambiguity. Let's use your second requirement to illustrate what is still needed.

PM: That is the security requirement. We want multiple levels of data access.

TE: Yes, but how many levels do you want? How do you want the access levels to be determined? The problem is that the requirement is at a high level of abstraction. The granularity is too large to allow us to detect the differences that indicate the risks for each choice we may take.

PM: What do we need to do to this requirement?

TE: We need to apply an iterative process, breaking it down into more specific requirements, then breaking each of those down again, repeating until we reach individually testable requirements. The first breakdown might include:

- Distinguishing among no access, read-only access, and write access.
- Determination of access rights by user ID and password.
- Determining access rights by project, determining by expense items vs. revenue items, or by forecast vs. actual.
- Giving the system security manager the ability to change access rights by individuals or groups of users.

PM: It is a matter of specificity vs. generality?

TE: Correct. Eventually, we must have requirements such as, "The product shall permit the system security manager to change a user's level of access to the revenue data for an existing project, from any level to any level, while the user is viewing such data."

PM: Is that possible for most database applications?

TE: I am not proposing this as an actual requirement, but giving it as an example of a testable requirement. You have to decide whether it is a requirement.

PM: There is clearly a lot of work to do; what about the interface requirement?

TE: That is not really a conformance requirement, but more of an interoperability requirement. You might require that the product is able to export selected data in a particular format; you cannot require that another product, such as a spreadsheet program, accept that format. The way your requirement is stated, you are trying to do both.

Interoperability Testing vs. Conformance Testing

PM: Isn't interoperability testing part of conformance testing?

TE: It can be, depending on how you view the system. If you are merely interested in the functional requirements of the Financial Accounting Package, the answer is no.

Interoperability testing focuses on the interfaces among different products, so the functionality of other products comes into play. If you view a collection of products as a complete system, and your functional requirements are stated for the entire system, the components' interoperability is part of conformance testing. Otherwise, I think of interoperability testing as a separate activity.

PM: Is any part of the interface requirement related to conformance testing?

TE: I think so. For example, most spreadsheet programs have the capability of reading textual tables of data where the column entries are separated by commas or by tabs. You can require that your package is able to output comma-delimited text files representing tables of data (although you would need to state it with more specificity). We could develop a conformance test for such a requirement. It is the spreadsheet program's ability to accept the comma-delimited format that we cannot handle as a conformance issue.

PM: I think I see where to go with the interoperability issue. Now let's focus on conformance testing of the single product. You have implied that it might need to be pretty extensive. How do I find the time and money?

Risk of Not Testing

TE: The question is whether you can risk skipping conformance testing. You are looking at a major investment in this Financial Accounting Package. You have seen the consequences of trying to save money and time up front at the risk of spending more money and more time later on. Never mind the other consequences of corrupted or lost data. It is like a design-and-build activity; investment of resources early in the process mitigates against risks later, when costs and consequences are more significant. I strongly recommend including requirements development and testing in your budget and schedule.

PM: It is clear to me that this conformance testing is beyond my level of expertise. Where do I go to get help?

Software Test Team

TE: That depends on your organization. There might be a suitable testing group within your agency. On the other hand, you might want to bring in a non-government group. You should expect better results if the test team has no interest in the outcome. You want a test team that has no legal or financial ties to the prospective suppliers or to your acquisition effort. You are trying to reduce risk, and that means you want the most objective viewpoint possible.

PM: What services should I expect from a testing group?

TE: First, it should work with you to derive functional requirements from your screening requirements. At each stage of this iterative process, it should work to ensure that its understanding of the requirements is the same as yours. Second, it should decompose the functional requirements into individual test cases, with pass/fail criteria for each case. Third, it should perform the actual testing, using the deployment system or an exact replica of it. Fourth, it should produce the required deliverable products.

PM: Your first expectation relieves my mind. I was worrying about this process of deriving testable requirements, but you are saying that is one of the test team's jobs, working with

us. That is the source of expertise that I am missing.

TE: Yes, this is the point where cooperation is the most important. The test team has the expertise to derive testable requirements, but only your organization can say, "Yes, that is what we really want."

PM: What are the deliverable products that I should ask for?

Software Testing Deliverables

TE: First and foremost, a complete test plan. This document would probably be delivered several times, and would eventually include the original requirements, the functional requirements with traceability to the screening requirements, the test cases, procedures for executing the test cases, and a complete schedule for the testing effort.

Second, a test log should be provided documenting the steps as actually performed. Each individual tester should keep a complete diary of his or her testing activities. The test log should be constructed by integrating these individual testing diaries. The test report should have an executive summary providing enough information to allow informed decision making.

PM: Have we covered everything? Is there anything else I need to consider?

TE: There is one thing you should be prepared to face. It is quite possible that none of the candidate products will pass every test case. You should consider some prioritization or some plan for providing missing functionality if no product has absolutely everything that you need. That is a knotty problem, and its solution is not exactly in the center of my area of expertise.

PM: Tom, I appreciate your advice, and I think I have no choice but to accept it. If we decide to look to an outside organization for testing services, I will certainly include your company on the list. Thanks again for coming by.

