\ascii
attach ,paper /UN=PW00336
ses. format paper

USERS GUIDE: SOFTWARE DEVELOPMENT METHODOLOGY (SDM)
For Advanced Systems Release 2

Author: Pete Warburton

Approved: _____

_____

_____

_____

DISCLAIMER:
This document is an internal working paper only. It is subject to change and does not necessarily represent any official intent on the part of CDC.

12.38.55.

CODE	LINE
I	1
S	2
	3
	4
	5
	6
	7
	8
	9
	10
	11
	12
	13
SI	14
SI	15
S	16
	17
	18
	19
	20
	21
	22
	23
	24
	25
	26
	27
SSSTU	28
	29
	30
	31
	32
	33
	34
SSTU	35
	36
SSTU	37
	38
SSTU	39
	40
SSTU	41
	42
SSI	43
I	44
	45
	46
	47
	48
	49
	50
	51
	52

REVISION DEFINITION SHEET

REV	DATE	DESCRIPTION
A	81/09/09	Partial draft for internal review.
B	81/09/21	Draft for internal review.

CODE	LINE
NI	1
S	2
SI	3
SSTTT	4
SI	5
SSTTT	6
STTT	7
STTT	8
STTT	9
STTT	10
STTT	11
STTT	12
STTT	13
STTT	14
STTT	15
STTT	16
STTT	17
STTT	18
STTT	19
STTT	20
STTT	21
STTT	22
STTT	23
STTT	24
STTT	25
STTT	26
STTT	27
STTT	28
STTT	29
STTT	30
STTT	31
STTT	32
STTT	33
STTT	34
STTT	35
STTT	36
STTT	37
STTT	38
STTT	39
STTT	40
STTT	41
STTT	42
STTT	43
STTT	44
STTT	45
STTT	46
STTT	47
STTT	48
STTT	49
	50
	51
	52

1.0 SDM OVERVIEW AND SUMMARY

1.0 SDM_OVERVIEW_AND_SUMMARY

SH1

1.1 INTRODUCTION

H2

The purpose of this SDM USERS GUIDE is to set forth the procedures, techniques, and tools to be used by SDD (Sunnyvale Development Division) personnel in developing Advanced Systems Release 2 software products.

I

The primary purpose of an SDM is to assure that a project will meet requirements on schedule and within budget, as agreed to between management and the project.

SI

The SDM proposed in this document is an evolutionary outgrowth of SDMs in use in Control Data since the early 1970s [Peterson 1973, Metzger 1973].

SI

This document, to the extent that it conflicts with Corporate Standard 1.01.106 "Software Development Model", constitutes a proposal to update the methodology of that Standard, insofar as that Standard is applicable to the development of Systems Software.

SI

While this document is concerned with current practices in SDD, it is more concerned with how current practices can be shaped into a coherent and systematic SDM.

1.2 WHAT_IS_SDM?

H2

SDM is the family of internal documents, techniques, and tools by which requirements become design and design becomes releasable code supported by published external documents.

I

For the Sunnyvale Development Division, requirements, design, implementation, evaluation, and publication activities are recorded in the following family of documents:

SI

S

a. Requirements documents (controlled via DCS) SSTTJ
   - AD/R (CY180 Architectural Requirements/Objectives) SSTTJ
   - SIS (System Interface Specification) STTJ
   - GDS (General Design Specification) STTJ

1.0 SDM OVERVIEW AND SUMMARY
1.2 WHAT IS SDM?

- DR (Design Requirements)	STTJ	1
- ERS (External Reference Specification)	STTJ	2
	S	3
b. Design documents (controlled via DCS)	SSTTJ	4
- GID (General Internal Design)	SSTTJ	5
	S	6
c. Implementation documents (code controlled by code transmittal and PSR procedures, documents via DCS)	SSTTJ	7
		8
- PL (Program Library of documented source code)	SSTTJ	9
- PSR corrective code	STTJ	10
- Baseline documentation changes	STTJ	11
- IMS (Internal Maintenance Specification)	STTJ	12
	S	13
d. Evaluation documents (PSTP controlled via DCS, others controlled by management procedures)	SSTTJ	14
		15
- PSTP (Product Set Test Plan, for each code Release)	SSTTJ	16
		17
- BER (Build Evaluation Report)	STTJ	18
- Approved Reviews (by Development and Evaluation) of Publications drafts of manuals.	STTJ	19
		20
	S	21
e. Publications (controlled by Publications procedures)	SSTTJ	22
- Reference Manuals	SSTTJ	23
- Operators Guides	STTJ	24
- Installation Handbook	STTJ	25
- Users Guides	STTJ	26
- "Instant" Reference booklets	STTJ	27
	S	28
While the generation of these documents is basically chronological due to logical dependencies inherent in the order given above, there is also an iterative process at work because as we learn more, we may have to revise previous documents. That is, requirements "drive" design and design "drives" code. However, refinement of design can lead to revision of requirements, and refinement of code can lead to revision of design (and sometimes, revision of requirements).	SSI	29 30 31 32 33 34 35 36 37 38
1.3 BASELINE DOCUMENTS, DAP'S, BSL'S, QSS'S, AND RSE'S	H2	39 40
Many of these documents are referred to as Baseline Documents, which are of two kinds: internal and external, subject to different sets of policies.	I	41 42 43 44
Internal Baseline documents are:	SI	45
- AD/R	SSTTJ	46
- SIS	STTJ	47
- GDS	STTJ	48

- DR STTJ 1
- ERS STTJ 2
- GID STTJ 3
- PSTP STTJ 4
- IMS STTJ 5
S 6

External Baseline documents are products of Publications: SI 7
- Reference manuals SSTTJ 8
- Operators guides STTJ 9
- Installation Handbook STTJ 10
S 11

DAPs are usually generated during the Analysis and Design activities, each DAP addressed to a particular issue. The author of a DAP should identify in the DAP the section(s) of baseline document(s) which will be modified if the DAP is approved. The content of an approved DAP should result in a BSL (baseline change) with change pages to an internal (possibly external) baseline document. S 12
13
14
15
16
17
18
19

QSSs (Quotation Special Software) and RSEs (Request Software Enhancement) which become features of standard software should be handled as are DAPs. A BSL with change pages for affected documents should be generated. SI 20
21
22
23
24
25

1.4 SOFTWARE DEVELOPMENT PHASES H2 26

27

Initial development phases are product/project oriented for a given version or release: SI 28
29
- Feasibility Phase SSTTJ 30
- Definition Phase STTJ 31
- Analysis Phase STTJ 32
- Design Phase STTJ 33
- Implementation Phase STTJ 34

These phases (plus the Feature Test Plan, which is product oriented) are covered in the Project Plan. SI 35
36
37

Concluding development phases are Product Set oriented toward a particular release: SI 38
39
- Evaluation Phase SSTTJ 40
- Publications Phase STTJ 41
- Release-activity Phase STTJ 42
S 43

In the past, maintenance has sometimes been considered a follow-on phase. However, for Advanced Systems, AD&C is directing that maintenance be handled in the same way that a new version would be: Go back to the Feasibility Phase and cycle again through all phases in an orderly manner. This SI 44
45
46
47
48

procedure should aid the preservation of structural integrity, which tends to erode over time [Belady 1979, VanHorn 1980].

