

# Horus IMSETY

Software Verification and Validation Plan  
Version 0.5 1st May 2007

<b>Project team:</b>	Jeroen Keiren	0569081
	Frank Koenders	0575629
	Thijs Nugteren	0574426
	Joeri de Ruiter	0578312
	Stijn Stiefelhagen	0579816
	Carst Tankink	0569954
	Pim Vullers	0575766
	Freek van Walderveen	0566348
<b>Project manager:</b>	Egbert Teeselink	
<b>Senior management:</b>	L. Somers	TU/e (HG 7.83)
	M. v.d. Brand	TU/e (HG 7.44)
<b>Adviser:</b>	R.J. Bril	TU/e (HG 5.09)
<b>Customer:</b>	E. v. Breukelen	ISIS

Computer Science, Eindhoven University of Technology, Eindhoven

### **Abstract**

This is the Software Verification and Validation Plan (SVVP) for the IMSETY project. This project is part of the Software Engineering Project (2IP40) and is one of the assignments at Eindhoven University of Technology. The document complies with the SVVP description from the Software Engineering Standard [1], as set by the European Space Agency. This document describes the activities concerning the project's verification and validation. It describes which actions have to be performed to approve documents and code. The appendices describe the necessary actions for each phase in the project.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Purpose . . . . .	1
1.2	Scope . . . . .	1
1.3	List of definitions . . . . .	1
1.4	List of references . . . . .	2
<b>2</b>	<b>Verification overview</b>	<b>3</b>
2.1	Organization . . . . .	3
2.1.1	Internal reviews . . . . .	3
2.1.2	External reviews . . . . .	3
2.1.3	Audits . . . . .	4
2.1.4	Tests . . . . .	4
2.2	Schedule . . . . .	5
2.3	Resources . . . . .	5
2.4	Project responsibilities . . . . .	5
2.5	Tools, techniques and methods . . . . .	5
<b>3</b>	<b>Administrative procedures</b>	<b>6</b>
3.1	Anomaly reporting and resolution . . . . .	6
3.2	Task iteration policy . . . . .	6
3.3	Deviation policy . . . . .	6
3.4	Control procedures . . . . .	7
3.5	Standards . . . . .	7
<b>4</b>	<b>Verification activities</b>	<b>8</b>
4.1	Reviews . . . . .	8
4.1.1	Internal reviews . . . . .	8
4.1.2	External reviews . . . . .	9
4.2	Formal proofs . . . . .	10
4.3	Tracing . . . . .	10
<b>5</b>	<b>Verification reporting</b>	<b>12</b>
<b>A</b>	<b>Review report template</b>	<b>13</b>
A.1	General review information . . . . .	13
A.2	Problems . . . . .	13
A.2.1	Legend . . . . .	13
A.3	Background information . . . . .	14
A.3.1	Major discussions . . . . .	14
A.3.2	Decisions . . . . .	14
A.4	Metrics . . . . .	14

<b>B UR Phase</b>	<b>15</b>
B.1 The User Requirements Review . . . . .	15
B.2 Requirements for User Requirements . . . . .	15
<b>C SR Phase</b>	<b>16</b>
C.1 The Software Requirements Review . . . . .	16
C.2 Requirements for Software Requirements . . . . .	16
<b>D AD Phase</b>	<b>17</b>
D.1 The Architectural Design Review . . . . .	17
D.2 Design Quality . . . . .	17
<b>E DD Phase</b>	<b>19</b>
E.1 The Detailed Design and Production Review . . . . .	19

# Document status sheet

<b>Document title</b>	Software Verification and Validation Plan
<b>Document identifier</b>	IMSETY/doc/SVVP
<b>Author(s)</b>	Stijn Stiefelhagen, Pim Vullers
<b>Version</b>	0.5
<b>Document status</b>	Internally approved

<b>Version</b>	<b>Date</b>	<b>Author(s)</b>	<b>Summary</b>
0.1 (Revision 185)	26-02-2007	Stijn Stiefelhagen, Pim Vullers	First version up for internal review.
0.2 (Revision 292)	28-02-2007	Pim Vullers	Corrected defects found during internal review.
0.3 (Revision 600)	02-04-2007	Pim Vullers	Added review purpose and updated AD phase appendix.
0.4 (Revision 610)	02-04-2007	Pim Vullers	Fixed remarks of the internal review.
0.5 (Revision 990)	01-05-2007	Pim Vullers	Updated the DD phase appendix.

