

DosGetNews()

April, 1989
Vol. 2, No. 4

OS/2 COMPONENT TEST GROUP

by Moshe Dunie
Manager, OS/2 Component Test

Dan Hinsley recently asked why the test managers are the only OS/2 managers with grey hair. Actually, I received my first grey hairs during my years as a development manager. Hopefully, this article will give you some idea of why life in OS/2 Component Test is not grey despite some grey hairs.

The OS/2 Component Test organization consists of four groups: PM CT Development (Pat Tharp), Kernel CT Development (Jeffpa), I/O CT Development (LarryBa) and Run (Jodyg/Nancys). The first three groups are test development groups staffed by Software Design Engineers, while the Run group carries out test execution tasks.

The Component Test (CT) goal is to provide 100% functional coverage for each component in a controlled environment, extensive errorcondition testing, complete boundary and limit testing, and component stress testing.

An OS/2 test developer is given the responsibility to develop software applications that systematically verify that a given component performs as specified in the System Design. In a way, the test developer is the first application writer for a new

OS/2 functionality. However, while the typical application writer needs to know only the interface to the OS (API), the OS/2 test developer needs to know the OS internals in order to assure proper test coverage and to isolate failures correctly.

The Component Test cycle for every new OS functionality includes five phases:

The first phase is a Design phase. It starts with reading the OS/2 Design Workbook as well as the minds of the OS designers. The goal is to verify the System Design. The CT Exit date is firm, however, the design specifications are often in a state of flux. Moreover, design specifications have a tendency to change significantly on the day the test developer completes the coding of his test software. Building an operating system is inherently different from a construction project (although existing project management tools fail to recognize this). You don't redesign the foundation while you are building the roof and you do not raise a DCR to your home builder to expand the capacity of all your pipes after the painting is done. DCRs often hit Component Test badly, as they require unplanned test development/modification effort by a very scarce resource that is already committed to a very tight schedule. Historical data show that the

number of new/modified lines of test code required on average by a DCR is higher than the number of new/modified OS code. Consequently it is extremely important that Development keeps CT in the loop of design changes. Participation in 2.0 memory management design reviews has proven very helpful and sitting in design reviews is encouraged. The outcome of the design stage is a software design of the test application and an IT1 document that lists the test case variations to be performed and the expected results. The design phase requires strong software engineering skills and sound test judgement. You cannot test every combination. Yet the test application needs to properly exercise and stress the component code and uncover subtle errors.

The coding phase is a significant part of the CT cycle. It is followed by the Debug phase, which is particularly challenging since the Test Developer needs to be able to debug his code in an unstable OS environment. At this stage any code failure could be either in the test code or in the OS, and the Test Developer needs to trace down OS code with the Kernel debugger isolating the detected failures. The start of the Debug phase is dependent on the availability of the new OS/2 functionality in a private or public driver.

As long as the OS fails to pass the newly developed test variations, the Test Developer owns test execution, isolating failures, preventing false PTRs and verifying PTR fixes. This is the DRun phase. Its duration depends on the number of detected failures and the PTR fixing rate.

At the point when the new OS functionality is reasonably stable, the Run team takes over ownership of test execution. It runs regression tests until each component meets the CT Exit criteria of 100% of variations attempted on a public driver with 95% of the variation successfully executed for each component and an overall success rate of 98%. CT Exit requires all severity 1 and 2 PTRs closed on public build drivers and severity 3 and 4 PTRs answered and negotiated with IBM's System Assurance and System Test. The Run team main-

tains the test run results.

Finally, I would like to focus on our Random Test Generator (RTG). The RTG is a large PM application designed to test the Engine by calling its functions in a random order with random arguments, given the large number of relevant combinations. Since equally weighting all functions and all types of arguments may lead to many uninteresting tests, we created control files (.CTL) which specify the probability of various events. We can pick a specific group of functions to call, call some more than others, initialize our tests with different functions, and then even tailor the types of arguments we want. We can test mainstream functionality, or go for error and boundary conditions as well. To reproduce a particular problem, we attach the seed for our random number generator to a control file to produce a script file (.SCR). These files always do the same tests. file (.SCR).