a. Feasibility Phase — SSSTTJ
- Deliverable documents are: — SSTTJ
  - Project Plan, chapters 1 (Definition Phase Plan) and 7 (References) — SSTTJ
  - GDS (first version) or other documentation describing the product in general terms for PLM and Marketing approval — STTJ
- The Feasibility phase begins when Management initiates the preparation of a GDS or equivalent documentation for submission to PLM and Marketing. — SSTTJ
- The Feasibility Phase concludes when all deliverable documents are approved. — STTJ
- GDS (at least a first version) is preferable to an ad hoc document because a GDS will be produced later any way, based upon the ad hoc documents. However, conditions vary among projects, and ad hoc documents may be more suitable to particular circumstances than a GDS. — STTJ
- The purpose of the Feasibility phase is to determine that there is a need in the CDC product line for the proposed product, and to reach a general consensus upon the requirements for, and the architecture of, the proposed product. — STTJ

S

b. Definition Phase — SSTTJ
- Deliverable documents are: — SSTTJ
  - Project Plan, chapter 2 (Analysis Phase Plan) — SSTTJ
  - GDS (final version) — STTJ
- The Definition phase begins when management initiates the preparation of either deliverable document. — SSTTJ
- The definition phase concludes when all deliverable documents are approved. — STTJ
- The purpose of the Definition Phase is to define features, performance, and architecture of the product in sufficient detail to provide direction to the Analysis phase, during which all requirements will be explicated. — STTJ

S

c. Analysis Phase — SSTTJ
- Deliverable documents are: — SSTTJ
  - Project Plan, chapter 3 (Design Phase Plan) — SSTTJ
  - DR — STTJ

	CODE	LINE
- ERS	STTJ	1
- GID chapters 1, 2, 3, and 5 (Analysis Spec and Data Dictionary)	STTJ	2
		3
- The Analysis Phase begins when management initiates the preparation of one or more of the Analysis Phase deliverable documents, based upon evidence that the GDS is sufficiently stablized to provide direction for the Analysis Phase.	SSTTJ	4
		5
		6
		7
		8
- The Analysis Phase concludes when all deliverable documents are approved.	STTJ	9
		10
- The purpose of the Analysis phase is to make explicit all feature requirements, performance requirements, interface requirements, and to insure that the proposed product architecture supports all known requirements and all envisioned future features and future requirements.	STTJ	11
		12
		13
		14
		15
		16
	S	17
d. Design Phase	SSTTJ	18
- Deliverable documents are:	SSTTJ	19
- Project Plan, chapter 4 (Implementation Phase Plan) and chapter 5 (Feature Test Plan)	SSTTJ	20
		21
		22
- GID chapter 2 (Design Spec and revised Data Dictionary)	STTJ	23
		24
- Internal and external document BSLs required by Publications for manuals supporting releasable code.	STTJ	25
		26
		27
- The Design phase begins when management initiates the preparation of either deliverable document.	SSTTJ	28
		29
- The Design phase concludes when all deliverable documents have been approved.	STTJ	30
		31
- The purpose of the Design phase is to document explicitly the design of the product prior to coding.	STTJ	32
		33
		34
	S	35
e. Implementation Phase	SSTTJ	36
- Deliverable documents are:	SSTTJ	37
- Sourse code PL	SSTTJ	38
- IMS	STTJ	39
- Reviews of drafts of (external) baseline manuals	STTJ	40
		41
- The Implementation phase begins when management initiates it.	SSTTJ	42
		43
- The Implementation phase concludes when deliverables are approved.	STTJ	44
		45
- The purpose of the Implementation phase is to generate releasable code that meets all requirements and to provide I&E and Pubs with	STTJ	46
		47
		48

documentation supporting the successful completion of their tasks.		1 2
	SS	3
f. Evaluation Phase	SSSTTJ	4
- Deliverable documents are:	SSTTJ	5
- Project Plan chapter 5, Feature Test Plan	SSTTJ	6
- PSTP (Product Set Test Plan, formerly System Test Plan in SDD)	STTJ	7 8
- System Test Plan (ARPD)	STTJ	9
- Test base programs and data	STTJ	10
- BER (Build Evaluation Report)	STTJ	11
- Testing activities are of two kinds: preparing test plans and tests, and testing code.	SSTTJ	12 13
- Feature Test planning begins with the preparation of the Project Plan chapter 5 (Feature Test Plan), and continues with the generation of test code and data.	SSTTJ	14 15 16 17
- Product Set Test planning begins when management initiates the preparation of the PSTP (based upon all Feature Test Plans of all products of the set), and continues with the generation of test code and data.	STTJ	18 19 20 21 22
- System Test planning begins when management initiates the preparation of the System Test Plan.	STTJ	23 24 25
- Testing of code begins when management initiates the testing of transmitted PL or PSR code for a release.	STTJ	26 27 28
- Testing phase for a release concludes when management accepts the BER and approves code for release.	SSTTJ	29 30 31
- The purpose of the Feature Test Phase is to insure that the product code performs correctly according to functions specified in the requirements documentation and the publications.	STTJ	32 33 34 35
- The purpose of Product Set Testing (SDD) is to insure that the versions of products in the to-be-released set function together correctly.	STTJ	36 37 38
- The purpose of the System Test Release activity is to insure that all of the software of the Release operates together as a system and meets performance requirements.	STTJ	39 40 41 42
	S	43
g. Publications Phase	SSTTJ	44
- Required deliverable documents are:	SSTTJ	45
- Manuals Test Plan	SSTTJ	46
- Reference Manuals (or Release Revision packets)	STTJ	47 48

	CODE	LINE
- Operator Guides	STTJ	1
- Installation Handbook	STTJ	2
- Optional deliverable documents are:	SSTTJ	3
- Users Guides	SSTTJ	4
- Instant Reference booklets	STTJ	5
- The Publications phase for a release begins when management initiates preparation of (or revision of) a document, following receipt from Development or Design of supporting documentation (e.g., an ERS) to warrent publications activity and providing resources are available.	SSTTJ	6 7 8 9 10 11
- The Publications phase for a code release ends with submission of manual originals to Corporate Printing.	STTJ	12 13 14
- The purpose of the Publications Release activity is to support released code with external user manuals.	STTJ	15 16 17
	S	18
h. Release-activity Phase	SSTTJ	19
- Deliverables	SSTTJ	20
- PLs available from SMD (Software Manufacturing Division)	SSTTJ	21 22
- External Publications manuals are available from LDS (Literature Distribution Service)	STTJ	23 24
- All Release Bulletins are available:	STTJ	25
SAB (Software Availability Bulletin)	SIJ	26
SRB (System Release Bulletin)	SIJ	27
FAM (Feature Abstract Memorandum)	SIJ	28
- The Release Phase begins when management initiates steps to move the deliverables from Development, Evaluation, and Publications to the organizations that distribute the deliverables to customers.	SSTTJ	29 30 31 32
- The Release Phase concludes when release materials are delivered to customers.	STTJ	33 34
- The purpose of the Release Phase is to insure that customers receive timely and coordinated service in connection with new releases.	STTJ	35 36 37 38 39 40 41 42 43 44 45 46 47 48

2.0 DEVELOPMENT PHASES

2.0 DEVELOPMENT PHASES

SSH1

2.1 FEASIBILITY PHASE

H2

The purpose of the Feasibility Phase is to explore the feasibility of a proposed product or product enhancement from the joint viewpoint of Marketing, PLM, and Development. "Feasibility" here means "market feasibility" (is there a profitable market for the proposed product?) rather than "engineering feasibility" (can the product be built to specifications?), which is explored in the Definition Phase.

