Software Development Standards for Project Cost Management Information System

Development of a large-scale software project requires the deployment of software standards. Department of Defense and Military Standards currently in use for software projects were developed for life-cycle software development efforts. Some modification to these standards was necessary for use in a prototyping environment. This paper reports the standards adapted for and being used by the Project Cost Management Information System.
18. SUBJECT TERMS. (Continued).

Configuration management
Data dictionary
Data normalization
Fourth generation language
Prototyping
Quality assurance
Relational database management system
Software engineering
Software project development
Preface

This miscellaneous paper describes software development standards used for prototype development of a relational database management software project. The paper also provides information regarding the application of development standards to current development efforts. Funding for this paper was provided by the US Army Engineer Waterways Experiment Station (WES).

The Federal, Military, Army, and Department of Defense Standards cited were used to establish the necessary software development standards for the Project Cost Management Information System (PCMIS). Other references are listed following the main text.

The information in this paper was compiled at WES by Margaret B. Wright, Systems Modernization Unit (SMU), Computer Science Division (CSD), Information Technology Laboratory (ITL). Mr. Warren Bennett was Chief of SMU, Dr. Jerome Mahloch was Program Manager for PCMIS, Dr. Windell Ingram was Chief of CSD, and Dr. N. Radhakrishnan was Chief of ITL. The efforts of Bobbie Morrow, Information Management Division, who requested copies of the various Federal Standards, are gratefully acknowledged.

Commander and Director of WES was COL Larry B. Fulton, EN, and Technical Director was Dr. Robert W. Whalin.
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preface</td>
<td>1</td>
</tr>
<tr>
<td>Introduction</td>
<td>3</td>
</tr>
<tr>
<td>Background</td>
<td>3</td>
</tr>
<tr>
<td>Quality Assurance Standards</td>
<td>4</td>
</tr>
<tr>
<td>User Workstations</td>
<td>9</td>
</tr>
<tr>
<td>References</td>
<td>10</td>
</tr>
<tr>
<td>Appendix A: Glossary of Terms</td>
<td>A1</td>
</tr>
<tr>
<td>Appendix B: Examples of Project Cost Management</td>
<td></td>
</tr>
<tr>
<td>Information System Plan Forms</td>
<td>B1</td>
</tr>
</tbody>
</table>
SOFTWARE DEVELOPMENT STANDARDS FOR PROJECT COST
MANAGEMENT INFORMATION SYSTEM

Introduction

The US Army Engineer Waterways Experiment Station (WES) Corporate Data Base Project (CDB) is using software engineering methods and a structured approach for development of management information systems. These software engineering methods are implemented on the Project Cost Management Information System (PCMIS) as part of the WES CDB. The standards used are based not only on the structured approach but also on Department of Defense (DoD) and Military standards. Modifications were made to typical life-cycle standards so they apply to a fourth generation language (4GL)* based information system being developed in a prototyping environment.

Project standards encompass quality assurance, design analysis, development, documentation, testing, screen/report standards, and security. Configuration management (CM) standards are developed, but implementation has proven very difficult in a prototyping environment. CM implementation is planned after successful system testing. The use of standards provides consistency in reporting and is a valuable tool for the assurance of a quality product.

Background

3. The PCMIS project is being developed using a relational database management system (RDBMS). The final product is a series of user-interactive screens. These screens are engineered to provide program managers and principal investigators a timely means of managing their project funds and a user friendly interface to the Finance and Accounting portion of the Corps Engineers Management Information System (COEMIS). A prototyping approach is being used to develop and validate the screens.

4. PCMIS has enjoyed a high level of user involvement. This level of user involvement introduces a greater degree of success (based on user involvement) in developing a successful information system.

* For convenience, special terms used in this paper are identified in the Glossary of Terms presented in Appendix A.
satisfaction) and some interesting challenges in the implementation of the usual design and development standards. Communication of requirements and specifications from the users to the design team was facilitated through the use of formal meetings between the design team and members of the user committee. Design requirements were translated by the design team into an online data dictionary. The development team receives the requirements in a standardized format. This documentation is also useful for design and database verification, quality assurance, integration testing, and for future maintenance of the system. This automated data dictionary is the baseline for system development and maintenance.