Every new OS/2 release is the outcome of a participative effort. We take pride in Microsoft commitment to excellence and work in close cooperation with Development, Program Management and Project Management in order to assure a high quality product.

DosGetNews()

Editor
Madeline Weise

DosGetNews() is a monthly news service published for and about Microsoft's Advanced Operating Systems Department. Its intent is to inform employees of departmental projects and activities. Contributions are welcome. Please see the editor in 2/2118 (email madelw).

Documentation, Process and Me

by Janine Harrison
Documentation and Process Coordinator

A lot of people look at me and ask "What do you do?". I think that this is a very good question as I am constantly asking it myself. For the most part, my job involves coordinating the Development Documentation for OS/2. I have also expanded recently into the issues of Process.

On the documentation side, I work with IBM (Boca and Hursley) to have (or maybe I should say ATTEMPT to have) up-to-date documentation. With Hursley, I am constantly asking them to send me updated on-line copies. They always tend to send hard copy and the cycle gets repeated to get on-line copies. I also ask questions regarding where information is in the spec or where it went from the 1.0 or 1.1 spec.

MS and IBM Boca have a much better working relationship. We use the PCV main-frame in Boca to pass document files. Both of us contribute to updating a document, rather than with Hursley where we have very little ability to contribute. I get lots of help from Madeline Weise in uploading and downloading files, plus help with building the documents once we receive the files.

The procedure for generating documentation is very different depending on which version of Script

was originally used in its production. If the doc is in ISIL or Book Master (supersets of Script), the files have to be edited and the incompatible Script commands removed or changed into something Dwscrip can handle. There is also the age old problem of stack space using Dwscrip since it is written in Basic. In general, if the document was written using VM Script, then it can be built with only a few changes to the sources.

Printing is another aspect altogether. Currently I 'borrow' the Net User Ed group's printer to handle all of the large jobs. The specs and the workbooks range in size from 200 to 1500 pages. Once a master of the document is printed, I get invaluable help from Marina Pierce and Jami Sherer to get the copies made and for distribution. I am currently organizing the documentation so that there should be hard copy available at all times. In the past, this has not always been the case.

Distribution is an area where I would like to have suggestions. Marina and I have tried several approaches: sign-up sheets, email to her, or email to me when there is a new document or when an update is available for distribution. None of these have worked as smoothly as we would like. If you have a better idea, I would appreciate hearing it.

After/before/during the documentation jobs comes Process. Peter Neupert appointed me the contact for process issues with IBM. So far I have worked with them on a new DCR process for 2.0, a new Design/Development process for OS/2 and have recently become involved with the IBM higher level process discussions.

I have been negotiating with Boca and Hursley to reach agreement on a common DCR template and on making the DCR processes at these sites similar. (Currently there is no way to make them the same.) In 1.1 and 1.2, PM used one template and the Kernel used another one. So anyone who wished to raise a DCR had to first figure which system to raise the DCR, get the correct template filled out and then deal with the different process steps to get the DCR approved. We had an "agreed" template, but Hursley has been having trouble with it recently. I have asked Richard Tait and Steve Wells to help sort this out. On the Boca side, the DCRs seem to moving as smoothly as one could hope.

Boca wrote a "Joint" OS/2 Design/Development Process Document. Joint being Boca and MS. After reviewing the document and getting in-house comments, I met with John McAdaragh and Bill Miller to resolve our differences. As a result we have decided to create a skeletal version of this document on the TOOLS Cruiser disk in Boca. We have an agreed Table of Contents and Bill is in the

process of placing those sections from the old document to which we were in agreement into the new format. My involvement in the IBM higher level process is derived from this document and how it fits into their ESD Systems Process document.

I have several other projects going. These include the following:

- * putting together information for new hires on documentation.
- * finding out about an IBM PM app which uses Book Master files.
- * trying to find better ways to handle the increasing load of documentation for OS/2.
- * resolving documentation questions and problems
- * staying on top of all the documentation.

I am always open to suggestions, ideas and any problems related to documentation. I know a big "bitch" is using Dwscrip. I am constantly looking for a better tool to use, but currently my hands are tied. I would also appreciate hearing about new documentation you might receive from sources within IBM.