

TITLE: "TESTING 1, 2, 3..."

by Tim Bryce
Managing Director
M. Bryce & Associates (MBA)
P.O. Box 1637
Palm Harbor, FL 34682-1637
United States
Tel: 727/786-4567

E-Mail: timb001@attglobal.net

WWW: <http://www.phmainstreet.com/mba/>

Since 1971: *"Software for the finest computer - the Mind"*

"Most beta test programs are designed more for promoting the product as opposed to actually testing it."

- Bryce's Law

INTRODUCTION

If you tuned into my August 22nd broadcast of *"Management Visions"* you heard me whine about how poorly commercial software is being tested by the major vendors. Normally, shoddy workmanship results in clients distrusting the products and vendors. Surprisingly, this is not the case in the IT Industry. Instead of holding vendors accountable by their customers, sloppy workmanship has become an acceptable form of behavior in this field.

Let me share with you the standard approach to testing as used by today's software companies:

1. Announce a general release of the product to the press.
2. Write a section of code and test/debug it.
3. Assemble the code and issue it to customers in "Beta" form.
4. Let the customers play with it and report problems as they occur.
5. From this, the vendor develops and prioritizes a punchlist of corrections.
6. The punchlist is addressed with the product delivery date in mind.
7. The first release of the product is issued with much fanfare (even though the punchlist has not been fully corrected).
8. More errors are reported by the customer base and added to the punchlist.

9. The second release of the product is issued as a fixpak. This release, in reality, represents the finished product.

The intent of beta code is more for public relations and publicity as opposed to delivering a viable product offering. Instead of thoroughly testing their products themselves, vendors let their customers do it for them. It shouldn't be this way.

I don't have a problem with the beta-test concept as such, so long as the vendor has legitimately tested the product themselves prior to issuing the release. They're not. Is it that they are incapable of testing it themselves? Maybe. Most likely though, it is cheaper to have outsiders test it for you. However, beta testing represents a disorganized test of the product. Something much more structured should be done prior to this by the vendor.

A SYSTEM IS A PRODUCT - THE "EXPLOSION/IMPLOSION" PROCESS

As we have discussed on more than one occasion in these bulletins, a system is a product that can be engineered and manufactured like any other product. Let's think about the design of a product, such as an automobile. The overall product is first broken down into a series of assemblies, such as the chassis, body, engine, drive train, and so on. Each assembly is then "exploded" (sub-divided) into subassemblies, and components.

After the product has been divided top-down into its assemblies, parallel teams of designers concurrently refine their portion of the product with little interaction between assembly design teams.

Parts such as nuts and bolts are carefully identified, classified, and reused not only within the product but with other products. One classic example of this is years ago when General Motors was able to slip a Pontiac Engine into an Oldsmobile. This demonstrated the power of standardization and reusability of parts.

After a product has been fully designed, the product assembly line "implodes" the product (bottom-up) with components making up subassemblies, and then the subassemblies into assemblies, and then assemblies into the final product. Again, think about the assembly line for automobiles where the pieces and parts of the car are built in smaller increments until the final assemblies are put together and the key is turned and the car is driven off the assembly line.

(continued on page 2)

(continued from page 1)

This approach to product design is obviously an old one, but it can be easily adapted to the design and development of an information system.

Under "PRIDE"-ISEM an information system is a 4-level hierarchical structure consisting of the system (representing the product) which is decomposed into sub-systems representing business processes. Each sub-system is divided into procedures (both administrative and computer). Administrative procedures are broken into operational steps or manual tasks. Computer procedures are broken into programs.

As in the product analogy, the system is designed top-down into sub-systems. Once this has been done, parallel teams of designers can work concurrently developing separate sub-systems.

When the final pieces and parts have been designed, they are developed and tested "bottom-up." For example, unit tests are performed on programs, a string test of the programs within a computer procedure is then performed; this is followed by a test of all of the procedures in each sub-system, and finally parallel testing of all of the sub-systems in the system is performed. If successful, the key is turned and the car is driven off of the assembly line.

The so-called "parts" of the system-product are the data components representing data elements, records, and files. Like product parts, the data resources should be carefully identified, classified, and reused not only within the system, but with other systems, thereby promoting integration.