I

For some products, such as those for which there is agreement to meet an existing ANSI standard, the Feasibility and Definition phases are relatively brief. For other products, such as Data Management and Networks, there may be much effort required to define a product well enough to provide design direction for the preparation of Analysis Phase documents (DR, ERS, etc.). This pre-Analysis activity may not divide cleanly between Feasibility and Definition, but generally Development activity on a GDS sufficiently detailed to win approval cannot begin until PLM and Marketing have established the market feasibility for the proposed product or proposed product enhancement.

SI

If the proposal is deemed feasible, then Development deliverable documents of the phase are:

SI

- Project Plan chapters 1 and 7.

SSTTJ

- GDS (initial version) or other documentation describing the product in general terms, for Marketing and PLM approvals.

STTJ

S

The intent of the documents is to provide direction to the Definition Phase.

SI

The primary activity of the feasibility phase will probably be an exchange of memos among interested parties concerning features, architecture, performance, and interfaces to other products which constitute a goal for a feasible product. To be of permanent value, the outcome of this exchange of memos should be recorded in the GDS or other documents to be approved by PLM, Marketing, and AHPD and/or SDD.

SI

2.2 DEFINITION PHASE

Development deliverables of the definition Phase are:
- Project Plan chapter 2 (Analysis Phase)
- GDS (final version)

The purpose of the Definition Phase is to firm up the decisions of the Feasibility Phase into a coherent set of requirments (features, performance, architecture, interfaces to other products) in a approved GDS. The object (or goal) is to provide a definition of the product and to provide design direction for the Analysis Phase. Prior to the beginning of the Definition phase, design direction is not firm enough to result in an approved GDS, for PLM and Marketing are still determining the market feasibility of the proposed product. There may also be budgetary considerations that restrict the resources available to prepare a GDS, and these considerations may also delay the transition from the Feasibility Phase to the Definition Phase.

Reugirements analysis is one of the most difficult of all software development activities [Boehm 1979].

Requirements analysis is an art, not a science, which seems to use the following sort of dialectical process:

1. The designer or design team, on the basis of the best and most complete information available, proposes to the customer(s) a design thought to meet all requirements in an optimum fashion.
2. The customer says the design will not do because..., and another requirement which the designer was unaware of (and possibly the customer too unaware of before thinking about it) crawls out of the woodwork.
3. The designer reworks the design, possibly from scratch, but more likely by patching it, and goes back to step 1.
4. The customer says that will not due because..., and back to step 2.

.. and the process iterates on and on.

If the designer is lucky, the process terminates in a coherent set of requirements (features, performance, architecture, interfaces to other products).

However, every requirement has a price and if the price is too high (low priority item conflicts with a high priority item, architectural structure is compromised, implementation cost is too much, etc.), the "requirement" ceases to be a requirement, no matter how tenaciously held theretofore.

2.3 ANALYSIS PHASE

2.3.1 INTRODUCTION

The purpose of the Analysis Phase is to finalize requirements and to carry design far enough to insure there are no design problems in meeting the requirements spelled out in the DR, the ERS, and the Analysis Spec portion of the GID.

Development deliverables are:
- Project Plan chapter 3
- DR
- ERS
- GID chapters 1, 2, 3, and 5 (Analysis Spec)

If the software research literature is correct in claiming that a requirements bug caught after delivery of a product to a customer costs 270 times as much to fix as a coding bug and that a design bug costs 90 times as much to fix [McCabe 1980], then Control Data should be able to save many maintenance dollars by doing a better job of generating and reviewing requirements and design documents.

SASD (Structured Analysis/Structured Design, [DeMarco 1978, Yourdon 1978]) emphasizes the difference between data flow analysis (a definition and requirements function) and structured design (a design function).

Experience with SASD during development of Advanced Systems Release 1 products resulted in very few products doing both data flow and structure charts. Projects converting from CY170 had worked out their requirements during CY170 development and had little need of data flow analysis. Structure charts, on the other hand, turned out to be useful for documenting design, though some projects found other techniques, such as state tables, of more value.

For Release 2, there are several techniques available, each with advantages and disadvantages relative to the differing needs of various projects.

Beyond Release 2, it may be possible to use a specification language (such as BSL [Barber 1981]) for analysis/design.

2.3.2 SA (STRUCTURED ANALYSIS) H3

Structured Analysis (as defined and described by DeMarco) offers useful techniques of decomposition, data transformations, and data dictionary. I

Decomposition is the technique of summarizing an entire program in a one-page context chart (to show data flow interfaces to other programs) and a one-page level 0 DFD, and then decomposing each process in the level 0 DFD into level 1 DFDs, and so on down to as many levels as are necessary to define each bottom-level process in structured English. SI

Data transformation is technique of showing (with decomposition of data: files into records, records into segments, segments or tables into data elements) how output data is derived (directly or indirectly) from intermediate files or tables and input data, and how intermediate files and tables are updated from input data. SI

A data dictionary defines all data elements and data aggregations. SI

Structured Analysis can be of great help to a project in defining the functions to be performed and insuring that interface requirements of users and other products are understood by the project and can be implemented by the project. SI

SA is supported by computer tools. The Data Dictionary [DCS ID=ARH3980] supports data descriptions and process descriptions. SASD Graphics [DCS ID=ARH3981] supports Data Flow diagrams. SI

2.3.3 IA (INFORMATION ANALYSIS) H3

Information Analysis offers the capability of defining data and the forms of data permissible during data transformations. It does not offer decomposition techniques, though IA can be used to define data at any level from the most abstract to the most detailed. Nor does IA offer the capability to define data transformations (i.e., the algorithm by which an output or file item is constructed from input or other file items). I

Information Analysis may be useful for projects whose main task is to describe a data base (e.g., the IADT project, the SI

2.0 DEVELOPMENT PHASES
2.3.3 IA (INFORMATION ANALYSIS)

SASD database to support Graphics and Data Dictionary, the Corporate Traffic project).

IA is not supported by computer tools. IAF and ADAM are available for implementation, but are rather complex to use as Analysis tools. S

2.3.4 STATE TABLES H3

State tables are a useful tool for complex programs where the reaction to a given input is a function of the internal state of the program. State tables have been used by Networks (to define protocol-driven programs) and Fortran/VS (to define symbol table processing). State tables can be very helpful in uncovering error cases, end cases, and infrequent cases that may be overlooked in the course of design, because the technique forces a look at all possible inputs for all possible states. I

While there is no computer tool specifically supporting state tables, the Graphics structure chart capability can be used. SI

2.3.5 DECISION TABLES H3

Decision tables can be useful, for the same reasons that state tables can be. Essentially, a decision table is appropriate for a program that has only one state for a given set of inputs. For these cases, all data input/output cases can be defined. I

In the computer industry, there are COBOL-related decision table tools, but none seems widely used in Control Data. SI

2.3.6 STRUCTURED TESTING H3

Structured Testing [McCabe 1980] offers several techniques for checking requirement specifications: I

- Cause and Effect graphs (pp II-11, II-12): For each cause mentioned in the ERS or Analysis Spec, there should be one or more causes; for each effect there should be one or more causes; and these should be coherent (specified by non-conflicting and/or conditions). SSTTJ
- Specification reviews (pp II-18 thru II-26) to insure specifications are complete and coherent. STTJ

