



**UNCLASSIFIED**

NLR-CR-2007-702

## **Role of Requirements in ATM Operational Concept Validation (RORI-OCV)**

Final Report (D3)

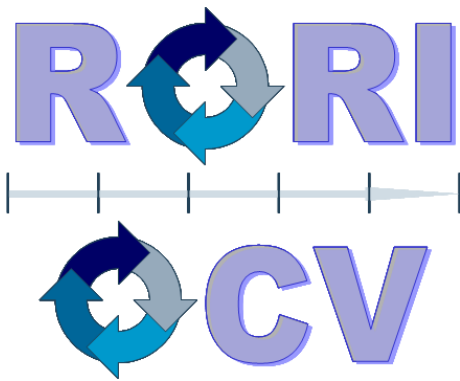
J. Teutsch, Y.A.J.R. van de Vijver and J.G. Braakhuis



## Executive summary

# Role of Requirements in ATM Operational Concept Validation (RORI-OCV)

Final Report (D3)



### Problem area

This document provides the final report (D3) for the EUROCONTROL project Role of Requirements in ATM Operational Concept Validation (RORI-OCV), also known as Task Requirement Sheet (TRS) 11093DK/07.

The project is concerned with an analysis of the role of requirements during the different lifecycle phases of Operational Concept Validation (OCV). To this end, a survey of current established practices in requirements engineering is performed. This work results in a number of recommendations concerning definition and interpretation of terms, application of relevant standards, and a strategy for the development of

requirements and their evolution along the lifecycle as defined within the E-OCVM and OCVSD framework.

### Description of work

In summary, the following tasks are carried out within the RORI-OCV project:

- Describing and selecting characteristic examples of the relevant standards from the military, as well as engineering and ATM organisations and giving recommendations regarding their application within the framework of Operational Concept Validation as set out within the OCVSD and the E-OCVM.

### Report no.

NLR-CR-2007-702

### Author(s)

J. Teutsch  
Y.A.J.R. van de Vijver  
J.G. Braakhuis

### Report classification

UNCLASSIFIED

### Date

December 2007

### Knowledge area(s)

ATM & Airport Simulation & Validation

### Descriptor(s)

REQUIREMENTS  
REQUIREMENTS ANALYSIS  
OPERATIONAL CONCEPT  
VALIDATION  
AIR TRAFFIC MANAGEMENT (ATM)  
E-OCVM

- Giving recommendations concerning the definition and interpretation of terms related to Operational Concept Validation and Requirements Engineering on the basis of well-known sources and the selected characteristic examples of relevant standards.
- Establishing a strategy for the development of requirements and their evolution along the lifecycle as defined within the E-OCVM and OCVSD framework, including ownership issues, from the selected characteristic examples of relevant standards.
- Analysing selected examples of European R&D Operational Concepts including those delivered by the Eurocontrol EATM units with respect to the use of and in particular the development and documentation of requirements.
- Deriving conclusions on possible differences between the desired and current situation of the role of operational concept, requirements, and specification generation and documentation in the lifecycle of concept development and validation.
- Providing recommendations for changes or additions in the OCVSD and the E-OCVM on the basis of the

established strategies for operational concept, requirements, and specification generation and documentation.

### **Results and conclusions**

The main result of the RORI-OCV study is the definition of a requirements development strategy consisting of the following three processes:

- The Requirements Elicitation or Capture Process aims at eliciting the initial requirements from stakeholders.
- The Requirements Analysis or Specification Process is performed throughout the R&D relevant phases of the E-OCVM life cycle to iteratively transform the initial raw requirements into a clear, consistent, and stable system specification that can be used as the basis for the industrialisation of the system in later phases of the life cycle.
- The Requirements Management Process is responsible for maintaining the set of requirements throughout all the iterations, together with the rationale and supporting documentation for each of the changes made.

An important recommendation considering the management of requirements is the identification of an actor called central validation or development

manager, who oversees all development activities and integrates requirements defined at different levels of detail and within different operational or technical domains.

Another important recommendation is to have a strong participation of industry in the early phases of the R&D life cycle in order to ensure that the specifications established at the end of the R&D life cycle phases really fulfil the expectations of the ATM industry that needs to bring products from R&D stages into operation.

Furthermore, recommendations for further studies were given concerning the following topics:

- Interviews with ANSP experts at different system levels and within different operational domains regarding their view on requirements.
- Industry expectations at the end of the R&D life cycle phases, the level of detail of R&D system specifications at this stage, and the consequences of the customer-supplier relationship on the development of requirements.
- Visualisation and modelling tools supporting elicitation and analysis of functional requirements.
- System development processes, the role of NASA

Technology Readiness Levels (TRL), and the relation with E-OCVM life cycle phases.

- Consequences of different ATM system architecture models on the requirements development processes.

Finally, it was recommended to determine a level of system complexity and unpredictability, regarding the operational outcome of simulations with humans in the loop, which defines the domain of validation. In this domain it will be difficult to verify a concise requirement so that hypotheses for operational objectives must be validated instead. Defining such a boundary in general terms should help when establishing a verification strategy within the E-OCVM framework.

### **Applicability**

This document will be used as input to the FAA-EUROCONTROL Working Meeting in Berlin in November 2007. It will be the task of FAA-EUROCONTROL Action Plan 5 (Validation) to elaborate a verification strategy which considers this study and which fits into the E-OCVM framework.



**UNCLASSIFIED**  
NLR-CR-2007-702

# **Role of Requirements in ATM Operational Concept Validation (RORI-OCV)**

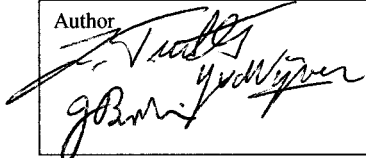

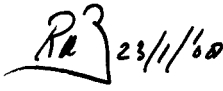
Final Report (D3)

J. Teutsch, Y.A.J.R. van de Vijver and J.G. Braakhuis

No part of this report may be reproduced and/or disclosed, in any form or by any means without the prior written permission of the owner.

Customer                   Eurocontrol  
Contract number        T07/11093/DK/1207/03  
Owner                     Eurocontrol  
Division                 Air Transport  
Distribution             Public  
Classification of title   Unclassified  
December 2007

Approved by:

Author  Y.A.J.R. van de Vijver	Reviewer 	Managing department 
---	---	--

**UNCLASSIFIED**



## Summary

This document provides the final report (D3) of NLR regarding the main activities carried out within the project entitled 'Project Management Plan RORI-OCV (D1)' (RORI-OCV), also known as EUROCONTROL Task Requirement Sheet (TRS) 11093DK/07.

The project is concerned with an analysis of the role of requirements during the different lifecycle phases of Operational Concept Validation (OCV). To this end, the activity foreseen by NLR will be comprised of a survey of current established practices in requirements engineering. This work will result in a number of recommendations concerning definition and interpretation of terms, application of relevant standards, and a strategy for the development of requirements and their evolution along the lifecycle as defined within the E-OCVM and OCVSD framework.

The project runs from August 2007 until December 2007 and is divided into five work packages (WP) each addressing one of the following activities:

- NLR Project Management and Reporting (WP0)
- Survey of Relevant Standards and Terminology (WP1)
- Requirement Development Strategy in the OCV Lifecycle (WP2)
- Analysis of European R&D Operational Concepts (WP3)
- Recommendations for OCVSD and E-OCVM (WP4)

This document contains the essence of all work packages mentioned above. Following the survey of standards and terminology, a requirement development strategy with links to the OCV lifecycle was developed. The document also addresses the analysis of European R&D operational concepts and investigates how requirements were established.

Recommendations for both the OCVSD and the E-OCVM are given concerning the integration of a requirement development strategy and a better understanding of the different development streams for operational concepts and technology running in parallel with validation.

The main result of the RORI-OCV study is the definition of a requirements development strategy consisting of the following three processes:

- The Requirements Elicitation or Capture Process aims at eliciting the initial requirements from stakeholders.
- The Requirements Analysis or Specification Process is performed throughout the R&D relevant phases of the E-OCVM life cycle to iteratively transform the initial raw



requirements into a clear, consistent, and stable system specification that can be used as the basis for the industrialisation of the system in later phases of the life cycle.

- The Requirements Management Process is responsible for maintaining the set of requirements throughout all the iterations, together with the rationale and supporting documentation for each of the changes made.

An important recommendation considering the management of requirements is the identification of an actor called central validation or development manager, who oversees all development activities and integrates requirements defined at different levels of detail and within different operational or technical domains.

Another important recommendation is to have a strong participation of industry in the early phases of the R&D life cycle in order to ensure that the specifications established at the end of the R&D life cycle phases really fulfil the expectations of the ATM industry that needs to bring products from R&D stages into operation.

Furthermore, recommendations for further studies were given concerning the following topics:

- Interviews with ANSP experts at different system levels and within different operational domains regarding their view on requirements.
- Industry expectations at the end of the R&D life cycle phases, the level of detail of R&D system specifications at this stage, and the consequences of the customer-supplier relationship on the development of requirements.
- Visualisation and modelling tools supporting elicitation and analysis of functional requirements.
- System development processes, the role of NASA Technology Readiness Levels (TRL), and the relation with E-OCVM life cycle phases.
- Consequences of different ATM system architecture models on the requirements development processes.

Finally, it was recommended to determine a level of system complexity and unpredictability, regarding the operational outcome of simulations with humans in the loop, which defines the domain of validation. In this domain it will be difficult to verify a concise requirement so that hypotheses for operational objectives must be validated instead. Defining such a boundary in general terms should help when establishing a verification strategy within the E-OCVM framework.



## Contents

<b>1</b>	<b>Introduction</b>	<b>13</b>
<b>2</b>	<b>Survey of Terminology and Relevant Standards</b>	<b>16</b>
2.1	Survey of Terminology used in Selected Sources	18
2.1.1	Concept-related Terminology	19
2.1.2	Requirement-related Terminology	22
2.1.3	Additional Terminology and Definitions	28
2.2	Survey of Relevant Systems Engineering Standards	34
2.2.1	Analysis of Standard IEEE 1362 for IT System Definition and ConOps	35
2.2.2	Analysis of Standard IEEE 830 on Software Requirement Specifications	35
2.2.3	Analysis of Standard IEEE 1233 on System Requirement Specifications	36
2.2.4	Analysis of Standard IEEE 15288 on Systems Engineering	37
2.2.5	Analysis of Standard ECSS-E-10/E-40 on Systems/Software Engineering	37
2.2.6	Analysis of Standard ED-79/ED-12B/ED-80 (Systems/Software/Hardware)	38
2.2.7	Analysis of Standard ED-78A (Approval of Air Traffic Services)	40
2.2.8	Development Life Cycles	44
2.3	Conclusions on the Survey of Terminology and Relevant Standards	48
<b>3</b>	<b>Requirement Development Strategy in the OCV Life Cycle</b>	<b>51</b>
3.1	Operational Concept Validation Life Cycle	51
3.2	Actors, Roles and Responsibilities in the OCV Process	58
3.3	Processes within the Requirement Development Strategy	64
3.3.1	Requirements Capture Process	64
3.3.2	Requirements Analysis Process	65
3.3.3	Requirements Management Process	67
3.3.4	Architectural Design	68
3.4	Summary	71
<b>4</b>	<b>Analysis of European R&amp;D Operational Concepts</b>	<b>72</b>
4.1	Analysis of European Commission Project EMMA	72
4.1.1	Project Background of EMMA	72
4.1.2	Applied Methods for Concept Development and Validation in EMMA	73
4.1.3	Role of Requirements in EMMA	77
4.1.4	Conclusions on the EMMA Requirements Development Process	77



4.2	Analysis of EATM Project FASTI	79
4.2.1	Project Background of FASTI	79
4.2.2	Applied Methods for Concept Development and Validation in FASTI	80
4.2.3	Role of Requirements in FASTI	80
4.2.4	Conclusions on the FASTI Requirements Development Process	81
4.3	Analysis of Industry Project VICTORIA	82
4.3.1	Project Background of VICTORIA	82
4.3.2	Applied Methods for Concept Development and Validation in VICTORIA	82
4.3.3	Role of Requirements in VICTORIA	83
4.3.4	Conclusions on the VICTORIA Requirements Development Process	83
<b>5</b>	<b>Conclusions and Recommendations</b>	<b>85</b>
5.1	Conclusions of the Study	86
5.2	Recommendations for AP5 and the OCVSD	90
5.3	Recommendations for the E-OCVM	94
	<b>References</b>	<b>98</b>
	<b>Appendix A Analysis of Terminology</b>	<b>105</b>
A.1	Definition of ‘Operational Concept’	105
A.2	Definition of ‘Concept of Operations’	106
A.3	Definition of ‘Concept of Use’	106
A.4	Definition of ‘Operational Procedures’	106
A.5	Definition of ‘Operational Requirements’	107
A.6	Definition of ‘User Requirements’	107
A.7	Definition of ‘Functional and Non-functional Requirements’	108
A.8	Definition of ‘System Specifications’	109
A.9	Zachman Enterprise Architecture Framework™	110
	<b>Appendix B Analysis of System Engineering Standards</b>	<b>111</b>
B.1	IEEE Std 610.12-1990 (Glossary of Software Engineering Terminology)	111
B.2	IEEE Std 1362-1998 (Guide for IT System Definition and ConOps)	113
B.3	IEEE Std 1233-1998 (Guide for System Requirements Specifications)	122
B.4	ESA standard ECSS-E-40 - Software	125
B.5	IEEE 15288 on systems engineering	127
B.5.1	Stakeholder Requirements Definition Process	127
B.5.2	Requirements Analysis Process	130



B.6	ESA ECSS-E-10 on systems engineering	133
B.6.1	Requirement Engineering	133
B.6.2	Documentation	136
B.6.3	Requirements and Recommendations for the Wording	138
B.7	ESA ECSS-E-40 on Software Engineering	139
B.7.1	Software Requirements Analysis	139
B.7.2	Requirements baseline (RB)	141
B.8	EUROCAE ED-12B on Airborne Software Engineering	143
B.8.1	Software Requirements Process	143
B.9	EUROCAE ED-79 on Airborne Systems Engineering	145
B.9.1	Requirements Capture	145
B.9.2	Derived Requirements	147
B.9.3	Validation of Requirements	148
B.10	EUROCAE ED-80 on Airborne Hardware Engineering	152
B.10.1	Requirements Capture Process	152
B.10.2	Conceptual Design Process	154
B.10.3	Hardware Design Standards	154
B.10.4	Hardware Design Data	155
B.11	EUROCAE ED-78A Approval Guidelines	155



## Abbreviations

ACC	Area Control Centre
ACI	Airports Council International
ADS-B	Automatic Dependent Surveillance Broadcast
AEA	Association of European Airlines
AECMA	European Association of Aerospace Industries
AFAS	Aircraft in the Future Air Traffic Management System
AFM	Aircraft Flight Manual
AGFA	Air-Ground Functional Architecture
AIP	Aeronautical Information Publication
AL	Aeronautical Lexicon
ANSP	Air Navigation Service Provider
AOC	Airline Operational Control (Centre)
AP	Action Plan
API	Application Program Interface
ARP	Aerospace Recommended Practices
ASA	Automated Support to Air Traffic Services
A-SMGCS	Advanced Surface Movement Guidance and Control Systems
ATC	Air Traffic Control
ATM	Air Traffic Management
ATS	Air Traffic Services
BETA	Operational Benefit Evaluation by Testing A-SMGCS
C-ATM	Co-operative Air Traffic Management
CAA	Civil Aviation Authorities
CAATS	Co-operative Approach to ATS
CANSO	Civil Air Navigation Services Organisation
CASCADE	EUROCONTROL ADS-B Implementation Co-ordination Programme
CMMI	Capability Maturity Model Integration
CNS	Communication Navigation Surveillance
CONOPS	Concept of Operations
COTS	Commercial Off-The-Shelf
COU	Concept of Use
CRD	Co-ordinated Requirements Determination
D	Deliverable
DDF	Design Definition File
DEVAM	Development of EATCHIP/EATMP Validation Methodologies



DJF	Design Justification File
DoDAF	Department of Defense Architecture Framework (US)
DRD	Document Requirements Definition
EA	Enterprise Architecture
EASA	European Aviation Safety Agency
EATCHIP	European ATC Harmonisation and Integration Programme
EATM	European Air Traffic Management
EATMP	European Air Traffic Management Programme
ECAC	European Civil Aviation Conference
ECSS	European Co-operation for Space Standardisation
EIA	Electronic Industries Association
EMMA	European Airport Movement Management by A-SMGCS
E-OCVM	European Operational Concept Validation Methodology
ER	Essential Requirement
ESA	European Space Agency
ETSO	European Technical Standard Orders
EUROCAE	European Organisation for Civil Aviation Equipment
FAA	Federal Aviation Administration (US)
FAR	Federal Aviation Regulations
FAST	FAA Acquisition System Toolset
FASTI	First ATC Support Tools Implementation
FDPS	Flight Data Processing System
FHA	Functional Hazard Analysis
FMEA	Failure Modes and Effect Analysis
HMI	Human Machine Interface
IA	Interoperability Assessment
IATA	International Air Transport Association
ICAO	International Civil Aviation Organisation
ICD	Interface Control Document
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IFATCA	International Federation of Air Traffic Controller Associations
INTEROP	Interoperability
IR	Implementation Rule
IRD	Interface Requirements Document
ISO	International Organisation for Standardisation
JAR	Joint Aviation Requirements



MAEVA	Master ATM European Validation Plan
MASPS	Minimum Aviation System Performance Standards
MEL	Minimum Equipment List
MMI	Man-Machine Interface
MONA	Monitoring Aids
MOPS	Minimum Operational Performance Specifications
MOTS	Modified Off-The-Shelf
MS	Microsoft™ Corporation
MTCD	Medium-term Conflict Detection
NAF	NATO Architecture Framework
NASA	National Aeronautics and Space Administration
NATO	North Atlantic Treaty Organization
NLR	Nationaal Lucht- en Ruimtevaartlaboratorium
NOTAM	Notice to Airmen
OATA	Overall ATM/CNS Target Architecture
OCD	Operational Concept Document
OCV	Operational Concept Validation
OCVSD	Operational Concept Validation Strategy Document
ORD	Operational Requirements Document
OPA	Operational Performance Assessment
OSA	Operational Safety Assessment
OSD	Operational Services and Environment Definition
OSEIC	Operational Service Environment Information Capture
OTS	Off-The-Shelf
PDR	Preliminary Design Review
PF	Pilot Flying
PNF	Pilot Not Flying
PSSA	Preliminary System Safety Analysis
R&D	Research and Development
RA	Requirements Analysis
RB	Requirements Baseline
RC	Requirements Capture
RCP	Required Communication Performance
RFP	Request for Proposal
RM	Requirements Management
RORI-OCV	Role of Requirements in Operational Concept Validation
RTCA	Radio Technical Commission for Aeronautics



SDD	System Description Document
SEM	Systems Engineering Manual (FAA)
SES	Single European Sky
SESAR	Single European Sky ATM Research
SOA	Service-oriented Architecture
SP	Sub-Project
SPOR	Services Procedures and Operational Requirements
SPR	Safety and Performance Requirements
SRD	System Requirements Document
SRR	System Requirement Review
SRS	Software Requirements Specification
SYRS	System Requirements Specification
SYSCO	System Supported Co-ordination
TMA	Terminal Manoeuvring Area
TP	Trajectory Predictor
TRD	Technical Requirements Document
TRL	Technology Readiness Level
TRS	Task Requirement Sheet
TS	Technical Specification
UAC	Upper Area Control
UML	Unified Modelling Language
V	Validation Life Cycle Phase (as defined in the E-OCVM)
VAMS	Virtual Airspace Modelling and Simulation (NASA)
VARTAN	Validation Reporting Template Analysis by NLR
VGH	Validation Guideline Handbook
VICTORIA	Avionics Technology Demonstrator
WG	Working Group
WP	Work Package



This page is intentionally left blank.



## **1 Introduction**

The project entitled 'Role of Requirements in Operational Concept Validation' (RORI-OCV), also known as TRS 11093DK/07 (Ref. [1]), describes an activity that is meant to support EUROCONTROL-FAA Action Plan 5 (AP5) in the establishment of a 'Validation and Verification Strategy'. Action Plan 5 is part of the collaboration of EUROCONTROL and the U.S. Federal Aviation Administration (FAA) in ATM R&D under Memorandum of Co-operation NAT-I-3454.

The major deliverable of AP5 is the Operational Concept Validation Strategy Document (OCVSD) which currently collects experiences of the different contributors made in the establishment of the European Operational Concept Validation Methodology (E-OCVM) and the outcome in terms of best practices of commonly organised workshops for validation practitioners on the topics of human-in-the-loop simulations and scenario use. Within AP5 the general feeling is that the work on the Validation Strategy has reached a mature level and that the work can now be extended towards the topic of a Verification Strategy.

AP5 participants analysed the topic and concluded that in order to complete a verification strategy it would be necessary to have a common understanding of the role of requirements in the R&D process. Therefore, as a first step, there needs to be a proper understanding of the definition and use of requirements including ownership and documentation of requirements as well as their application within the lifecycle of operational concept validation as described in both the OCVSD (Ref. [18]) and the E-OCVM (Ref. [15]). AP5 participants agreed to initiate this work with a survey of current established practices in requirements engineering in relation to Operational Concept Validation. The RORI-OCV project provides the specification of this survey. The results of the survey will be used to organise a workshop for practitioners to consolidate the role of requirements in the Operational Concept Validation process and to subsequently develop the Verification Strategy.

As mentioned above, the Operational Concept Validation process is outlined in both the OCVSD and the E-OCVM. The documents define a lifecycle with pertaining maturity phases for the development of an Operational Concept. The final output of the outlined process is a description of such an 'Operational Concept' including information about whether it is valid (i.e. validated to be fit for purpose) or not. That description is the subject of continual discussion.



There is no single common view in the R&D community of how to describe an Operational Concept and how that description is further used in the subsequent concept development lifecycle phases by ANSPs and Industry (e.g. in the procurement process). Therefore, recommendations for the content and level of detail of an Operational Concept at the end of the Operational Concept Validation lifecycle are needed. The validated Operational Concept is made available to ANSPs and Industry as basis for implementation. On the European level those Operational Concepts need to be described in a sufficiently generic way as the local implementations need being adapted to the local conditions. From a US perspective operational concepts are refined until they contain sufficient detail to initiate the requirements development process

TRS 11093DK/07 questions the role of Operational Concepts as a suitable or complete description of a potential end system that will allow ANSPs and Industry together build a globally interoperable future. R&D (the Operational Concept Validation community) needs to understand how to communicate effectively the detail of new concepts such that ANSPs and Industry can understand how to build and implement the developed concept.

A first study about improving the transfer from Operational Concept Validation to implementation was already carried out (see Ref. [23]). One of the main conclusions of this study was that end results should be gathered and re-checked by a so-called focal point. This already shows one of the dilemmas of the current end result of R&D: while a concept might be described in sufficient detail, it usually is very difficult to come to conclusive validation results due to a lack of traceability and documentation of requirements and subsequent interpretation of the results with respect to these requirements. The fact that a focal point is called for, might be closely linked to the problem that research in Europe usually is divided among many R&D organisations, some of which are closely linked to an ANSP that is looking at the specific needs of local implementations. Other conclusions in the mentioned study directly concern standards and definition of requirements as well as configuration control and change management aspects. Therefore, the RORI-OCV project will specifically look into these issues, i.e. the application of relevant standards (military, engineering, ATM organisation), and a strategy for the development of requirements and their evolution along the lifecycle as defined within the E-OCVM and OCVSD framework, including ownership of the requirements.

In a preparatory working meeting the AP5 participants felt that many terms related to the definition of an operational concept and the definition of requirements are still used with inconsistent interpretations. Consequently the term 'requirements' is avoided in the OCVSD whereas in the E-OCVM some sort of requirements are being implicitly used in the validation



phases of the concept development lifecycle as prototypes and pre-operational implementations are being developed and used. Thus, it will also be one of the tasks of the proposed study to produce recommendations concerning definition and interpretation of terms which could be partially based on sources like the EUROCONTROL Aeronautical Lexicon on the OneSky website, or the EATMP glossary of terms.

In summary, the following tasks are carried out within the RORI-OCV project:

- Describing and selecting characteristic examples of the relevant standards from the military, as well as engineering and ATM organisations and giving recommendations regarding their application within the framework of Operational Concept Validation as set out within the OCVSD and the E-OCVM,
- Giving recommendations concerning the definition and interpretation of terms related to Operational Concept Validation and Requirements Engineering on the basis of well-known sources and the selected characteristic examples of relevant standards,
- Establishing a strategy for the development of requirements and their evolution along the lifecycle as defined within the E-OCVM and OCVSD framework, including ownership issues, from the selected characteristic examples of relevant standards.
- Analysing selected examples of European R&D Operational Concepts including those delivered by the Eurocontrol EATM units with respect to the use of and in particular the development and documentation of requirements
- Deriving conclusions on possible differences between the desired and current situation of the role of operational concept, requirements, and specification generation and documentation in the lifecycle of concept development and validation.
- Providing recommendations for changes or additions in the OCVSD and the E-OCVM on the basis of the established strategies for operational concept, requirements, and specification generation and documentation.

The document at hand is the draft final report of the RORI-OCV project that will be used to initiate discussions at the AP5 working meeting in Berlin in November 2007. It will contain the surveys of terminology and systems engineering standards and will present a proposal on how to describe the actors and processes involved in developing operational concepts and requirements along the lifecycle of operational concept validation as described in the E-OCVM. Current working practices in major European R&D validation activities will be considered in this description.



## **2 Survey of Terminology and Relevant Standards**

Currently many terms relating to operational concept validation and the associated requirements capturing and management process are still used with inconsistent interpretations. Therefore, this chapter investigates the use of the following terminology:

- Operational Concept
- Concept of Operations
- Concept of Use
- Operational Procedures
- Operational Requirements
- User Requirements
- Functional and Non-functional Requirements
- Technical or Technology Requirements
- System Requirements
- System Specifications

The review of these terms and a recommendation for their use and definition is based on the following well-known sources:

- The EUROCONTROL Aeronautical Lexicon (via OneSky: <https://extranet.eurocontrol.int/>)
- The EATMP glossary of terms (via EATM website <http://www.eurocontrol.int/eatm/>)
- The NASA glossary of terms (via VAMS website <http://www.vams.arc.nasa.gov/>)
- IEEE 610 standard glossary of software engineering terminology (Ref. [24])

Additionally, two sources were referenced that came into view during the process of making the terminology survey. They are the E-OCVM glossary, and a glossary that is part of the FAA Acquisition System Toolset (FAST).

The survey of relevant systems engineering standards focuses on the development phases relating to operational concept validation, i.e. phases V1 to V3 as described in the E-OCVM (Ref. [15]). For the survey, an appropriate selection of documents was made, addressing these phases in particular. Among others, the selection is based on the fact that during the last decade, unification and harmonisation of standards have taken place. The IEEE has adopted ISO/IEC standards, and specific military standards, such as MIL-STD-498, have been officially phased out in favour of these harmonised standards (although they are still used to the present day).



Therefore, the most applicable set of standards considered are the following IEEE standards:

- IEEE 15288 (adoption of ISO/IEC 15288) on systems engineering
- IEEE 12207 (adoption of ISO/IEC 12207) on software life cycle processes
- IEEE 1233 on system requirement specifications
- IEEE 830 recommended practice for software requirements specifications
- IEEE 1362 guide of information technology, system definition, concept of operations (CONOPS) document

In addition to these IEEE standards, the following standards and documents were also considered for analysis:

- ESA ECSS E-10 on systems engineering and ECSS E-40 on software engineering (both are tailored versions of the according IEEE standards for space projects)
- FAA SEM (Systems Engineering Manual)
- EUROCAE ED-78A, ED-79, ED-12B and ED-80
- CAATS Best Practices Manual
- Improvement of E-OCVM Transition V3 to V4 Study Report (see Ref. [23])

Aircraft manufacturers often have more detailed, company-specific standards for (parts of) the development lifecycle. Most of these detailed standards are based on the more generally available standards. For instance, the Airbus A380 fly-by-wire system has been developed to ED-79/ARP 4754 level A, its computer software to ED-12B/DO-178B, and its computer hardware to ED-80/DO-254. These general standards are considered as well as part of the survey activity.

The definition of terms is closely related with the activities for analysis of the relevant systems engineering standards. Thus, the definitions and possible differences in their understanding will be highlighted in the beginning of this document before continuing with the description and analysis of the standards.