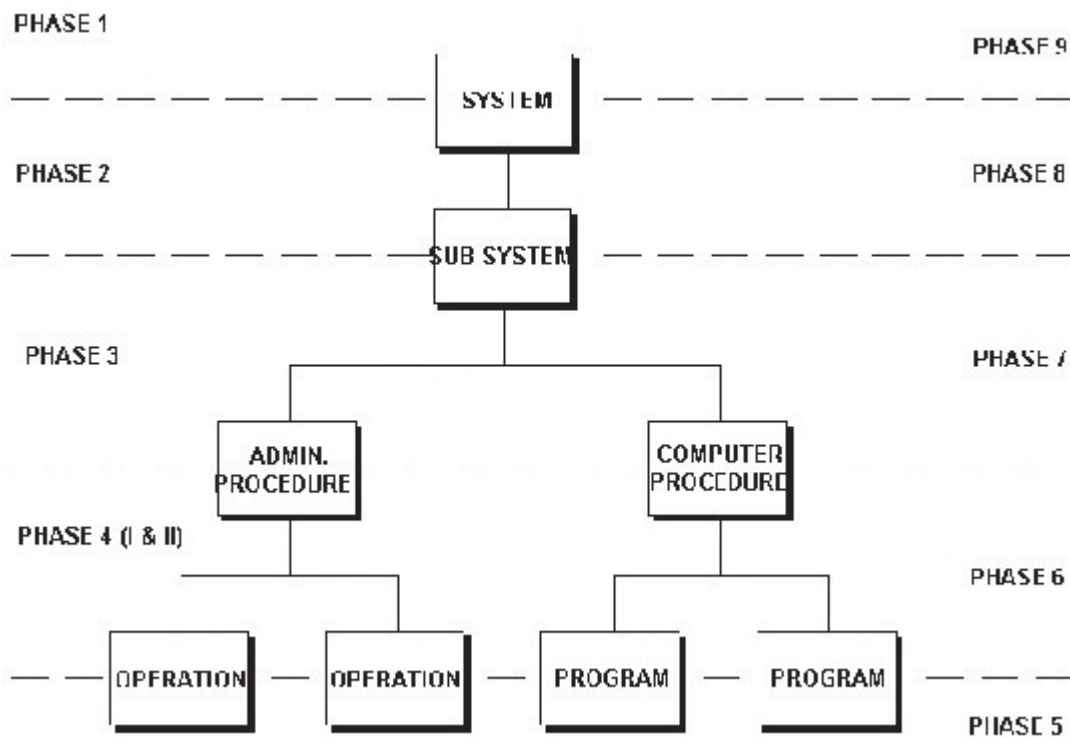
The point is, like products, systems are designed by explosion (top-down if you will) and tested and implemented by implosion (or bottom-up).

TOP-DOWN TEST PLANS

The early phases of "PRIDE"-ISEM are primarily used to design the system into refined components. They are also used to specify the testing criteria for the later phases; to illustrate.

PHASE 2 - SYSTEM DESIGN - The Test Plan at this level will be used later in Phase 8 and should cover how parallel testing of the sub-systems should be performed. Key file maintenance sub-systems should be tested and installed prior to simple "read-only" sub-systems.

(continued on page 3)



(continued from page 2)

PHASE 3 - SUB-SYSTEM DESIGN - The Test Plan at this level will be used later in Phase 7 and should address human/machine interaction. Since inputs and outputs should be fully defined by this phase, the Test Plan should include:

- Design standards to be observed for Graphical User Interfaces and report layouts. This includes the use of Help text.
- Default values and validation rules for collecting data.
- Ease of use (intuitiveness).
- The processing of optional parameters.
- Verify the overall processing logic.
- Benchmark tests.

PHASE 4-II - SOFTWARE ENGINEERING - The Test Plan at this level will be used later in Phase 6 and should check operational dependencies between programs and stress testing. Input/File/Output layouts should be verified as well as volume testing. The assemblage of test data should be included.

BOTTOM-UP TESTING

The remaining phases in "PRIDE"-ISEM are used to test the system from the bottom-up.

PHASE 5 - SOFTWARE MANUFACTURING - Following the production of the executable program, the programmer assembles pertinent test data to test and debug the individual program. To do so, the programmer consults the Test Plan, Program Specifications, and pertinent Software Structure Diagrams resulting from Phase 4-II. This becomes the "roadmap" for testing the program.

There are a variety of testing and debugging tools available on the market. Use of such tools should be encouraged.

When the program has been tested to the satisfaction of the programmer, the results are reviewed with the computer procedure designer and Quality Assurance. This review assures that all standards have been followed, and the program has been tested to conform to specifications. The test data is saved for later reference and the test results are filed for future reference.