# Document change report

<b>Document title</b>	Software Verification and Validation Plan
<b>Document identifier</b>	IMSETY/doc/SVVP
<b>Date of changes</b>	01-05-2007

<b>Section number</b>	<b>Reason for change</b>
E	DD phase about to begin

# Chapter 1

## Introduction

### 1.1 Purpose

This document describes procedures concerning the testing of the delivered products (product documents and software) of the IMSETY project for compliance with the requirements. The requirements that the software has to be verified against can be found in the product documents URD [14], SRD [9], ADD [3] and DDD [4]. The modules to be verified and validated are defined in the AD phase. The goal of verifying and validating is to check whether the software product to be delivered conforms to the requirements of the client and to ensure a minimal number of errors in the software. This project document is written for managers and developers of the IMSETY project.

### 1.2 Scope

In the IMSETY project a system has to be designed to support scientific experiments in space. This system, with user documentation, developer documentation and a demonstration model of an experiment (in software) has to be developed during this project. The documents need to be verified against their direct predecessor. The produced code will be tested according to the test plans, verifying them against their corresponding documents.

### 1.3 List of definitions

Table 1.1 contains the definitions of all used terms, acronyms and abbreviations in this document.

<b>AD</b>	Architectural Design
<b>ADD</b>	Architectural Design Document
<b>ATP</b>	Acceptance Test Plan
<b>Baseline</b>	A baseline is a document or a product that has been formally reviewed and agreed upon, and is a bas
<b>CI</b>	Configuration item
<b>DD</b>	Detailed Design
<b>DDD</b>	Detailed Design Document
<b>ECP</b>	External Contacts Person
<b>ITP</b>	Integration Test Plan
<b>PM</b>	Project Manager
<b>QAM</b>	Quality Assurance Manager
<b>SCMP</b>	Software Configuration Management Plan
<b>SM</b>	Senior Management
<b>SPMP</b>	Software Project Management Plan
<b>SQA</b>	Software Quality Assurance
<b>SQAP</b>	Software Quality Assurance Plan
<b>SR</b>	Software Requirements
<b>SRD</b>	Software Requirements document
<b>STP</b>	System Test Plan
<b>SVN</b>	Subversion
<b>SVVP</b>	Software Verification and Validation Plan
<b>TR</b>	Transfer
<b>UR</b>	User Requirements
<b>URD</b>	User Requirements document
<b>UTP</b>	Unit Test Plan

Table 1.1: List of definitions

## 20 1.4 List of references

- [1] European Space Agency. ESA Software Engineering Standards. Technical report, ESA, February 1991.
- [2] Horus. Acceptance Testplan, March 2007.
- [3] Horus. Architectural Design Document, March 2007.
- 25 [4] Horus. Detailed Design Document, March 2007.
- [5] Horus. Integration Testplan, March 2007.
- [6] Horus. Software Configuration Management Plan, March 2007.
- [7] Horus. Software Project Management Plan, March 2007.
- [8] Horus. Software Quality Assurance Plan, March 2007.
- 30 [9] Horus. Software Requirements Document, March 2007.
- [10] Horus. Software User Manual, March 2007.
- [11] Horus. Software Verification and Validation Plan, March 2007.
- [12] Horus. System Testplan, March 2007.
- [13] Horus. Unit Testplan, March 2007.
- 35 [14] Horus. User Requirements Document, March 2007.
- [15] Tom Verhoeff. Design Review Checklist, 2002. URL [http://wwwis.win.tue.nl/2M390/rev\\_des.html](http://wwwis.win.tue.nl/2M390/rev_des.html).
- [16] Tom Verhoeff. Requirements Review Checklist, 2002. URL [http://wwwis.win.tue.nl/2M390/rev\\_req.html](http://wwwis.win.tue.nl/2M390/rev_req.html).