2.3.7 DATA FLOW ANALYSIS VERSUS STRUCTURE CHART ANALYSIS SH3

It seems to be a matter of individual temperament that some programmers prefer data flow analysis while others prefer structure design. Few programmers seem temperamentally equipped to view both as equally useful. This difference seems to have roots in a preferred position either that control flow is the logical consequence of data flow, or that data flow ought to be the logical consequence of control flow (i.e., which has logical precedence: data flow or control flow? which is the boss, from a requirements point of view?). I

The challenge of the Analysis Phase is to make sure that data flow requirements are understood prior to detailed design, otherwise the detailed design may not be able to support the requirements of the program. Hence DeMarco's plea to set aside design until data flow has been analysed to the point where those specifying requirements have agreed that the proposed specifications meet the requirements. SI

The crucial point is that the DR and ERS not be subject to modifications during the design phase, due to either management or the project having overlooked or misunderstood requirements. SI

2.4 DESIGN PHASE H2

The purpose of the Design Phase is to complete design prior to Implementation (coding and unit test). I

Development deliverables are: SI
- Project Plan chapter 4 SSTTJ
- GID (final version) STTJ
- BSLs for internal and external baseline documents STTJ

S

Evaluation deliverable: Project Plan chapter 5 SI

Publications deliverable: Manuals Test Plan SI

SD (Structured Design) is the principal methodology of design, as spelled out by Yourdon and Constantine. SI

SD is supported by the SASD Graphics for SCTs (structure charts) and the SASD Data Dictionary for module descriptions. Module descriptions should be detailed enough so that there is no ambiguity or open question encountered by the programmer SI

who translates the module description into code meeting CYBIL coding standards. This does not necessarily mean that the module descriptions are so detailed that each structured English statement is the equivalent of one or a few lines of code.

Structured Testing [McCabe 1980] provides guidelines for SI
reviewing design documents (pp IV-20 thru IV-36).

It is recommended that each project prepare a Project Notebook SI
setting forth procedures that all project members are expected to adhere to (e.g., "NOS/VE Project Procedures and Conventions").

2.5 IMPLEMENTATION PHASE H2

The purpose of the Implementation phase is to generate code I
which has been reviewed and unit-tested (Development), to generate test programs and data (Evaluation), and to generate drafts of external manuals (Publications).

Development deliverables are: SI
- Source Code PL SSTTJ
- IMS STTJ

Evaluation deliverables are: Test programs and data SSI

Publications deliverables are: Drafts of external manuals SI

Coding and code reviews wil be done in accordance with SI
SDD/ARPH coding staandards and procedures.

The project should insure that the procedures of the Project SI
Notebook are adhered to (or revise the procedures so that they are adhered to).

2.6 EVALUATION PHASE H2

Historically, the function of Software Evaluation has been to I
detect errors before a customer did, so Software Development could correct bugs before the software was submitted to an acceptance test or installed at a user's site [Metzger 1973].

Within the perspective of SDM, the function is somewhat SI
different.

2.0 DEVELOPMENT PHASES
2.6 EVALUATION PHASE

	CODE	LINE
While some persons look to proper design to result in bugless code and a program that never had any bugs is a better program than one in which the bugs have been fixed [Mills 1976], others believe that the proper function of Evaluation is to pinpoint the origin of errors in the development process so as to debug the development process [Deming 1981].	S	1
		2
		3
		4
		5
		6
		7
"During July 1981, Dr W Deming, the man whose ideas inspired the revolution in quality in Japanese industry conducted a four-day seminar for Control Data. He said:	SI	8
		9
		10
- 85% of product defects arise from the process that produces the product, not from the workers who immplement the process.	SSTTJ	11
		12
		13
- Everyone is already doing his "best". If you want fewer defects, you have to find a better process.	STTJ	14
		15
- If you reach your current level of defects through test and rework, you can find a process that	STTJ	16
		17
-- achieves the same level of defects directly, without test and rework and	SSTTJ	18
		19
-- is more profitable than your current process.	STTJ	20
- If you search for it, you can eventually find a process that	SSTTJ	21
		22
-- produces no defects	SSTTJ	23
-- is more profitable than your current process.	STTJ	24
- The best use of your testing process is to determine the capability of your process (its inherent defect level) so that you can improve it." [Huntwork 1981, page 5.2.1]	SSTTJ	25
		26
		27
		28
If these remarks are to be taken seriously, then for Release 2 the various test plans should address how Evaluation will determine which part of the development process is contributing to each error encountered. In the literature of Software Engineering, these problems are discussed in [Boehm 1975] and [Boehm 1976], among other places.	SSI	29
		30
		31
		32
		33
		34
		35
Test plans are:	SI	36
- Project Plan chapter 5 Feature Test	SSTTJ	37
- PSTP (Product Set Test Plan), SDD	STTJ	38
- System Test Plan, AHPD	STTJ	39
	S	40
Sources of errors to be identified include:	SI	41
- Requirements activity	SSTTJ	42
- Design activity	STTJ	43
- Implementation activity	STTJ	44
- Publications activity	STTJ	45
- Evaluation activity	STTJ	46
	S	47
Within each of these activities, a possible source of error	SI	48

might be:
- omission or oversight — SSTTJ
- misunderstanding — STTJ
- poor documentation in a baseline document — STTJ
- poor documentation or documentation in an inappropriate document — STTJ
- "fell between the cracks" and some aspect of SDM is deficient — STTJ

2.7 PUBLICATIONS PHASE — SH2

The Publications and Graphics Division has procedures for generating external baseline manuals and other manuals for the planned code release. — I

Development management and Publications management work together to establish a schedule such that both groups can meet their commitments for release. — SI

Major items of the interface between Development and Pubs have been mentioned in the phases above: — SI

- Internal baseline documents must arrive in Pubs on schedule inorder that Pubs prepare draft manuals on schedule. — SSTTJ
- BSLs to external baseline documents must arrive in Pubs on schedule inorder that Pubs prepare draft manuals on schedule. — STTJ
- Pubs drafts of manuals must arrive in Development on schedule inorder that Development and Evaluation can get reviewed drafts back to Pubs in time for Pubs to make changes and still meet the Release schedule. — STTJ

S

The key document in providing this schedule is the Pubs "Manual Test Plan" prepared by Pubs during the Design Phase, with appropriate input from Development management. — SI

2.8 RELEASE-ACTIVITY PHASE — H2

Timely release of materials to customers entails much coordination among Development, Evaluation, Publications, Software Manufacturing, and Literature Distribution. — I

The procedures for these activities are spelled out in the SDD Mini-procedures Handbook. — SI

3.0 DOCUMENTS

3.0 DOCUMENTS

SH1

For each document, a brief description is given, followed by a table of contents. Where appropriate, these skeletal tables of contents are based on CDC Standard 1.01.100 "Programming Project Management Standards".

SI

NOTE for any document containing a glossary: The ANSI Dictionary for Information Processing (ANSI X3/TR-1-77) defines technical terms not defined in the glossary of the document.

SI

3.1 PROJECT_PLAN

H2

Purpose: To describe an activity in terms of how it is to be done, when it will be done, what the cost will be, what other projects are constrained, and what are constraining projects.

SSTU

The Project Plan is a management document rather than a technical document. It should include a minimum of technical detail about the product.