PHASE 6 - SOFTWARE TESTING - The purpose of this phase is to test and correct errors in the overall computer procedure. This is sometimes called a "string test," "job stream test," or "stress test." Whereas the "unit test" performed in Phase 5 was concerned with data format,

the emphasis in Phase 6 is on volume testing.

It is the objective of the Computer Procedure Designer to assure that the procedure meets the sub-system specifications. In addition to testing for correct processing logic, they also check the operational considerations of timing, re-start, etc.

Following testing, the results are reviewed with Quality Assurance and the Sub-System Designer. This assures compliance with all standards and that the Sub-System requirements are met. When the testing is satisfactorily accomplished, the test data is filed for future reference.

PHASE 7 - SUB-SYSTEM TEST - Testing in Phase 7 is performed in a similar manner as in Phase 6, except at a higher level. Whereas Phase 6 "Software Testing," is performed by Software Engineering for all of the programs within the computer procedure, Phase 7 is performed by Systems Engineering for all of the procedures in the sub-system, both administrative and computer.

During Computer Procedure Design, Software Engineering prepared the computer procedure that will be used by DP Operations. In order to test this procedure, it is recommended that DP Operations actually perform the tests under Software Engineering's direction. In this way, the operating procedures are tested along with the programs used to implement the procedure. Often the operating people will point out areas for procedure improvement.

During Phase 3, Systems Engineering specified a testing criteria for the overall sub-system. This included the validation of input/output processing, along with anticipated transaction volume for the sub-system. Phase 7, therefore, is used to verify the test criteria and benchmarks.

Following Phase 7, the sub-system is essentially ready for operation. However, considerable care should be exercised to avoid a premature installation of the sub-system for day-to-day operation. This situation could cause panic conditions if other parts of the system are not ready. This is why Phase 8 is used to release sub-systems.

PHASE 8 - SYSTEM OPERATION - A final test of the system is performed where the various sub-systems are tested in parallel. This is similar to the testing in Phases 6 and 7, but at the highest level. It is not uncommon to invite user personnel and DP operations to participate in

(continued on page 4)

(continued from page 3)

the test. The Administrative Procedure Manuals and Computer Run Books, as created in the various Phases 4-I's and 4-II's, are used as part of the formal walk-through. The test data as created in previous phases is assembled and used as the basis for the test. Where available, actual data should be tested to simulate actual system situations. If any errors are discovered, they are corrected immediately and testing is resumed. Changes to the sub-systems may also be proposed; however, these should be considered carefully. In some instances, it may be more appropriate to suspend any modifications or improvements temporarily until use of the new system has settled into routine operation.

CONCLUSION

Whether you use our "PRIDE"-ISEM or not, the concept of defining refined levels of test plans "top-down", and performing corresponding tests "bottom-up" is a simple and effective approach. It offers a structure that is easy to manage and provides for a rigorous test of the system. Test data generators and debugging aids are useful and should be encouraged, but simple organization and a methodical approach to testing can have a more dramatic effect. Following this, if you still want to perform a beta test, you have my blessing.

Always remember this, most beta test programs are designed more for promoting the product as opposed to actually testing it. Think about it; its a "no-brainer" for the vendor. Customers begin to talk about the product while assuming the testing costs of the vendor. Even if something were to go wrong (and frequently does), the vendor is not liable and simply shrugs off the snafus due to beta code.

I used to beta-test software products for vendors, but I no longer have the time nor inclination to do the manufacturer's work for them anymore. Further, I no longer rush out to buy the latest release of "any" software product; I have been burned too many times by the vendors. As far as I'm concerned, the software vendors really need to clean up their act when it comes to testing. If they really want us to test their products for them, let us know where we should send the bill.

END

"PRIDE" Special Subject Bulletins can be found at:

<http://www.phmainstreet.com/mba/mbass.htm>

They are also available through the "PRIDE Methodologies for IRM Discussion Group" at:

<http://groups.yahoo.com/group/mbapride/>

You are welcome to join this group if you are so inclined.

The "Management Visions" Internet audio broadcast is available at:

<http://www.phmainstreet.com/mba/mv.htm>

"PRIDE" is the registered trademark of M. Bryce & Associates (MBA) and can be found on the Internet at:

<http://www.phmainstreet.com/mba/pride/>

Copyright © MBA 2005. All rights reserved.