**2.1 Survey of Terminology used in Selected Sources**

The survey of terminology started with a simple collection of definitions from abovementioned sources. During this process it was found that most of the definitions of terms that could be found in the Aeronautical Lexicon (AL) of EUROCONTROL came from the EATMP glossary and from the EUROCONTROL Operational Concept Document (OCD), the latter being linked to the Single European Sky ATM Research (SESAR) and the Overall Target Architecture Activity (OATA) project. Thus, both SESAR (WP2.4) and OATA refer to the OCD terminology.

Generally, terms and definitions from the Aeronautical Lexicon and from the NASA (VAMS) and FAA (FAST) sources were more focussed on the operational concept, while the IEEE 610 standard contained several definitions of requirements, constraints and specifications related to software development.

When making the survey, an additional topic of discussion was the definition of two terms related with industry-related processes. They are:

- Enterprise Architecture
- Service-oriented Architecture

These terms will also be defined in the following. The consequences of the described processes for the operational concept validation lifecycle, though, will not be discussed as part of the survey. They will be considered when discussing the proposed requirement development strategy in Chapter 3.

The table below gives an overview of the sources (see description above) in which definitions of the identified terms could be found, indicated by crosses (X). Crosses in brackets mean that only related terms are described. Abbreviations for the sources can be found in Appendix A.

<b>Term</b>	<b>AL</b>	<b>AL</b>	<b>AL</b>	<b>AL</b>	<b>E-OCVM</b>	<b>VAMS</b>	<b>FAST</b>	<b>IEEE 610</b>
	<b>S E S A R</b>	<b>O C D</b>	<b>E A T M</b>	<b>O A T A</b>				
Operational Concept	X	X	X		X	X		
Concept of Operations	X	X		X				



Term	AL S E S A R	AL O C D	AL E A T M	AL O A T A	E-OCVM	VAMS	FAST	IEEE 610
Concept of Use							X	
Operational Procedures			X					
Operational Requirements	X		X				(X)	(X)
User Requirements			X					
Functional Requirements	X		X				(X)	X
Non-functional Requirements	(X)							(X)
Technical Requirements								(X)
System Requirements								(X)
System Specifications								X

Table 2-1: Overview of Terminology and Sources

**2.1.1 Concept-related Terminology**

The list of terms related to the definition of an operational concept comprises the following:

- Operational Concept
- Concept of Operations
- Concept of Use
- Operational Procedures



For each of these terms the sources were referenced and commonalities and differences were worked out. The exact definitions with references of sources can be found in Appendix A of this document.

Starting with the term 'Operational Concept' it was found that the three European definitions, coming from SESAR (identical to EUROCONTROL OCD description), EATM and the E-OCVM in their general definition are identical. They describe the operational concept as being a 'high-level description' or 'a broad outline' of an 'operational structure' or a 'set of ATM components' which need to meet 'a set of high-level user requirements'. While EATM only talks about an 'operational structure', the E-OCVM clearly mentions 'ATM components'. SESAR goes into more detail and adds that it is not only a set of ATM components but also a description of the way in which 'they are organised and operated'.

Differences emerge in the second part of each of the definitions which in all cases tries to give more detail. However, while both the EATM and E-OCVM definitions talk about additional content of the operational concept, SESAR mentions exclusions. The latter seems to be a confusing element of the definition as the exclusions leave room for speculation about the actual contents. The SESAR definition states that an operational concept does not include a description of the air navigation infrastructure or a description of a technical system and its use. The other two definitions, however, mention that an operational concept does indeed include a description of airspace organisation, operational procedures and associated operational requirements for system support. The E-OCVM even goes as far as stating that it should additionally contain information on involved actors and their tasks and responsibilities as well as enablers, events and drivers of these events, processes and their relation to each other, and information flows.

Thus, looking at the differences in definition it seems like SESAR is addressing another audience than the two other definitions, probably audiences with two different levels of profundity. Clearly, the E-OCVM definition addresses validation experts who usually need the mentioned contents as input for their activities.

Considering the conclusions above, plus the fact that the American definition from VAMS is going into the same direction as the EATM and E-OCVM definitions by clearly identifying a list of operational services which should be described as part of the operational concept description, it seems that the E-OCVM definition is the most usable definition from the standpoint of validation and should therefore be considered in the following. However, before such a conclusion can be made the other terms should be looked at as well.



Regarding the term ‘Concept of Operations’ two different definitions were found. The first definition again is from SESAR (or EUROCONTROL OCD) and describes a concept of operations as a description of how an operational concept is applied including an identification of functions and processes, interactions and information flows as well as actors with their roles and responsibilities. In that regard it is interesting to notice that while excluding these topics from the definition of the operational concept, SESAR includes them now specifically in the definition of a concept of operations. Both EATM and the E-OCVM do not define such a term.

The second definition of the term comes from OATA. This definition, however, is different from the SESAR definition and seems to be an alternative for the operational concept definition. According to OATA the concept of operations documents the purpose of the ATM system and assists in the system requirements identification process, describing the operational concept with its characteristics and behaviour from the point of view of a user.

Here it seems that the SESAR definition is a bit clearer about the contents of a concept of operations. However, while EATM and E-OCVM seem to see a concept of operations as part of the operational concept, SESAR makes a clear distinction between the two.

The term ‘Concept of Use’, although being a familiar term within the European validation community, could not be found in any of the European resources. Also VAMS or IEEE did not have a definition for the term. As a conclusion an internet search was performed, which resulted in the finding of a definition from the FAA Acquisition Management System (FAST). Here, a concept of use has a similar definition as the earlier definitions for the concept of operations. The definition talks about the concept of use as being an explanation of how new capabilities will function within the existing operational environment from a user standpoint. Thus, the roles and responsibilities of key participants should be defined. Operational issues that systems engineers need for the development of requirements should be explained and procedural issues identified.

From this, it can only be concluded that the terms ‘Concept of Operations’ and ‘Concept of Use’ are used in the same context. Keywords are system interoperability, actors with roles and responsibilities, and procedural issues. The definition of these elements should in both cases lead to support for systems engineers in defining system requirements.

Finally, the definition for the probably most straightforward term ‘Operational Procedures’ was only found in the EATM reference. Operational procedures are described as consisting of the

contents of ATC operational manuals incorporating international, national and local rules and regulations, procedures and working practices.

In resume, it can be stated that an Operational Concept is seen as a high-level description of ATM system elements that need to address a high-level set of user requirements. Depending on either the audience or the user of an operational concept description, it either includes or excludes a description of user-related information on system interoperability, data and information flows, actors with roles and responsibilities and operational procedures which would support the system requirement capturing process. Such a more detailed description is referred to as either Concept of Operations or Concept of Use. The Operational Procedures are defined to consist of abovementioned rules, regulations, procedures and working practices.

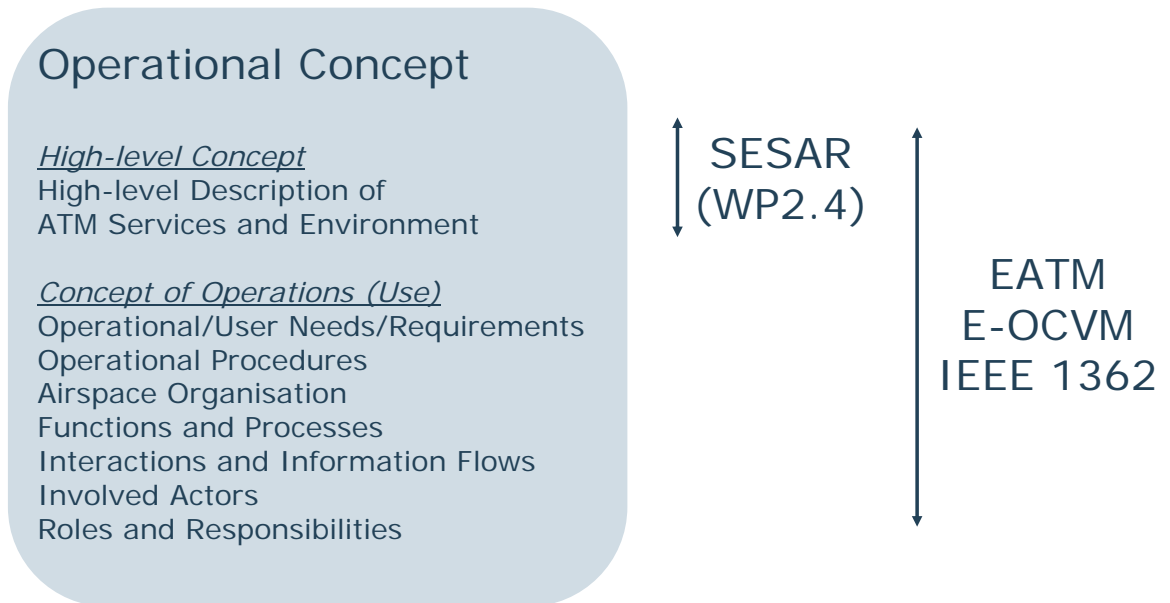


Figure 2-1: Overview of Terminology related to Operational Concept

**2.1.2 Requirement-related Terminology**

The list of terms related to the definition of requirements comprises the following:

- Operational Requirements
- User Requirements
- Functional and Non-functional Requirements
- Technical or Technology Requirements
- System Requirements
- System Specification



For each of these terms, plus several additional terms encountered in the process, the sources were referenced and commonalities and differences were worked out. The exact definitions with references of sources can again be found in Appendix A of this document.

Looking at the list of terms it seems obvious that all the definitions can only be certain flavours of the definition of a requirement, some of them only being used in a specific context, others being part of a larger group of similar types of requirements. Thus, it seemed the most logical approach to start from the general definition of a requirement, and only then look at the different flavours, preferably from top to bottom, i.e. starting with the most general group of requirements.

As mentioned before, the IEEE definitions were considered to be the most valuable source of information. The IEEE 610 standard gives three descriptions for a requirement. It is either a condition or capability needed by a user to solve a problem or achieve an objective, or it is a condition or capability that must be met or possessed by a system or system component to satisfy a formally imposed document (standard, specification, contract etc.), or it is considered a documented representation of a condition or capability as described above. As such, the term 'Requirement' would refer to different views or rather different actors or activities in the process of concept validation. The first definition would then be the viewpoint of the user who requires the ATM system component considered in the operational concept to have a certain capability or meet a certain condition. The second definition would take the point of view of a system engineer who would have to develop a system or system component according to certain formally imposed conditions and capabilities. The third definition clearly concerns the management of requirements and links the term to an identifiable item for a condition or capability in a formal document.

The general definition found in the FAA Acquisition Management System (FAST) clearly refers to the IEEE definitions and takes the viewpoint of the user with the FAA being that user. A clear reference is also given to the documentation that is expected to describe the requirements.

Also SESAR has a general description of the term 'Requirement'. This description talks about 'characteristics' that identify 'needed accomplishment levels' rather than system capabilities although the same is certainly meant. Instead of mentioning 'conditions' that must be met SESAR also talks about 'objectives for a given set of conditions' that must be 'achieved' thereby giving a definition that is less rigid.



Together with general definitions of the term ‘Operational’ the IEEE definitions lead to a definition of an operational requirement. IEEE gives three definitions. The term ‘Operational’ pertains to a system or component that is either ready for use or is installed in its intended environment but it can also pertain to the environment itself in which the system or component is intended to be used. Thus, in conjunction with the term ‘Requirement’ only the last definition makes sense, so that an ‘Operational Requirement’ in terms of the IEEE standard is a condition or capability needed by the user in an environment in which a system or a system component is intended to be installed. The system or component must meet the condition or possess the capability to satisfy the description of such a condition or capability in a formally imposed document (such as a specification or a standard).

SESAR has its own description of an operational requirement and relates it very much to an ATM problem by constituting that it is a statement of the operational attributes of a system for the effective and/or efficient provision of air traffic services to users. In effect, this is again a less rigid formulation clearly addressing the ATM system and its services as the system that the requirements are imposed on.

EATM goes even further by relating it to a traffic demand situation for an air traffic control sector. In that sense, the definition is much too specific and problem-oriented to be further considered in this analysis. It does not refer to a system or system component that could address other issues than sector management.

Surprisingly, a definition of ‘User Requirements’ could not be found in any of the sources. The reason for this could be that the understanding of the term is probably so obvious that a clear definition has not been asked for yet. At least, EATM defines the term ‘User’, stating that it denotes the aggregate of organisations, people, automated systems, infrastructure, procedures, rules and information, which receive services from an ATM system, but are not part of it. According to EATM, there are two major categories of users, aviation users and non-aviation users. The term ‘Aviation Users’ refers to the aircraft operators. Typical examples of ‘Non-aviation Users’ are physically or functionally adjacent Air Navigation Systems, air defence, law enforcement agencies, customs, etc.

Considering this definition it is clear that the term user requirements must be seen just as broad as the term user, meaning that requirements could be imposed from any of these users. This stresses the importance of the requirements capturing process that is expected to take place at the very beginning of operational concept validation and should consider all kinds of aviation and non-aviation users. In actual systems engineering these rather high-level requirements (the



EATM User Requirements Document actually describes ATM stakeholder needs) must then be translated to functional requirements for the respective system that is proposed as a solution to the related ATM problem.

This leads to the following term that needs to be defined. Based on the IEEE requirement definition, a 'Functional Requirement' is what the user of the system or the system component wants the system to do. It thus specifies a function that a system or system component must be able to perform. The SESAR and EATM definitions are identical. They see a functional requirement as an operational requirement that determines what function a system should perform. In that way, the SESAR, EATM and IEEE definitions are identical. SESAR and EATM go a bit further by specifying that a functional requirement can usually be expressed by a verb applying to a type of data, such as 'display aircraft position'.

FAST does not describe the term but refers to a 'Functional Baseline' which describes the functional, interoperability, and interface characteristics, of a system or a component and the verification required to demonstrate the achievement of those characteristics. This, however, shows a different approach as it means that in some cases there could also be functional constraints, especially when there is a situation of 'technology push'.

This dilemma can be best understood when looking at the term 'Non-functional Requirement' as opposed to the term 'Functional Requirement'. No definition of that term could be found in any of the sources. However, the IEEE document mentions a number of other requirements, which could be placed into this category of requirements. They are:

- Design Requirement
- Implementation Requirement
- Interface Requirement
- Physical Requirement
- Performance Requirement

Looking at the different definitions for these requirements it seems that, even though these requirements could be operationally motivated and based on system user input, they have the ability to constrain the types of solution that will meet the purely functional requirements. Consequently, the IEEE definition of the term 'Design Requirement' states that it is a requirement that 'specifies or constrains the design of a system or system component'. This means that such requirements could eventually be motivated quite differently, depending on the circumstances. SESAR describes the design and development process of a system as a 'set of processes that transforms requirements into specified characteristics or into specifications of a



product, process or system'. In that way it is left open whether design and development imposes constraints or whether design and development are ideally translating the functional requirements. However, it is clear that the design requirements to a large extent determine the characteristics and specifications of the final product.

Quite similarly, IEEE describes the term 'Implementation Requirement' as a requirement 'that specifies or constrains the coding or construction of a system or system component'.

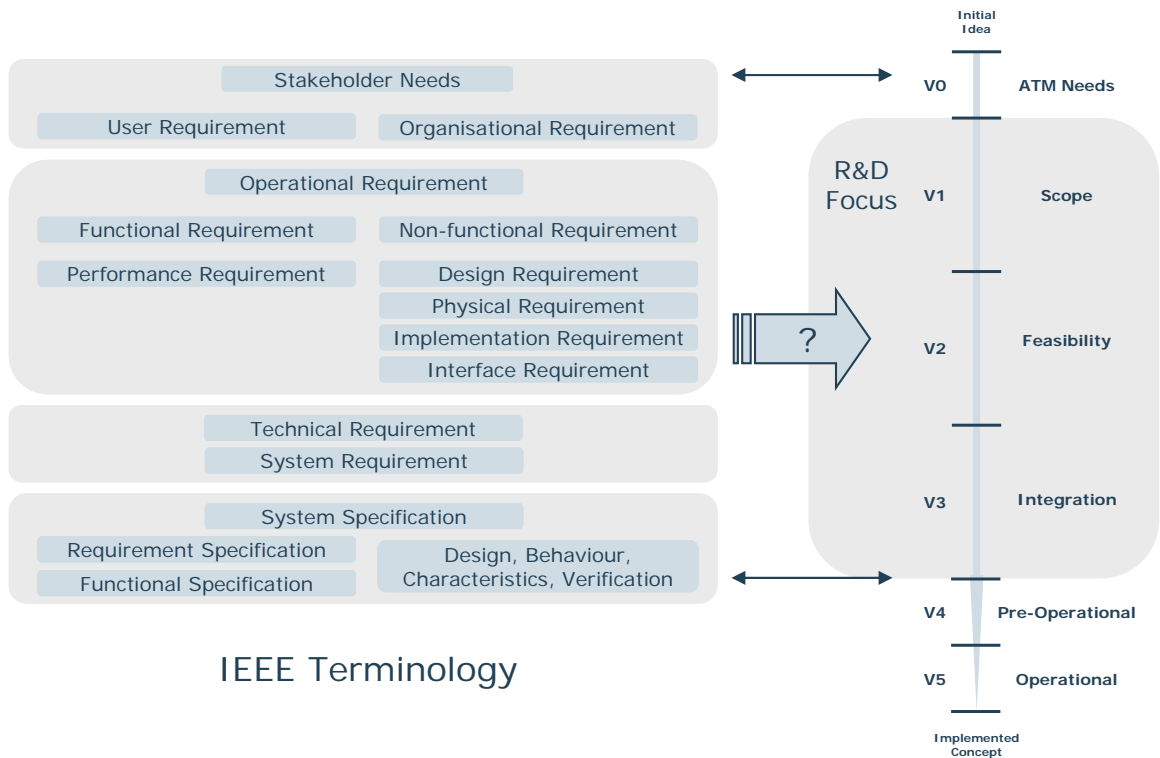
The term 'Interface Requirement' is defined as a requirement 'that specifies an external item with which a system or system component must interact, or that sets forth constraints on formats, timing, or other factors caused by such an interaction'. With that definition IEEE points to another term which is 'Interoperability Requirements'. The latter can be seen as an alternative to the term 'Interface Requirement'.

Also the term 'Physical Requirement' is defined by IEEE. According to this source the term 'specifies a physical characteristic that a system or system component must possess, e.g. material, shape, size, weight'. Again it is left open what the motivation for such a constraint is. It could be related to the function itself, to the user, the operational environment or even to legal and business considerations. In any case, it narrows down the number of possible solutions as does a design requirement.

Finally, the term 'Performance Requirement' is defined by IEEE as a requirement 'that imposes a condition on a functional requirement, e.g. a requirement that specifies the speed, accuracy or, or memory usage with which a given function must be performed'. SESAR finds a similar definition for the term. It is described as 'the extent to which a mission or function must be executed, generally measured in terms of quantity, quality, coverage, timeliness or readiness'. Again the motivation could be manifold and either related or completely unrelated to the functional requirements for the system.

No matter whether the abovementioned requirements or constraints are functional or non-functional requirements, they all can be summarised by the term 'System Requirements' or 'Technical Requirements' and 'Technology Requirements' as they clearly pertain to systems or system components, especially given the background of the IEEE document which is very much software-related. As has been mentioned earlier, they also determine to a large extent the characteristics and specifications of the final product.

This leads to the final term that will be analysed in this document, the term ‘System Specification’. According to IEEE a system specification is a ‘document that specifies, in a complete, precise, verifiable manner, the requirements, design, behaviour, or other characteristics of a system or component, and, often, the procedures for determining whether these provisions have been satisfied’. The latter refers to the verification activities of the system. IEEE then details this definition by giving separate definitions for the terms ‘Requirement Specification and ‘Functional Specification’. A requirement specification is described as a ‘document that specifies the requirements for a system or component’. It is further elaborated that such a document typically includes ‘functional requirements, performance requirements, interface requirements, design requirements and development standards’. A functional specification thus is a ‘document that specifies the functions that a system or component must perform’ and is ‘often part of a requirements specification’.



IEEE Terminology

Figure 2-2: Overview of IEEE Terminology related to Requirements and Specifications

In conclusion, Figure 2-2 gives an overview of all requirement-related terms (mainly from IEEE Ref. [24]) and their approximate position along the concept validation life cycle (as described in Ref. [15]). While stakeholder needs are assessed at the very beginning of the life cycle in V0, it is the task of the requirements capturing process at the beginning of V1 to translate these needs into initial user requirements. Additionally, there may be organisational requirements being



derived from, for example, organisational structures, available resources, and regulations. All these requirements are captured and documented as operational requirements of the system considered. Operational requirements are usually subdivided into functional and non-functional requirements. While performance requirements are usually related with functional requirements, all other (non-functional) requirements mentioned in IEEE standard 610 are constraining the solution(s) by imposing requirements on the design, physical appearance, technical implementation and interoperability (interfaces) of the system or system component under consideration. Also abovementioned organisational issues, such as budgets and project schedules can be part of these non-functional requirements.

In V1 most capturing activities will take place. This includes the translation of operational requirements into technical and system requirements. In V2 requirements will be refined, for example, based on the results of verification activities and validation platform acceptance tests that need to take place before V2 validation activities or based on the V2 validation activities on acceptability, usability and feasibility of the proposed concept(s) and system(s). In V3 the operational concept and the prototype system should be consolidated in order to be able to make performance assessments for validation purposes. This final validation activity should eventually end in consolidated system specifications, which are usually composed of a requirement specification and a functional specification as well as specifications of design, behaviour and characteristics of the system. The necessary steps for verification of the system requirements should also be described in this document based on the experiences made in the concept and technology development activities being carried out in parallel with the validation activities.

Although the specifications should consolidate all validation results, it is not quite clear what the required level of detail is, i.e. whether the specifications must and actually can include all issues which are expected for a successful implementation. After all, they are still research specifications. There still needs to be clarity considering the industrial contribution that is given within the pre-operational phase V4 and at the transition from V3 to V4. An answer to this open question will depend on issues such as stakeholder involvement in the validation process and whether the strategy for technology development is in line with validation activities or not.

### **2.1.3 Additional Terminology and Definitions**

During the discussion of terminology and the possible solution strategies for requirement development along the E-OCVM lifecycle several additional terms were mentioned that have their origin in SESAR related activities.

Special attention was given to terms relating to system architectures. They are:

- Enterprise Architecture (EA), in particular the so-called Zachman framework, and



- Service-oriented Architecture (SOA)

In the following the general concept of both architectures is briefly introduced for a broader understanding.

The term ‘Enterprise Architecture’ (EA) describes the structure and the behaviour of the processes, information systems, personnel and sub-units in an organisation in line with the core goals and the strategic direction of the organisation. It provides information to the business decision makers on how complex organisations are structured, how they function, and which technology supports those functions. Relationships between business and technology are modelled in such a way that key dependencies are exposed from the underlying complexity and so can be used to support decisions. In other words, a critical path of dependencies between the operational domain and technology is presented. In combination with data models, organisation structures and standards the enterprise architecture allows answering otherwise intricate questions, provided the dependencies between domains are all known. Increasing complexity in information technology can be seen as a driver for the development of models for the enterprise architecture (see Ref. [1]).

Usually, two types of enterprise architectures are discerned: the ‘as-is’ model and the ‘to-be’ model. While the first model is used to understand the dependencies in the current organisation, the second model is usually applied to determine the impact of change. Given the complexity of such models it is sensible to have a formal way of categorising the information content in so-called frameworks. Currently, the best-known framework is the Zachman framework that defines a number of views on the enterprise model (see Ref. [40]). The complete model with views and pertaining elements can be found in Appendix A of this document.

The term ‘Service-oriented Architecture’ (SOA) describes an architectural style that supports service orientation. It is interpreted in many ways in current literature and due to its popularity among software engineers the abbreviation SOA is used to brand products and technology, which adds to the confusion. However, a general understanding is that SOA establishes an architectural model that aims to enhance the efficiency, agility, and productivity of an enterprise by positioning services as the primary means through which solution logic is represented in support of the realisation of the strategic goals associated with service-oriented computing. The latter is usually confused with the architecture. It is therefore very important to make a clear distinction between what SOA actually is and how it relates to service-oriented computing elements.



As a form of technology architecture, a SOA implementation can consist of a combination of technologies, products, APIs, supporting infrastructure extensions, and various other parts. The actual face of a deployed service-oriented architecture is unique within each enterprise. However, it is typified by the introduction of new technologies and platforms that specifically support the creation, execution, and evolution of service-oriented solutions (see Ref. [6]).

Since both terms described above are mentioned by SESAR in relation with possible system architectures that should be considered when designing the ATM system of the future they are of special interest, however, their analysis is out of the scope of this project and could be the topic of a different study. Nevertheless, the outlook can be given that issues regarding the architecture of a system of systems, such as ATM, are definitely of highest importance for V1 as they have an enormous impact on the basic operational requirements.

Other identified terms being related to both operational concept and requirement were identified. Terms related to EUROCAE were:

- High Level ATM Concept
- Principles of Use
- Operational Application Description
- System Description
- Equipment Specification

Terms related to SESAR/Episode-3 were:

- Method of Operation
- Principles of Operation

Finally, terms related to the certification within an EASA-ETSO (European Technical Standard Order) were:

- Community Specification
- Essential Requirement
- Implementation Rules
- Regulatory Requirement

Since all these terms were not identified before the start of the project a deep analysis is certainly out of the scope of this project within the limits of budget and effort specified in the contract. However, a quick evaluation of their meaning and relevance is given in the following.



Unfortunately, none of the additional terms could be found in the referenced sources. The only related term that could be found in the IEEE standard was the term ‘system’. It is described as a ‘collection of components organised to accomplish a specific function or set of functions.

Regarding the ‘High Level ATM Concept’ the notion is that it is closely related to SESAR definition of an operational concept, meaning that it includes a description of the system in its environment and how it is operated, however, not further detailing it down to the level of a concept of operations.

‘Principles of Use’ are related to the basic rules for the use of certain system component. In that regard it is not surprising that the term is used within EUROCAE.

An ‘Operational Application Description’ is related to the previous term as it refers to the description of the operational use of a system, probably containing an operational procedure description. As such, it should be part of a concept of operations.

A ‘System Description’, however, seems to be unrelated to the operational procedures and purely describes the system or system component and its integration in the operational environment from a technical viewpoint.

Finally, an ‘Equipment Specification’ is clearly a synonym for a system specification, though the word ‘equipment’ could go down to the use of a pen.

In ‘Method of Operation’ the word ‘method’ indicates that there is a certain procedural standard closely linked to the system. The term is therefore comparable to the term ‘Principles of Use’. At least, it sounds like a synonym for ‘standard operational procedure for the use of a system’ on a general level.

The same goes for the term ‘Principles of Operation’. Again this indicates that there are basic rules for operation of the system, probably introduced in order to discuss procedures on a higher level.

EASA-ETSO terms are related to certification issues, which usually fall into V4 of the E-OCVM life cycle model. However, it was found that the three terms ‘Community Specification’, ‘Essential Requirement’ and ‘Implementation Rules’ are mainly used within the Single European Sky (SES) Programme in order to describe a process, in which a specification document (the community specification) is used in order to ensure compliance of a system or system component with SES regulations in the form of so-called ‘Essential Requirements’ and ‘Implementation Rules’. EUROCAE describes ‘Community Specifications’ as system specifications which are used by the European Commission, under the framework of the Single European Sky legislation, as a means of ensuring compliance to the legislation. As such, they



implement the ‘Interoperability Regulations’ of the European Commission, as is specified in Ref. [3]:

‘Article 4.1.a of the Interoperability Regulation lays down that Community Specifications may be European Standards for systems or constituents, together with the relevant procedures, drawn up by the European standardisation bodies in co-operation with EUROCAE, on a mandate from the Commission in accordance with Article 6. 4 of Directive 98/34/EC of the European Parliament and of the Council of 22-Jun-1998 laying down a procedure for the provision of information in the field of technical standards and regulations and pursuant to the general guidelines on co-operation between the Commission and the standardisation bodies signed on 13-Nov-1984.’

Further it is found in Ref. [3]:

‘Article 4.2 of the Interoperability Regulation lays down that compliance with the Essential Requirements and/or Implementing Rules for interoperability shall be presumed for systems, together with the associated procedures, or constituents that meet the relevant Community Specifications and whose reference numbers have been published in the Official Journal of the European Union.’

These statements give the context of the terms ‘Essential Requirements’ and ‘Implementation Rules’ and also give a notion of what is meant by ‘Regulatory Requirement’. The latter poses a constraint on a system that is developed in the context of SES based on the Interoperability Regulations of the European Commission. The system has to comply with what is called a ‘Community Specification’ that is developed by the European standardisation bodies together with EUROCAE. A community specification contains ‘Implementation Rules’ (IRs) and ‘Essential Requirements’ (ER) that must be met in order to comply with the specification.