## 40 Chapter 2

# Verification overview

## 2.1 Organization

The QAM checks the verification and validation of the activities of the IMSETY project. Therefore the QAM attends every internal or external review. If the QAM is not available the vice-QAM  
45 will take his place, this means that every time the QAM is mentioned it can also be the vice-QAM. If the QAM runs into problems he reports them to the PM. The PM needs to verify that these problems are resolved.

The IMSETY project uses the following methods of verification and validation:

### 2.1.1 Internal reviews

50 In order to keep the quality of our documents up to standards they will be subject to internal reviews.

The team carrying out the internal review of a technical or management document will at least consist of the following persons:

- The QAM. He will make the review document.
- 55 • One of the authors of the document.
- At least one other member of the IMSETY team, not part of the authoring team.
- The adviser/PM may also be present if necessary.

### 2.1.2 External reviews

When a document has been internally accepted it should have the desired quality. Having the  
60 right amount of quality does not mean that the document conforms to the customers expectations. Therefore an external review is held.

An external review can only take place after the document has been approved by the adviser. Any documents sent to the adviser will have to be accepted internally first. The external reviews of management documents will be done by the SM. The team carrying out the external review of  
65 a technical document will consist of the following people:

- The adviser (if available).
- At least one the author(s) (of the document to be reviewed).
- The QAM. He will make the review document.
- At least one other member of the IMSETY team.
- 70 • When necessary (URD [14], SRD [9], ATP [2]) also the customer.

### 2.1.3 Audits

Audits are reviews that assess compliance with software requirements, specifications, baselines, standards, procedures, instructions, codes and licensing requirements. Physical audits check that all items identified as being part of the configuration are present in the product baseline. A  
 75 functional audit checks that unit tests, integration tests and system tests have been carried out and records their success or failure. Functional and physical audits can be performed before the release of the software (ESA Software Engineering Standard Section 4.2.5 [1]). The SM is allowed to audit the IMSETY project to check if the procedures, as described in the management documents SPMP [7], SQAP [8], SVVP [11] and SCMP [6] are followed. Audits are not routine  
 80 checks, but the SM can request them.

The following rules apply to all audits:

- Only the SM can request audits.
- Audit requests must be directed to the PM.
- In the audit request the following information must be included:
  - 85 – Names of the auditors (at least two persons)
  - Possible dates of the audit
  - Purpose of the audit
  - Items that will be checked
- The audit is attended by a the QAM, the PM and possibly others as indicated by SM.
- 90 • The results of the audit are reported by a group member in a written report to the PM and the QAM within one week of the audit. This report must contain the following information:
  - Date of the audit
  - Participants in the audit
  - Checked items
  - 95 – Conclusion
  - Recommendations

### 2.1.4 Tests

The testing is done following the ESA Life cycle verification approach [1]. The following test plans (plans that outline the approach to testing) can be found as separate documents:

- 100 • ATP (Acceptance Test Plan) [2]
- STP (System Test Plan) [12]
- ITP (Integration Test Plan) [5]
- UTP (Unit Test Plan) [13]

The ATP [2] has to be approved by the client, as it will define the terms on which the final  
 105 product will be accepted. The results of the tests are presented to the PM and the QAM. The produced code and product documents must also be tested to assure that all the requirements are met. This can be found in section 4.3 and is documented in the appendices C, D and E.

## 2.2 Schedule

Before any document is delivered or externally reviewed it will be reviewed internally. The delivery  
110 dates of documents can be found in the SPMP [7].

Tracing matrices are part of the corresponding documents. At the end of each phase a final  
trace will be performed to ensure that all elements of the document can be traced back to preceding  
documents and vice-versa. The results of this final trace will be enclosed in the document for the  
corresponding phase.

## 115 2.3 Resources

The verification and validation will happen in the same situation as described in the SPMP [7].

## 2.4 Project responsibilities

Some roles as defined in the SPMP [7] have responsibilities related to verification and validation.  
These responsibilities are:

120 Project Manager:

- The work is adequately inspected.

Quality Assurance Manager:

- Assuring that the requirements of the documents are adhered to.
- Assuring the documents conform to the specified layout and contain the proper information.
- 125 • Lead the review sessions.
- Manage the test runs.

