
N A S A

N A S A

```

*****
*
*
*
*   U S L / D B M S   N A S A / P C R & D
*
*
*   W O R K I N G   P A P E R   S E R I E S
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*****

```

Report Number

DBMS.NASA/PC R&D-13

The USL/DBMS NASA/PC R&D Working Paper Series contains a collection of formal and informal reports representing results of PC-based research and development activities being conducted by the Computer Science Department of the University of Southwestern Louisiana pursuant to the specifications of National Aeronautics and Space Administration Contract Number NASW-3846.

For more information, contact:

Wayne D. Dominick

Editor

USL/DBMS NASA/PC R&D Working Paper Series
Computer Science Department
University of Southwestern Louisiana
P. O. Box 44330
Lafayette, Louisiana 70504
(318) 231-6308

DBMS.NASA/PC R&D-13

WORKING PAPER SERIES

(NASA-CR-184545) USL/DBMS NASA/PC R AND D
PROJECT SYSTEM TESTING STANDARDS Final
Report, 1 Jul. 1965 - 31 Dec. 1967
(University of Southwestern Louisiana.
Lafayette. Center for Advanced Computer

N89-14565
Unclas
G3/82 0183563

N A S A

N A S A

USL/DEMS NASA/PC R&D PROJECT

SYSTEM TESTING STANDARDS

Srinu Kavi

Dennis R. Moreau

Lin Yan

The University of Southwestern Louisiana
Computer Science Department
Lafayette, Louisiana

October 12, 1984

DEMS.NASA/PC R&D-13

- 1 -

WORKING PAPER SERIES

This document establishes a set of system testing standards to be used in the development of all "C" software within the NASA/PC R&D Project. Testing will be considered in two phases, namely, the program testing phase and the system testing phase. The objective of these standards is to provide guidelines for the planning and conduct of program and software system testing.

(1) Desk Checking

- Syntax Check:

- Indentation discipline followed.
- Every "{" should match every "}".
- Spelling of identifiers and builtin function names.
- All variables are declared and not previously declared differently.
- Comments begin with "/*" and end with "*/".
- Each left parenthesis has a corresponding right parenthesis.
- An even number of double quotes (") per statement.
- If a " needs to be output, \ should precede ", like \".

- Semantic Check:

Does the program do what its title says it does? Take time to check everything carefully. Check for each loop termination. "Walk through" the program with sample cases of test data.

(2) Test Planning and Strategy

- Test Plan:

The program test plan is the documentation of the planned sequence of tests, test cases, and expected results of each series of tests. This plan must be reviewed and approved by the project leader.

- Sequence of Tests:

Top down testing is recommended, i.e., major routines, main-flow of the logic, and sections involving a less complex level of detail are tested first; the sequence might be:

File opening and closing
Start-of-program housekeeping
Major input and output handling routines
Major control routines
Update routines
Output processing
Major error and exception routines
Minor error and exception routines

- Actual results vs. expected results

When the testing is completed, the actual results of the tests must be summarized for comparison to expected results, in order to demonstrate that the program is completely tested.

(3) Test data

Prior to the coding of each program, test data must be generated. The use of a standard naming convention for such test data is : <name>.tst for program <name>.C. The standard naming convention is not necessary when the program requires a name dependent input segment. For any desired portable software program, one needs to issue portable machine-independent test data.

(4) Compilation

Batch files should be used for compilation within development directories. These batch files should contain all routinely needed parameters for the compiler and librarian/linker invocations. Once the system is complete, a single batch file which performs a complete system installation from source code should be created and named "make.bat" emulating the UNIX standard.

SYSTEMS TESTING

The objective of systems testing is to find and correct any remaining performance and/or logic bugs that might exist in the system, in linkages between programs, in user procedures, data preparation, error detection and correction, and output distribution.

(1) The System Test Plan**- Responsibilities and Authorizations**

Each programmer is responsible for testing and documenting his program(s). Then, the project leader must check the documentation and conformance to standards. If satisfied, systems testing should be planned. (If it is a team project, the entire project team should review and approve it first.)

- Test Objectives**- Control Objectives**

- Error Types
- Recovery checkpoints
- Acceptance of files (file formats) and records
- Acceptance of input

- Processing Objectives

- Valid and invalid combinations of transactions
- Conditions
- Parameters

- Output

- Message lengths (Max. 65 chars)
- Message formats

- Schedule

At the beginning of a project, a precise schedule may not be possible. But after a project is tested by the programmer(s), a schedule should be fixed for systems testing. It should be planned at least two weeks in advance.

(2) Test File(s)

There are three possible sources of test data for systems testing

- Test data already prepared by programmers
- Live cases supplied by the user
- Contrived test data

The cases should be organized as a series of progressively more complex and comprehensive sets. Each set should be capable of being used separately and in combination with others, and combinations with exception conditions and error conditions.

(3) Linkage Tests

Tests for the interfaces between programs.

(4) Input/Output Tests

Tests for input generation procedures and output distribution procedures.

(5) User Acceptability

Final series of tests must be made by the user.

(6) Verification

All results should be compared with pre-prepared expected results. Discrepancies must be corrected.

(7) Authorization and Handover

When this point is reached, all (known) errors have been found and corrected, and test results documented, copies of the documentation must be delivered to the project manager.

1. Report No. <i>IN</i>		2. Government Accession No. <i>183583</i> 117401		3. Recipient's Catalog No.	
4. Title and Subtitle USL/NGT-19-010-900: USL/DBMS NASA/PC R&D PROJECT SYSTEM TESTING STANDARDS				5. Report Date <i>DATE</i> October 12, 1984 <i>OVERRIDE</i>	
7. Author(s) SRINU KAVI, DENNIS R. MOREAU, AND LIN YAN				6. Performing Organization Code	
9. Performing Organization Name and Address University of Southwestern Louisiana The Center for Advanced Computer Studies P.O. Box 44330 Lafayette, LA 70504-4330				8. Performing Organization Report No.	
12. Sponsoring Agency Name and Address				10. Work Unit No.	
				11. Contract or Grant No. NGT-19-010-900	
				13. Type of Report and Period Covered FINAL; 07/01/85 - 12/31/87	
				14. Sponsoring Agency Code	
15. Supplementary Notes					
16. Abstract					
<p>This Working Paper Series entry establishes a set of system testing standards to be used in the development of all "C" software within the NASA/PC R&D Project. Testing will be considered in two phases, namely, the program testing phase and the system testing phase. The objective of these standards is to provide guidelines for the planning and conduct of program and software system testing.</p> <p>This report represents one of the 72 attachment reports to the University of Southwestern Louisiana's Final Report on NASA Grant NGT-19-010-900. Accordingly, appropriate care should be taken in using this report out of the context of the full Final Report.</p>					
17. Key Words (Suggested by Author(s)) System Testing Standards, Software System Testing, PC-Based Research and Development			18. Distribution Statement		
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 5	22. Price*