In Ref. [3] the term ER is used more or less interchangeably with the term IR which leads to the conclusion that the difference is merely in the way they are formulated. A number of compulsory ERs are also given in this reference. They are:

- Seamless Operation
- Safety
- Civil-military Co-ordination,
- Support of New Concepts of Operation
- Environmental Constraints
- Principles Governing the Logical Architecture
- Principles Governing the Construction of Systems



Such regulatory requirements can thus be seen as ‘Organisational Requirements’ as mentioned in Chapter 2.1.2 of this document. Analysing their contents, however, would be out of the scope of this investigation.



## 2.2 Survey of Relevant Systems Engineering Standards

The scope of the E-OCVM (phase V1 to V3) corresponds with the concept phase and first part of the requirements phase of a software lifecycle as defined in the IEEE standards (cf. 610, 12207, Ref. [24] and [28]). Therefore, further investigation of the standards has concentrated on these parts of the lifecycle.

Taking this into account, the following standards are of particular interest to this research activity:

- IEEE 1362 guide for IT system definition and ConOps (Ref. [27])
- IEEE 830 on software requirement specifications (Ref. [25])
- IEEE 1233 on system requirements specifications (Ref. [26])
- IEEE 15288 on systems engineering (Ref. [29])
- ESA ECSS-E-10 on systems engineering (Ref. [11])
- ESA ECSS-E-40 on software engineering (Ref. [12])
- EUROCAE ED-12B on airborne software engineering (Ref. [7])
- EUROCAE ED-79 on airborne systems engineering (Ref. [9])
- EUROCAE ED-80 on airborne hardware engineering (Ref. [10])
- EUROCAE ED-78A on guidelines for the regulatory framework of provision and use of ATS supported by data communication (Ref. [8])

Main aspects regarding requirements are:

- Standardised hierarchy in requirement documents
- Standardised structure of requirement documents
- Standardised attributes of requirements
- Rules on wording of requirements
- Standard process for requirements management and development
- Requirements traceability
- Requirements integrity
- Requirements reporting
- Requirements reviews

Main aspects regarding validation are:

- Purpose is to check product against stakeholder requirements
- Conformance needs to be demonstrated
- Non-conformance parts need to be isolated
- Validation results need to be reported



### **2.2.1 Analysis of Standard IEEE 1362 for IT System Definition and ConOps**

The ConOps approach (IEEE 1362) provides an analysis activity and a document that bridges the gap between the user's needs and visions and the developer's technical specifications. In addition, the ConOps document provides the following:

- A means of describing a user's operational needs without becoming bogged down in detailed technical issues that shall be addressed during the systems analysis activity.
- A mechanism for documenting a system's characteristics and the user's operational needs in a manner that can be verified by the user without requiring any technical knowledge beyond that required to perform normal job functions.
- A place for users to state their desires, visions, and expectations without requiring the provision of quantified, testable specifications. For example, the users could express their need for a 'highly reliable' system, and their reasons for that need, without having to produce a testable reliability requirement. [In this case, the user's need for 'high reliability' might be stated in quantitative terms by the buyer prior to issuing a request for proposal (RFP), or it might be quantified by the developer during requirements analysis. In any case, it is the job of the buyer and/or the developer to quantify users' needs.]
- A mechanism for users and buyer(s) to express thoughts and concerns on possible solution strategies. In some cases, design constraints dictate particular approaches. In other cases, there may be a variety of acceptable solution strategies. The ConOps document allows users and buyer(s) to record design constraints, the rationale for those constraints, and to indicate the range of acceptable solution strategies.

### **2.2.2 Analysis of Standard IEEE 830 on Software Requirement Specifications**

From the IEEE Recommended practice for software requirements specification (IEEE 830), the following points should be considered carefully in the remainder of this research:

- The practice strongly recommends that a Software Requirements Specification (SRS) is jointly prepared by supplier and customer; this would advocate a strong(er) participation of industry in the early stages of validation.
- The practice specifically mentions site adaptation requirements as a section of the document 'that should:
  - Define the requirements for any data or initialization sequences that are specific to a given site, mission, or operational mode (e.g., grid values, safety limits, etc.);
  - Specify the site or mission-related features that should be modified to adapt the software to a particular installation.'

### 2.2.3 Analysis of Standard IEEE 1233 on System Requirement Specifications

The IEEE Guide for developing system requirements specifications (IEEE 1233) focuses more on activities that one would expect in the E-OCVM phases V4 and V5. However, the system requirements specification development process described in this standard seems most appropriate to the type of activities performed in phases V2 and V3 of the E-OCVM:

‘The system requirements development process, in general, interfaces with three external agents - the customer, the environment, and the technical community. Each of the external agents is described in the text below. Figure 2-3 shows the interactions among the various agents necessary to develop a System Requirements Specification.

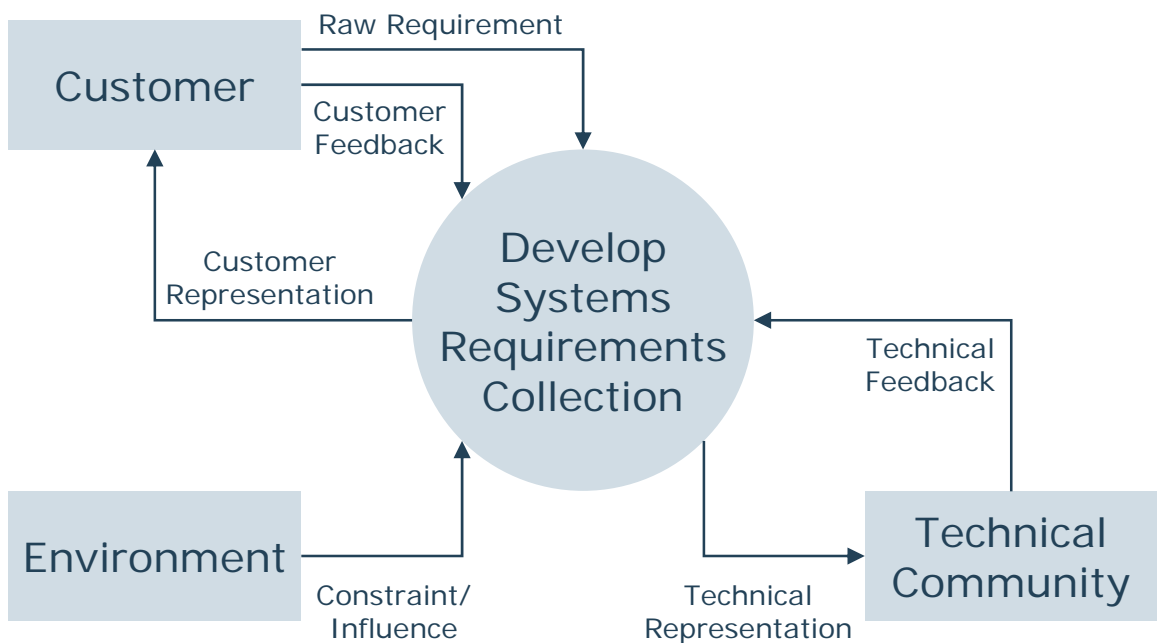


Figure 2-3: Development Process for a System Requirements Specification

Raw requirements: Prior to the SyRS process the customer has an idea for a system, for a process improvement, or for a problem to be solved. At this point, any initial concept for a system may be imprecise and unstructured. Requirements will often be intermingled with ideas and suggestions for potential designs. These raw requirements are often expressed in initiating documents similar to the following:

- a) Concept of operations. This type of document focuses on the goals, objectives, and general desired capabilities of the potential system without indicating how the system will be implemented to actually achieve the goals.



- b) System concept. This type of document includes concept of operations information, but will also include a preliminary interface design for the system and other explicit requirements. ...'

These latter documents are typical results of phases V0 and V1, and therefore the process described in the figure and text above can be 'translated' to map the phases V2 and V3 in which through prototyping and concept/technology development these raw requirements are gradually moulded into a consistent specification.

#### **2.2.4 Analysis of Standard IEEE 15288 on Systems Engineering**

IEEE 15288 (also known as ISO/IEC 15288) provides a common systems engineering process framework covering the life cycle of man-made systems. This life cycle spans the conception of ideas through to the retirement of a system. It provides the processes for acquiring and supplying systems. In addition, this framework provides for the assessment and improvement of the life cycle processes. The processes in this International Standard form a comprehensive set from which an organisation can construct system life cycle models appropriate to its products and services.

The document defines 25 systems engineering processes in four categories: enterprise, agreement, project and technical processes. Two technical processes directly relate to the role of requirements, they are the Stakeholder Requirements Definition Process and the Requirements Analysis Process. The first process is intended to identify the stakeholders, elicit their needs and provide a basis for defining system requirements. The second process is to transform the stakeholder, requirement-driven view of desired services into a technical view of a required product that could deliver those services.

The document provides example product life cycle stages, comprising: concept, development, production, utilisation, support, and retirement stages. They can be considered equivalent to the E-OCVM life cycle phases, but the E-OCVM puts more emphasis on the concept phases and does not cover the three last stages (utilisation, support and retirement). The concept stage can be mapped on steps 0 (need), 1 (scope) and 2 (feasibility) of the E-OCVM life cycle. Step 3 (integration) would map on the development stage.

#### **2.2.5 Analysis of Standard ECSS-E-10/E-40 on Systems/Software Engineering**

The European Space Agency (ESA) maintains a large set of interrelated management, engineering and product assurance standards for space projects and applications. They are very



concise prescriptions written for suppliers that deliver systems or services to ESA and therefore have a clear customer oriented focus. The following three documents are analysed:

- ECSS-E-10 part 1B ‘Requirements and process’
- ECSS-E-10 part 6A ‘Functional and technical specifications’
- ECSS-E-40 part 1B ‘Software - Part 1: Principles and requirements’

When comparing with the E-OCVM, it is clear that the documents do not cover needs, and scoping phases. Because of the clear customer-supplier relationship, the requirements process starts at the systems requirements level. Still, it does provide information on documentation of mission statement, operational concept, and functional specification. The ECSS documents can be used as a reference for requirements related topics such as:

- Traceability
- Wording
- Specification trees
- Documentation
- Baselining

In particular, on the topic of documentation, the standard contains some artefacts that may be useful to this research. One of these is the Design Justification File: ‘During the software requirements and architecture engineering process, the result of all significant trade-offs, feasibility analyses, make-or-buy decisions and supporting technical assessments are documented in a design justification file (DJF).’ The use of a Design Justification File (DJF) could be very useful in phases V2 and V3 of the E-OCVM in order to maintain a history of decisions, and their rationale, during the prototyping and experiment activities. Together with the final set of requirements at the end of phase V3, this DJF could be a deliverable to phase V4 to smoothen this transition (see also recommendations in Ref. [23]).

#### **2.2.6 Analysis of Standard ED-79/ED-12B/ED-80 (Systems/Software/Hardware)**

The EUROCAE guidance on systems (ED-79), software (ED-12B) and hardware (ED-80) is intended for safety critical airborne applications rather than systems engineering in general. However, these two areas have many interrelations and as a result, these documents do cover a significant part of systems engineering in general.

The ED-79 (also known as ARP 4754) is focused on safety assessment in the development of complex systems. It describes a requirements capture process in order to highlight relationships with failure condition classifications and assignment of development assurance levels. It is possible that this provides relevant insights for the relationship between requirements and

E-OCVM phase V2 (feasibility) because this phase entails checking for safety issues. Another relevant part of the ED-79 is the description of how to validate requirements, i.e. checking completeness and correctness. This is detailed with a list of questions that need to be answered when performing the validation.

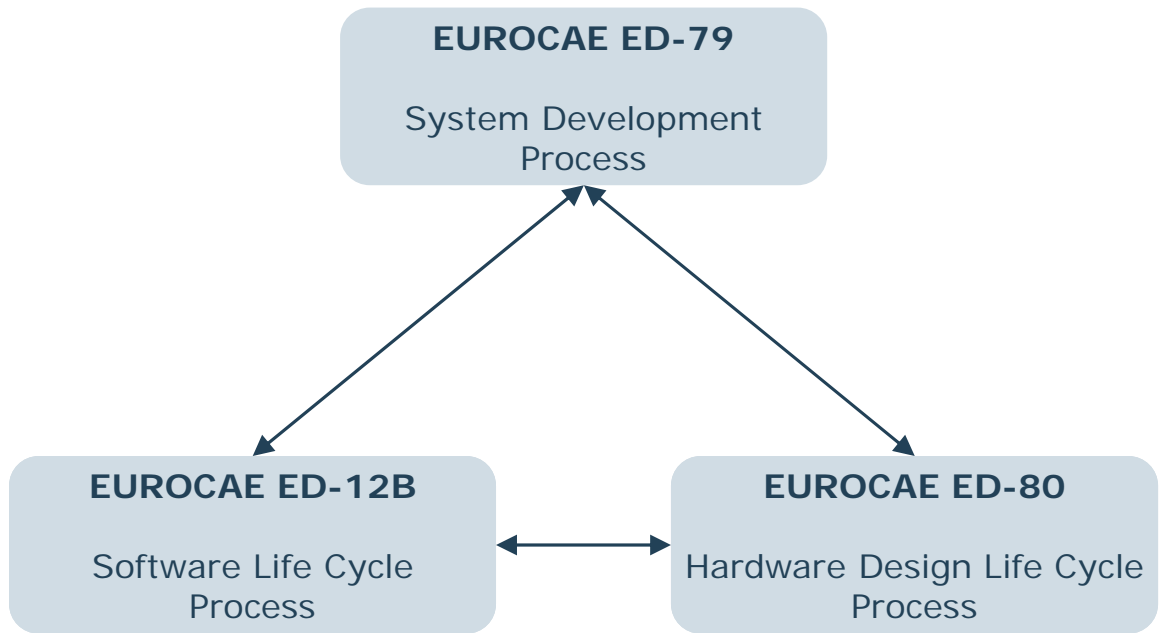


Figure 2-4: Overview of Certification Guidance by EUROCAE

The ED-12B (also known as RTCA DO-178B) provides clear guidance on what a supplier should do when developing airborne safety critical software. It covers various processes of the development life cycle including a software requirements process. It should be noted that it assumes there exists a set of high-level requirements which need to be detailed at the software level. This limits the applicability of the process for RORI-OCV, but it can be considered as a checklist for a more generic requirements process.

The ED-80 (also known as RTCA DO-254) provides clear guidance on what a supplier should do when developing airborne safety critical hardware. It provides a reference hardware development life cycle starting with a requirements capture process. Since it is the hardware equivalent of ED-12B, the same limitations apply. In addition to process descriptions the ED-80 identifies two data items relevant to requirements processes. The first data item is identified as requirements standards that may be used during the requirements capture process to define the



rules, procedures, methods, guidance and criteria for developing the requirements. The second data item are the actual hardware requirements resulting from the requirements capture process.

### **2.2.7 Analysis of Standard ED-78A (Approval of Air Traffic Services)**

The EUROCAE document specifying ‘Guidelines for Approval of the Provision and Use of Air Traffic Services Supported by Data Communications’ is one of the most frequently referenced documents in European Commission projects and EUROCONTROL initiatives that deal with the introduction of communication technology for ATM (e.g. AFAS, CASCADE). The reason for this lies in the fact that this guidance material describes a regulatory process for the approval of such technology, so that the application of ED-78A in research and development projects is seen as a prerequisite for a successful continuation of the life cycle in phases V4 and V5.

In general, ED-78A describes a structure of standardised documents that are required for the approval process. The document mainly has been written as a guideline for ‘stakeholders and approval authorities involved in the operational implementation of the provision and use of ATS supported by data communications’, such as ‘ATS providers, ATS equipment manufacturers, supporting service providers’, and ‘aircraft and equipment manufacturers and operators’. This means that, at first glance, there is no direct link to concept validation, especially as approval processes for technology relate to later phases in the life cycle. However, since both concept development and technology development processes are interwoven with the concept validation process, these guidelines also have an impact on concept validation activities.

Apart from describing a regulatory framework the purpose of ED-78A is to ‘recommend minimum acceptable criteria for approval’ as related to aircraft design, operator operations, and ATS provider operations. The criteria are described the form of ‘process objectives and guidance for evidence’, such as applied standards or ‘results of verification’. Again this shows that ED-78A is closely related to technology development and associated verification activities are seen as a main source for finding evidence.

Part of the scope of ED-78A is described as the provision of ‘means to establish the operational, safety, performance, and interoperability requirements for ATS supported by data communications, to assess their validity, and to qualify the related CNS/ATM system’. With this definition ED-78A puts itself at the core of the requirement development process, albeit on the rather low level of system component development.

In order to understand the complete scope of ED-78A, it is useful to have a look at the different processes described in the document. They are:



- Approval Process
- Entry-into-Service
- Operations
- Co-ordinated Requirements Determination

The approval process is one of the basic elements in the guidelines and is of less interest for E-OCVM phases V1 to V3. Nevertheless, ED-78A stresses that especially the processes for co-ordinated requirements determination (or requirement capturing) and the technical and operational qualification of evidence (or validation and verification) eventually play a major role in that process and that there are several feedback loops between the processes before the approval process is finished. The feedback loops make the approval process (and essentially all processes of life cycle phases V4 and V5) the overall governing process, in which concept and technology development as well as verification and validation activities take place. This is shown in Figure 2-5, which is a simplified version of a figure in ED-78A (see also Appendix B).

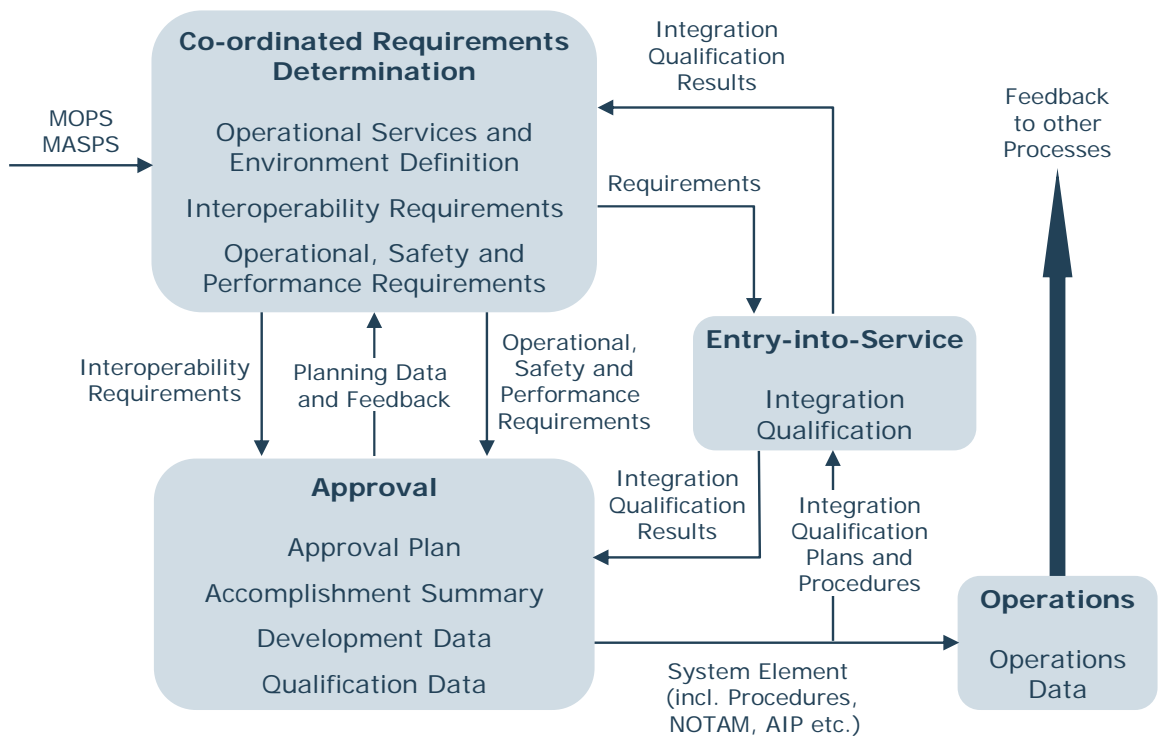


Figure 2-5: Relationship of Processes in ED-78A

A closer look will be taken at the process for co-ordinated requirements determination (CRD) and its relation with the processes for qualification of evidence. According to ED-78A the CRD process consists of:

- Operational Services and Environment Information Capture (OSEIC)
- Operational Safety Assessment (OSA)
- Operational Performance Assessment (OPA)
- Interoperability Assessment (IA)

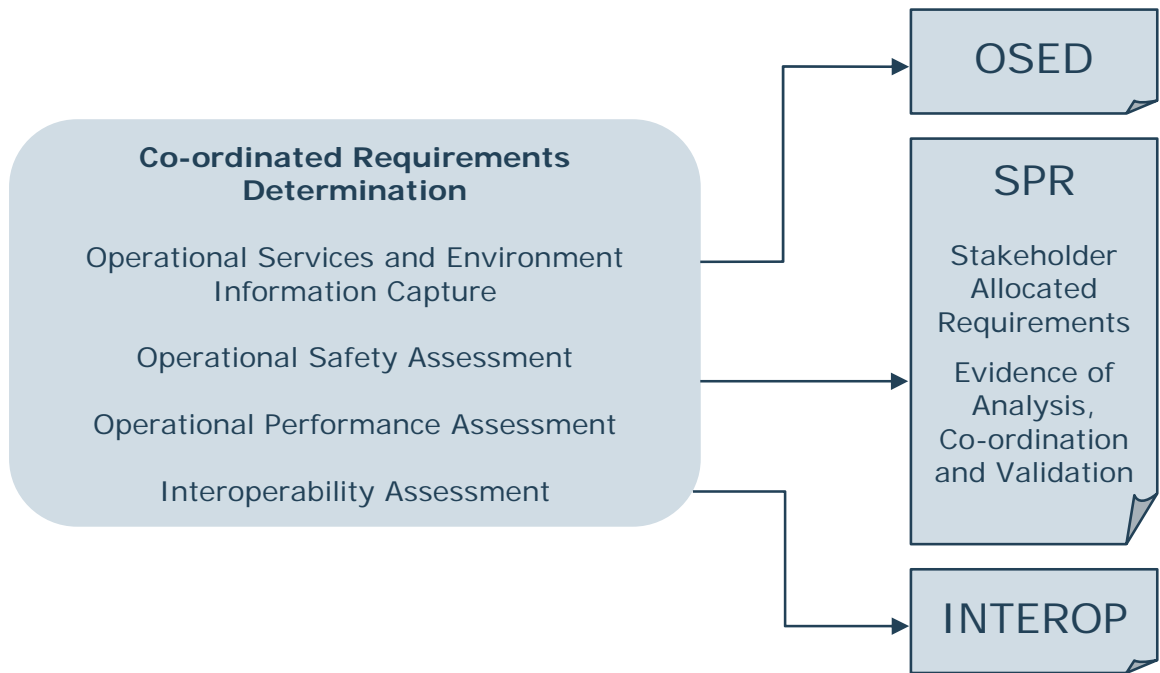


Figure 2-6: Output of CRD Process

The OSEIC leads to the production of a so-called Operational Services and Environment Definition (OSED), which captures service descriptions, including operational processes, operational performance expectations, selected technologies, and characteristics of the intended operational environments (Figure 2-6). This means that there is a close relation between the definition of an OSED and an OCD or a ConOps (cf. Section 2.1.1).

The OSA, OPA and IA identify, co-ordinate, allocate, and validate the operational, safety, performance and interoperability requirements, and update the OSED, as necessary. Operational, safety, and performance requirements are captured in the Operational, Safety, and Performance Requirements (SPR) standard and interoperability requirements are captured in the Interoperability Requirements (INTEROP) standard (Figure 2-6).



Generic activities defined for the OSA, OPA, and IA tasks are:

- Identification of Requirements
- Co-ordination of Requirements
- Allocation of Requirements
- Validation of Requirements

An interesting remark in ED-78A regarding both SPR and INTEROP standards is that the requirements they contain are validated prior to the publication of the standards, meaning that the results of the generic activities above need to be contained in the final versions of these documents.

What is important with respect to the E-OCVM and the previously defined terminology is that the SPR definition in ED-78A refers to the ‘objectives and allocated requirements’ for a ‘specific operation’ while the INTEROP definition refers to ‘technical, interface, and related functional requirements for a specific technology’. In that sense, a clear distinction is made between operational and technical aspects, and separate processes are dedicated to capturing the requirements from the OSED. When looking at the requirements capturing process as part of a case-building activity, this distinction might not be as obvious. For the specific case of ATS data communication ED-78A considers this a valid approach, though.

The generic requirement determination processes (for operation, safety, performance and interoperability) described in ED-78A are seen as co-ordinated processes under participation of all stakeholders, so that their initial result may be on a high level. The processes are further broken down in ED-78A by specifying Development and Qualification processes ‘ that are performed by each stakeholder to ensure that their element of the CNS/ATM is produced with assurance that operational, safety, performance and operability objectives and requirements from the SPR and INTEROP standards are met’. Thus, the document again makes a clear distinction, this time between validation of objectives and requirements on a high-level in the CRD process and verification and validation of a subset of requirements at a system component level in the development and qualification processes.

Regarding the subject of validation as part of the qualification process (Chapter 5.2 of ED-78A), which also entails verification, process assurance and configuration management, some guidance is given. It is noted that the validation objectives to be met concern validation of requirements produced at the organisational level, which points to stakeholder related cases, validation of a risk mitigation strategy, and validation of HMI requirements. Much more is said about verification and the tests for requirements to be performed. Considering the use of the



term validation in the document, which is defined as examining whether requirements for a specific use are correct, it can indeed be assumed that validation is seen as use case related testing of the operation. Emphasis is put on HMI validation while safety aspects, such as a system element safety assessment is seen as a verification activity.

In conclusion, it should be mentioned that the processes described in ED-78A are very complex and no clear description of the validation processes in life cycle phases V2 and V3 that would lead to a refinement of requirements is given. Instead the ATS system component under consideration is broken down into stakeholder relevant sub-components that each have to comply with higher-level requirements as must be shown in a qualification process involving verification and validation. Nevertheless, a number of processes and generic tasks are described that could prove very useful for an understanding of the requirements capturing and developing process. For a deeper understanding of ED-78A and the exact definition of processes and other standard elements the reader is referred to Appendix B.11.

### **2.2.8 Development Life Cycles**

This section provides background information on development life cycles. Most of the system/software engineering standards described before do not prescribe or exclude any particular development life cycle. However, a survey of terms and standards for system engineering cannot be considered complete without mentioning development life cycle models, since most of the standards usually contain some kind of guideline for mapping the requirements of the standard onto popular life cycle models.

There are many approaches to developing complex systems. In order to appreciate the differences between these approaches it is necessary to understand the interrelations between the development processes. Development life cycles are characterised by the way they organise the interactions between technical development processes. Development life cycles have evolved over time because people adapted them to solve problems they encountered with existing life cycles. Three basic types of life cycle models may be distinguished:

- Waterfall Model
- V-Model
- Spiral Model

A waterfall model (Figure 2-7) defines a sequential development approach. The number of steps may differ, but in general the sequence contains the following processes: analysis, design, implementation, and test. In some cases it is preceded by a process such as feasibility analysis or requirements capture, and in some cases it is followed by a process such as maintenance or deployment.

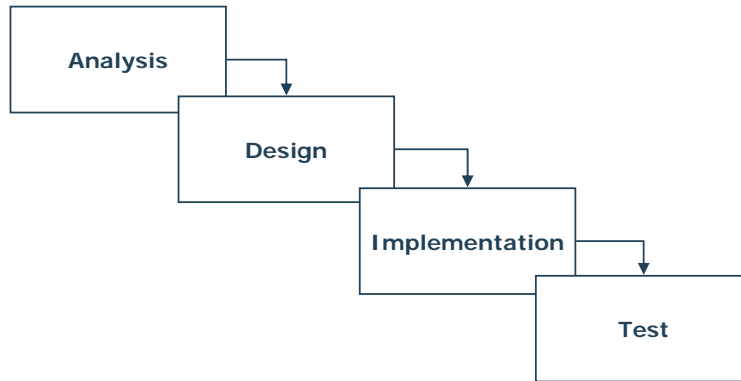


Figure 2-7: Development Life Cycle - Waterfall Model

There are several weaknesses of this approach. First, there is a risk that problems are discovered late in the development, because the waterfall model only considers integration and test issues at the end of the life cycle. Secondly, the design has to be finalised before specific implementation problems are discovered. Coping with these problems late in the development is likely to incur delays and increase cost (e.g. redesign, requirements change). Therefore, only for projects with limited risk in terms of development and final product acceptance the waterfall model can be used, e.g. routine development of conventional solutions.