Librarian / Configuration Manager:

- Tag documents that have been committed for review, and the approved versions if changes  
where needed.

## 130 2.5 Tools, techniques and methods

By using SVN there is a common repository, accessible through the internet, where any project  
members is able to view, update and remove files.

By the use of Trac the project members is able to see who and what changes have been made  
on SVN in the past. For more information about this read the SCMP [6].

## 135 Chapter 3

# Administrative procedures

### 3.1 Anomaly reporting and resolution

140 Everything that does not conform to the standards it should be up to, or does not conform to requirements it should conform to, is an anomaly. Procedures for anomaly resolution can be found in the SQAP [8]. Furthermore, it is the task of the QAM to monitor whether the procedures as defined in the management plans (SPMP [7], SCMP [6], SQAP [8] and SVVP [11]) are followed. This is done during team meetings, reviews and by randomly checking configuration items. Findings are reported to the PM. It then is the responsibility of the PM to enforce compliance with defined procedures.

### 145 3.2 Task iteration policy

Every task performed is to be reviewed as described in chapter 4. If, during a review, problems are discovered concerning the correct conclusion of the task a decision is made concerning the iteration of the task. Guidelines are provided for the following cases:

- 150 • The team responsible was unable to complete their task, most probably because of one of the risks as described in section 3.3 of the SPMP [7]. In this case, it is the responsibility of the QAM to solve the problem and make sure the task is completed as described in the SPMP [7]. If the QAM is unable to do so, he must report this to the PM. If problems arise concerning the dependencies between tasks these are to be reported to the PM.
- 155 • A structural error was found in the execution of the task, for example the output of a piece of code that does not comply with the requirements.  
In this case, the team that is responsible performs the task again. If necessary the PM schedules extra man-hours.
- 160 • An item was forgotten during the execution of a task.  
Depending on the severity of this item the QAM will decide whether a redo of the task is needed, or only the forgotten item needs to be fixed. This case will most probably occur in processing review remarks.

### 3.3 Deviation policy

During the IMSETY project, the procedures described in the management documents are followed. However, if in the QAM's opinion, this endangers the completion of the project then the QAM can 165 decide to deviate from these procedures. If the decision is made to deviate from the procedures described in the management documents, the PM must be informed of such a deviation.

### 3.4 Control procedures

Procedures assuring that configuration items are not accidentally or deliberately changed are described in the SCMP [6].

### 170 3.5 Standards

Before both internal and external reviews, the authors certify that the document is according to ESA Software Engineering standard [1], and that the document complies with the standard layout as detailed in SCMP [6].

# Chapter 4

## Verification activities

### 4.1 Reviews

Review procedures are held during all phases of the IMSETY project. Configuration items are reviewed in the phase they are delivered; an overview of which item is delivered in which phase can be found in the SPMP [7]. All project and product documents have one of the following statuses:

- Draft (initial status)
- Internally approved with proposed changes
- Internally approved
- Conditionally approved
- (Externally) approved

Note that approved technical documents are not modified (unless the completion of the project is endangered). With respect to the approved management documents only appendices for every phase are added during the project. The appendices are approved during review meetings.

For a document to become (internally) approved it has to be reviewed. Here internal and external reviews of technical and management documents (Section 2.1.1 respectively 2.1.2) are distinguished. For more information about reviews, see [16, 15].

As noted above each document starts with the draft status. Once there has been a internal review it either becomes Internally approved or needs some changes. When there are only minor changes needed the document will be internally approved when the QAM has confirmed that the changes have been made. If it concerns major changes a new internal review will be needed.

Internally approved documents can be scheduled for external review. During this review the document can reach the highest status of externally approved if there are no defects. If there are only minor defects the document may be conditionally approved, these defects need to be solved to retrieve the highest status. If major defects show up during the review the document needs to be changed to solve these defects. Because this involves great changes it should pass a new internal review before it may be subject to another external review.

#### 4.1.1 Internal reviews

For the organization of internal reviews see section 2.1.1. See appendix A for the review report template.

Table 4.1 shows the action list for the preparation and execution of the internal reviews of documents. T is the time of the review meeting.