Quality Assurance Standards

5. Quality assurance (QA) standards are primarily based on DoD Standard (TL) 2168 Software Quality Program. (Department of Defense 1988b). They are expanded in the project QA plan to include project-specific guidelines. Reports are identified and formats are established. Quality assurance is provided to ensure compliance to explicit functional and performance requirements, explicit documented development standards, and implicit characteristics that are expected of PCMIS. Quality assurance encompasses procedures for the effective application of methods and tools, formal technical reviews, testing strategies and techniques, procedures for change control, procedures for assuring compliance to standards, and measurement and reporting mechanisms. It is a planned, systematic pattern of actions to provide adequate confidence that PCMIS conforms to established technical requirements.

6. Quality assurance guidelines are based on the following standards:
   a. Defense System Software Quality Program
   c. Quality Assurance Terms and Definitions
   d. Defense System Software Development
   e. Software Product Standards

7. The following reports, taken from the online data dictionary, are applicable to QA for the purpose of determining and monitoring quality of
requirements analysis, systems interface, software development, and data base design:

a. All (entity) relationships.
b. Function definition.
c. Function decomposition.
d. Functions implemented by modules.
e. Modules which implement a function.
f. Module definition.
g. Module summary.
h. Modules and the tables/columns used.
i. Function-module cross reference.
j. Tables, columns, and indexes.
k. Tables/columns which are used in a module.

Requirements analysis/design standards

8. Project requirement analysis started with formal user requirement meetings where user requirements were communicated to project design analysts. PCMIS analysts used Yourdan-DeMarco Software Engineering techniques to represent project data flows.

9. Analysis and design project guidelines are based on standard entity relationship (E/R) methodology. A fully attributed E/R diagram is the established baseline for documentation in the database dictionary. The relational database design is taken to third normal form and then "custom-fit" to project specifications based on performance, efficiency, code complexity, and designer experience considerations. This process is called data normalization.

10. The following reports are used for analysis/design baseline:

a. Interview minutes.
b. Data flow diagram.
c. Work flow diagram.
d. Entity relationship diagram.
e. Functional hierarchy.
f. Requirements list.

Documentation standards

11. Department of Defense Automated Information Systems Documentation Standards DoD STD 7935.1 (1986) forms the basis for documentation standards for PCMIS. This standard specifically provides guidelines for the development (and revision) of information systems. The Data Base Specification section applies to DBMS.
12. Specific guidelines are further established in the introduction of the PCMIS User's Manual. Standard key definitions are also included here and apply only to the specific production environment. The user's manual establishes a standard format for screen documentation and presentation.

13. The online database dictionary provides much of the project documentation with the dictionary employing standard software engineering documentation methods. Additional documentation guidelines are discussed in other individual sections of this paper and are subject specific.

Testing standards

14. Testing guidelines are documented in the PCMIS software Test Plan and based on the following standards:

a. Defense System Software Quality Program


c. Defense System Software Development

d. Software Product Standards

These standards are modified to apply to prototyping. PCMIS testing guidelines cover the areas of field/screen testing, code design testing, user validation, and system test. The first three areas are at the unit/integration level of testing or lower. Acceptance testing is conducted by the users prior to production. Module (unit) testing is performed by the module programmer and includes the following:

a. Verification of all computations using nominal, singular, and extreme data values.

b. Verification of all data input options.

c. Verification of all data output options and formats, including error and information messages.

d. Exercise of all executable statements at least once.

e. Test of options at branch points.

System test includes a parallel test with the actual data and results from COEMIS fed to PCMIS. The resulting database will then be examined to prove the system.

15. Testing guidelines include: (a) recommendation of personnel requirements for the different levels of testing; (b) required facilities and hardware; and (c) required support software.
16. Several test reports and their formats are identified in the test plan as the standard for testing documentation. They include the following report forms (copies included in Appendix B).

a. Module screen signoff checklist (for programmers).
b. Software validation form (to report code problems).
c. Software test matrix (match requirements to test).
d. Test procedures (test description and data).
e. Software evaluation report (for test evaluation).

The software validation form is used to report any problems with code, code walk-throughs, break sessions, and user validation. A section for comments and correction signoff is included.

Screen/report standards

17. Screen standards include guidelines for screen and menu layouts and screen security. A screen layout is set forth to spell out: the program identifier, border characteristics, and general layout for all screens. Specific placement of key fields, such as date, screen title, and screen code, is indicated in the screen layout.

18. Menu screen standards include placement of menu options and instructions to be included for menu options. Standard menu options are numeric, although entry of a specific alphanumeric screen code allows a user to skip through the normal menu paths.

19. Entry and display screen standards set forth guidelines for consistency in (a) function key definitions, (b) error messages, and (c) help messages. Error messages and help messages that are common throughout are standardized.