Variations on the waterfall model exist, such as the incremental model where several builds of the product are developed.

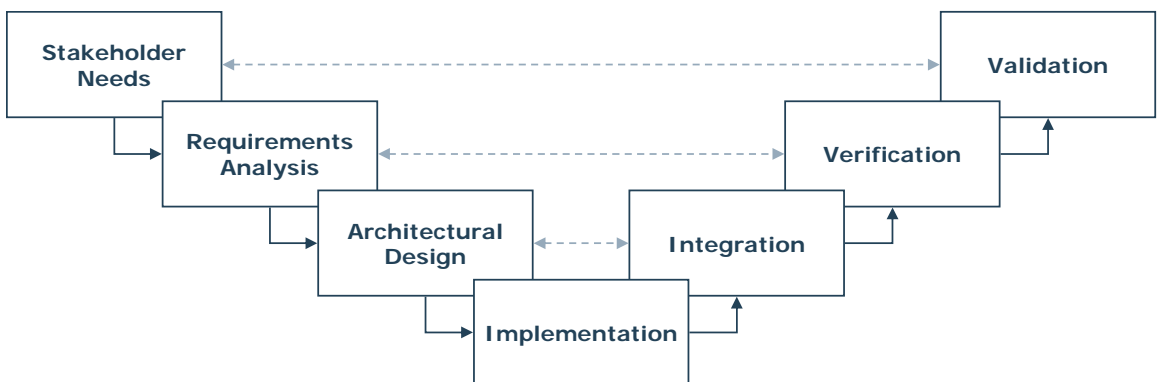


Figure 2-8: Development Life Cycle - V-Model

A V-model (Figure 2-8) defines a development approach that connects design processes with verification processes. At the start of the project, the stakeholder needs are identified to ensure



that the final product is valid for its intended environment and that the product will meet the stakeholder expectations. Early in the project it is verified that requirements are testable, and test requirements are written to prepare for verification. The product is designed taking integration aspects into account and it is integrated in accordance with its design.

The V-model can be viewed as a modified waterfall model, because it basically contains the same processes and only adds relationships. In this way it tries to discover problems early in the project when it is less costly to solve them. A weakness of this model is that it does not provide guidance on how to deal with unforeseen properties of the product.

Variations on the V-model exist, such as the W-model where additional focus is on improving testing activities.

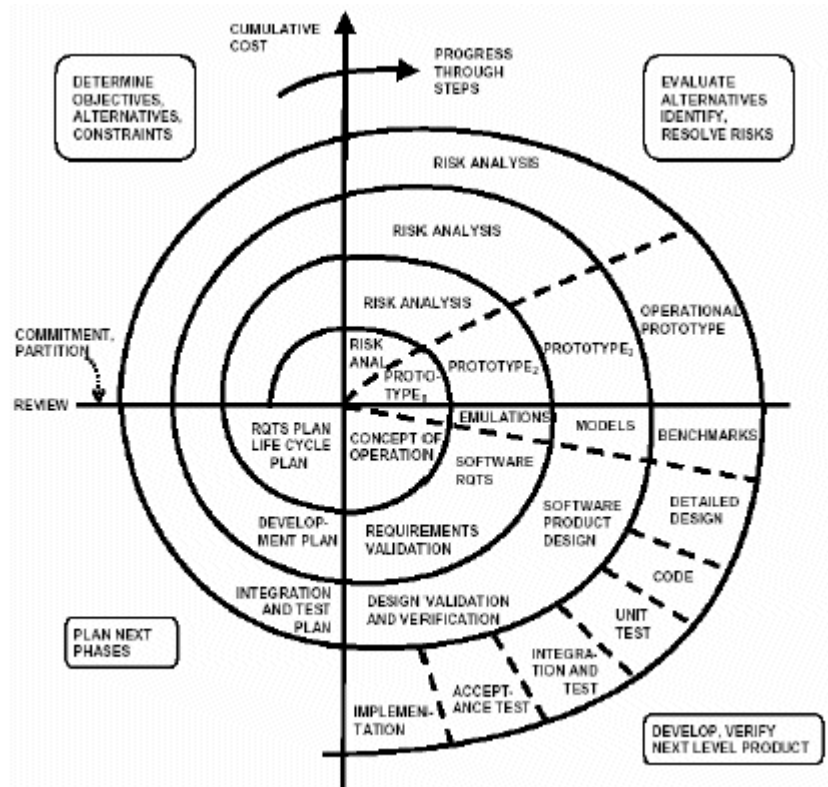


Figure 2-9: Development Life Cycle - Spiral Model

A spiral model (Figure 2-9) defines an iterative development approach. Each of the iterations deliver a prototype of the product of increasing maturity by performing activities such as defining objectives, analysing risks, planning the development, and modelling the product. After the final prototype is ready, the real product is developed. This approach allows the customer or

stakeholders to provide feedback early in the project. The original spiral model as shown in Figure 2-9 has been developed by Boehm (cf. Ref. [2]).

Variations on the spiral model exist, emphasising different aspects and tailoring it to different processes. Figure 2-10 shows how a spiral approach can be applied to a V-model resulting in four iterations. They are:

1. Study:  
The stakeholder requirements are analysed and a consolidated set of requirements is validated by means of a review by the stakeholders.
2. Concept:  
A product architecture is developed together with a simulation model of the product. Using this model the requirements are verified and the overall concept is validated.
3. Prototyping:  
A detailed design is developed together with a prototype of the product. Using the prototype the architecture is verified and the detailed design is validated.
4. Realisation:  
The real product and its documentation is developed using the previous baselines of the requirements, architecture and design.

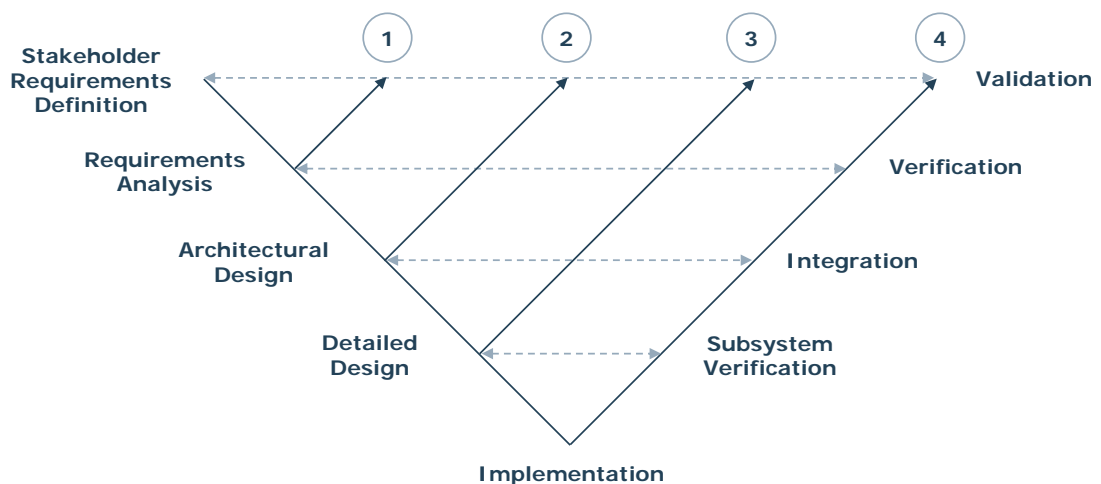


Figure 2-10: Spiral Approach applied to a V-Model



### **2.3 Conclusions on the Survey of Terminology and Relevant Standards**

One of the main outcomes of the survey of terminology is that there are many terms for similar, but not always identical, artefacts or activities that are often tailored to a specific problem domain or organisation (see especially Section 2.1.3). Thus, in most cases it will be necessary to ask the specific user of such a term or the related organisation for a clear definition.

Nevertheless, the survey of terminology tried to concentrate on the commonalities in definitions and therefore a number of general conclusions about the use of concept and requirement related terms can be drawn.

Regarding the definition of an operational concept it is useful to distinguish between high-level concept documents that give a high-level description of ATM services and environment and Concepts of Operation or Use that provide more detail on user needs and requirements, the way that the system parts are operated, organisational issues, functions and processes, interactions and information flows, involved actors and their roles and responsibilities. Operational procedures are defined to consist of abovementioned rules, regulations, processes and working practices.

Regarding requirement-related definitions there is a basic distinction between stakeholder needs (sometimes also called user requirements) and requirements.

Stakeholder needs may be inconsistent, technically or financially impossible, and may contain implementation details, design decisions or any other kind of statement on the system or function to be developed. Stakeholder needs are input to the requirements analysis process. They are sometimes called 'raw requirements' and are often documented in a Concept of Operations (ConOps) or a similar document (OCD, OSED, COU).

The first activity in a requirements analysis process is to transform the identified needs into requirements. These requirements must be clear and consistent, without unnecessarily limiting the possible set of design solutions. The complete set of (validated) requirements is the basic part of the system or software specification, and is usually documented in a system or software requirements specification document. The complete specification should also contain information on functional aspects, design, behaviour, and other characteristics of the system or software together with the necessary tests that have to be carried out for verification of the requirements.

Requirements are often divided in categories based on system-external distinguishable properties, such as origin (user requirements, environment requirements, interface requirements, regulatory requirements etc.), or based on system-internal distinguishable properties, such as functional/non-functional requirements, performance requirements, quality requirements, security requirements etc.



Although the specifications available at the end of phase V3 should consolidate the requirements and the validation results, it is not quite clear what the required level of detail actually is. After all, they are still research specifications that might or might not be in line with technology development. Therefore, an additional aspect in the R&D phases is the relationship between customers and suppliers of the technology concerned and the kind of technology being developed (component, sub-system, system, system of systems, enabler etc.).

Considering the additionally identified terminology on system architectures, it is interesting to notice that none of the terminology survey references or standards is explicit about them, although the topic is currently heavily debated within SESAR. Additional standards for architectures could be referenced in order to obtain a clearer picture of their role in operational concept validation, yet, such an analysis is out of the scope of this study. It is expected, though, that issues regarding the architecture of a system of systems, such as ATM, are of highest importance for V1 as they have an enormous impact on the basic operational requirements.

The proposed general requirement development process can be divided into two major processes:

- Requirements Elicitation or Capture Process
- Requirements Analysis or Specification Process

While raw requirements are the output of the capture process, the output of the analysis phase is a specification with (validated) requirements for the system or software. Thus, in general, these two requirement engineering processes govern the operational concept validation life cycle from phase V1 through phase V3 (see Figure 2-2).

A useful representation of the general requirement development process is given in IEEE Standard 1233. This representation is depicted in Figure 2-3. Since all standards that were part of the survey put emphasis on the development of technology, necessary concept development and validation activities are not directly included in any of the process representations.

Therefore, it seems to be necessary to at least include the concept development community as an important actor in the process. It would also be possible to include validation experts at this point to also make a distinction between feedback obtained from the development of the concept and its gradual validation. Both processes are certainly closely related. Eventually, this would result in a representation as shown in Figure 2-11.

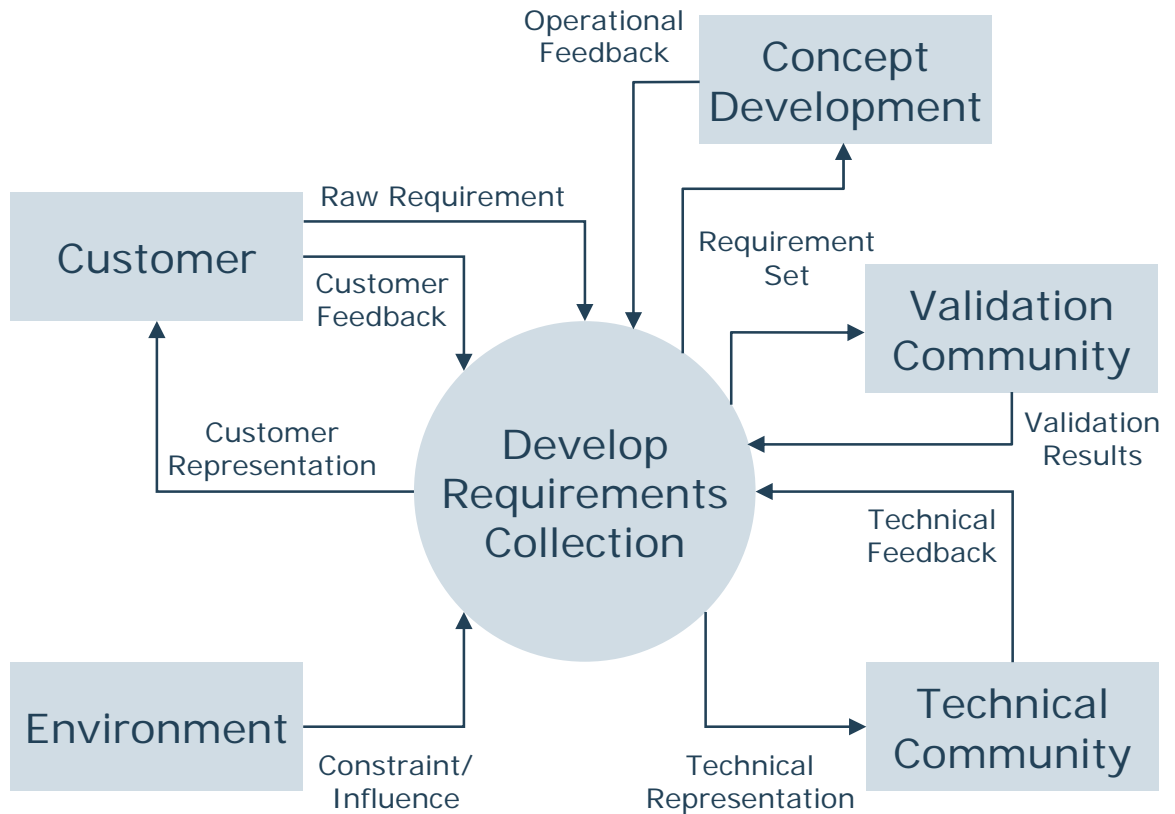


Figure 2-11: General Requirement Development Process

The following sections in this document will discuss the operational concept validation life cycle as described in the E-OCVM (Ref. [15]) and will look at integrating the requirement development processes with that life cycle.



### 3 Requirement Development Strategy in the OCV Life Cycle

This chapter discusses the operational concept validation life cycle as described in the European Operational Concept Validation Methodology (E-OCVM) [15] and describes a consolidated requirement development strategy, based on the survey of terminology and standards from the previous section, within that life cycle. An important aspect in describing an appropriate strategy will be the identification of relevant actors and their roles and responsibilities and the identification of complementary processes in the related domains of operational concept development and technology development. Furthermore, a common and useful approach is to separate activities associated with the capturing of stakeholder needs as requirements and activities associated with the analysis and further development of these requirements. Responsibilities and activities related to the administration of requirements are commonly grouped into a single process as well.

Therefore, the following process structure for a requirement development strategy is proposed:

- Requirements Capture
- Requirements Analysis
- Requirements Management

These processes will be elaborated in subsequent sections of this chapter.

#### 3.1 Operational Concept Validation Life Cycle

The E-OCVM in its current form describes three aspects of validation that in their entirety provide a structured, iterative and incremental approach to operational concept validation. While the Structured Planning Framework, which is based on the MAEVA Validation Guideline Handbook (Ref. [34]), facilitates planning issues for R&D programmes, the Concept Life Cycle Model (Figure 3-1) gives an overview of the complete life cycle of an ATM R&D operational concept and indicates the type of validation activities necessary within those phases that represent the main focus of ATM R&D.

As such the model can also be complemented with a view on how concept validation interfaces with technology or product development and concept development, the latter being very closely related with the different validation life cycle phases. It also offers the possibility to have a closer look at how the third aspect, the Case-based Approach, relates to the other processes. This approach integrates validation results into key cases that directly address stakeholder issues about ATM system performance and behaviour. An investigation on best practices for this approach is part of the European Commission project Co-operative Approach to ATS (CAATS).

The present study looks at the current E-OCVM life cycle model and tries to incorporate abovementioned requirements engineering processes into the model. At the same time it tries to make use of the best practices of case-based approaches concerning a requirement development strategy.

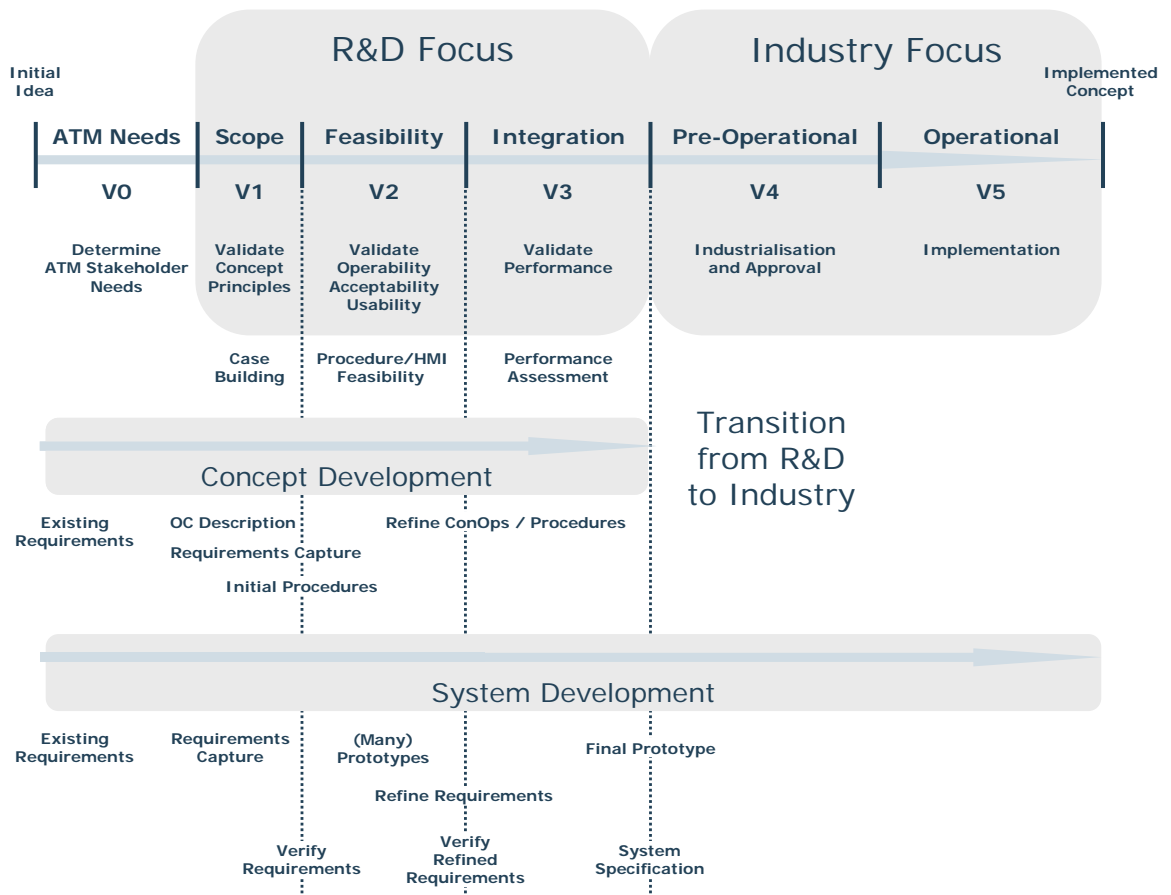


Figure 3-1: Operational Concept Validation Life Cycle Model (E-OCVM)

The OCV life cycle, as described in the E-OCVM, consists of the following six phases:

- ATM Needs: Identification of ATM performance needs and barriers
- Scope: Identification of possible benefit mechanisms through concept description
- Feasibility: Development and exploration of concept until it is operationally feasible
- Integration: Integration of required functionality into pre-industrial prototypes
- Pre-Operational: Transformation of prototypes into industrial products
- Operational: Implementation of products and operational procedures



The topic of this chapter, the incorporation of a requirement development strategy into the E-OCVM life cycle model, covers the first four phases of the life cycle. Stakeholder needs are identified in V0 as part of a requirement capture process. The needs are used as input for the requirement development process that takes place within the R&D related phases of the life cycle, i.e. from V1 to V3. The requirement development process ends with a consolidated set of requirements in form of a specification in V3. Throughout the considered life cycle phases a requirement management process takes place that has to maintain requirements and help identifying possible inconsistencies.

A similar approach to detailing activities has been taken in the case-related best practices of the CAATS project (Ref. [4]). For the safety aspect four different main categories were identified, namely Safety Regulation, Safety R&D and Safety Assessment Methodologies, and Safety Management. In each of these categories activities with a different scope and different objectives take place. Safety regulation looks at the necessary requirements that must be met in order to comply with the specifications of a certain regulator and therefore can be regarded as a sub-process of the requirement capture process. Safety R&D supports the development of new ATM concepts and technology by developing new ideas, standards and methodologies for safety validation, and safety assessment methodologies offer existing strategies for validating safety of elaborated systems and procedures or any other proposed change in operation. Thus, the last two categories are part of the requirement refinement process that takes place in phases V1 to V3. Finally, safety management is a continuing parallel process that must ensure coherency and traceability of all safety-related activities. Therefore, this process compares well with the general requirements management process.

Human factors activities described in CAATS loosely fit in such a categorisation approach as well. Human-machine interaction and failure recovery experiments assess the elaborated systems and procedures regarding safety aspects (overlap with safety activities), and especially usability and acceptability (thus feasibility). Other activities focus on actors, communication, procedures, roles, responsibilities, and training, i.e. requirements that have to do with organisational issues.

In summary, the CAATS best practice categories can all be illustrated as being part of the requirements capture, analysis, and management processes. Thus, the processes sketched in the requirement development strategy should be seen as a framework that stakeholder-specific cases can be built on.

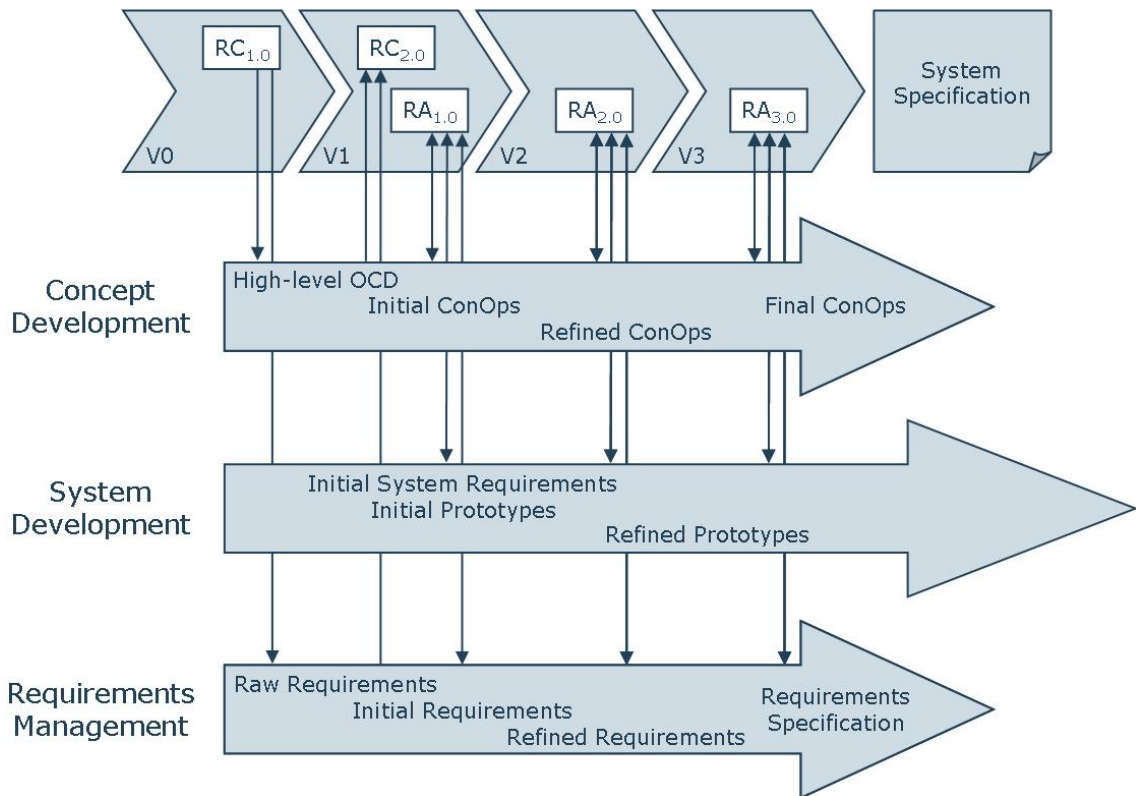


Figure 3-2: Overview of Proposed Processes for a Requirement Development Strategy

In order to give a first impression of the processes associated with the requirement development strategy, Figure 3-2 presents a high-level view of the Requirements Capture (RC), Requirements Analysis (RA), and Requirements Management (RM) processes within the E-OCVM lifecycle.

In phase V0, a first requirements capture process will be performed (RC1.0) to gather the initial, high-level stakeholder needs and to initiate communication with the concept and technology development teams.

At the start of phase V1, a second iteration of requirements capture (RC2.0) is performed to include early feedback from concept and technology development and to complete the requirements capture process. At this point in time, a complete list of needs or raw requirements will be available. This list is both input to the first iteration of requirements analysis (RA1.0), and requirements management. During the first iteration of the requirements analysis, interaction with concept and technology development will take place in order to scope the operational concept and to build the various stakeholder-related cases. At the end of this iteration, a set of requirements consistent with the scope of the operational concept and the various cases will be available as input to both the mainstream activities and the concept and technology development streams. Because this interaction is important for the quality of the end

product, a central role (e.g. a Programme Manager or a Development Strategy Manager) is highly recommended. This person should make sure that the information between the different work streams is shared at the appropriate time, and, ideally, that the different work streams are executed synchronously, e.g. at predefined and pre-planned synchronisation points within an overall development strategy.

In phase V2, the concept is iteratively developed and evaluated. Each of the iterations of the mainstream activity consists of a requirements analysis activity and an according requirements management activity, which is not elaborated in Figure 3-2, though, because of the continuous character of the management activity. Figure 3-3 shows the iterations in the requirement analysis processes in phase V2. It also demonstrates that there might be many concepts and systems to be considered in the beginning, but that the selection process that mainly will happen in phase V2 (although it cannot be excluded for the other phases of the life cycle) will eventually lead to a reduced number of solutions.

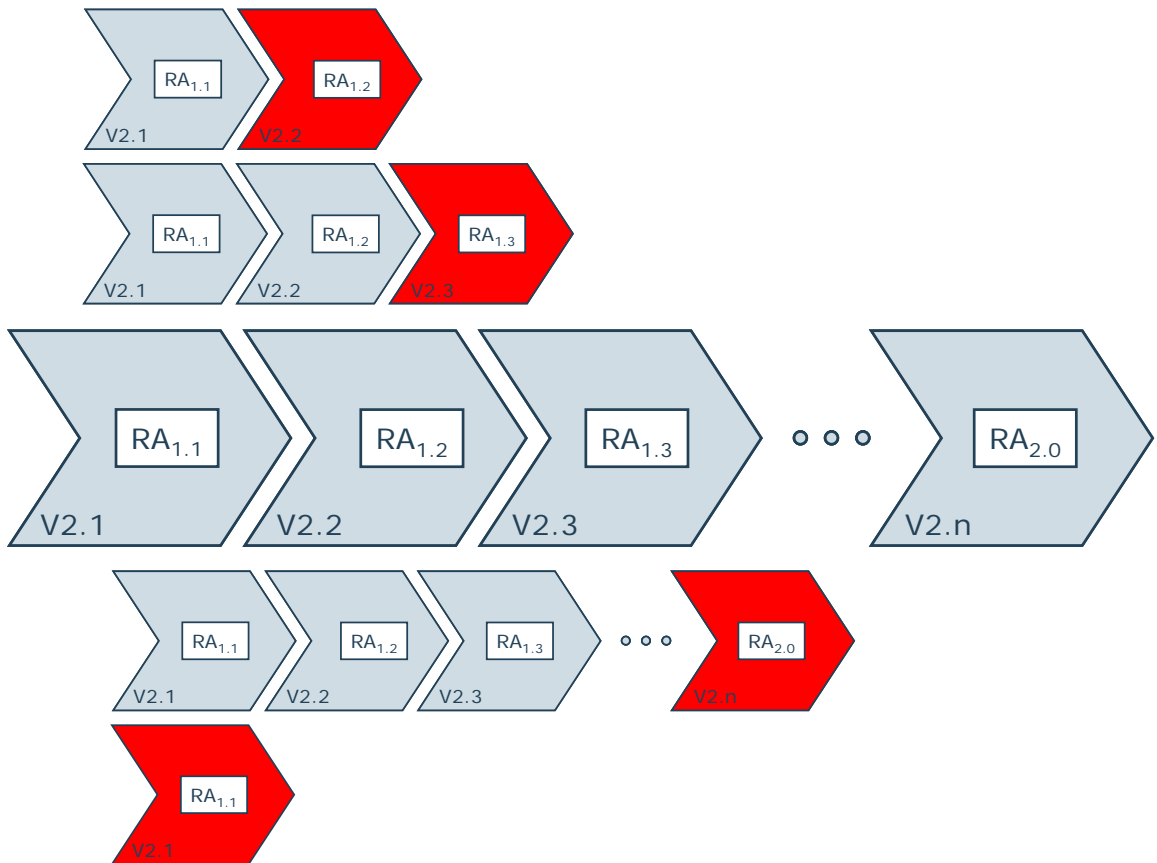


Figure 3-3: Selection Process in Phase V2 of the Life Cycle



The final iteration ends with an execution of the requirements analysis (RA2.0) and requirements management process to wrap up the phase V2 changes in requirements, and to make a clear transfer of requirements to the next phase, and to the next activities in the concept and technology development streams.