SS

The Project Plan is included in this USERS GUIDE in order to:

SS

- Standardize the format among SDD projects

SSTTJ

- Indicate the sequence in which chapters should be written, and indicate the chronological relationship of Product Plan chapters with other documents

STTJ

S

Content: The project plan is the controlling project document and contains several parts. These include (all may not be required for a given project):

SSTU

Chapter 1--Definition Phase Plan SSTU
Chapter 2--Analysis Phase Plan STU
Chapter 3--Design Phase Plan STU
Chapter 4--Implementation Phase Plan STU
Chapter 5--Feature Test Plan STU
Chapter 6--Post Mortem STU
Chapter 7--References STU

3.0 DOCUMENTS
3.1 PROJECT PLAN

		CODE	LINE
Audience:	Managers	SSTU	1
	Planners	S	2
	Interfacing projects	S	3
	System test	S	4
	Development project	S	5
	Quality Assurance	S	6
			7
Owner:	SDD Management	SSTU	8
			9
Author:	Development Project/Product Design (chapters 1,2,3,7)	SSTU	10
			11
	Development Project (Chapter 4)	STU	12
	Evaluation (Chapter 5)	STU	13
	Development Project/Product Design/Evaluation (Chapter 6)	STU	14
			15
			16
Comments:	All the planning documents, as well as the post mortem, are included in this one plan. This makes the project plan more complete and meaningful. Since it is organized into chapters, the audience can go directly to the part that is of interest. Most of the chapters above are based on a document that used to be stand-alone. Due to the fact that these were stand-alone, a great deal of redundancy was noted. Collapsing the documents into one eliminates this problem.	SSTU	17 18 19 20 21 22 23 24 25 26
			27
	The Definition Plan describes objectives, deliverables, and schedules for the definition phase. The Analysis Plan does likewise for the Analysis phase. The Design Plan consists of objectives, milestones, and resources needed for the design activity. The Implementation Plan contains similar types of information, plus constraints, risks, unit testing plans or direction, and System Integrated Test (SIT) plans, if required. Descriptions of individual unit tests in the form of a matrix or a list will be produced by the project and/or the design team. These details need not be part of the IPP. The Feature Test Plan describes the activities to be performed by Evaluation to verify functional capabilities of a given product or feature, as well as activities required to verify the product performance requirements as specified in the AD/R and the DR. The Feature Test Plan also lists resource requirements, constraints, risks, and testing milestones. Plans for performing System Integrated	SS	28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48

3.0 DOCUMENTS
3.1 PROJECT PLAN

Test (SIT) cycles should also be included, if appropriate. SIT plans should be in response to the SIT plans outlined in Chapter 4 of the project plan. As with the IPP, specific test descriptions and/or a test matrix are provided by the evaulation project or by the design team as a separate working document; these details need not be part of the FTP. The post mortem is an informal document that describes what went right with the project, what went wrong with the project, and what could have been done to rectify bad situations in the project.

Each chapter of the project plan can be considered either as a stand-alone document or as a part of the whole. Chapters are completed and distributed at different points in time, and, in the case of the Feature Test Plan, are authored by different people. Note that information is not repeated in each of the chapters. For example, for each chapter that contains milestones, the choice of milestones should be only those needed by people other than the author and the author's manager, for example, interdependency milestones. In chapters 1, 2, and 3 only start and complete dates may be required. Intermediate milestones are not of general interest and quickly become obsoleted by the PERT. SS

Table of Contents: SST

1.0 Definition Plan SSSTTJ
1.1 Introduction STTJ
Introduction to and summary of chapter 1. Relevant documents can be listed here or in chapter 7. Can contain a short technical description of the product, especially if the GDS does not yet exist. SIJ
1.2 Deliverables STTJ
Project Plan chapter 2 (Analysis Plan), GDS, and any other deliverables. SIJ
1.3 Milestones STTJ
Dates for start, DCS submittal, and approval of each deliverable document. SIJ
1.4 Resources and Schedule STTJ
Identify person/months of effort for each calender month for each deliverable. SIJ
Identify any other resources need for the

3.0 DOCUMENTS
3.1 PROJECT PLAN

```
          phase.                                                          1
1.5       Constraints                                          STTJ       2
          Identify any constraints upon schedule and           SIJ        3
          resources.  (These constraints apply to the                     4
          phase resources and schedule, not to the                        5
          product.)                                                       6
                                                                          7
2.0       Analysis Phase Plan                                  SSTTJ      8
2.1       Introduction                                         STTJ       9
          Introduction to and summary of chapter 2.            SIJ        10
2.2       Deliverables                                         STTJ       11
          Project Plan chapter 3 (Design Plan), DR,            SIJ        12
          ERS, GID chapter 1 (Analysis Spec and Data                      13
          Dictionary), and any other deliverables.                        14
2.3       Milestones                                           STTJ       15
          Dates for start, DCS submittal, and approval         SIJ        16
          of each deliverable document.                                   17
2.4       Resources and Schedule                               STTJ       18
          Identify person/months of effort for each            SIJ        19
          calender month for each deliverable.                            20
          Identify any other resources need for the                       21
          phase.                                                          22
2.5       Constraints                                          STTJ       23
          Identify any constraints upon schedule and           SIJ        24
          resources.  (These constraints apply to the                     25
          phase resources and schedule, not to the                        26
          product.)                                                       27
                                                               STTSKI     28
3.0       Design Phase Plan                                    PTJSTTJ    29
3.1       Introduction                                         STTJ       30
          Give an abstract of chapter 3.                       SIJ        31
3.2       Deliverables                                         STTJ       32
          List what (documentation including GID               SIJ        33
          chapter 2, etc.) will be produced as a                          34
          result of the Design phase.                                     35
3.3       Objectives                                           STTJ       36
          State all the major goals that are to be             SIJ        37
          accomplished during the design phase.  State                    38
          considerations that will affect design, such                    39
          as SIS, implementation language, etc.                           40
3.4       Methods                                              STTJ       41
          State how (in procedural terms) the design           SIJ        42
          will be done.                                                   43
3.5       Constraints.                                         STTJ       44
3.6       Resources                                            STTJ       45
3.7       Milestones.                                          STTJ       46
          List milestones for the design phase only.           SIJ        47
                                                                          48
```

3.0 DOCUMENTS
3.1 PROJECT PLAN

		CODE	LINE
4.0	Implementation Phase Plan	SSTTJ	1
4.1	Introduction	STTJ	2
	Give an abstract of chapter 4.	SIJ	3
4.2	Deliverables	STTJ	4
	State what will be delivered by this phase, such as software, documentation, etc.	SIJ	5 6
4.3	Objective	STTJ	7
	State what is to be accomplished during the implementation phase.	SIJ	8 9
4.4	Overview	STTJ	10
	Give a brief product description (refer details to documents which contain all the details, including chapter 1.). State assumptions, list a glossary if needed.	SIJ	11 12 13 14
4.5	Schedule (Or phase plan for implementation)	STTJ	15
	Discuss the methodolgy of implementation, and what will occur during successive phases of the implementation. Discuss SIT plans.	SIJ	16 17 18
4.6	Unit testing.	STTJ	19
	Describe how unit testing will take place during the different phases of implementation, and whether unit tests will be salvageable as candidates for a system test base.	SIJ	20 21 22 23 24
4.7	Contingencies, dependencies, risks.	STTJ	25
4.8	Resources	STTJ	26
	State resource requirements - number of people needed (at different phases if possible), machine time, and anything else that affects implementation progress.	SIJ	27 28 29 30
4.9	Milestones	STTJ	31
	State milestones only for the implementation plan.	SIJ	32 33 34
5.0	Feature Test Plan	SSTTJ	35
5.1	Introduction	STTJ	36
	Give an abstract of chapter 5.	SIJ	37
5.2	Product features to be tested.	STTJ	38
5.3	Testing Methodology	STTJ	39
5.3.1	Testing Approach	STTJ	40
	Discuss the approach to be used to select which features are to be tested and which (if any) are not to be tested, and identify which are to be tested and which are not to be tested.	SIJ	41 42 43 44 45
5.3.2	Features of the product not testable.	STTJ	46
5.3.3	Documentation	STTJ	47
5.3.4	Test Base code reviews	STTJ	48