Nr.	Actor	Action	Time
1	QAM	Set a date for the internal review of the document	-
2	Author	Deliver the paper review version of the document to the reviewers	T - 2 workdays
3	Reviewer	Inspect the document and annotate	Before T
4	Reviewer	Discuss all errors other than language errors	T
5	Author & QAM	Write down all necessary changes	T
6	Reviewer	Decide if the document can be approved, provided the stated changes are made	T
7	QAM	If the document cannot be approved, an appointment for a new review meeting is made	T
8	Author	Collect annotated documents	T
Only when the document is rejected do actions 9 and 10 apply.			
9	QAM	See to it that the stated remarks are handled properly by the team delivering the document	After T
10	QAM	Grant the document the status internally accepted if all requested changes are made	After T

Table 4.1: Internal review actions

### 4.1.2 External reviews

For the organization of external reviews see section 2.1.2. Table 4.2 shows the action list for the preparation and execution of the external reviews of documents. T is the time of the review meeting. This procedure is only for the external review of product documents. The metrics of the external review will be sent to the SM. We will use the same format as in internal reviews for external reviews (see appendix A).

Nr.	Actor	Action	Time
1	ECP	Set a date and place for the external review of the document	After internal acceptance
2	Author	Deliver the paper version of the document to all reviewers (with a reply address)	T - 5 workdays
3	Reviewer	Inspect the document and write down all errors explicitly	Before T
4	Reviewer	Deliver remarks to the moderator	Before T
5	QAM	Inspect remarks	Before T
6	Author	Lead the meeting and keep discussions to the point	T
7	QAM	Document everything that is discussed during the review	T
8	Reviewer	Discuss all comments that need explanation or discussion	T
9	Author	Collect the remarks on the documents	After T
10	Reviewer, QAM	Decide the status of the document at the end of the meeting. There are three possible outcomes: the document is rejected and a new appointment is made the document is accepted and the status Approved is granted	After T
11	QAM	Make minutes of the review, and hand these together with the remarks of the reviewers to the Senior Management. Also make sure they will go to the configuration management system	After T
Only when the document is rejected do actions 12 and 13 apply.			
12	QAM	See to it that the remarks are handled properly by the team responsible for the document	After T
13	QAM	Grant the document the status Approved if all reviewers inform that their remarks are handled properly, eventually after another review if the remarks included major changes	After T

Table 4.2: External review actions

## 4.2 Formal proofs

Formal proofs will not be used. The SVVP will be updated concerning formal proofs when needed.

## 4.3 Tracing

215 During the IMSETY project the relation between the input and the output of a phase must be checked several times. A traceability table as result of the final trace is included in the output document of the phase. In this table the CI is traced to the input of the phase. During the software life cycle it is necessary to trace:

- 220 • User requirements to software requirements and vice versa, this is documented in the SVVP/SR (Appendix C).
- Software requirements to component requirements and vice versa, this is documented in the SVVP/AD (Appendix D).
- Component requirements to DD requirements and vice versa, this is documented in the SVVP/DD (Appendix E).
- 225 • Integration tests to architectural units and vice versa, this is described in the integration test plans [5]. These tests are performed during the DD-phase.
- Unit tests to the modules of the detailed design, this is described in the unit test plans[13]. These tests are performed during the DD-phase.

- 230 • System tests to software requirements and vice versa, this is described in the system test plans [12]. These plans are executed during the DD-phase.
- Acceptance tests to user requirements and vice versa, this is described in the acceptance test plans [2]. These tests are executed during the TR-phase.

To support traceability, all requirements are uniquely identified.

235 Requirements are identified as follows: DCat-RCat-No (without dashes) where DCat defines the category of the requirement. Allowed values are:

- UR (User Requirement URD [14])
- SR (Software Requirement SRD [9])

RCat defines the subcategory of the requirement. The allowed values for RCat depend on the value of DCat. When DCat has the value UR, allowed values for RCat are:

- 240 • CAR (Capability Requirement)
- COR (Constraint Requirement)

When DCat has the value SR, allowed values for RCat are:

- FUR (Functional Requirement)
- EFR (Extra-functional Requirement, such as Reliability and Maintainability requirements)

245 'No' is a number of each requirement in ascending order, starting at 1. Examples:

- URCOR7 identifies constraint requirement seven of the user requirements.
- SRFUR12 identifies functional requirement twelve of the software requirements.