In phase V3, similar to phase V2, for each of the iterations of building, consolidating and testing, a (small) requirements analysis and requirements management activity must be performed to ensure that late insights or developments are reflected in the requirements. At the end of phase V3, one last (major) requirements analysis (RA3.0) and requirements management activity must be performed to check and consolidate the final list of requirements and accompanying documentation. The output of this final activity will then be the system specification, which among others contains the final requirements specification. The exact level of detail of such a specification depends on various issues, such as stakeholder involvement, the kind of technology being developed and the conditions for transfer to phase V4.

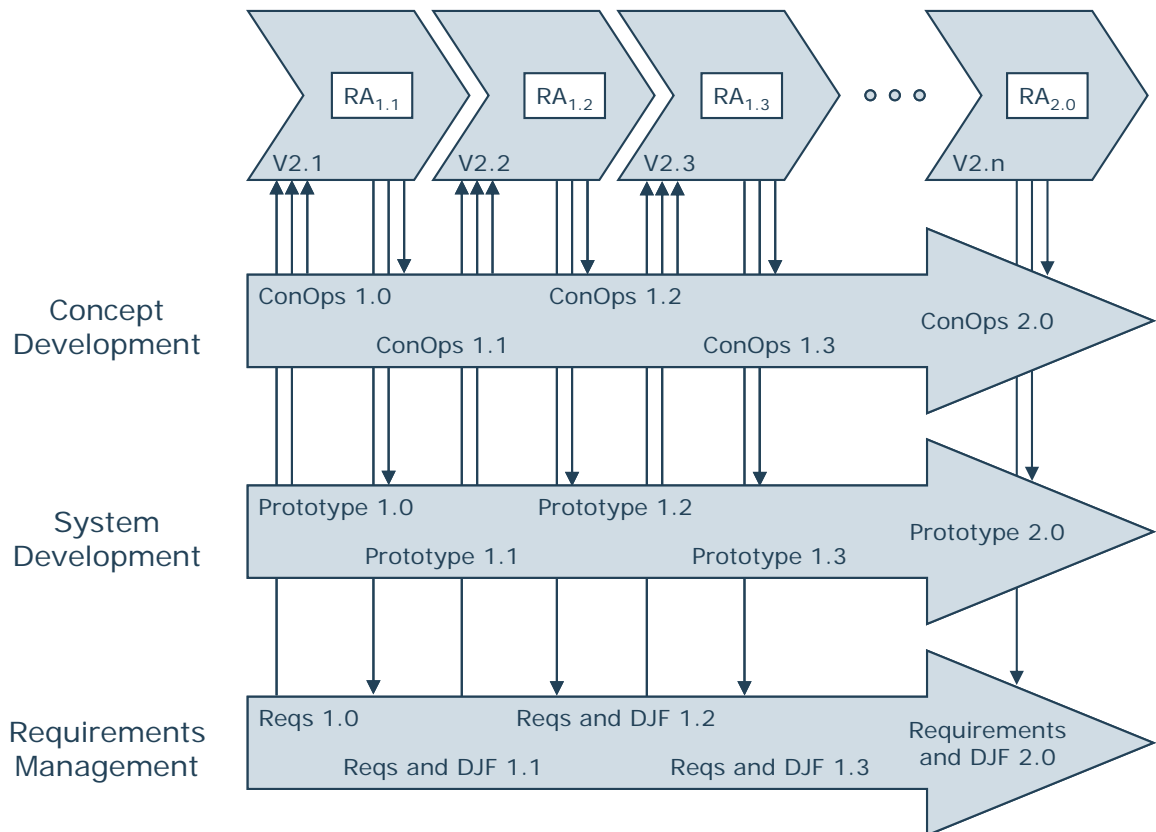


Figure 3-4: Iterative Processes in Phase V2 of the Life Cycle



Phase V2 in the E-OCVM lifecycle is often an iterative phase in which the concept is developed and evaluated in several steps. Figure 3-4 shows, in more detail, the relations between the E-OCVM life cycle with concept and technology development, and the proposed requirements processes during this phase. The starting points are the initial ConOps, prototypes and requirements coming out of phase V1. These starting points should be labelled version 1.0. They are validated in a first iteration in phase V2 (V2.1) and the impact of the results of the validation on the requirements will be analysed in an execution of the requirements analysis process (RA1.1). The output of this process is fed back to the other streams resulting in updates and refinements of the ConOps, prototypes, and requirements (which should be labelled version 1.1). A Design Justification File (also labelled version 1.1) will be added to the requirements document to capture all decisions and their rationale made during this iteration. These version 1.1 artefacts are input to the next iteration, which follows the same workflow, resulting in versions 1.2 of all artefacts. This workflow may be repeated as often as necessary. Finally, after the last iteration of phase V2, the final versions of all artefacts are labelled version 2.0 and are transferred to Phase V3.



### 3.2 Actors, Roles and Responsibilities in the OCV Process

As indicated in the previous sections of this document, the general processes in the requirement development strategy are related to concept development, technology development, requirements management and the validation mainstream from phases V0 to V3. This separation insinuates that there are related actors with certain roles and responsibilities for each of these processes. Indeed, from an R&D standpoint, these are the usual actors directly involved in the validation process. Additionally, the VARTAN study (Ref. [38]) suggests that there are different levels of detail and managerial responsibility within each of the phases of the life cycle which provide a structure to R&D validation activities initiated by the European Commission and EUROCONTROL. It would stretch the scope of the present study to go into the details of the VARTAN project. However, the general assumption is that responsible actors can be identified for making decisions on a strategy level (move from one life cycle phase to the next), on a programme level (preferably addressing particular key performance areas), on a project level (focusing on a confined set of activities to find evidence for improvements in a certain key performance area), and on a validation exercise level (focusing on the conducting of a particular validation exercise). In order to simplify this view, the current strategy assumes that there is at least one validation manager who has programme manager capabilities and negotiates with stakeholders on the progress along the life cycle phases. At the lower project and exercise levels this validation manager is supported by members of the validation team who fill in the structure of the VARTAN study.

Thus, the following actors, which are seen as internal participants to the OCV process, are considered in the suggested requirement development strategy:

- Validation Experts (with the Validation Manager):  
The validation manager is responsible for carrying out a validation programme focussing on finding evidence of improvements in one or more key performance areas. The validation manager negotiates with decision makers (stakeholders) on the status of a validation programme and discusses possible ways to progress from one life cycle phase to the next. The validation manager is supported by a validation team consisting of various validation and simulation experts and also a number of dedicated experts for key performance areas. The team needs to carry out the validation plan in order to find evidence for hypotheses that postulate a certain improvement in a key performance area effectuated by a development in both operational concept and applied technology. Training will be an important element of the tasks of the validation team. In the same way that technology will be verified before a validation exercise can take place, it will be necessary to verify that the operational team involved in the validation exercises is capable of understanding and executing their roles and tasks appropriately.



- **Requirement Development Experts (with the Requirement Development Manager):**  
Since the elaboration and development of requirements play a major role in the validation process and the transition from one life cycle to another, the validation manager should work together very closely with the actor responsible for requirements management. Ideally, it should be the same person. The requirement development team should consist of dedicated experts in any of the considered key performance areas (case-builders) and participants from both technical and concept development teams (e.g. the respective team manager). The team will capture ATM needs (e.g. in brainstorming sessions with stakeholders) and formulate them as raw requirements. Feedback from both concept and technology development should lead to a better understanding of how to transform these raw requirements into operational requirements and related organisational requirements. In the course of the validation mainstream activities requirements are further analysed and refined based on the validation results obtained. The final set of requirements will be documented by the requirement development team in the requirement specification which is part of a system specification.
- **Concept Development Experts (with the Concept Development Manager):**  
The concept development team starts its work in V1 with the development of a high-level concept based on the identified stakeholder needs and further transforms the high-level concept to an initial Concept of Operations (ConOps) based on the developed raw requirements. The concept development team needs to closely co-operate with both the technology development team and the validation team as well. The technology team will give input on the operational options and possibilities for the concept and the validation team will be able to perform an initial validation of the high-level concept and the initial ConOps by investigating the potential of the concept to fulfil the identified ATM needs. In the course of the validation activities in phases V2 and V3 mainly the ConOps is further refined regarding the operational environment, procedures and roles and responsibilities of operational personnel. Results of this refinement should find their way into the writing of the requirements specification at the end of V3.
- **Technology Development Experts (with the Technology Development Manager):**  
The technology development team focuses its attention on the development of the technology necessary to improve the defined operations in such a way that the identified ATM needs are fulfilled. This means that they will play a rather inactive role in V0. In phase V1 they need to identify technical solutions for the operational options documented in the high-level concept and initial ConOps document. Therefore, the technology development team will stay in close contact with the concept development team. They also have a look at the initial requirements produced by the requirement development team and will translate the operationally and organisationally motivated requirements into technical

or system requirements. In that way, the technology development team should be able to produce one or several prototypes addressing all identified operational options and supporting the operational staff in such a way that necessary operational improvements can be achieved. It is one of the main tasks of the team to guarantee that the prototypes are built according to the current set of requirements. Thus, verification activities between phases V1 and V2 and between phases V2 and V3 are necessary before any of the validation activities in V2 and V3, assessing feasibility and performance of the operation, can take place. Between phases V2 and V3 the selection of a final prototype, fulfilling the refined set of requirements, should take place in order to be able to validate operational performance. At this stage there could also still be more than just one prototype. In that case the final validation phase should compare both prototypes. As a final activity the technology development team must be involved in the writing of the requirements specification.

Finally, the actual participants of validation exercises (operational experts, controllers, pseudo-pilots etc.) must not be overlooked as internal actors of the validation processes. However, they should play a minor role in any kind of activity that is concerned with analysing results.

Actors external to the validation processes are mainly ATM stakeholder groups, standardisation and regulation bodies and several other groupings of involved actors.

Usually, the following ATM stakeholders play a major role:

- **Aircraft Operators:**  
Airlines will usually indicate a number of interested actors depending on the scope of the operational concept and the validation activities taking place. They could be the airline pilot flying (PF), who is the responsible of controlling the aircraft, the pilot not flying (PNF), who supports the PF in controlling the aircraft, the Airline Operational Control Centre (AOC) in charge of controlling all aircraft operations of the airline, and airline vehicles moving in apron areas. Other aircraft operators, such as GA, helicopter operators and the military will have similar actors and roles that need to be considered.
- **Air Traffic Controllers:**  
Controllers are usually the main actors in ATM related validation as they have to execute the defined operations with the developed technology and will be the actors responsible for achieving the improvement. Air traffic controllers work in the tower and air traffic control centres. Depending on the scope of operational concept validation different controllers could be involved. In the tower there are clearance controllers (departure manager), taxi controllers (ground controller), runway controllers, a manager controller in charge of



managing several controller positions, and a supervisor controller responsible for the tower working environment. In ATC centres they could be approach controllers and area controllers. ATC centre controllers usually have two different roles, namely tactical sector controller (directly communicating with the pilots), and planning sector controller (with a wider scope and for co-ordination with neighbouring sectors). The military will have similar actors to be considered.

- **Airport Authorities:**  
Depending on organisational issues (important for determining organisational requirements) airport operators may have different roles and tasks. In general, they provide a large set of services to the airport users. In some cases ground controllers and taxi controllers may be part of such an organisation (e.g. Fraport) but in general airport operations units will provide and co-ordinate other services, such as follow-me drivers, marshals, fire and rescue brigades, and police and security operations.
- **Handling Operators:**  
Handling operators can be part of the airport authorities but are usually seen as separate stakeholders being responsible for passenger transport, aircraft service vehicles, and baggage trains and trucks.
- **Regulators:**  
ICAO, the International Civil Aviation Organisation, is the global regulator for aircraft and airport operation. ICAO produces standards and recommended practices, MASPS and MOPS for many of the technologies developed for civil aviation. On a local level, operations in each country are regulated by the Civil Aviation Authorities (CAA/FAA). Also the European Commission together with EUROCONTROL and EASA, the European Aviation Safety Agency, can be seen as a regulator. As has been described in Section 2.1.3, special regulations are in place for the Single European Sky Programme. These regulations need to be considered when developing new systems. Furthermore, Section 2.2.7 showed that there are further regulations for the approval of systems laid down by the standardisation bodies together with EUROCAE and RTCA. These regulations need to be considered in the approval processes taking place in V4 and V5, however, the foundation for approval is indeed laid in earlier phases of the life cycle, so that these regulations must not be overlooked in the requirements capture process.

Other Organisations will only be mentioned in this section of the document for completeness. They are sometimes involved in operational concept validation as they can have an impact on the (political) decision making. Such organisations are: Association of European Airlines (AEA), Airport Council International (ACI), European Association of Aerospace Industries (AECMA), Civil Air Navigation Services Organisations (CANSO), International Air Transport

Association (IATA), International Federation of Air Traffic Controller Associations (IFATCA), and different unions.

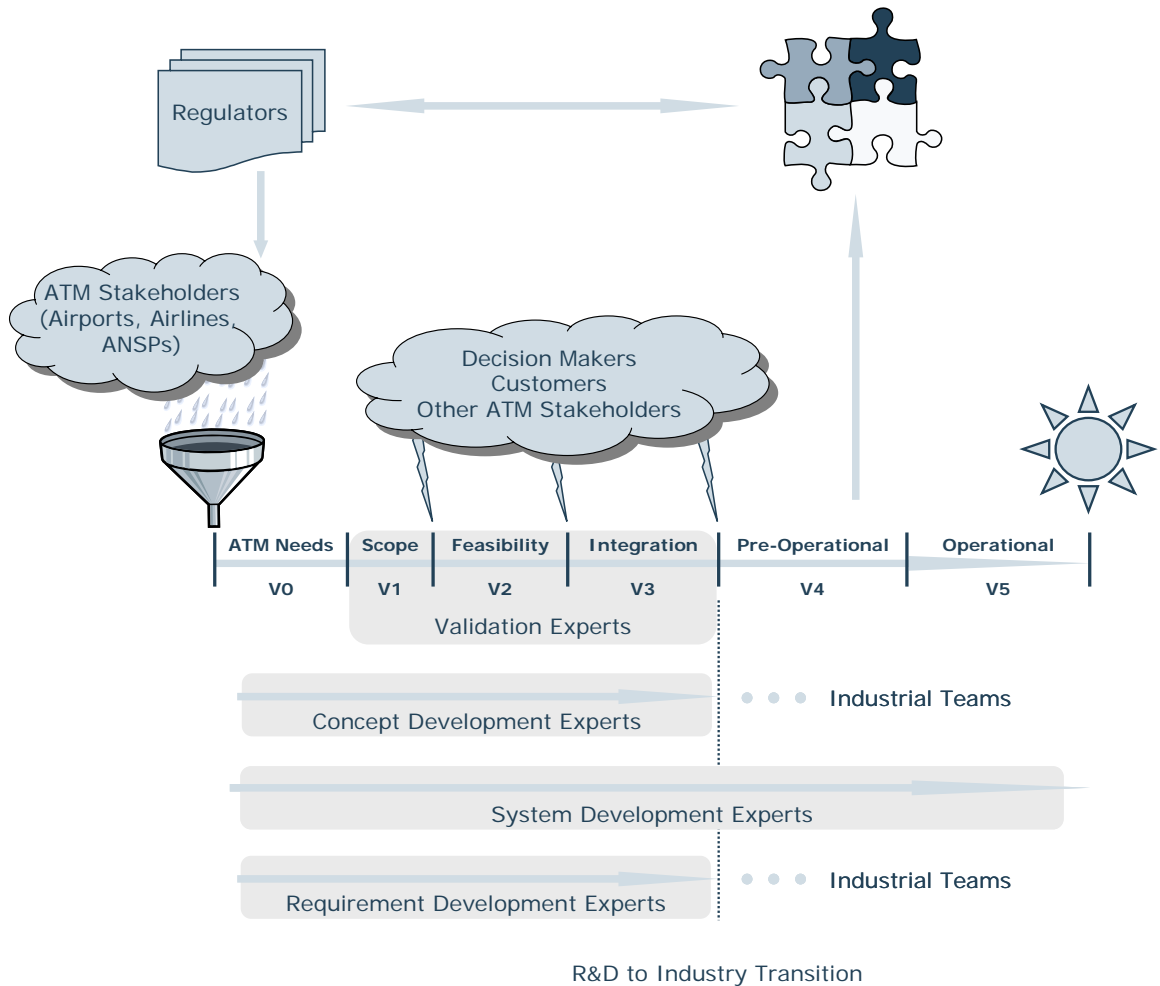


Figure 3-5: Actors in Operational Concept Validation

Figure 3-5 gives an overview of actors described in this section. Internal actors are shown along the life cycle phases of operational concept validation. This also shows the dilemma that technology development activities continue after the R&D related phases, which makes it difficult to differentiate between technology development with a clear R&D focus and with an industry focus. In many cases technology already achieved a pre-operational status while concept development is still in the initial R&D phases.

Furthermore, the figure shows that when making requirement development activities part of the operational concept validation activities they would extend these activities down to phase V0 in which needs are identified and a first translation from needs to requirements takes place. Thus, combining both activities and staffing them with the respective experts could lead to a more



competent validation team and would enhance the links between identification of requirements and validation objectives.

It is an open question who the decision maker or customer is. This question can only be answered depending on the context in which validation takes place. Usually, one of the ATM stakeholders (which could be a regulator) is the decision maker. However, when constituting this role it needs to be clear that the decision maker does indeed have both authority and capability to make a decision for the ATM stakeholder community. The role of validation is to provide sufficient evidence to the decision maker for making the decision to promote the set of identified requirements and one or several concepts and prototypes fulfilling these requirements from one phase of the life cycle to the next.



### **3.3 Processes within the Requirement Development Strategy**

As has been pointed out at the beginning of this chapter, the requirement development strategy is based on three different processes carried out by the requirement development team which could be part of the validation team:

- Requirements Capture
- Requirements Analysis
- Requirements Management

For each of these processes, the following sections will present the relevant standards that can be used as guidelines and will give a short description of the contents.

#### **3.3.1 Requirements Capture Process**

Generally, the Requirements Capture (RC) process is based on the following standards:

- IEEE 15288: the stakeholder requirements definition process
- EUROCAE ED-78A: the co-ordinated requirements determination process

The purpose of the Requirements Capture process proposed within the E-OCVM framework is to identify ATM stakeholder needs (V0) and transform them into raw requirements for a system that can provide the services needed by ATM users and other stakeholders in a defined environment (V1). The requirement development team identifies stakeholders, or stakeholder groups, involved with the system throughout its life cycle, and their needs and desires. Based on the outcome of this process the concept development team should be able to produce a high-level concept description specifying the necessary environment and the system components that play a role in this environment. The technology development team should contribute to the environment description by making suggestions regarding possible technical solutions for the identified operational needs.

The upcoming lists of outcome and activities proposed for the RC process are based on the initial steps of the requirements definition process as described in IEEE 15288, and parts of the ED-78A process description for developing concept and requirement documentation.

The following information, which commonly is contained in a single document, should be the outcome of the RC process:

- a) Documented ATM stakeholder needs.
- b) Documented applicable regulations and legislation.
- c) Documented raw requirements based on the stakeholder needs.



- d) A high-level concept description including the system environment (as part of the parallel process for concept development).
- e) Specification of the required system characteristics and constraints on a system solution (as part of the parallel process of technology development).

The RC process should comprise the following activities:

- 1) Identification of stakeholders (V0):  
This includes, but is not limited to, organisations involved in the development, qualification, operation, and approval of the CNS/ATM system.
- 2) Identification of stakeholder needs (V0):  
This involves the identification of ATM needs through stakeholder consultation.
- 3) Translation of needs into raw requirements (V0 to V1):  
Formulation of raw requirements based on ATM stakeholder needs and identified organisational requirements due to regulation and legislation processes. This includes consultation with regulators and stakeholders.
- 4) Identification and definition of system constraints (V0 to V1):  
Specific constraints can apply such as Minimum Operational Performance Standards (MOPS) and Minimum Aviation System Performance Standards (MASPS) or other issues concerned with legislation (e.g. on safety and security).
- 5) Specification of other non-functional requirements (V0 to V1):  
The list of such requirements can be manifold. Section 2.1.2 gives examples.

### **3.3.2 Requirements Analysis Process**

Generally, the Requirements Analysis (RA) process is based on the following standards:

- IEEE 15288: the requirements analysis process
- EUROCAE ED-78A: the co-ordinated requirements determination process
- IEEE 1362-1998: the definition and documentation of scenarios

The purpose of the Requirements Analysis process proposed within the E-OCVM framework is to transform the stakeholder needs for operational services into a technical view of a required product that could deliver those services. This process builds a representation of a future system that will meet stakeholder requirements and that, as far as constraints permit, does not imply any specific implementation. It results in measurable operational and technical requirements that specify what characteristics the system is to possess and with what magnitude in order to satisfy stakeholder requirements.

The common set of operational requirements must express the intended interaction that the system will have with its operational environment and that are the reference against which each



resulting operational service is validated in order to confirm that the system fulfils the identified needs.

The upcoming lists of outcome and activities proposed for the RA process are based on the both the requirements definition process and the requirements analysis process as described in IEEE 15288, and parts of the ED-78A process description for developing concept and requirement documentation.

The following information, which commonly is contained in a single document, should be the outcome of the RA process:

- a) A documented set of initial, refined, or final operational requirements (depending on the applicable life cycle phase) based on a previously identified set of raw, initial or refined requirements and intermediate validation results in each life cycle.
- b) The operational concept description consisting of the previously mentioned high-level concept and environment description and a initial, refined, or final Concept of Operations (depending on the applicable life cycle phase) concentrating on a description of the use of the system and applicable procedures (as part of the parallel process of concept development).
- c) A specification of a set of initial, refined, or final technical requirements (depending on the applicable life cycle phase) and attributes for a system solution (as part of the parallel process of technology development). This includes constraints that will affect the architectural design of a system and the means to realise it.
- d) A basis for verifying that the initial, refined, or final technical requirements are satisfied is defined (as part of the parallel process of technology development).
- e) A basis for validating the conformance of the services is defined (as part of the parallel process of validation).

The RA process should comprise the following activities:

- 1) Analyse elicited requirements:  
Analysis includes identifying and prioritising the conflicting, missing, incomplete, ambiguous, inconsistent, incongruous or unverifiable requirements.
- 2) Identify the functional boundary of the system.
- 3) Identify required functions of the system.
- 4) Identify implementation constraints.
- 5) Identify technical and quality measures.
- 6) Identify critical qualities of the system.
- 7) Analyse requirements integrity.



8) Resolve requirements problems:

This includes requirements that cannot be realised or are impractical to achieve. This is typically performed as a feasibility analysis during V2 of the E-OCVM.

9) Feed back requirements to stakeholders.

10) Confirm requirements correctness, for instance, in the context of E-OCVM, by validation activities to find evidence and provide feedback.

11) Define scenarios and identify user interaction:

For more information on scenarios refer to IEEE Std 1362-1998 (Chapter 4.6).

12) Produce operational and technical requirement documentation (see SPR and INTEROP definitions in ED-78A) based on the concept documentation (see OCD in section 2.1.1 or OSED description in ED-78A).

The activities described above are given in a logical order, but can be performed partially in parallel, and also iteratively within one instance of the process. As described in the previous sections, the Requirements Analysis Process as a whole will also be performed iteratively throughout the phases of the E-OCVM, and may also be performed multiple times within one phase, depending on the chosen overall development strategy. The proposed requirements development strategy does not intend to prescribe or exclude any of the development life cycles discussed in Chapter 2. But, given the fact that the E-OCVM is concerned with validation, the spiral approach applied to a V-model, as described in Chapter 2.2.8, should be considered for organising the validation and requirements activities. However, this does not imply that the overall systems development strategy should be the same.

Visualisation and modelling, for example with the help of UML, may be helpful tools for analysing functional requirements. Such techniques could include use cases, function trees and matrices and could also evolve into complete operational scenarios.

### **3.3.3 Requirements Management Process**

The proposed Requirements Management process is based on the following standards:

- CMMI: the requirements management key process area
- EUROCAE ED-80: the requirements capture process
- ECSS E-40: on the Design Justification File (DJF)

The CMMI (<http://www.sei.cmu.edu/cmmi>), which was not analysed in the survey, is a model for improving and appraising the performance of development organisations. It stands for 'Capability Maturity Model Integration'. Concepts covered by this model include systems engineering, software engineering, integrated product and process development, and supplier sourcing as well as traditional concepts such as process management and project management.



One of the key process areas of this model is requirements management. The purpose, outcomes and activities of the Requirements Management process proposed below is based on the description of this key process area in the CMMI. Since this description is very general, details from the ED-80 requirements capture process have been injected to make the description more practical.

The purpose of the Requirements Management (RM) process proposed within the E-OCVM framework is to maintain the operational and technical requirements (including possible organisational requirements due to compliance with regulations for approval, certification, safety etc.) of the services, systems and system components and to identify inconsistencies between those requirements and the plans and work products.

The following should be the outcome of the RM process for each of the life cycle phases:

- a) The requirements are recorded consistent with all correctness criteria.
- b) The traceability of the requirements is consistent and complete.
- c) All requirements have been agreed upon.
- d) A log of relevant requirements changes is available (see also DJF in ECSS E-40).

The RM process should comprise the following activities:

- 1) Record identified requirements together with appropriate attributes.
- 2) Obtain an understanding of requirements.
- 3) Provide derived requirements produced to the appropriate process.
- 4) Provide requirements omissions and errors to the appropriate process for resolution.
- 5) Obtain stakeholder commitment to requirements.
- 6) Manage requirements changes.
- 7) Maintain bi-directional traceability of requirements.
- 8) Identify inconsistencies between project work and requirements.

### **3.3.4 Architectural Design**

Although the architectural design process is out of the scope of this study, this section provides some background information on the process in order to clarify the relationship between requirements and architecture.

The processes within a development approach have relationships that depend on the selected life cycle. In general, there is a strong relationship between requirements analysis and architectural design whereby the system requirements and design are defined at increasing detail in an iterative fashion. Architectural design explains how the system is structured in various ways. A



common approach is to break down the system into sub-systems and to take each sub-system and break it down to the next level until sufficient detail is reached. During this process, high-level system requirements need to be allocated to lower level sub-systems. Interfaces of sub-systems are defined by allocating functional requirements to the various sub-systems, effectively determining the input-output relationships within the system.