3.0 DOCUMENTS
3.1 PROJECT PLAN

		CODE	LINE
5.3.5	SIT plans (if any)	STTJ	1
5.4	Performance Testing	STTJ	2
	Discuss what testing will be done to verify DR specifications.	SIJ	3 4
5.5	Constraints, risks, dependencies (people, machine configurations, tools)	STTJ	5 6
5.6	Resource requirements	STTJ	7
5.7	Deliverables	STTJ	8
	Testing software and documentation to be delivered as a result of this plan.	SIJ	9 10
5.8	Transmittal Criteria	STTJ	11
	This is transmittal criteria for the product to be tested. Include final release criteria for the product also (from DR).	SIJ	12 13 14
5.9	Milestones	STTJ	15
	List milestones for the entire test base creation and testing of the product. Include milestones for test base availability and transmittal.	SIJ	16 17 18 19
		S	20
Appendix-A	Feature versus test matrix.	SSTTJ	21
A.1	Test Base Content	STTJ	22
A.1.1	Current tests	STTJ	23
	Discuss the size and content of the test base, without going over each test in detail. Cover such things as: how large the test base is, perpetuation of tests from the old test base, on what medium and in what form the test base is, a general categorization of the tests in the test base, and the test case naming convention. A blow by blow account of each test can be given in the test base matrix.	SIJ	24 25 26 27 28 29 30 31 32 33 34 35 36
A.1.2	Modifications and conversions	STTJ	37
	Discuss classes of tests that will be modified, dependent on development, schedules, and other criteria.	SIJ	38 39 40 41
A.1.3	Enhancements to existing tests	STTJ	42
	Features that need to be covered but are not by the current test base.	SIJ	43 44 45
A.2	Feature versus test matrix	STTJ	46
	List tests by name and feature(s) tested by each test.	SIJ	47 48

3.0 DOCUMENTS
3.1 PROJECT PLAN

6.0	Post Mortem	SSSTTJ	1
	This chapter is the result of meetings with	SIJ	2
	development, integration, evaluation,		3
	product design, and publications personnel		4
	who were involved with the project. Topics		5
	to be covered should include:		6
	- Analysis Phase/Design Phase of the	SSTTJ	7
	project		8
	- Implementation Phase	STTJ	9
	Strategy	SIJ	10
	Positive Aspects	SIJ	11
	Negative Aspects	SIJ	12
	Code Reviews	SIJ	13
	Staffing/Machine Usage	SIJ	14
	Schedules	SIJ	15
	Interactive Usage	SIJ	16
	Tools	SIJ	17
	Special Factors	SIJ	18
	Release Mechanics	SIJ	19
	- I&E	STTJ	20
	Test Strategy	SIJ	21
	Other test topics	SIJ	22
	Special Feature Testing	SIJ	23
	Installation Decks	SIJ	24
	- Publications	STTJ	25
	- Conclusions/Suggestions/Actions	STTJ	26
	- Total Project Cost Data	STTJ	27
		S	28
7.0	References	SSTTJ	29
	List all documents, memos, etc. relevent to	SIJ	30
	each and all chapters of the Project Plan.		31
			32
			33

3.2 AQ/R (ARCHITECTURAL OBJECTIVES AND REQUIREMENTS) H2 34

35

Purpose:	To specify high-level requirements on a system-wide as well as product-wide scale; to be used as input to the DR and the test plans.	SSTU	36 37 38 39
Content:	Describes System in general terms, describes major functional elements and characteristics of the system in specific terms, and furnishes detailed specifics of the sytem definition.	SSTU	40 41 42 43 44
Audience:	Managers	SSTU	45
	Product Design	STU	46
	Development projects	STU	47
	Corporate reviewers	STU	48

	Evaluation	STU	1
	Publications	STU	2
			3
Owner:	A D & C	SSTU	4
			5
Author:	A D & C	SSTU	6
			7
Table of Contents: (See AD/R, A1688)		SST	8
			9
			10
3.3 SIS		H2	11
			12
Purpose:	To insure a uniform interface across the operating system and the product set.	STU	13 14
			15
Content:	Covers product-to-product, product-to-user, system-to-user, and product-to-operating system interfaces.	SSTU	16 17 18
			19
Audience:	Managers	SSTU	20
	Product Design	STU	21
	Development projects	STU	22
	Corporate reviewers	STU	23
	Evaluation	STU	24
	Publications	STU	25
			26
Owner:	Product Design/Advanced Systems Design	SSTU	27
			28
Author:	Product Design	SSTU	29
			30
Enforcer:	A D & C	SSTU	31
			32
Table of Contents: (See SIS, S2196)		SST	33
			34
			35
3.4 GDS		H2	36
			37
Purpose:	To document prioritized objectives and design direction, for a given product that should be met but are not official commitments. This document should address multiple releases of a product, i.e. the product's life cycle.	SSTU	38 39 40 41 42
			43
Contents:	The GDS encompasses design direction, performance prediction, and test direction as currently found in three separate documents. The GDS serves as input to the feature test plan, the analysis and design specifications, the ERS, and the performance	SSTU	44 45 46 47 48

test plan. 1
2
Audience: Development projects SSTU 3
Product Design STU 4
System test STU 5
Publications STU 6
7
Owner: Product Design/Advanced Systems Design SSTU 8
9
Author: Product Design/Advanced Systems Design SSTU 10
11
Table of Contents: SST 12
1.0 Introduction SSTU 13
Give an abstract describing the product and this document. STU 14
15
2.0 Glossary STU 16
3.0 Major Product Interfaces STU 17
Discuss the external and internal interfaces of the product. STU 18
19
4.0 Major Product Features STU 20
Discuss the major product features. STU 21
5.0 Standards STU 22
Discuss standards, such as ANSI, SIS, AD/R, which affect this product. STU 23
24
6.0 Publications STU 25
7.0 Performance Considerations STU 26
Discuss the primary performance objectives with regard to the design of this product. STU 27
28
8.0 Compatibility STU 29
Discuss compatibility across predecessor and possible successor products, and with elements or concepts of the overall system (such as system control language compatibility). STU 30
31
32
33
34
9.0 Migration STU 35
Discuss migration/conversion impact, and what means will be available to ease conversion/migration from predecessor to this product. STU 36
37
38
39
10.0 Test Direction STU 40
Discuss general testing strategy. STU 41
42
43

3.5 QR H2 44
45
Purpose: To document the commitment by the division (product design, development, system test and publications) to produce software products that meet stated SSTU 46
47
48

requirements.