## Chapter 5

# Verification reporting

250 For the validation of product documents (apart from the URD [14]) a part will be added to the document concerning its validation. This means it should be stated against which document it has been validated and the resulting traceability table. The validation should be done by the authors of the document.

255 A verification report is written as a result of a test. It is written by the people performing the test of the product document or code. It contains the following items:

- Unique reference number of the test plan
- Problems discovered and, if available, solutions to these
- Acceptance or disapproval of the product document or code. In case of disapproval, accompanied with a short explanation of the reasons for disapproval

260 For the verification and validation of the entire IMSETY project, progress meetings are held with the SM according to the SPMP [7].

# Appendix A

## Review report template

### A.1 General review information

Project: IMSETY  
Document:  
Version:  
Type of review:  
Date:  
Start time:  
End time:  
QAM: Jeroen Keiren or Frank Koenders  
Author(s):  
Non-author participants:  
Review outcome

### 265 A.2 Problems

Nr.	Ref.	Type	Remark	Severity	Action
-----	------	------	--------	----------	--------

#### A.2.1 Legend

**Number** Sequence number of remark starting at 1.

**Reference** Reference to page/section and line number(s)

**Type :**

270

**M** Missing

**X** Extra/superfluous

**A** Ambiguous

**I** Inconsistent

275

**U** Improvement desirable (unclear)

**S** Not conforming to standards

**R** Risk-prone

**F** Factually incorrect

**N** Not implementable

280 **E** Editorial

**Severity** Major or Minor, when in doubt leave open

**Action** Decision about future action:

- No action
- Local change
- 285 • Global change
- Investigate

## A.3 Background information

### A.3.1 Major discussions

Abstract containing most important information concerning major review discussions.

290 **A.3.2 Decisions**

List of decisions made during the review session.

## A.4 Metrics

The following numbers of problems have been identified during the review:

Major	
Minor	
Total	

## 295 Appendix B

# UR Phase

### B.1 The User Requirements Review

The outputs of the User Requirements Definition Phase are formally internally and externally reviewed in the User Requirements Review (UR/R). It ensures that the URD states the user requirements clearly and completely and a general description of the processes to be supported (the environment) is present.

The SPMP, SCMP, SVVP and SQAP are only internally reviewed.

### B.2 Requirements for User Requirements

User requirements should be realistic, that is:

- 305 • Clear.
- Verifiable.  
"The product shall be user friendly" is not verifiable.
- Complete.
- Accurate.  
310 Among other things, the URD is inaccurate if it requests something that the user does not need, for example a superfluous capacity.
- Feasible.

In the URD each requirement must ...

- ... have a unique identifier.  
315 Traceability in other phases depends on this identifier.
- ... have a priority.  
Essential requirements have to be met for the software to be acceptable.
- ... be marked if unstable.  
Unstable requirements may depend on feedback from later phases.

## 320 Appendix C

# SR Phase

### C.1 The Software Requirements Review

The outputs of the Software Requirements Definition Phase are formally reviewed in the Software Requirements Review (SR/R). Internal reviews are held before the formal external SR/R takes  
325 place. It ensures that:

- The SRD states the software requirements clearly, completely and in sufficient detail to start the design phase.
- The STP [12] is an adequate plan for system testing the software in the DD phase.

### C.2 Requirements for Software Requirements

330 In the SRD each requirement must ...

- ... have a unique identifier.  
Traceability in other phases depends on this identifier.
- ... be marked essential or not.  
Essential requirements have to be met for the software to be acceptable.
- 335 • ... have a priority if the transfer will proceed in phases.
- ... be marked if unstable.  
Unstable requirements may depend on feedback from other phases.
- ... have references that trace it back to the URD.  
340 A software requirement is caused by one or more user requirements.
- ... be verifiable

Besides the requirements the SRD must contain a traceability matrix containing trace from the software requirements to the user requirements and vice-versa. This matrix should be complete, meaning all requirements should be traceable.