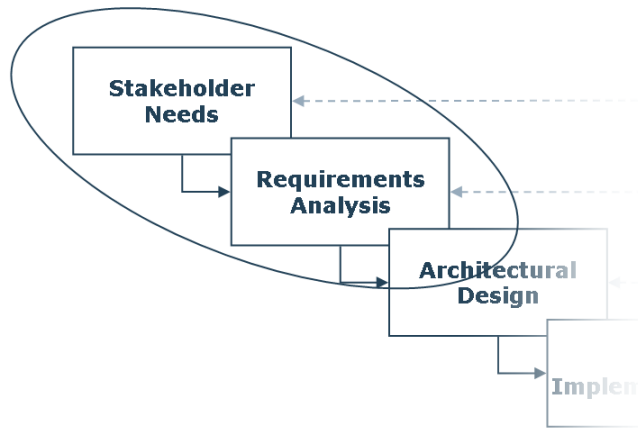


Figure 3-6: Project Focus and Relationship between Requirements and Architecture

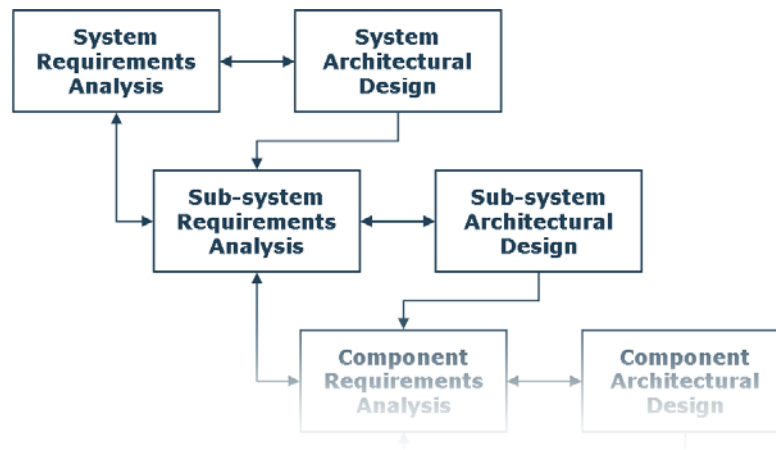


Figure 3-7: Iterative Nature of Requirements and Architecture



When defining new functions that will be part of a 'system of systems' it is common to start with the overall architecture of the existing system of systems, rather than the requirements of a specific system. In this way it is possible to define a new capability of the overall system of systems and to find which modifications and which systems are needed to support this capability.

Various standardised approaches exist to model a system of systems. These standards are called architecture frameworks. Some examples include: Zachman Framework, Department of Defense Architecture Framework (DoDAF) and NATO Architecture Framework (NAF). Recently, EUROCONTROL has defined the Overall ATM/CNS Target Architecture (OATA) as a high-level design of an integrated ATM 'system of systems' across all ECAC states that will supersede the current collection of individual national systems. Finally, the Single European Sky ATM Research (SESAR) project also follows an architecture-driven approach, and a separate work package (WP2.4) has been created in which the overall system architecture will be defined.



### **3.4 Summary**

In this chapter, both the Operational Concept Validation life cycle of the E-OCVM and the involved actors have been described, and a requirement development strategy within that lifecycle has been proposed. This proposal is based on the survey of the terminology and standards in the previous chapter. The proposed strategy consists of three requirement processes: Requirements Capture, Requirements Analysis, and Requirements Management, for which detailed activities have been described.

The Requirements Capture process aims at eliciting the initial requirements from stakeholders and is performed in phases V0 and V1 of the E-OCVM lifecycle. In phase V0, the process is performed to elicit and capture high-level stakeholder needs. In phase V1, these needs are transformed into raw requirements for a system that can provide for these needs in a defined environment.

The Requirements Analysis process is performed throughout E-OCVM phases V1 to V3 to iteratively transform these raw requirements into a clear, consistent, and stable system specification that can be used as the basis for the industrialisation of the system (phases V4 and V5). In each of these iterations, requirements are fed to other development streams, such as concept development and prototype development to continuously improve both the concept and the prototypes. Also in each of these iterations, evidence and feedback from validation activities is received and used in the requirements analysis to continuously improve and further detail the requirements, ultimately leading to a feasible and consistent set of requirements, i.e. the specification (at the end of phase V3).

During these iterations of Requirements Analysis, a third process is performed in parallel, called the Requirements Management Process. This process is responsible for maintaining the set of requirements throughout all the iterations, together with the rationale and supporting documentation for each of the changes made. This history, together with the final specification, should be considered as an important deliverable at the beginning of phase V4 which better explains to Industry why the specification is as it is. This also concerns the issue regarding the level of detail of such specifications as mentioned in previous chapters (see Section 2.3).



## **4 Analysis of European R&D Operational Concepts**

This chapter analyses a selection of European R&D projects with regard to requirements development by describing the project background, applied methods for concept development and validation, and the role of requirements within the project. At the end of each section conclusions on the appropriateness of the process will be given.

### **4.1 Analysis of European Commission Project EMMA**

#### **4.1.1 Project Background of EMMA**

The ‘European Airport Movement Management by A-SMGCS’ (EMMA) integrated project is set within the Sixth Framework Program of the European Commission (Directorate General for Energy and Transport) and looks at Advanced Surface Movement and Guidance Systems (A-SMGCS) as a holistic approach for changes in airport operations. It builds on the experiences of earlier projects such as ‘Operational Benefit Evaluation by Testing A-SMGCS’ (BETA). With BETA new technologies for data extraction, digitising, data fusion, data link and multilateration became available. Although A-SMGCS progressed from a demonstration status to a full operational system, the complete proof of benefit of A-SMGCS was missing. Therefore, EMMA aims at setting the standards for A-SMGCS systems and their operational usage, safety and interoperability while also focussing at the benefit expectation in Europe.

In order to achieve this ambitious goal, EMMA is subdivided into two project phases (EMMA-1 and EMMA-2). In the first phase, which took place from 2003 until 2006, an implementation of A-SMGCS Level I and II was looked at as an initial step. While the Level I implementation merely seeks to enhance safety and efficiency on the ground by means of additional surveillance services, the Level II implementation already looks at an automated control service which helps controllers to detect potentially dangerous conflicts on runways and restricted areas. In the second phase of the EMMA project, which runs from 2006 until 2009, the focus was extended to a full-level A-SMGCS. This means that higher-level A-SMGCS functionality will be implemented. In a first step, this allows for the sharing of traffic situation awareness among pilots and drivers on the airport and the introduction of an automated routing function. In a second step functions will be improved with conflict resolution advisories for controllers and the up-link of a validated route planning to pilots and drivers.

The EMMA-1 project was structured in six different sub-projects (SP). There were three ground-related sub-projects and one on-board-related sub-project representing the three different test sites (Prague, Milan Malpensa, and Toulouse) and the on-board site. These four sub-projects were autonomous and independent and were linked by two additional activities or



sub-projects, namely the definition of the concept and the establishing of a consolidated methodology for validation and verification of technical sub-systems. The EMMA-2 project is structured identically.

**4.1.2 Applied Methods for Concept Development and Validation in EMMA**

EMMA-1 quite rigorously followed the ED-78A approach (see Section 2.2.7). The EMMA-1 Detailed Work Plan (Ref. [32]) describes the approach to concept development. It states that EMMA-1 made a comprehensive review and assessment of all A-SMGCS related work previously performed. Based on this work the users of the system (ANSPs) developed an enhanced A-SMGCS concept of operations.

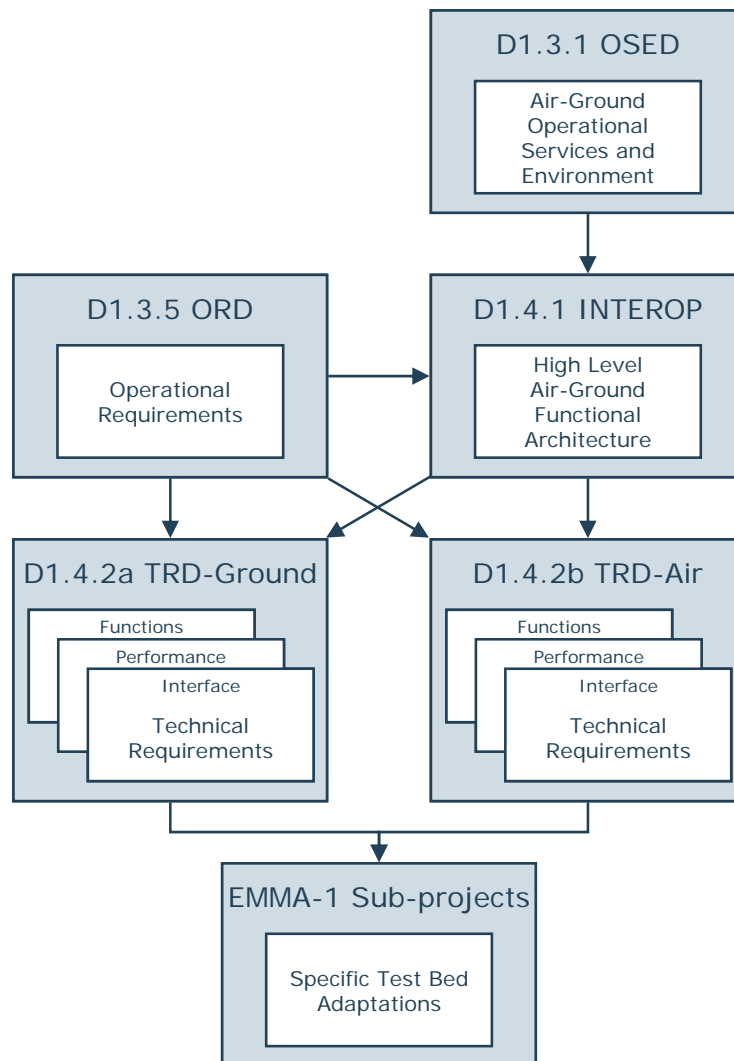


Figure 4-1: Relationship between EMMA-1 Concept Documentation



This concept of operations was based on existing EUROCONTROL concept documentation and was further elaborated, including a description of services and the A-SMGCS air and ground environment in the EMMA-1 Operational Environment and Service Description (OSED), and the documentation of operational requirements in the ORD, as shown in Figure 4-1.

In EMMA-2 a similar approach to concept development and requirements elaboration was chosen. However, this approach considered earlier results from EMMA-1 as input. The approach is described in the EMMA-2 Services, Procedures and Operational Requirements (SPOR) document (Ref. [33]).

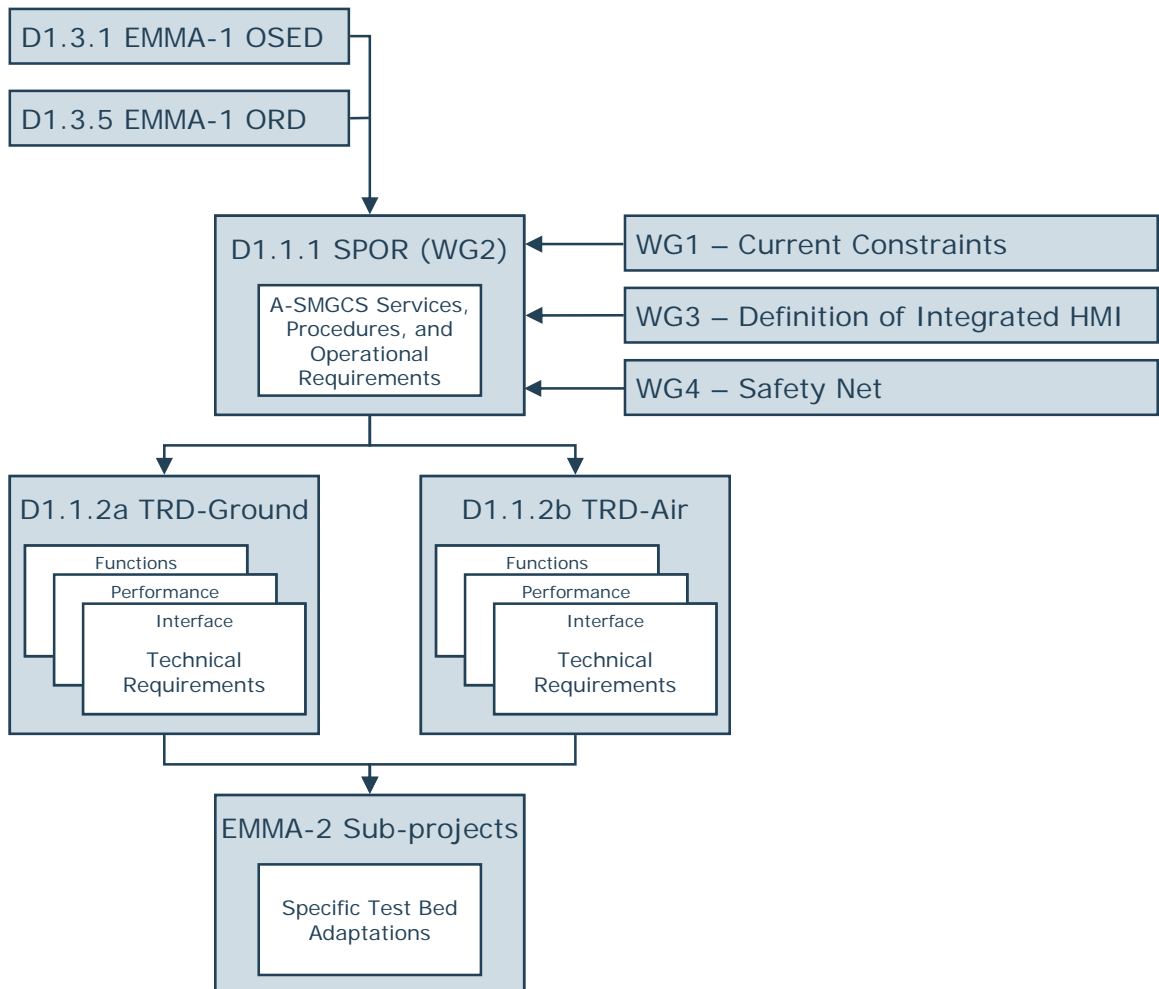


Figure 4-2: Relationship between EMMA-2 Concept Documentation



The SPOR is indeed the central document in EMMA-2 concept development and consolidates both results from EMMA-1 and results from EMMA-2 working groups, considering current operational constraints for controllers, pilots and vehicle drivers (WG1), safety net technology (WG4), and the definition of an integrated HMI (WG3). In comparison with the EMMA-1 documents, the SPOR combines the contents of an OSED, ORD and INTEROP document and lays emphasis on the integration of services, especially at the front end for controllers, pilots, and vehicle drivers (see Figure 4-2).

Regarding validation, the EMMA-1 project mainly followed the methodology described in the Master European Validation Plan (MAEVA) project (Ref. [34]). This means that the stepped validation approach described in MAEVA was followed in the general validation strategy as described within the Generic Verification and Validation Masterplan (cf. Ref. [39]). Since verification was not described in the MAEVA guidelines, the participants of the EMMA project provided their own definition of both verification and validation in order to clearly discern the two activities (Ref. [39]):

- *Verification* is testing against predefined technical specifications, technical functional testing ('did we build the system right?').
- *Validation* is testing against operational requirements (as defined by stakeholders and written down in the OSED document of EMMA SP1), man-in-the-loop, ATM procedure testing, case studies ('did we build the right system?').

In EMMA-2, which mainly tried to follow the E-OCVM guidelines for validation and therefore considered a life cycle model for concept validation, no relevant changes were applied to these definitions. However, it was realised that indeed there are different life cycle phases requiring more or less rigour in the validation approach. Nevertheless, most of the simulation activities were again regarded as life cycle phase V3 validation activities and verification activities were seen as tests of technology (enablers) with respect to predefined specifications such as MASPS and MOPS (Ref. [41]).

Starting from the V-shape approach as described in IEEE and ESA standards (see also Ref. [13]), the EMMA validation strategy places verification at the bottom of the V-shape before sub-system integration. This is different from Figure 2-10, where there are additional verification activities for higher integrated systems. EMMA, however, made a clear distinction between the lower and the higher level, meaning that higher level verification activities were rather seen as life cycle phase V2 validation activities. Anyhow, all major simulation activities in EMMA-1 and EMMA-2 were seen as life cycle phase V3 validation activities.

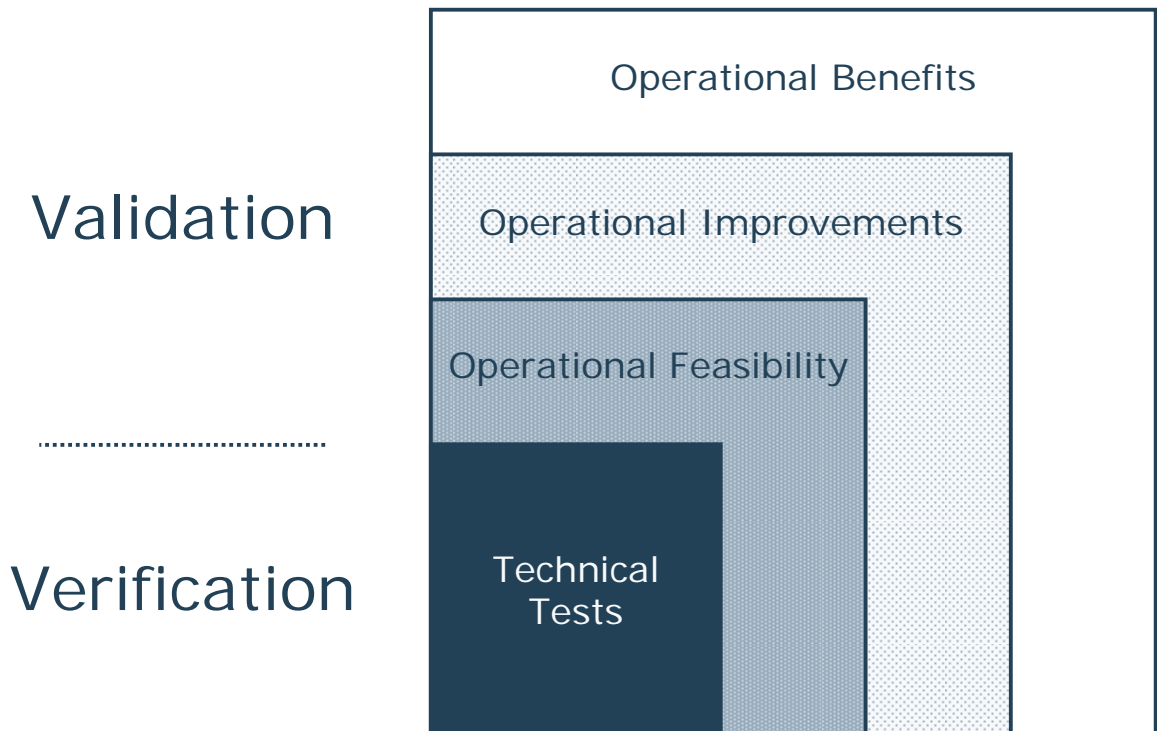


Figure 4-3: EMMA Definition of Verification and Validation Stages

Generally, the following stages in verification and validation were identified:

- Technical tests were conducted in order to assess the technical performance of A-SMGCS equipment and therefore represent the verification subject.

Validation activities were split into three major building blocks, namely:

- Operational feasibility addressing the definition of the operational use of the equipment in accordance with the system performance assessed during verification. This stage includes the 'operational verification' and 'system parameter tuning' activity as well as 'system usability' aspects. These activities are necessary before further validation of the system with respect to possible improvements and benefits can take place.
- Operational improvements (capacity, efficiency, safety, Human Factors) are investigated when both system requirements and user requirements are met by the system as verified and evaluated in the previous stages. In this stage the performance of the specific ATM concept (possibly related with new technology) can be assessed.
- Operational benefits are looked at in a last stage. Only when it has been verified that the system is working properly according to all technical and operational requirements and



when it has been validated that there will be operational improvements, it will be possible to translate such improvements into monetary terms.

Thus, in conclusion all major activities concerning the assessment of operational requirements were seen as validation activities, while all technical tests as part of the assessment of technical requirements were seen as verification activities.

#### **4.1.3 Role of Requirements in EMMA**

According to the EMMA-1 concept development work plan (Ref. [32]) the OSED and the Operational Requirements Document (ORD) were established independently based on all material available from previous projects and from the EUROCONTROL A-SMGCS project. Both the OSED and ORD then served as input for the INTEROP document (later in the project called AGFA, air-ground functional architecture document) and the technical requirement documents, which were split into air and ground technical requirements. Eventually, the requirement documentation fed the adaptation activities for the different test platforms.

The same approach was chosen in EMMA-2 where OSED, ORD and INTEROP documents were input for the SPOR document, which consolidated previous results with identified current constraints and offered solutions in the form of integrated front end systems and associated procedures. Again there was a clear cut between operational and technical requirements for air and ground which were extracted from the SPOR document.

Although the EMMA-1 project did not foresee any refinement of requirements and did not consider the life cycle phases of validation, a feedback loop from the validation activities to the concept development activities occurred in the form of updated versions of the concept documents (OSED, ORD, TRD, AGFA and test site operation documents). EMMA-2 currently has not specified any updates in their project planning.

#### **4.1.4 Conclusions on the EMMA Requirements Development Process**

The EMMA project strictly followed the ED-78A approach regarding document structure which is mainly due to the fact that technology development, especially in EMMA-1, did already progress to very late stages in the life cycle. Thus, the focus of EMMA was indeed to produce documents required for approval and certification, such as OSED, INTEROP and ORD and validation was seen as a means to test whether the proposed system fulfils the operational requirements.

Nevertheless, several participants within the EMMA consortium were also involved in the major European validation activities, such as MAEVA, CAATS and the development of the E-OCVM itself, so that the general approach to validation could be elaborated in a more detailed way. This led to abovementioned validation plan documents (Ref. [39] and Ref. [41]) in



which all test and simulation activities were categorised according to Figure 4-3 which is a bit closer to a life cycle approach described in the E-OCVM.

The process of requirement elaboration, however, remains unclear<sup>1</sup>. The methodology to obtain technical requirements was described as the process “to extract operational, functional and performance requirements from relevant documentation and map them onto an architectural framework” according to the TRD. However, it seems that this process was not a collaborative effort between concept developers, technology developers, and validation experts, but rather happened in isolation by technology experts. The methodology to obtain operational requirements is even more obscure. The ORD mentions a number of basic documents such as the OSED, and high-level EUROCONTROL documents, but the processes of requirement capturing and analysis are not described in much detail.

---

<sup>1</sup> This mainly concerns EMMA-1 as many of the activities in EMMA-2 are currently underway.



## **4.2 Analysis of EATM Project FASTI**

### **4.2.1 Project Background of FASTI**

The FASTI programme was initiated as a co-ordinated and European-wide approach to the implementation and rapid deployment of an initial set of automated tools that support controller tasks. The main objective of the FASTI programme is to ‘co-ordinate the implementation and deployment of controller system support and tools, as required, across ECAC by 2012 in a harmonised way’.

The principal aims of the FASTI programme are to:

- Increase sector capacity, improve flight efficiency and maintain safety to meet forecast airspace user demand;
- Provide implementation support to ANSPs, ATC Regulators and Industry in order to harmonise and expedite the deployment of controller automated system support;
- Establish appropriate controller system support capabilities, common performance levels across the European ATM Network and ATC procedures;
- Act as an enabler for future implementation of automated support to ATC currently under development.

Traditional controller working methods and procedures are expected to be subject to change in order to become compatible with and to accommodate the added system support. An operational concept document outlining the scope of FASTI applications, assumptions, dependencies and operational environments as well as descriptions of controller roles and working methods required for the successful operational implementation of such tools was prepared in July 2006 by the FASTI Operational Focus Group (see Ref. [20]).

The FASTI programme focuses on three support tools that have been investigated in previous studies within the Automated Support to Air Traffic Services (ASA) programme, and are considered sufficiently mature for implementation:

- Medium Term Conflict Detection (MTCD)
- Monitoring Aids (MONA)
- System Supported Co-ordination (SYSCO)

The orientation work performed for establishing the operational concept document showed that there might be considerable consequences for current working methods that do not build on this kind of system support. This is due to the required changes in presenting data which changes the controllers’ mental model and also due to the necessary system entries for accurately feeding the predictive functions of the tools.



The operational scope of application of FASTI support tools is en-route and extended TMA airspace within the ECAC region being managed by Area Control Centres (ACC). The relationship between functionality of a tool and the various concepts of use is elaborated within FASTI.

The implementation of required Trajectory Prediction (TP) is outside the scope of the FASTI programme. Nevertheless, it is identified as a key enabler as TP is a core function of the Flight Data Processing System (FDPS). Furthermore, the programme intends to specify TP performance requirements, such as availability, integrity and continuity.

Another key enabler identified by FASTI for successful operation of future tools is the Human Machine Interface (HMI). The HMI must enable easy access to flight data and the input of ATC clearances by controllers, thus permitting trajectory updates. HMI requirements and implementation guidance, associated with the programme scope, are also part of the programme deliverables.

#### **4.2.2 Applied Methods for Concept Development and Validation in FASTI**

The applied methods for concept development and validation in FASTI have been captured in a Validation Plan, the first edition of which has been released in December 2006 (see Ref. [21]). The document defines the FASTI operational concept validation activities on the programme level. This includes validation of the necessary enablers. It defines the necessary projects to achieve the FASTI validation objectives and the approach to consolidate the project's findings on the programme level. The decision whether to go forward or not with the 'Implementation and Ops Planning' of the FASTI programme is taken after the completion of the 'Initial Implementation and Ops Validation' of the FASTI Operational Concept.

For the individual projects there will be separate Project Level Validation Plans.

The E-OCVM life cycle model is used as the planning framework for the Programme Level Validation Plan.

The plan further describes the place of the FASTI programme in the European research and development strategy, quantifies the ATM needs, describes the operational concept, programme objectives, validation stakeholders, and the validation expectations and objectives.

#### **4.2.3 Role of Requirements in FASTI**

Operational requirements for Medium-Term Conflict Detection (MTCD) and supporting tools such as Monitoring Aids (MONA) were developed in the late 1990s in the framework of EATCHIP and have undergone a series of demonstrations and trials, culminating in live trials at Malmö ACC, Rome ACC and Maastricht UAC as part of the Automated Support to ATS (ASA)



programme. In addition, a number of major Air Navigation Service Providers (ANSP) developed and prototyped MTCD and MONA tools according to their own specific needs. In parallel, the ATM systems industry developed MTCD and MONA functionality in line with the EATCHIP (later EATM) guidelines, and delivered these systems to a number of ANSPs.

In order to pool the experience of EUROCONTROL and national ANSPs, a FASTI Baseline Description document has been developed to provide a comprehensive set of guidelines, including validated operational requirements, for consideration in the acquisition or development of MTCD and MONA tools (see Ref. [19]). The commonalities in the implementations as well as their specialisations were studied. Links with the driving objectives and constraints were established, in order to explain how the objectives influence the operational requirements. Three generic levels of capability were identified corresponding to the degree of delegation to the system to automate the conflict detection task. The operational requirements were accompanied by implementation guidelines based on the experience of the contributing systems. Operational requirements were formulated in a generic way, as much as possible. The guidelines describe how the implementation might be adapted according to the driving objectives.

#### **4.2.4 Conclusions on the FASTI Requirements Development Process**

The FASTI Baseline Description document, which clearly separates the generic operational requirements and the specific implementation guidelines (which are in fact local, detailed implementation requirements derived from the generic operational requirements), is a good example of how to document requirements and how to maintain traceability between the more general operational requirements, and the implementation specific requirements resulting from validation exercises and pre-operational prototypes.

### 4.3 Analysis of Industry Project VICTORIA

#### 4.3.1 Project Background of VICTORIA

The VICTORIA project aimed at demonstrating the validity of new avionics technologies and systems based on open standards. Within the project a validation platform has been developed for integration of standardised components, technologies and tools in an open, modular and improved aircraft electronic system. The project was led by Thales Avionics and involved over thirty European companies including avionics industries, research establishments, aircraft manufacturers and universities.

The VICTORIA project started in 2001 and ended in 2004, partly funded by the European Commission under the 5th Framework Programme for ‘Competitive and Sustainable Growth’.

#### 4.3.2 Applied Methods for Concept Development and Validation in VICTORIA

VICTORIA followed development and validation processes as defined in two top-level documents:

- Project Development Plan (Ref. [35])
- VICTORIA Program General Process (Ref. [30])

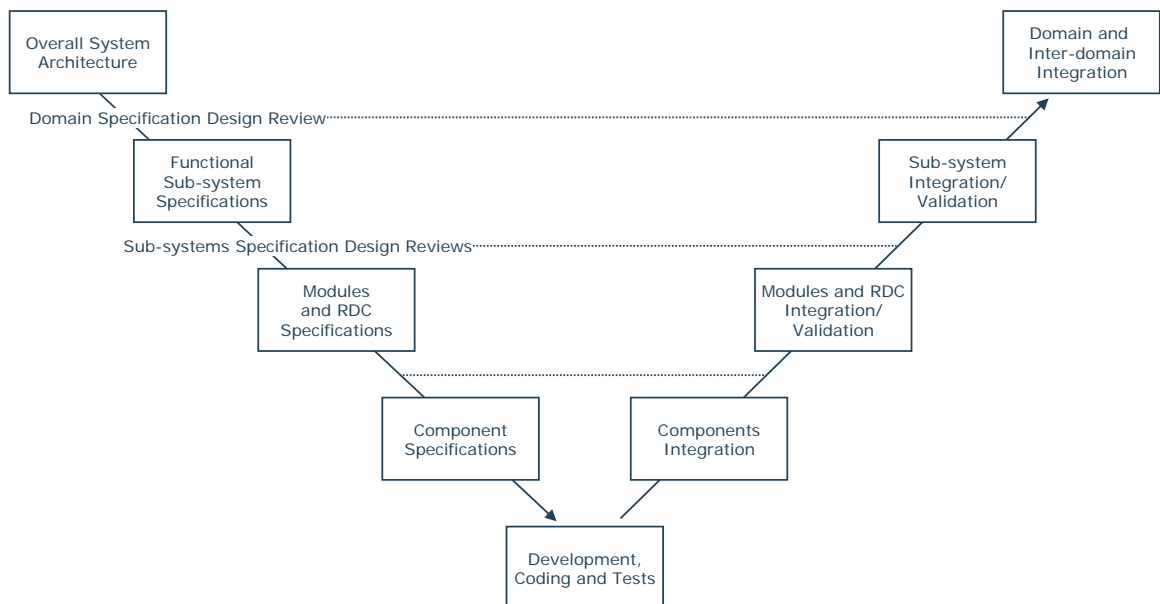


Figure 4-4: Development Approach of Project VICTORIA

These processes have been developed as part of the project work, basing them on existing standards such as Airbus Directives. The selected development approach is in accordance with a V-model, as shown in Figure 4-4.