TE: Thanks, Pete. See you later.

Summary

Why do I need conformance testing? In the world of electronic commerce, we are totally reliant upon software to create, process, transmit, and store critical information. We must be able to trust this software to correctly and reliably provide the required functionality.

Why are my preliminary screening requirements unsatisfactory? Screening requirements help to identify software that offers functionality similar to what is required, but we need assurance that the offered functionality performs as designed or as advertised to satisfy our needs. Screening requirements are given at a high level of abstraction, such as "The product shall be Y2K compliant." Testable requirements are derived from screening requirements, often with a testing team's help, and address specific items (e.g. "The product must accept Feb. 29, 2000 as a valid date and must reject with a warning message Feb. 29, 1900.")

Is interoperability testing part of conformance testing? Whereas conformance testing attempts to determine whether a product meets its functional requirements, interoperability testing focuses on the interfaces among different products. If a collection of products is viewed as a complete system, and functional requirements are stated for the entire system, interoperability of the components is part of conformance testing. Otherwise, we view interoperability testing as a separate, related, activity.

How do I find the time and money for conformance testing? In the case of a major investment of COTS software, you must consider the consequences of not doing conformance testing.

If your agency discovers that an already purchased product does not meet the agency's needs, money and time have been wasted. Recovery from this situation is likely to cost far more money and take far more time than would have been consumed by testing. Requirements development and testing must be planned as part of the acquisition cycle.

How do I acquire conformance testing services? That depends on your organization. The conformance testing team should be a third party, not associated with the acquisition group or the vendor. Conformance testing could be performed by a team outside the organization or from a separate division of the acquiring agency. It is important that the testing team have no financial, legal, or other dependence upon the acquisition team.

What services should I expect from my testing organization? It should help derive functional requirements from the screening requirements, ensure that its understanding of these requirements is the same as that of the acquisition team, decompose functional requirements into individual test cases, identify pass-fail criteria for each test case, perform the testing, and produce the required deliverable products.

What deliverable products should I expect from testing? The test team should provide a complete test plan, including requirements, test cases, test products, and schedule; a test log documenting the steps performed; a test report showing the individual pass/fail reports; and an executive summary sufficient for the acquisition team to make a decision. ♦

About the Authors



William H. Dashiell is a computer scientist at the DoD National Imagery and Mapping Agency. He has worked on the development of software testing by statistical methods using binomial models, coverage designs, mutation testing, and usage models. He has contributed to the development of conformance and testing protocols for federal, national, and international information technology standards. He has a bachelor's degree in business administration and in education, a master's degree in education technology, and a doctorate in mathematics education from the University of Maryland. He also has a master's degree in computer science from Hood College in Maryland.

William H. Dashiell, Ph.D.
U.S. Government; NIMA
Voice: 703-874-8219
Fax: 703-874-8841
Adwd@netkconnect.net



Phil Brashear is a senior systems engineer at EDS in Dayton, Ohio, where he leads the EDS Conformance Testing Center and Performance Software Quality Assurance on various DoD projects. He directed compiler testing efforts and test suite maintenance at CTA Inc. from 1989-97. He performed compiler validations and directed test suite enhancements at SofTech Inc. from 1986-89. Prior to that, he was a member of the mathematics and computer science faculties at Eastern Kentucky University. He has a bachelor's degree in mathematics education from the University of Kentucky, a master's degree in mathematics from Northwestern University, and completed course work and exams for a doctorate in mathematics at the University of Georgia.

EDS Conformance Testing Center
P.O. Box 24593
Dayton, Ohio 45424-0593
Voice: 937-237-4510
Fax: 937-237-4660
Internet: www.eds-conform.com

Coming Events

September 18-19

The Internet Challenge—The Utility Response to a .Com World www.tdworld.com/marketing/interchall.htm

September 26-28

2nd Computer Security & Information Assurance Conference
www.certconf.org

October 15-19

Object Oriented Programming Systems Languages and Applications Conference (OOPSLA 2000)
www.acm.org/events



October 23-25

Symposium on Operating Systems Design and Implementation
www.usenix.org/events/osdi2000

October 30-31

3rd International Conference on Practical Aspects of Knowledge Management (PAKM 2000)
www.do.isst.fhg.de/workflow/events/index_e.html



November 10

Information Outlook 2000 (Australian Computer Society)
www.acs.org.au/act/events/io2000/index.html

November 16-17

ACM Conference on Universal Usability
www.acm.org/sigchi/cuu

December 4-7

International Conference on Power System Technology
www.ee.uwa.edu.au/~aipps/powercon



December 11-13

Global Development Network Conference
www.gdnet.org

January 18-19

Measurement Science Conference
www.msc-conf.com/findex.html#cfp2001.html

January 25-27

21st Annual National CSIE Conference
www.ryerson.ca/~csie/2001/engindex.html

January 30-February 2

CIEC 2001 Odyssey: Industry & Education Engineering
www.asee.org/conferences/html/ciec2001.htm

February 7-9

Network and Distributed System Security Symposium
www.isoc.org/ndss01/call-for-papers.html

April 29-May 3

STC  *Software
Technology
Conference*

www.stc-online.org