20. Context sensitive help is provided for each screen with programmer defined messages specific to all allowable function keys. Automatic help messages for unused function keys are turned off to prevent confusion and to present a clearer help screen. Help messages for each enterable field are also available, in the form of an online help facility. Field help is not automatically displayed, in an effort to prevent redundancy and improve system performance. A predefined function key is used to access field help messages.

21. Hardcopy reports follow similar layout guidelines for key fields. Guidelines are included for titles, headings, footers, and page numbers. Reports are no more than 132 characters in width. Repeating fields that are subordinate to a key field are indented. For example, a report may show
several work units, and the subunits associated with each work unit are indented below it.

**Security standards**

22. Security guidelines are established for access limitation, online availability, data integrity, backup/recovery, data retention, and historical data management in PCMIS Data Base Control and Security Issues Plan. The security plan identifies system constraints and establishes baseline positions based upon user requirements.

23. Each user is assigned an individual username and password not echoed on the screen. This allows for unique identification of each person granted access. Unique user identification provides an audit trail and is used to perform access control. Classes of users are authorized to perform certain activities or to access certain data. All access for update, deletion, and commonly accessed queries are made through PCMIS menu. When a user logs on, his user class or type is determined and access privileges are established accordingly. Database queries only are allowed outside the PCMIS menu structure to allow managers flexibility in accessing their data.

24. Online availability addresses the hours that PCMIS will be available to users and further states that maintenance is performed after normal working hours. Guidelines for user notice are established in the event the system must be down during working hours.

25. Guidelines to ensure database integrity are established. Database integrity checks are developed and run to verify database accuracy. It is the database administrator’s responsibility to maintain database integrity in conformance to the E/R model. Database backup and recovery methods determine the backup interval and recommended procedures for recovery operations. Specific backup/recovery standards are tailored to the production environment. Other environment specific standards include reorganization of the database due to maintenance or performance requirements.

26. Data retention and archival of historical data are covered by Army Regulation AR 25-400-2 (Headquarters, Department of the Army, 1987). Regulations are used as required minimum and modified for upward reporting capability.
Database naming standards and conventions

27. The abbreviation standard adhered to for data element naming is AR 310-50 (Headquarters, Department of the Army, 1985). All applicable abbreviations included in the standard are used. Any words not included in the standard are constructed using the conventions described in the Headquarters, Department of the Army, 1 JUNE 1987 letter entitled "Data Element Naming Conventions." The data element naming conventions referred to in this letter apply to all Corps Information Systems Modernization Program projects.

Keyboard mapping standards

28. Keyboard mapping standards for PCMIS depends on the production machine. The development is taking place in an IBM block mode environment, therefore the keyboard mapping reflects this. Keyboard mapping varies depending on the communications protocol used. There is one set of keyboard symbols for developers using terminals with direct connections. A second set is used by members of the development team using a software emulation package and a hardware connection. A third set is used by people who dial into the system using a communications package such as Procomm.

29. Each software package maps the keyboard differently. During development, the standard being used is based on direct connection to the development machine. A decision concerning standard keyboard mapping must be made prior to production, after the production machine has been set up, and protocol software has been tested.

User Workstations

30. User workstations could be "dumb" terminals connected directly to the production machine, or PC's. Due to the large number of PC's currently in use on the WES campus, PCMIS is standardizing on PC's.

31. PC communications to the production machine will be via the WES network for the most part. VistaCom, a CDC product, is the communications/emulation package under consideration, because the production machine will be a CDC computer. One software communications package will be the standard established for all user interaction. This standard protocol will require specific user training, but will "travel" with the user regardless of the workstation type used to access the database.
References


Headquarters, Department of the Army. 1985 (Nov). "Authorized Abbreviations and Brevity Codes," Army Regulation 310-50, Washington, DC.


Appendix A: Glossary of Terms

1. **Baseline** is a specification or product that has been reviewed and agreed on and thereafter serves as the basis for further development. A baseline can be changed only through change control procedures (Evans 1987).

2. **Data dictionary** is: (a) a collection of the names of all data items used in a software system, together with relevant properties of those items, such as length and representation; (b) a set of definitions of the data flows, data elements, files, databases, and processes referred to in a data flow diagram (Evans 1987).

3. **Data flow diagram** is a graphic representation of a system, showing data characteristics, relationships, and logical flow of data as links between the software functional elements (Evans 1987).

4. **Data normalization** identifies redundant data that may exist in the logical data structure, determines unique keys needed to access data items, and helps to establish necessary relationships between data items. Three levels of normalization, called normal forms, can be achieved. Normalization is used to simplify the logical data structure (Pressman 1987).