The applied validation process has several levels, whereby validation at one level can be part of verification at the next integration level:

1. Verification of modules (sub-subsystem level). The responsible partner can decide which method to apply for this verification activity. Applicable standards may include EUROCAE ED-14 (environmental test procedures), ED-12B (airborne software), and ED-80 (airborne hardware).
2. Validation of the modules as part of a domain (subsystem level). This activity is part of the domain verification. VICTORIA has six domains, such as utilities, energy, cockpit, etc. For each domain an Experimental System Test Plan (Ref. [31]) has been written based on the VICTORIA Program General Process.
3. Validation of the inter-domain integration (system level) against the project objectives. The project objectives to be met have been captured in the VICTORIA Program General Process. They include architecture, system and process objectives. At the end of the project the integrated validation system is scored against all objectives. By aggregating the scores an evaluation is made of the overall compliance.

#### **4.3.3 Role of Requirements in VICTORIA**

The VICTORIA validation system consists of many components that have been developed with different methodologies. Pre-existing components, or components that function as a prototype for a future product, often have been developed in accordance with Airbus Directive 200 (ABD200 Requirements and Guidelines for the Systems Designers). This directive states that a System Requirements Document (SRD) and a System Description Document (SDD) should be written. However, VICTORIA has not applied this directive because of intellectual property rights and other constraints such as time and money. Instead, each component has been described in a Functional Specification Document with limited scope and depth compared to the SRD and SDD. As a result, mainly functional and interface requirements have been identified and managed.

Non-functional requirements such as reliability, availability, and maintainability are not specified, since they are not applicable to the VICTORIA validation. To create the functional specification, some companies have applied a requirement specification standard such as MIL-STD-961D (Ref. [5]).

#### **4.3.4 Conclusions on the VICTORIA Requirements Development Process**

Although the technology differs between VICTORIA and E-OCVM, the concept validation approach has similarities. It combines existing systems and newly developed systems in order to validate whether a new concept can meet certain objectives. In order to successfully develop



and integrate this validation system, it is necessary to capture and manage requirements in accordance with a defined process and to state objectives that are the basis of validation.

In addition, it is important to understand the hierarchical nature of systems and to realise that validation at one level can be part of verification at the next integration level. It is therefore useful to realise that other branches of technology face similar challenges and have found solutions that could have a spin-off for the E-OCVM.



## 5 Conclusions and Recommendations

This chapter summarises the conclusions of the RORI-OCV study and gives recommendations for both the OCVSD and the E-OCVM regarding the integration of a requirement development strategy and a better understanding of the different development streams for operational concepts and technology running in parallel with validation. In that regard, it is important to understand the differences between the two documents which have not been dealt with yet in much detail and are therefore briefly discussed.

Generally, the E-OCVM is based on the methodological approach described by the Master European Validation Plan (MAEVA) project in its Validation Guideline Handbook (VGH, see also Ref. [34]). The MAEVA VGH suggests a five-step approach to validation that was originally defined in the EUROCONTROL Development of EATCHIP/EATMP Validation Methodologies (DEVAM) project. As refined and expanded, these five steps form the core of the VGH. In order to address each of the steps at the appropriate level and for the appropriate audience, the VGH contains three parts, each describing the steps at different levels of detail. Despite the general acceptance of the VGH within the ATM community, several adaptations of the framework were proposed within European R&D initiatives (especially the Gate-to-Gate and C-ATM projects) concentrating on the initial approach to validation activities and the related life cycle of the concept or technology to be validated. EUROCONTROL together with the CAATS project team (responsible for co-ordinating validation approaches within the current European Commission Framework Programme) consolidated all change proposals in the first edition of the E-OCVM, which has become the mandatory methodological framework for concept validation within the European ATM community. The current version of the E-OCVM (Ref. [15]) addresses validation at the highest two levels of the MAEVA stepped approach, and additionally introduces a concept life cycle model (as discussed in this document) as well as a so-called case-based approach. The latter integrates validation exercise results into key cases that directly address stakeholder issues about ATM performance and behaviours. Since the detailing of the case-based approach is one of the tasks of the CAATS project in its second phase and is an ongoing activity, this study has been concentrating on the E-OCVM life cycle model. Thus, recommendations concerning the E-OCVM focus on the interpretation and detailed analysis of the operational concept validation life cycle.

The OCVSD (as part of the activities within the FAA-EUROCONTROL Action Plans for real-time and fast-time assessments, AP5 and AP9) builds on the results of the E-OCVM development efforts, but concentrates on organisational issues and harmonisation of validation approaches between Europe and the United States. It identifies three key groups affected by ATM concept development namely human system operators (air and ground), engineering partners (including



R&D teams and industry), and organisations and decision making groups. Recommendations concerning the OCVSD will therefore concentrate on the different actors in operational concept validation and the possible consequences for their roles and responsibilities.

The upcoming section of this document summarises the conclusions that can be derived from the main activities of the project, i.e. the survey of standards and terminology, the proposal for a requirement development strategy and the analysis of the European R&D projects. The final two sections then provide recommendations for the OCVSD and the E-OCVM with a focus on the two aspects mentioned above, i.e. the role of requirements within the life cycle and the consequences for the organisational aspects, including actors and their roles and responsibilities. The recommendations are repeated and highlighted after each paragraph describing them.

### **5.1 Conclusions of the Study**

One of the main outcomes of the survey of terminology is that there are many terms for similar, but not always identical, artefacts or activities that are often tailored to a specific problem domain or organisation. Thus, in most cases it will be necessary to ask the specific user of such a term or the related organisation for a clear definition.

Regarding the definition of an operational concept it is useful to distinguish between high-level concept documents that give a high-level description of ATM services and environment and Concepts of Operation or Use that provide more detail on user needs and requirements, the way that the system parts are operated, organisational issues, functions and processes, interactions and information flows, involved actors and their roles and responsibilities. Operational procedures are defined to consist of abovementioned rules, regulations, processes and working practices.

Regarding requirement-related definitions there is a basic distinction between stakeholder needs (sometimes also called user requirements) and requirements. Stakeholder needs ('raw requirements') may be inconsistent, technically or financially impossible, and may contain implementation details, design decisions or any other kind of statement on the system or function to be developed. The requirements analysis process transforms the identified needs into requirements. These requirements must be clear and consistent, without unnecessarily limiting the possible set of design solutions. Eventually, requirements are refined through verification and validation processes that will lead to specifications. The specifications available at the end of research and development activities should consolidate the requirements and the validation results. However, it is not quite clear what the required level of detail is. After all, these specifications are still research specifications that might or might not be in line with technology



development. Therefore, an additional aspect in the R&D phases is the relationship between customers and suppliers of the technology concerned and the kind of technology being developed.

Considering the additionally identified terminology on system architectures, it was noticed that none of the terminology survey references or standards is explicit about them, although the topic is heavily debated within SESAR. It is expected that issues regarding the architecture of a system of systems, such as ATM, will be of highest importance for the initial validation phases as they have an enormous impact on the basic operational requirements.

The survey of relevant systems engineering standards focused on the development phases relating to operational concept validation, i.e. phases V1 to V3 as described in the E-OCVM. For the survey, an appropriate selection of documents (from IEEE, ESA and EUROCAE) was made, addressing these phases in particular.

The IEEE standards proved to be useful in several aspects. IEEE 1362 provides a detailed description of a Concept of Operations (ConOps) document and the related analysis process. IEEE 830 describes the contents of a Software Requirements Specification (SRS) with a special emphasis on adaptation requirements and strongly recommends that the document is jointly prepared by supplier and customer, thereby advocating a strong participation of industry in the early stages of validation. IEEE 1233 and IEEE 15288 describe processes for requirements specification development and other systems engineering processes that are appropriate to the type of activities performed in phases V2 and V3 of the E-OCVM. Consequently they have been considered in the proposal for a requirement development process.

Concerning the ESA standards it was found that they are part of a large set of interrelated management, engineering and product assurance standards for space projects and applications and contain very concise provisions for suppliers. They are thus useful as a reference for topics such as traceability, wording and documentation of requirements. Furthermore, it was found that the described use of a Design Justification File (DJF) could prove to be very helpful in phases V2 and V3 of the E-OCVM in order to maintain a history of decisions, and their rationale, during prototyping and experiment activities.

EUROCAE standards ED-79, ED-12B and ED-80 also added to a clearer picture on requirement development processes. They contain details regarding requirements capturing for systems, software and hardware, albeit with a focus on safety critical airborne applications. ED-78A, which is one of the most frequently referenced guideline documents in the European



ATM R&D community, describes rather complex processes for approval of air traffic services supported by data communications. The structure and contents of suggested documents described in ED-78A has been maintained in many verification and validation projects, although the guidelines do not provide a detailed view of how verification and validation activities should be carried out.

The general requirement development process, which is proposed in this study as a result of the survey of standards, can be divided into two major processes, supported by a third process:

- The Requirements Elicitation or Capture Process aims at eliciting the initial requirements from stakeholders and is performed in phases V0 and V1 of the E-OCVM life cycle. In phase V0, the process is performed to elicit and capture high-level stakeholder needs. In phase V1, these needs are transformed into raw requirements for a system that can provide for these needs in a defined environment.
- The Requirements Analysis or Specification Process is performed throughout E-OCVM phases V1 to V3 to iteratively transform these raw requirements into a clear, consistent, and stable system specification that can be used as the basis for the industrialisation of the system (phases V4 and V5). In each of these iterations, requirements are fed to other development streams, such as concept development and prototype development to continuously improve both the concept and the prototypes. Also in each of these iterations, evidence and feedback from validation activities is received and used in the requirements analysis to continuously improve and further detail the requirements, ultimately leading to a feasible and consistent set of requirements, i.e. the specification (at the end of phase V3).
- The Requirements Management Process is responsible for maintaining the set of requirements throughout all the iterations, together with the rationale and supporting documentation for each of the changes made. This history, together with the final specification, should be considered as an important deliverable at the beginning of phase V4 which better explains to industry why the specification is as it is.

The analysis of actors and their roles and responsibilities within the validation process additionally concluded that it is desirable to have at least one person with a central role as validation manager or development manager overseeing all development processes with a gradual refinement of requirements. This means that such a manager can actually be the head of a group of requirement development engineers, where each member of this group is responsible for a given set of requirements on either a certain level in the organisation or a certain domain. The development manager then has the task to integrate all requirements processes and thereby align all process in system development.



The analysis of European R&D projects also resulted in several conclusions, which are briefly summarised:

- The EMMA project strictly followed the ED-78A approach regarding document structure. Due to the progress in technology development, EMMA was focussed on approval and certification issues. The OSED, INTEROP and ORD documents and the related validation activities were seen as a means to test whether the proposed system fulfils the operational requirements. There was a clear distinction between technical and operational requirements. The process of requirement elaboration (OSEIC) remains unclear.
- Within the FASTI project there was a clear separation between the generic operational requirements and the specific implementation guidelines (which are in fact local, detailed implementation requirements derived from the generic operation requirements). This is a good example of how to document requirements and how to maintain traceability between the more general operational requirements, and the implementation specific requirements resulting from validation exercises and pre-operational prototypes.
- The VICTORIA approach to validation combines existing systems and newly developed systems in order to validate whether a new concept can meet certain objectives. In order to successfully develop and integrate the system, it is necessary to capture and manage requirements in accordance with a defined process and to state objectives that are the basis of validation. In addition, it is important to understand the hierarchical nature of systems and to realise that validation at one level can be part of verification at the next integration level.



## **5.2 Recommendations for AP5 and the OCVSD**

Given the conclusions made in this study several recommendations apply to the OCVSD and the more organisational description of the validation processes.

The definition of a requirement given in the IEEE documentation (Ref. [24]) makes a distinction between stakeholder needs (sometimes also called user requirements) and requirements. Stakeholder needs (raw requirements, objectives) may be inconsistent, technically or financially impossible, and may contain implementation details, design decisions or any other kind of statement on the system or function to be developed. The requirements analysis process transforms the identified needs into requirements. These requirements must be clear and consistent, without unnecessarily limiting the possible set of design solutions.

This insinuates that indeed there is a different understanding of the term ‘requirement’ at different levels of the process (strategy, programme, project, and exercise levels, see also VARTAN study in Ref. [38]) and for different actors within the process (operational experts or systems experts). Thus, it is recommended to investigate the issue of different levels of requirement and the way these requirements are either verified or validated in order to see whether there is a link between the type and level of requirement and the activity that must be performed to find evidence that the requirement is fulfilled. Furthermore, the requirements as such might be categorised according to rationale and priority (mandate, essential). This element should also be considered.

A possible way to start such an investigation would be the performance of interviews with relevant stakeholders, such as ANSPs, airlines, and manufacturers. Since some of the stakeholders take part in AP5 working meetings, the AP5 working group could perform these interviews and address relevant specialists at different levels of the organisation.

As mentioned above another factor that should be considered is the role of the stakeholder, who could be one of the mentioned actors in the validation process. For example, the project EMMA made a clear distinction between operational and technical requirements. Technical requirements were elaborated from the operational requirements by the relevant experts and split into air and ground domain requirements.

In summary, this means that interviews need to consider the type of stakeholder (operational domain), the organisational level and therefore the level of detail of the requirements, and the role of the stakeholder within the validation process. The results of the interviews should then be integrated and conclusions should be drawn on the processes that lead to the necessary evidence for the requirements to be fulfilled.



It is recommended to investigate how to categorise different levels of requirement and the way these requirements are either verified or validated in order to see whether there is a link between the type and level of requirement and the activity that must be performed to find evidence that the requirement is fulfilled. Furthermore, the requirements as such might be classified according to rationale and priority (mandate, essential). For this purpose interviews with industry (ANSPs) should be performed. This recommendation concerns AP5 activities, which consider the definition of a verification strategy, in which abovementioned issues play a major role.

As a further recommendation a list of questions for interviews with ANSPs is given below:

1. What is your role in the organisation/level in hierarchy/function description?
2. Could you briefly describe your current and recent activities in the development of new concepts and/or functions/systems?
3. Have you been involved in Validation or Verification activities?
  - a. What do you consider to be *Validation*?
  - b. What do you consider to be *Verification*?
4. What is to your opinion the role of requirements in these activities?
5. What process has been followed to capture and analyse these requirements?
  - a. Did it turn out be adequate?
  - b. What were the strong parts of the process?
  - c. What were the problems with the process, if any?
  - d. Would you use the same process again, or would you change it? Why, and how?
6. How were the requirements documented?
  - a. Was the documentation adequate?
  - b. Were you able to transfer all knowledge on the function/concept through these requirements?
  - c. If no, what additional means of documentation or knowledge transfer did you use?
  - d. What problems were encountered later in the development due to insufficient requirements documentation, if any?
7. What would you recommend other ANSPs who are developing concepts/systems/functions?
8. What pitfalls would you like to warn other ANSPs about?

The list above recommends interview questions concerning abovementioned issues that need to be investigated by AP5 to get to a definition of a verification strategy.

Another issue that was mentioned at several places in this document is the level of detail of specifications at the end of the final R&D phase V3 and the transfer of the information



contained in these specifications to industry, i.e. the transfer from phase V3 to V4. Since the validation activities are not necessarily in line with technology development, specifications might only address the operational system as such and consequences for technical enablers, sub-systems or system components would still need to be elaborated based on the specifications given. A similar issue concerns system architecture, i.e. the way and level of detail at which architecture and design issues are addressed within a specification document. Since any analysis of either of these topics was outside the scope of this study, the recommendation is given to perform additional studies on the expectations of industry regarding specifications of architecture, design and requirements of systems at the end of the R&D related phases in the validation life cycle.

It is recommended to perform additional studies (e.g. in the form of interviews with representative industries) on the expectations of industry regarding specifications of architecture, design and requirements of systems at the end of the R&D related phases in the validation life cycle. A better understanding of these issues might lead to a better transition from V3 to V4 and a better transfer of knowledge from R&D to operation.

This recommendation leads to another topic that has also been addressed during the survey of standards, namely stakeholder participation and, in particular, industry participation in R&D. According to standard IEEE 830 it is highly recommended to have a strong participation of industry in the early phases of the R&D life cycle in order to ensure that the specifications mentioned above really fulfil the expectations of the ATM industry that needs to bring products from R&D stages into operation.

Industry participation from the beginning of the life cycle is highly recommended (see also IEEE 830) in order to further mitigate transition problems between V3 and V4.

A related issue is the relationship between customer and supplier. In order to clearly understand the roles of stakeholders within the validation life cycle and the development life cycles for operational concepts and systems, it will be important to identify customer-supplier relationships. These can be determined by having a closer look at the requirements, which are formulated by the customer in order to bring about a desired result. The supplier will need to prove that the developed solution or system indeed complies with the requirement. Just as in the case of requirements, there will thus be different types of customer-supplier relationships. Therefore, a final recommendation considering OCVSD-related investigations would be to perform a study that looks at the role of industry in the early phases of the life cycle and the



different types of customer-supplier relationships that might lead to different types of specifications and documents at the end of phase V3. Such a study could be combined with a study on the expectations of industry regarding specifications.

It is recommended to perform a survey on typical relationships between ATM customers and suppliers. This should help to get a better understanding of the reasons for transition problems between V3 and V4.



### 5.3 Recommendations for the E-OCVM

The major recommendation regarding the E-OCVM is the proposed strategy for requirement development along the validation life cycle. The strategy consists of two major processes, Requirements Capturing and Requirements Analysis, which are supported by a third process, called Requirements Management:

- The Requirements Elicitation or Capture Process aims at eliciting the initial requirements from stakeholders (V0 and V1)
- The Requirements Analysis or Specification Process is performed throughout E-OCVM phases V1 to V3 to iteratively transform these raw requirements into a clear, consistent, and stable system specification that can be used as the basis for the industrialisation of the system (phases V4 and V5). As mentioned in the previous section the level of detail of such a specification and the related industry expectations still need to be investigated.
- The Requirements Management Process is responsible for maintaining the set of requirements throughout all the iterations, together with the rationale and supporting documentation for each of the changes made.

If the strategy described in this study is to be adopted it will be important to work on the issues already mentioned, namely industry participation in early life cycle phases and industry expectations for V4 as well as clear customer-supplier relationships and according requirement specifications. Especially the management process described and further detailed in the standards could help to establish a clearer understanding by enhancing traceability of the requirement development processes.

It is recommended to improve the E-OCVM (or OCVSD depending on the focus of the improvement) with descriptions of the proposed processes for elicitation/capture, analysis/specification and management of requirements. They should be integrated in the life cycle model as indicated in this study.

As suggested earlier in the analysis of actors and roles it is also recommended to identify the central role of a programme manager who could be a validation or development engineer and whose task it is to integrate all requirements development processes through requirements management, in order to oversee the complete system development process. It will be important that this manager is supported by a group of requirements engineers (or validation managers) who are each responsible for requirements at a certain level of detail (e.g. component, sub-system, system) and within a certain domain (operations, airborne systems, ground systems, FDPS, HMI, architecture etc.).



In order to improve the requirements management process and the consistency of system development, it is recommended to define the central role of a programme manager supervising all development streams (supported by managers on each system and domain level) in the E-OCVM.

For the analysis of requirements it is recommended to have a look at visualisation and modelling tools, such as UML, which may prove to be helpful regarding functional requirements. Such techniques could include use cases, function trees and matrices and could also evolve into complete operational scenarios. An additional study could investigate these techniques, define their role and value within the analysis process, and, if applicable, compare the results which can be obtained.

It is recommended to find an agreement on the tools to be suggested for requirements capturing and analysis processes (UML, use cases, scenarios) in the E-OCVM.

Requirements elicitation and capture processes are described in IEEE 15288 but also in ED-78A. In that regard, a recommendation derived from the EMMA project would be to follow a document structure of OSED, INTEROP and ORD documents (and further related processes and document regarding safety) whenever approval and certification processes for ATS supported by data communication is required. However, from the experience gained in the analysis of the project, it will be important to clearly document ED-78A processes such as the requirement elicitation process (OSEIC) and the qualification process, which actually is the complete process of finding evidence through verification and validation in phases V1 to V3.

It is recommended to make a suggestion to EUROCAE to improve ED-78A with more detail on requirement elicitation process and qualification process and/or describe these processes in the E-OCVM and have EUROCAE make a reference to the E-OCVM.

Regarding the system development processes it was concluded that they do not always run in parallel with validation and operational concept development processes and depend largely on the system or technology under investigation (technical enabler, component development, system development, architecture development). System development often also depends on contractual constraints, existing technology, and national or company interests.

This eventually leads to the identified problems in the transition from V3 to V4 as it is very difficult to define the level at which a system can be specified when there are several system



components and sub-systems being at different stages of development. Again requirements management and gradual refinement of requirements along the life cycle phases should offer the solution. However, this means that it will be necessary to ensure that all system components and sub-systems are verified against the requirements identified at the beginning of each life cycle phase and after each of the iterations within a life cycle phase before validation of the integrated system can occur. This is due to the hierarchical nature of system development which leads to the view of validation being verification at the system level, meaning that there are consecutive processes of building, verifying, integrating, and validating as has been shown in the analysis of the VICTORIA project.

It is recommended to consider consecutive processes of building, verifying, integrating and validating due to hierarchical nature of systems when developing a verification strategy within the E-OCVM framework.

For the development of a verification strategy within the E-OCVM this means, that it should be determined at which level of the concept the system is considered so complex and unpredictable regarding the operational outcome with humans in the loop, that it will be difficult to verify a concise requirement and that a hypothesis for a certain objective must be validated instead.

It is recommended to determine a level of system complexity and unpredictability, regarding the operational outcome of simulations with humans in the loop, which defines the domain of validation. In this domain it will be difficult to verify a concise requirement so that hypotheses for operational objectives must be validated instead. Defining such a boundary in general terms should help in establishing a verification strategy within the E-OCVM framework.

A recommendation that is related with this issue is to also perform a study on the NASA Technology Readiness Levels (TRL), which represent the life cycle of a certain technology, and determine how these readiness levels relate to the E-OCVM life cycle phases. Especially, the transition point from phase V3 to V4 should be investigated in order to see at which point there is a comparable transfer of knowledge required from one readiness level to another and if anything is stated regarding the relationship between customer and supplier in that regard. Obviously, such a relationship always involves risk. To a certain extent such risk can be mitigated by stating clear requirements. However, there will also be the risk of over-specification which leads to unwanted effects. Also, it should be interesting to investigate how the hierarchical nature of systems is represented in the readiness levels.



It is recommended to analyse the relation between NASA TRL and the E-OCVM life cycle phases in order to get better understanding of industrial development and the transition from V3 to V4. This should also improve the understanding of system and technology development processes running in parallel with OCV.

A valuable conclusion derived from the European R&D project analysis is that it is certainly helpful to produce an operational baseline on a high-level which clearly separates the generic operational requirements and the specific implementation guidelines (which are in fact local, detailed implementation requirements derived from the generic operation requirements). The FASTI baseline is seen as a good example of how to document requirements and how to maintain traceability between the more general operational requirements, and the implementation specific requirements resulting from validation exercises and pre-operational prototypes. This means that apart from specifying requirements at different levels of system hierarchy it will also be useful to specify them on different operational levels.

It is recommended to consider a guideline in the E-OCVM (or OCVSD depending on the audience) to have an operational baseline produced by an organisation like EUROCONTROL in order to manage system requirements on a higher level and to separate general from implementation-specific requirements (as suggested in FASTI).

Finally, it was concluded that one of the imminent topics addressed by SESAR, namely the definition of system architectures, is not addressed in the standards that have been surveyed. The reason for this fact is that the investigated standards had a focus on requirements engineering, so that architecture issues were only considered regarding adaptation requirements for systems or system components having to fit within the given architecture. Thus, it is recommended to devote another study to the analysis of architecture models currently discussed in SESAR, such as the enterprise and service-oriented architectures. Such an analysis should elaborate the consequences of architecture models for the requirements capture and analysis processes described in this study.

It is recommended to suggest to SESAR to perform a study of the impact of the considered architecture models on the OCVSD/E-OCVM life cycle phases and the proposed requirement development processes in order to mitigate risks in terms of mismatches in system requirements.