## 345 Appendix D

# AD Phase

### D.1 The Architectural Design Review

The outputs of the Architectural Design phase are formally reviewed in the Architectural Design Review (AD/R). An SQA report is also input for the AD/R. Internal reviews are held before the  
350 formal external AD/R takes place. It ensures that:

- The ADD describes the optimal solution to the problem stated in the SRD.
- The ADD describes the architectural design clearly, completely and in sufficient detail to start the detailed design phase.
- The ADD is a good design for IMSETY.
- 355 • The ITP [5] is an adequate plan for integration testing the software in the DD phase.

### D.2 Design Quality

A good design is:

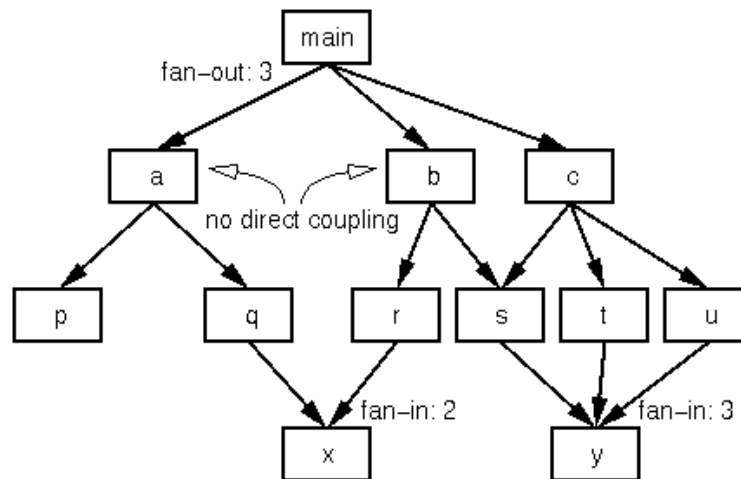
- Adaptable: it is easy to modify and maintain.
- Efficient: it makes a minimal use of available resources.
- 360 • Understandable: it is not only clear for the developers but also for outsiders.
- Modular: the components are simple and independent from each other:
  - A change in one component has minimal impact on other components.
  - A small change in requirements does not lead to system wide changes.
  - The effects of an error condition are isolated to its source component.
  - 365 – A component is understandable as a stand-alone unit, without reference to others.

Good components obey the information hiding principle: software design decisions are encapsulated so that the interface reveals as little as possible about its inner workings. For example, a component should hide how its data is stored: it can be in memory (in an array, list, tree, ...) or in a temporary file.

370 Some rules to choose components:

- Minimize coupling between components:
  - Minimize the number of items that are passed between components;

- Pass only the data that are needed (data coupling);
  - Do not pass a structure of which only a small part is being used (stamp coupling);
  - 375 – Avoid the use of control flags (control coupling);
  - Do not use global data (common coupling).
- Maximize cohesion inside a component: put elements into a component that are related to each other; they contribute for example to the same task.
  - Restrict fan-out: restrict the number of child components.
  - 380 • Maximize fan-in: reuse components as often as possible.
  - Use factoring: avoid duplication of functionality. Cut the common functionality from the components and put it into a reusable component.
  - Separate logical and physical functionality: top-level components must be separated from physical aspects (the data they deal with); the level of abstraction of a component must be according to its place in the hierarchy.
  - 385



structure chart showing fan-in and fan-out

# Appendix E

## DD Phase

### E.1 The Detailed Design and Production Review

390 The outputs of the Detailed Design and production Phase are formally reviewed in the Detailed Design and production Review (DD/R). An SQA report is also input for the DD/R. Internal reviews are held before the formal external DD/R takes place. It ensures that:

- The DDD [4] describes the detailed design clearly, completely and in sufficient detail to allow development and maintenance by software engineers not involved in the project.
- Modules have been coded according to the DDD [4].
- 395 • Modules have been verified according to the unit test specifications in the UTP [13].
- Major components have been integrated according to the ADD [3].
- Major components have been verified according to the integration test specifications in the ITP [5].
- 400 • The software have been verified against the SRD [9] according to the system test specifications in the STP [12].
- The ATP [2] specifies the test design, test procedures and test cases so that all user requirements in the URD [14] can be validated.
- The SUM [10] explains what the software does and instructs the users how to operate the software correctly.