Contents: See Corporate Standard 10:01:03:011 SSTU

Audience: Corporate reviewers SSTU
Managers STU
Development projects STU
System test STU
Publications STU
Product Management STU

Owner: PLM (Product Line Management) SSTU

Author: Product Design/Advanced Systems Design SSTU

Table of Contents: SST

There is no flexibility in the generation of this document. All sections listed below must be present in the DR, even if "not applicable". PLM has stated that they will not review a DR that does not conform to the CDC Corporate Standard for DRs. SSTU

1.0 Definition SSSTU
2.0 References STU
2.1 Interdependent Documents STU
2.2 Technical References STU
2.3 Standards STU
2.3.1 Control Data Standards STU
2.3.2 National, International and Industry Standards STU
3.0 Requirements STU
3.1 Functional Requirements STU
3.1.1 Functional Operational Features STU
3.1.2 RAM Features STU
3.1.3 Configurations STU
3.1.3.1 Minimum STU
3.1.3.2 Typical STU
3.1.3.3 Maximum STU
3.1.3.4 Test STU
3.1.4 Physical Characteristics (Hardware) STU
3.2 Performance Requirements STU
3.2.1 Operational Performance Characteristics STU
3.2.2 RAM Performance Characteristics STU
3.2.3 Maintenance/Installation STU
3.2.3.1 Preventive Maintenance (Hardware) STU
3.2.3.2 Customer Performed Maintenance STU
3.2.3.3 Capital Test Equipment STU

3.2.4	Calendar Life (Hardware)	STU	1
3.3	Compatibility Requirements	STU	2
3.3.1	Predecessor Products	STU	3
3.3.2	Companion Products	STU	4
3.3.3	Similar Products	STU	5
3.3.4	Competitive Products	STU	6
3.4	Interdependencies	STU	7
3.5	Cost Objectives	STU	8
3.5.1	Field Maintenance	STU	9
3.5.2	Manufacturing Costs (Hardware)	STU	10
3.6	Product Verification	STU	11
3.7	Delivery Support Requirements	STU	12
3.7.1	Product Support Manuals	STU	13
3.7.2	Special Packaging	STU	14
3.7.3	Release Media	STU	15
4.0	Master Project Authorization	STU	16
5.0	Attachments	STU	17
A.	Statement of Compliance	STU	18
B.	Standards Checklist	STU	19
C.	Product Restrictions	STU	20
D.	Other	STU	21
			22
See Corporate Standard 10:01:03:011 for more details.		SST	23
			24
			25
			26
3.6 ERS		H2	27
			28
Purpose:	To define in detail the external characteristics of a software product or feature and to specify the user/system interface. The ERS is used as input to the GID, the IMS, the feature test plan (Chapter 5 of the Project Plan), and to external user manuals. The DR and GDS are inputs to the ERS.	SSTU	29
			30
			31
			32
			33
			34
			35
Audience:	Managers	SSTU	36
	Development Project	STU	37
	Evaluation	STU	38
	Product Design	STU	39
	Publications	STU	40
			41
Owner:	Baseline Control Board	SSTU	42
			43
Author:	Development Project/Product Design	SSTU	44
			45
Table of Contents:		SST	46
		S	47
1.0	Introduction	SSTU	48

3.0 DOCUMENTS
3.6 ERS

	A brief statement describing the software and its purpose.	STU	1 2
2.0	References	STU	3
3.0	Feature Description	STU	4
3.X	Feature Name	STU	5
3.X.1	Abstract	STU	6
	Give a brief and concise description of the feature.	STU	7 8
3.X.2	Description	STU	9
	Completely define the feature in detail. Include a description of its function and possible usage, a definition of the variables and options applicable to the feature, results expected from correct use of the feature, dependencies of this feature on other features.	STU	10 11 12 13 14 15 16
3.X.3	Interfaces	STU	17
	Identify and discuss any component interfaces with the user, his program, or the operator that are created or affected by this feature. Include input and output formats of the feature.	STU	18 19 20 21 22
3.X.4	Aborts and Recovery	STU	23
	Discuss the manner in which the software and/or system will react in abort situations that are caused by this feature. Include reaction of this feature to system and user initiated aborts.	STU	24 25 26 27 28
3.X.5	Performance	STU	29
	Discuss how this feature will affect the performance of the component, software product or overall system, from an external point of view, if it is helpful for the user to know it. Don't get into internal details.	STU	30 31 32 33 34 35
4.0	Product-level Description	STU	36
4.1	Interfaces to other Software Products.	STU	37
	Discuss external references to other software.	STU	38 39
4.2	Restrictions and Limitations.	STU	40
	Discuss known restrictions and limitations introduced as a result of this program or enhancement, at the user, operator, and programmer level.	STU	41 42 43 44
5.0	Errors	STU	45
	List all error diagnostics for the product, including severity level, significance, and corrective action for the user to take for	STU	46 47 48

each error. 1
6.0 Glossary (optional) STU 2
Terms, abbreviations, or symbols which have special meaning in this document. STU 3
4
5
6

3.7 GID H2 7

8

Purpose: To describe the overall process performed by a software product or component. This description covers major processes, the flow of data through the product, and descriptions of the data objects that are manipulated, as well as documentation at the module level--structure of the modules and the information that each passes or accesses. SSTU 9

Content: The GID consists of the Analysis Specification (AS) and the Design Specification (DS). SSTU 17

Audience: Development Project SSTU 20
Design Team STU 21
Product Design STU 22
Evaluation STU 23

Owner: Product Desgn/Advanced Systems Design SSTU 25

Author: Development Project SSTU 27

Table of Contents: SST 29

1.0	Introduction	SSSTU 31
2.0	Analysis Specification	STU 32
2.1	Overview	STU 33
2.2	Data Flow Diagrams (DFDs)	STU 34
2.2.1	Context Diagram	STU 35
2.2.2	Level 0 and lower DFDs	STU 36
2.2.3	Process Descriptions	STU 37
2.3	Data Structure Diagrams	STU 38
3.0	Data Dictionary	STU 39
4.0	Design Specification	STU 40
4.1	Structure Charts	STU 41
4.2	Module Descriptions	STU 42
4.3	Data Structure Diagrams (if needed)	STU 43
4.4	Design Issues	STU 44
5.0	References	STU 45

46

For a detailed description of the elements of a GID, see DCS ID=S3855. SST 47
48

3.0 DOCUMENTS
3.8 PSTP (PRODUCT SET TEST PLAN, FORMERLY SYSTEM TEST PLAN)

3.8 PSTP (PRODUCT SET TEST PLAN, FORMERLY SYSTEM TEST PLAN) H2

Purpose: To list build schedules and testing plans for a given CCR or CPS release. SSTU

Content: This plan outlines testing plans and requirements for a given Product Set build (in SDD). Only information that is not covered in other documents is noted here. One example of this type of information is installation testing planning. Performance test descriptions and size of a feature test base are examples of information that should not be included, since that information is available elsewhere. SSTU

Audience: Managers SSTU
Publications STU

Owner: Evaluation SSTU

Author: Evaluation SSTU

Table of Contents: SST

This outline is extracted from the proposed CDC Standard for system test plans CDC-STD 1.01.110. Please refer to that standard if more details are desired. SSTU

1.0 Scope SSSTU
This section identifies the software covered by the test plan. STU
2.0 Applicable documents. STU
3.0 Test Approach STU
3.1 Testable conditions STU
This subsection identifies the conditions that are to be tested in the software covered by the test plan. Examples include: STU