5. **Prototyping** is the rapid development of a functional representation of a system capability that serves to provide a test bench on which system and user interface concepts can be tested prior to development (Evans 1987).

6. **4GL fourth generation language** is a higher level language that uses a nonprocedural approach to translate requirements to implementation. Fourth generation tools automatically generate source code based on the developer’s specification (Pressman 1987).
Appendix B: Examples of Project Cost Management
Information System Test Plan Forms

1. Module screen signoff checklist. The checklist (Figure B1) is completed by the programmer/analyst during verification of the coded modules. The list is then validated and verified for test readiness by the analyst in charge of testing, and the records are maintained for audit purposes.

2. Software validation form. The software validation form (Figure B2) is used to report code problems. It is used for code design walk-throughs, code break sessions, and user validation. The developer is responsible for correcting the deficiency and returning the form to be filed in the software validation notebook.

3. Software test matrix. The software test matrix (Figure B3) serves two purposes. When the appendix is delivered at the beginning of a test period, the matrix documents the modules to be tested and which of the tests from the test suite will be used to test functions implemented by each module. The matrix then will be used during the test to record the validation or exceptions to a test. A copy of the completed matrix will be included in the software evaluation report to verify any exceptions or corrections.

4. Test procedures. The software test procedure report (Figure B4) will be delivered as part of the Test Appendix. It will be prepared by the individual performing the test and used to record the immediate testing results. The beginning and ending test times will be noted.

5. Software evaluation report. The software evaluation report (Figure B5) will be completed by the tester and delivered to the test analyst no later than 3 days following a test period.
This checklist is filled out by the programmer/analyst during verification of each coded module. It is validated and verified for test readiness by the analyst in charge of testing. These work sheets are maintained for audit purposes.

SCREEN CODE: _______ DATE: _______
PROGRAMMER: _______ ANALYST: _______

TESTING STANDARDS

Software testing represents the ultimate review of specification, coding, and design. Test cases are constructed to zero in on areas where there is a high probability of finding errors. Field/Screen level testing will include:

All computations verified using nominal, singular, and extreme data values?

All data input options verified?

All data output options and formats, including error and information messages verified?

All executable statements exercised?

Options at branch points tested?

Entry of invalid data, escape sequences, and function keys?

Proper database storage verified where applicable?

SCREEN STANDARDS

Screen Standards have been developed for the PCMIS project. The programmer/developer is responsible for compliance with these standards.

Field identifiers clear?

Help messages clear? (use examples where possible)

Error messages clear?

All other screen standards including format satisfied?

Figure B1. Module/screen signoff checklist
I. **Purpose**

Use this form to record operational or functional defects observed in PCMIS software. This form may also be used to document suggestions for improvements or enhancements to PCMIS. Please observe the following guidelines:

- Always identify the screen/program module containing the defect or for which a suggestion is being made.
- Indicate if the defect/suggestion is related to the functional requirements being fulfilled by the screen/program module, the method in which the screen/program module operates, or is being used.
- Do not record multiple defects/suggestions on the same form. Use a new form for each defect observed or suggestion being made.

II. **Review Identification**

Name of Reviewer: ________________ Date: ________

III. **Software Identification**

Screen/Program Module Code: ______
Title: ________________________________________

IV. **Defect/Suggestion Classification**

Functional: ___ Operational: ___

V. **Description of Defect/Suggestion**

VI. **Person Assigned to for Resolution**

VII. **Resolution of Defect/Suggestion**

VIII. **Programmer Signature/Date Fixed**

Figure B2. Software validation form
<table>
<thead>
<tr>
<th>Tested Module</th>
<th>Functional Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Test Certification Signatures:

Quality Assurance:  
Programmer/Analyst:  

Code Review/Design:  
Programming Management:  

Comments:

Figure B3. Software test matrix
a. Test XXXXXXXXX.

b. Test input requirements: 

   Beginning Time:
   1. 
   2. 
   3. 
   4. 
   5. 
   6. 
   7. 
   8. 
   9. 
   10.

   Ending Time: ________

d. Test output analysis method:

e. Uniquely identified acceptance criteria:

Figure B4. Test procedures
Review Identification:

Project:
Date:
Location:
Time:

Product Identification:

Module:
Material Identification:
Producer:

Brief Description:

Code Test:

Code Test Evaluation Participants:

Problems or Defects Encountered with Recommended Action:

User Validation Test:

User Validation Test Participants:

Problems or Defects Encountered with Recommended Action:

Figure B5. Software evaluation report