Performance SSTU
Resource Usage STU
Stress Testing STU
Availability STU
Reliability STU
Installability STU
Maintainability STU
Operability STU

3.0 DOCUMENTS
3.8 PSTP (PRODUCT SET TEST PLAN, FORMERLY SYSTEM TEST PLAN)

	Usability	STU	1
	Compatibility	STU	2
	Security	STU	3
	Functional Operation (features)	STU	4
			5
3.2	Testing Selection	SSTU	6
	This subsection defines a rationale for selecting which of the conditions identified in the section 3.1 are or are not to be tested, and identifies which are to be tested and which are not to be tested. This section may refer to individual feature test plans for details.	STU	7 8 9 10 11 12 13
3.3	Testing Procedures	STU	14
	This subsection identifies the procedures that are to be used to execute tests, record results, report results, store test data and procedures, and document errors.	STU	15 16 17 18
4.0	Entrance and Exit Criteria	STU	19
	There are three sets of criteria to be specified. These are: 1) minimum criteria to be satisfied to enter and remain in the system testing phase, 2) the minimum criteria to be satisfied to exit the system testing phase, and 3) the criteria for the software to become certified. This section describes the criteria which apply.	STU	20 21 22 23 24 25 26 27
5.0	Resource Requirements	STU	28
	a) Personnel Requirements.	STU	29
	b) Hardware Requirements.	STU	30
	c) Software and Tools Requirements.	STU	31
	d) Other Requirements.	STU	32
6.0	Schedules/Costs	STU	33
7.0	Responsibilities	STU	34
	Each activity described in the plan must be assigned to specific organizations or individuals.	STU	35 36 37 38 39
3.9 IMS		H2	40
			41
Purpose:	To describe the design of a product at all levels. The IMS is a deliverable and is also used as a basis for product maintenance.	SSTU	42 43 44 45
Content:	The IMS consists of the final GID (Analysis Spec, Design Spec, and Data Dictionary).	SSTU	46 47 48

3.0 DOCUMENTS
3.9 IMS

Audience: Customers SSTU 1
Development Project STU 2
3
Owner: Development Project SSTU 4
5
Author: Development Project SSTU 6
7
Table of Contents: (See GID Table of Contents) SST 8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48

4.0 REFERENCES

4.0 REFERENCES H1

Barber 1981 — Milton Barber, "Introduction to BSL 1.0" (forthcoming). Available from the author, Control Data Corp., SVL160. SSTTJ STTJ

Belady 1979 — L A Belady and M M Lehman, "Characteristics of Large Systems", chapter 3 of Research Directions in Software Technology, edited by Peter Wegner, MIT Press, Cambridge, MA. SSTTJ

Boehm 1975 — Barry W Boehm, Robert K McClean, and D B Urfrig, "Some Experiences with Automated Aids to the Design of Large-Scale Reliable Software", IEEE Transactions on Software Engineering, March 1975. Reprinted in Tutorial: Software Testing and Validation Testing, edited by Edward Miller and William E Howden, IEEE Computer Society, 1978. SSTTJ

Boehm 1976 — B W Boehm, J R Brown, and M Lipow, "Quantitative Evalustion of Software Quality", Proceedings of the 2nd International Conference on Software Engineering, IEEE Computer Society, 1976. SSTTJ

Boehm 1979 — Barry Boehm, "Software Engineering as it is", Proceedings of the 4th International Conference on Software Engineering, IEEE Computer Society, Los Alamitos, CA. Reprinted in Software Engineering, edited by Herbert Freeman and Philip Lewis, Acedemic Press, NY, 1980. SSTTJ

DCS ID=ARH3980 — "SASD Data Dictionary Tools User's Handbook", Rev F, 81/07/31. SSTTJ

DCS ID=ARH3981 — "SASD Graphics User's Handbook", Rev C, 81/07/31. SSTTJ

DCS ID=S3855 — "Analysis/Design Specification Procedure", 79/07/27. SSTTJ

4.0 REFERENCES

DCS ID=S4042	"Document Extraction Utility", DAP, 81/07/20.	SSTTJ	1
DeMarco 1978	Tom DeMarco, Structured Analysis and System specification, Yourdon Inc. NY.	SSTTJ	3
Deming 1981	Dr W Edwards Deming, Management of Statistical Techniques for Quality and Productivity, handout for CDC course 3056, June 9-12, 1981, (CDC private).	SSTTJ	6
Huntwork 1981	"Software Testing and Quility Assurance", handout #PR3545/PR3546 by Paul K Huntwork and Alvin L Wilson for CDC course.	SSTTJ	11
McCabe 1980	Thomas J. McCabe, Structured Testing, handout used in SDD's Structured Testing course.	SSTTJ	15
Metzger 1973	Philip W Metzger, Managing a Programming Project, Prentice Hall, NJ.	SSTTJ	18
Mills 1976	H D Mills, "Software Development", IEEE Transactions on Software Engineering, December 1976.	SSTTJ	21
NOS/VE Project	"Procedures and Conventions"	SSTTJ	25
Peterson 1973	R A Peterson, J R Donaldson, and H J Squires, Software Development Guidelines, Control Data Corp. publication 60418500.	SSTTJ	27
VanHorn 1980	Earl C VanHorn, "Software Must Evolve", in Software Engineering, edited by Herbert Freeman and Philip Lewis, Acedemic Press, NY, 1980.	SSTTJ	31
Yourdon 1978	Edward Yourdon and Larry Constantine, Structured Design, Yourdon Press, NY.	SSTTJ	36

Table of Contents

1.0 SDM OVERVIEW AND SUMMARY 1-1
1.1 INTRODUCTION 1-1
1.2 WHAT IS SDM? 1-1
1.3 BASELINE DOCUMENTS, DAP'S, BSL'S, QSS'S, AND RSE'S . . . 1-2
1.4 SOFTWARE DEVELOPMENT PHASES 1-3

2.0 DEVELOPMENT PHASES 2-1
2.1 FEASIBILITY PHASE 2-1
2.2 DEFINITION PHASE 2-2
2.3 ANALYSIS PHASE 2-3
2.3.1 INTRODUCTION 2-3
2.3.2 SA (STRUCTURED ANALYSIS) 2-4
2.3.3 IA (INFORMATION ANALYSIS) 2-4
2.3.4 STATE TABLES 2-5
2.3.5 DECISION TABLES 2-5
2.3.6 STRUCTURED TESTING 2-5
2.3.7 DATA FLOW ANALYSIS VERSUS STRUCTURE CHART ANALYSIS . 2-6
2.4 DESIGN PHASE 2-6
2.5 IMPLEMENTATION PHASE 2-7
2.6 EVALUATION PHASE 2-7
2.7 PUBLICATIONS PHASE 2-9
2.8 RELEASE-ACTIVITY PHASE 2-9

3.0 DOCUMENTS 3-1
3.1 PROJECT PLAN 3-1
3.2 AO/R (ARCHITECTURAL OBJECTIVES AND REQUIREMENTS) 3-7
3.3 SIS 3-8
3.4 GDS 3-8
3.5 DR 3-9
3.6 ERS 3-11
3.7 GID 3-13
3.8 PSTP (PRODUCT SET TEST PLAN, FORMERLY SYSTEM TEST PLAN) 3-14
3.9 IMS 3-15

4.0 REFERENCES 4-1