## References

- [1] **Bailey, Ian,**  
*A Simple Guide to Enterprise Architecture,*  
Model Futures Ltd., London, 2006,  
<http://www.modelfutures.com/>
  
- [2] **Boehm, Barry W.,**  
*A Spiral Model of Software Development and Enhancement,*  
Computer Magazine (pp. 61-72),  
IEEE, New York, May 1988
  
- [3] **Bowen, David,**  
*EUROCAE View of the Community Specification Development Process,*  
EUROCAE Presentation, May 2007
  
- [4] **Co-operative Approach to Air Traffic Services (CAATS),**  
*CAATS Best Practices Manual,*  
Deliverable 1.8,  
Isdefe, Madrid, October 2005
  
- [5] **Department of Defense of the United States of America (DoD),**  
Standard Practice for Defense Specifications (MIL-STD-961D),  
U.S. Department of Defense, Philadelphia, March 1995
  
- [6] **Erl, Thomas,**  
*SOA: Principles of Service Design,*  
Pearson, Prentice Hall - Professional Technical References, London, July 2007  
<http://www.soasystems.com/>
  
- [7] **European Organisation for Civil Aviation Equipment (EUROCAE),**  
*Software Considerations in Airborne Systems and Equipment Certification,*  
ED-12B,  
EUROCAE, Paris, December 1992



- [8] **European Organisation for Civil Aviation Equipment (EUROCAE),**  
*Guidelines for Approval of the Provision and Use of ATS Supported by Data Communications,*  
ED-78A,  
EUROCAE, Paris, December 2000
  
- [9] **European Organisation for Civil Aviation Equipment (EUROCAE),**  
*Certification Considerations for Highly Integrated or Complex Aircraft Systems,*  
ED-79,  
EUROCAE, Paris, April 1997
  
- [10] **European Organisation for Civil Aviation Equipment (EUROCAE),**  
*Design Assurance Guidance for Airborne Electronic Hardware,*  
ED-80,  
EUROCAE, Paris, April 2000
  
- [11] **European Space Agency (ESA),**  
*ESA ECSS-E-10 System Engineering,*  
European Co-operation for Space Standardisation, Noordwijk, April 1996
  
- [12] **European Space Agency (ESA),**  
*ESA ECSS-E-40 Software Engineering,*  
European Co-operation for Space Standardisation, Noordwijk, April 1999
  
- [13] **European Space Agency (ESA),**  
*Guide to Software Verification and Validation,*  
Issue 1, Revision 1,  
European Space Agency, March 1995
  
- [14] **EUROCONTROL,**  
*Task Requirement Sheet T07/11093DK,*  
EATM Framework Contract for Outside Assistance,  
EUROCONTROL, Brussels, July 2007



- [15] **EUROCONTROL**,  
*European Operational Concept Validation Methodology (E-OCVM)*,  
Edition 2.0,  
EUROCONTROL, Brussels, February 2007  
<http://www.eurocontrol.int/valfor/>
  
- [16] **EUROCONTROL**,  
*EUROCONTROL Aeronautical Lexicon*,  
EUROCONTROL OneSky Website,  
<https://extranet.eurocontrol.int/>
  
- [17] **EUROCONTROL**,  
*EATMP Glossary of Terms*,  
EUROCONTROL, Brussels, June 2004  
<http://www.eurocontrol.int/eatm/>
  
- [18] **FAA/EUROCONTROL Memorandum of Co-operation**,  
*Action Plan 5: Validation and Verification Strategy -  
Operational Concept Validation Strategy Document*,  
Edition 2.0,  
EUROCONTROL, Brussels, March 2007  
<http://www.eurocontrol.int/valfor/>
  
- [19] **First ATC Support Tools Implementation (FASTI)**,  
*First ATC Support Tools Implementation (FASTI) Baseline Description*,  
Edition 1.0,  
FASTI Operational Focus Group, Brussels, September 2006  
<http://www.eurocontrol.int/fasti/>
  
- [20] **First ATC Support Tools Implementation (FASTI)**,  
*First ATC Support Tools Implementation (FASTI) Operational Concept*,  
Edition 1.0,  
FASTI Operational Focus Group, Brussels, July 2006  
<http://www.eurocontrol.int/fasti/>



- [21] **First ATC Support Tools Implementation (FASTI),**  
*First ATC Support Tools Implementation (FASTI) Validation Plan,*  
Edition 1.0,  
EUROCONTROL, Brussels, December 2006  
<http://www.eurocontrol.int/fasti/>
  
- [22] **Federal Aviation Administration (FAA),**  
*FAA Acquisition System Toolset (FAST),*  
FAA Acquisition Management System (AMS),  
FAA, Washington, July 2007  
<http://fast.faa.gov/>
  
- [23] **Hasenbanck, Peter and Dr. Elmar Blatt,**  
*Improvement of E-OCVM: Transition V3 to V4 - Study Report,*  
Edition 1.0,  
ATSMconsult, Vienna, December 2006  
<http://www.eurocontrol.int/valfor/>
  
- [24] **Institute of Electrical and Electronics Engineers (IEEE),**  
*IEEE Standard Glossary of Software Engineering Terminology,*  
IEEE Std 610.12-1990,  
IEEE, New York, December 1990
  
- [25] **Institute of Electrical and Electronics Engineers (IEEE),**  
*IEEE Recommended Practice for Software Requirements Specifications,*  
IEEE Std 830-1998,  
IEEE, New York, October 1998
  
- [26] **Institute of Electrical and Electronics Engineers (IEEE),**  
*IEEE Guide for Developing System Requirements Specifications,*  
IEEE Std 1233-1998,  
IEEE, New York, December 1998



- [27] **Institute of Electrical and Electronics Engineers (IEEE),**  
*IEEE Guide for Information Technology - System Definition - Concept of Operations (ConOps) Document,*  
IEEE Std 1362-1998,  
IEEE, New York, March 1998
- [28] **Institute of Electrical and Electronics Engineers (IEEE),**  
*Industry implementation of International Standard ISO/IEC 12207:1995,*  
IEEE Std 12207.2-1997,  
IEEE, New York, April 1998
- [29] **Institute of Electrical and Electronics Engineers (IEEE),**  
*Adoption of ISO/IEC 15288:2002 Systems Engineering - System Life Cycle Processes,*  
IEEE Std 15288-2004,  
IEEE, New York, 2005
- [30] **Itier, J.B.,**  
*VICTORIA Program General Process,*  
VICTORIA D2.2.1,  
Airbus France, Toulouse, October 2001
- [31] **Itier, J.B.,**  
*Experimental System Test Plan,*  
VICTORIA D2.2.2.1,  
Airbus France, Toulouse, March 2002
- [32] **Jakobi, Jörn,**  
*EMMA SP1 Detailed Work Plan,*  
EMMA-1 Deliverable D1.0.1,  
DLR, Braunschweig, April 2005
- [33] **Jakobi, Jörn,**  
*EMMA-2 A-SMGCS Services Procedures and Operational Requirements (SPOR),*  
EMMA-2 Deliverable D1.1.1,  
DLR, Braunschweig, October 2007



- [34] **A Master ATM European Validation Plan (MAEVA),**  
*Validation Guideline Handbook (VGH),*  
Issue 3.0,  
Isdefe, Madrid, April 2004
  
- [35] **Mautray, Gilles,**  
*Project Development Plan,*  
VICTORIA O8.1,  
Thales Avionics, Toulouse, July 2001
  
- [36] **National Aeronautics and Space Administration (NASA),**  
*NASA VAMS Project Definitions,*  
NASA Ames, Moffett Field, August 2004  
<http://www.vams.arc.nasa.gov/>
  
- [37] **Teutsch, Jürgen,**  
*Project Management Plan RORI-OCV (D1),*  
NLR-Memorandum ATAP-2007-098,  
NLR, Amsterdam, September 2007
  
- [38] **Teutsch, Jürgen, Roy Jansen and Roalt Aalmoes,**  
*VARTAN Final Activity Report - Establishing and Automating Validation Reporting*  
*Templates in the E-OCVM Framework,*  
NLR-CR-2006-046,  
NLR, Amsterdam, April 2006
  
- [39] **Teutsch, Jürgen, Doris Dehn and Herman Nijhuis,**  
*EMMA Generic Verification and Validation Masterplan,*  
EMMA-1 Deliverable D6.1.1,  
NLR-CR-2007-826,  
NLR, Amsterdam, December 2005  
<http://www.dlr.de/emma/>



- [40] **Zachman, John A.**,  
*A Framework for Information Systems Architecture*,  
IBM Systems Journal, Vol. 38, Nos. 2&3,  
IBM, Los Angeles, 1999
- [41] **Zografos, Konstantinos, Frans van Schaik and Jürgen Teutsch**,  
*EMMA-2 Validation Plan*,  
EMMA-2 Deliverable D6.1.1,  
NLR, Amsterdam, December 2007



## Appendix A Analysis of Terminology

The definitions in this appendix can be found in the following references:

- European Operational Concept Validation Methodology (E-OCVM) [15]
- The EUROCONTROL Aeronautical Lexicon [16] with sub-references for:
  - EUROCONTROL Operational Concept Document (OCD)
  - Single European Sky ATM Research (SESAR)
  - Overall Target Architecture Activity (OATA)
  - EATMP Glossary of Terms (EATM) [17]
- The NASA VAMS Glossary of Terms (VAMS) [36]
- IEEE 610 Standard Glossary of Software Engineering Terminology (IEEE) [24]
- FAA Acquisition System Toolset (FAST) [22]

### A.1 Definition of ‘Operational Concept’

Source	Description
SESAR/OCD	A high level description of a set of defined ATM components and the manner in which they are organised and operated which meet a given set of high level user requirements. Comment: the operational concept is neither a description of the air navigation infrastructure, nor a technical system description nor a detailed description of how a particular functionality or technology could be used.
EATM	Broad outline of an operational structure able to meet a given set of high level user requirements. It comprises a consistent airspace organisation, general operational procedures and associated operational requirements for system support.
VAMS	An operational concept describes what a specific set of air transportation system capabilities does or will do to provide specific operational services to an identified set of system users. These operational services include: 1) Flight Planning, 2) Separation Assurance, 3) Situational Awareness & Advisory, 4) Navigation & Landing, 5) Traffic Management - Strategic Flow, 6) Traffic Management - Synchronization, 7) Airspace Management, 8) Emergency/Alerting, and 9) Infrastructure/Information Management. An operational concept may be limited to a sub-set of these services; for example, the operational concept might be: the air transportation system provides separation assurance between aircraft.
E-OCVM	Description of a set of ATM components and the manner in which they are configured and operated. The ATM Operational Concept should address a specific ATM problem. A statement of the operational concept, generally provided by the development team, should provide information on the



	actors involved and their tasks and responsibilities, enablers, events and the drivers of the events, processes and their relation to each other, airspace organisation, information flows and procedures.
--	--

**A.2 Definition of ‘Concept of Operations’**

Source	Description
SESAR/OCD	A detailed description of how an operational concept is applied. It identifies the functions and processes, and their corresponding interactions and information flows; concerned actors, their roles and responsibilities.
OATA	Documents the purpose of the ATM system (and sub-systems) and assists in the identification of system requirements and high-level business needs that the system will satisfy. It describes the operational concept supporting the system as well as its characteristics and behaviour from a user’s point of view. In doing so any R&D issues should be highlighted.

**A.3 Definition of ‘Concept of Use’**

Source	Description
FAST	The concept of use explains how new capabilities will function within the existing operational environment and how they will satisfy the service need. It defines key elements of the required capability and the roles and responsibilities of key participants (e.g., controllers, maintenance technicians, pilots). It explains operational issues that system engineers must understand when developing requirements; identifies procedural issues that may lead to operational change; and establishes a basis for evaluating benefits. If proposed alternative solutions are significantly different from each other, more than one concept of use may be required. The concept of use is recorded in the Exhibit 300 Program Baseline Attachment 1: Program Requirements.

**A.4 Definition of ‘Operational Procedures’**

Source	Description
EATM	ATC Operations Manuals, incorporating international, national and local rules and regulations, procedures and working practices.



### A.5 Definition of ‘Operational Requirements’

Source	Description
IEEE	<p>Operational means:</p> <ol style="list-style-type: none"> <li>1) Pertaining to a system or component that is ready for use in its intended environment.</li> <li>2) Pertaining to a system or component that is installed in its intended environment.</li> <li>3) Pertaining to the environment in which a system or component is intended to be used.</li> </ol> <p>Requirement means:</p> <ol style="list-style-type: none"> <li>1) A condition or capability needed by a user to solve a problem or achieve an objective.</li> <li>2) A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents.</li> <li>3) A documented representation of a condition or capability as in 1 or 2.</li> </ol>
FAST	Requirement: Conditions or capabilities that must be met or exceeded by a system or component to satisfy agency needs. Requirements form the basis for a contract, standard, specification, or other formally imposed document.
SESTAR	<p>Requirement: Characteristics that identify the accomplishment levels needed to achieve specific objectives for a given set of conditions. So requirements are statements that prescribe a function, an aptitude, a characteristic or a limitation to be met by the ATM/CNS system under given environmental conditions.</p> <p>Operational Requirement: A statement of the operational attributes of a system needed for the effective and/or efficient provision of air traffic services to users.</p>
EATM	Operational Requirement: A stipulated demand placed on someone to fulfil an operational Air Traffic Control / Management need. Instructions which define the opening and closing times of the sectors necessary to guarantee a safe and orderly traffic flow. They are determined by the traffic volume, distribution and complexity.

### A.6 Definition of ‘User Requirements’

Source	Description
EATM	Users: Users refer to the aggregate of organisations, people, automated systems, infrastructure, procedures, rules and information, which receive services from the EATCHIP Phase III ATM System, but are not part of it



	<p>(i.e. not owned and/or operated by the EATCHIP Phase III ATM System). There are two major categories of Users: aviation users and non-aviation users. The term ‘Aviation users’ refers to the aircraft operators. Typical examples of Non-aviation Users are: physically or functionally adjacent Air Navigation Systems, air defence, law enforcement agencies, customs, etc.</p>
--	---

**A.7 Definition of ‘Functional and Non-functional Requirements’**

Source	Description
EATM/ SESAR	<p>Functional Requirement: Operational requirements that determine what function a system should perform. They can usually be expressed by a verb applying to a type of data, e.g. display aircraft position.</p>
IEEE	<p>Functional Requirement: A requirement that specifies a function that a system or system component must be able to perform.</p> <p>Design Requirement: A requirement that specifies or constrains the design of a system or system component.</p> <p>Implementation Requirement: A requirement that specifies or constrains the coding or construction of a system or system component.</p> <p>Interface Requirement: A requirement that specifies an external item with which a system or system component must interact, or that sets forth constraints on formats, timing, or other factors caused by such an interaction.</p> <p>Performance Requirement: A requirement that imposes a condition on a functional requirement, e.g. a requirement that specifies the speed, accuracy, or memory usage with which a given function must be performed.</p> <p>Physical Requirement: A requirement that specifies a physical characteristic that a system or system component must possess, e.g. material, shape, size, weight.</p>
SESAR	<p>Design and Development: Set of processes that transform requirements into specified characteristics or into specifications of a product, process or system.</p> <p>Performance Requirement: The extent to which a mission or function must be executed, generally measured in terms of quantity, quality, coverage, timeliness or readiness.</p>
FAST	<p>Functional Baseline: A functional baseline is the initially approved documentation describing a system’s or item’s functional, interoperability, and interface characteristics, and the verification required to demonstrate the achievement of those characteristics.</p>



**A.8 Definition of ‘System Specifications’**

Source	Description
IEEE	<p>Specification: A document that specifies, in a complete, precise, verifiable manner, the requirements, design, behaviour, or other characteristics of a system or component, and, often, the procedures for determining whether these provisions have been satisfied.</p> <p>Requirements Specification: A document that specifies the requirements for a system or component. Typically included are functional requirements, performance requirements, interface requirements, design requirements and development standards.</p> <p>Functional Specification: A document that specifies the functions that a system or component must perform. Often part of a requirements specification.</p>



A.9 Zachman Enterprise Architecture Framework™

	WHAT	HOW	WHERE	WHO	WHEN	WHY
	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION
<b>SCOPE</b> {contextual}  Planner	List of Things Important to the Business Entity = Class of Business Thing	List of Processes the Business Performs Process = Class of Business Process	List of Locations in Which the Business Operates Node = Major Business Location	List of Organizations Important to the Business People = Major Organizational Unit	List of Events/Cycles Significant to the Business Time = Major Business Event/Cycle	Lists of Business Goals/Strategies Ends/Means = Major Business Goal/Strategy
<b>BUSINESS MODEL</b> {conceptual}  Owner	e.g., Semantic Model Entity = Business Entity Relationship = Business Relationship	e.g., Business Process Model Process = Business Process I/O = Business Resources	e.g., Business Logistics System Node = Business Location Link = Business Linkage	e.g., Work Flow Model People = Organization Unit Work = Work Product	e.g., Master Schedule Time = Business Event Cycle = Business Cycle	e.g., Business Plan End = Business Objective Means = Business Strategy
<b>SYSTEM MODEL</b> {logical}  Designer	e.g., Logical Data Model Entity = Data Entity Relationship = Data Relationship	e.g., Application Architecture Process = Application Function I/O = User Views	e.g., Distributed System Architecture Node = (S Function (Processor, Storage, etc) Link = Line Characteristics	e.g., Human Interface Architecture People = Role Work = Deliverable	e.g., Processing Structure Time = System Event Cycle = Processing Cycle	e.g., Business Rule Model End = Structural Assertion Means = Action Assertion
<b>TECHNOLOGY MODEL</b> {physical}  Builder	e.g., Physical Data Model Entity = Segment/Table/etc Relationship = Pointer/Key/etc	e.g., System Design Process = Computer Function I/O = Data Elements/Sets	e.g., Technology Architecture Node = HW/System Software Link = Line Specifications	e.g., Presentation Architecture People = User Work = Screen Formats	e.g., Control Structure Time = Execute Cycle = Component Cycle	e.g., Rule Design End = Condition Means = Action
<b>DETAILED REPRESENTATIONS</b> {out-of-context}  Subcontractor	e.g., Data Definition Entity = Field Relationship = Address	e.g., Program Process = Language Statement I/O = Control Block	e.g., Network Architecture Node = Address Link = Protocol	e.g., Security Architecture People = Identity Work = Job	e.g., Timing Definition Time = Interrupt Cycle = Machine Cycle	e.g., Rule Specification End = Sub-condition Means = Step
<b>FUNCTIONING ENTERPRISE</b>	e.g.: DATA	e.g.: FUNCTION	e.g.: NETWORK	e.g.: ORGANIZATION	e.g.: SCHEDULE	e.g.: STRATEGY



## Appendix B Analysis of System Engineering Standards

### B.1 IEEE Std 610.12-1990 (Glossary of Software Engineering Terminology)

Applicable terms and definitions from this document:

**Software life cycle:** The period of time that begins when a software product is conceived and ends when the software is no longer available for use. The software lifecycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and checkout phase, operation and maintenance phase, and, sometimes, retirement phase.

Note: These phases may overlap or be performed iteratively.

Contrast with: **software development cycle.**

**Software development cycle:** The period of time that begins with the decision to develop a software product and ends when the software is delivered. This cycle typically includes a requirements phase, design phase, implementation phase, test phase, and, sometimes, an installation and checkout phase.

Contrast with: **software life cycle.**

Notes:

- (1) The phases listed above may overlap or be performed iteratively, depending upon the software development approach used.
- (2) This term is sometimes used to mean a longer period of time, either the period that ends when the software is no longer being enhanced by the developer, or the entire software life cycle.

**Concept phase:**

- (1) (IEEE Std. 1002-1987) The period of time in the software development cycle (*Note: software life cycle? see definitions*) during which the user needs are described and evaluated through documentation (for example, statement of needs, advance planning report, project initiation memo, feasibility studies, system definition, documentation, regulations, procedures, or policies relevant to the project).
- (2) (IEEE Std 1012-1986) The initial phase of a software development project, in which the user needs are described and evaluated through documentation (for example, statement of needs, advance planning report, project initiation memo, feasibility studies, system definition, documentation, regulations, procedures, or policies relevant to the project).



**Requirements phase:** The period of time in the software life cycle during which the requirements for a software product are defined and documented.

**Requirements specification:** A document that specifies the requirements for a system or component. Typically included are functional requirements, performance requirements, interface requirements, design requirements, and development standards.

**Software requirements specification (SRS):** Documentation of the essential requirements (functions, performance, design constraints, and attributes) of the software and its external interfaces (IEEE Std 1012-1986).

**Requirement:**

- (1) A condition or capability needed by a user to solve a problem or achieve an objective.
- (2) A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents.
- (3) A document representation of a condition or capability as in (1) or (2).

**Functional requirement:** A requirement that specifies a function that a system or system component must be able to perform.

Contrast with: design requirement, implementation requirement, interface requirement, performance requirement, and physical requirement.

**Design requirement:** A requirement that specifies or constrains the design of a system or system component.

Contrast with: functional requirement, implementation requirement, interface requirement, performance requirement, and physical requirement.

**Implementation requirement:** A requirement that specifies or constrains the coding or construction of a system or system component.

Contrast with: design requirement, functional requirement, interface requirement, performance requirement, and physical requirement.

**Interface requirement:** A requirement that specifies an external item with which a system or system component must interact, or that sets forth constraints on formats, timing, or other factors caused by such an interaction.

Contrast with: design requirement, functional requirement, implementation requirement, performance requirement, and physical requirement.



**Performance requirement:** A requirement that imposes conditions on a functional requirement; for example a requirement that specifies the speed, accuracy, or memory usage with which a given function must be performed.

Contrast with: design requirement, functional requirement, implementation requirement, interface requirement, physical requirement.

**Physical requirement:** A requirement that specifies a physical characteristic that a system or system component must possess; for example, material, shape, size, weight.

Contrast with: design requirement, functional requirement, implementation requirement, interface requirement, performance requirement.

*General note: The difference between software life cycle and software development cycle at the front is the concept phase. The phases V0 to V3 of the E-OCVM seem to match this concept phase, maybe including (part of) the requirements phase. Therefore, a more in-depth investigation of the role of requirements in this concept phase, or the role of the concept phase in establishing requirements should give an idea of a possible strategy for handling of requirements in the early E-OCVM phases. Therefore, in reviewing other standards, we should concentrate on this part of the software life cycle.*

*Note (2): The terms ‘Operational concept’, ‘Concept of operations’, ‘Concept of use’, ‘Operational requirements’, ‘User requirements’, ‘Non-functional requirements’, ‘System requirements’, ‘Operational procedures’, ‘System specification’, ‘Technical or technology requirements’ do not appear as such in the references.*

## **B.2 IEEE Std 1362-1998 (Guide for IT System Definition and ConOps)**

Abstract: The format and contents of a concept of operations (ConOps) document are described. A ConOps is a user-oriented document that describes system characteristics for a proposed system from the users’ viewpoint. The ConOps document is used to communicate overall quantitative and qualitative system characteristics to the user, buyer, developer, and other organizational elements (for example, training, facilities, staffing, and maintenance). It is used to describe the user organization(s), mission(s), and organizational objectives from an integrated systems point of view.



From the foreword (introduction to the guide, not actually part of it):

(p. iii) This guide does not specify the exact techniques to be used in developing the ConOps document, but it does provide approaches that might be used. Each organization that uses this guide should develop a set of practices and procedures to provide detailed guidance for preparing and updating ConOps documents. These detailed practices and procedures should take into account the environmental, organizational, and political factors that influence application of the guide.

...

The ConOps approach provides an analysis activity and a document that bridges the gap between the user's needs and visions and the developer's technical specifications. In addition, the ConOps document provides the following:

- A means of describing a user's operational needs without becoming bogged down in detailed technical issues that shall be addressed during the systems analysis activity.
- A mechanism for documenting a system's characteristics and the user's operational needs in a manner that can be verified by the user without requiring any technical knowledge beyond that required to perform normal job functions.
- A place for users to state their desires, visions, and expectations without requiring the provision of quantified, testable specifications. For example, the users could express their need for a 'highly reliable' system, and their reasons for that need, without having to produce a testable reliability requirement. [In this case, the user's need for 'high reliability' might be stated in quantitative terms by the buyer prior to issuing a request for proposal (RFP), or it might be quantified by the developer during requirements analysis. In any case, it is the job of the buyer and/or the developer to quantify users' needs.]
- A mechanism for users and buyer(s) to express thoughts and concerns on possible solution strategies. In some cases, design constraints dictate particular approaches. In other cases, there may be a variety of acceptable solution strategies. The ConOps document allows users and buyer(s) to record design constraints, the rationale for those constraints, and to indicate the range of acceptable solution strategies.

...

(p. iv) Users who formerly applied MIL-STD-498, Software Development and Documentation, and related standards will find that the ConOps document described in this guide is very similar to the operational concept description (OCD) included in MIL-STD-498.

...

Users of EIA/IEEE J-STD-016-1995, EIA/IEEE Interim Trial-Use Standard for Information Technology Software Life Cycle Processes Software Development Acquirer - Supplier Agreement will find that the ConOps document described in this guide is substantively identical to the OCD included in EIA/IEEE JSTD-016-1995.



...

(p. v) Developing the initial version of the ConOps document should be one of the first activities completed on a software project. As the project evolves, the nature of the work to be done and details of the work will be better understood. The ConOps document should be updated periodically to reflect the evolving situation. Thus, each version of the document should be placed under configuration control.

...

Use of a ConOps document was first documented in Lano, R. J., 'A Structured Approach for Operational Concept Formulation,' TRW SS-80-02, TRW Defense and Space Systems Group, Redondo Beach, CA, 1980. In 1992 the Software Systems Technical Committee of the American Institute of Aeronautics and Astronautics (AIAA), developed a standard for an OCD.

From the guide itself:

(p.2-3, definitions)

3.4 concept of operations (ConOps) document: A user-oriented document that describes a system's operational characteristics from the end user's viewpoint.

Synonym: operational concept description (OCD).

3.5 constraint: An externally imposed limitation on system requirements, design, or implementation or on the process used to develop or modify a system.

3.10 functionality: The capabilities of the various computational, user interface, input, output, data management, and other features provided by a product.

3.17 scenario: (A) A step-by-step description of a series of events that may occur concurrently or sequentially. (B) An account or synopsis of a projected course of events or actions.

3.26 user need: A user requirement for a system that a user believes would solve a problem experienced by the user.

(p.5, contents of ConOps)

Title page

Revision chart

Preface

Table of contents

List of figures

List of tables

1. Scope

1.1 Identification

1.2 Document overview

1.3 System overview



- 2. Referenced documents
- 3. Current system or situation
  - 3.1 Background, objectives, and scope
  - 3.2 Operational policies and constraints
  - 3.3 Description of the current system or situation
  - 3.4 Modes of operation for the current system or situation
  - 3.5 User classes and other involved personnel
  - 3.6 Support environment
- 4. Justification for and nature of changes
  - 4.1 Justification of changes
  - 4.2 Description of desired changes
  - 4.3 Priorities among changes
  - 4.4 Changes considered but not included
- 5. Concepts for the proposed system
  - 5.1 Background, objectives, and scope
  - 5.2 Operational policies and constraints
  - 5.3 Description of the proposed system
  - 5.4 Modes of operation
  - 5.5 User classes and other involved personnel
  - 5.6 Support environment
- 6. Operational scenarios
- 7. Summary of impacts
  - 7.1 Operational impacts
  - 7.2 Organizational impacts
  - 7.3 Impacts during development
- 8. Analysis of the proposed system
  - 8.1 Summary of improvements
  - 8.2 Disadvantages and limitations
  - 8.3 Alternatives and trade-offs considered
- 9. Notes
- Appendices
- Glossary

(p.6, Document overview, section 1.2 of ConOps doc.)

The purposes of a ConOps document will, in most cases, be:

- To communicate the user's needs for and expectations of the proposed system to the buyer and/or developer; or



- To communicate the buyer's or developer's understanding of the users' need and how the system shall operate to fulfil those needs.

However, a ConOps document might also serve other purposes, such as building consensus among several user groups, among several buyer organisations, and/or among several developers.

*Note: The structure of sections 3 (current system) and 5 (proposed system) are the same. For the current work, subsections 3.2 (5.2), operational policies and constraints, and subsections 3.3 (5.3), description of the system or situation, are the most relevant. The contents of the x.3 section should include the following items:*

(p.7)

- a) The operational environment and its characteristics;
- b) Major system components and the interconnection among those components;
- c) Interfaces to external systems or procedures;
- d) Capabilities, functions, and features of the current system;
- e) Charts and accompanying descriptions depicting inputs, outputs, data flows, control flows, and manual and automated processes sufficient to understand the current system or situation from the user's point of view;
- f) Cost of system operations;
- g) Operational risk factors;
- h) Performance characteristics, such as speed, throughput, volume, frequency;
- i) Quality attributes, such as: availability, correctness, efficiency, expandability, flexibility, interoperability, maintainability, portability, reliability, reusability, supportability, survivability, and usability; and
- j) Provisions for safety, security, privacy, integrity, and continuity of operations in emergencies.

*Note (cont'd): In addition to these sections, the section 4, justification for and nature of changes, seems also very relevant. The contents of the various subsections should be:*

(p.9, justification for changes, section 4.1 of ConOps:)

This sub-clause should:

- a) Briefly summarise new or modified aspects of the user needs, missions, objectives, environments, interfaces, personnel, or other factors that require a new or modified system;
- b) Summarise the deficiencies or limitations of the current system or situation that make it unable to respond to new or changed factors; and



- c) Provide justification for a new or modified system.
  - 1) If the proposed system is to meet a new opportunity, describe the reasons why a new system should be developed to meet this opportunity.
  - 2) If the proposed system improves a current operation, describe the rationale behind the decision to modify the existing system (e.g., to reduce life cycle costs or improve personnel efficiency).
  - 3) If the proposed system implements a new functional capability, explain why this function is necessary.

(p.10, description of desired changes, section 4.2 of ConOps:)

This description should include the following, as appropriate:

- a) Capability changes. Description of the functions and features to be added, deleted, and modified in order for the new or modified system to meet its objectives and requirements.
- b) System processing changes. Description of the changes in the process or processes of transforming data that will result in new output with the same data, the same output with new data, or both.
- c) Interface changes. Description of changes in the system that will cause changes in the interfaces and changes in the interfaces that will cause changes in the system.
- d) Personnel changes. Description of changes in personnel caused by new requirements, changes in user classes, or both.
- e) Environment changes. Description of changes in the operational environment that will cause changes in the system functions, processes, interfaces, or personnel and/or changes that should be made in the environment because of changes in the system functions, processes, interfaces, or personnel.
- f) Operational changes. Description of changes to the user's operational policies, procedures, methods, or daily work routines caused by the above changes.
- g) Support changes. Description of changes in the support requirements caused by changes in the system functions, processes, interfaces, or personnel and/or changes in the system functions, processes, interfaces, or personnel caused by changes in the support environment.
- h) Other changes. Description of other changes that will impact the users, but that do not fit under any of the above categories.

(p.10, priorities among changes, 4.3 of ConOps:)

... should classify and prioritise the features of the proposed system.

- a) Essential features. Features that shall be provided by the new or modified system. The impacts that would result if the features were not implemented should be explained for each essential feature.



- b) Desirable features. Features that should be provided by the new or modified system. Desirable features should be prioritised. Reasons why the features are desirable should be explained for each desirable feature.
- c) Optional features. Features that might be provided by the new or modified system. Optional features should be prioritized. Reasons why the features are optional should be explained for each optional feature.

(p.10, changes considered but not included, 4.4 of ConOps:)

This sub-clause identifies changes and new features considered but not included in 4.2 of the ConOps document, and the rationale for not including them. By describing changes and features considered but not included in the proposed system, the authors document the results of their analysis activities. This information can be useful to other personnel involved with system development, whether it be users, buyers, or developers should they want to know if a certain change or feature was considered, and if so, why it was not included. In software especially, there are few, if any, outward signs of what has been changed, improved or is still unsafe or unsecure (e.g., in certain scenarios or workarounds).

(p.11, assumptions and constraints, 4.5 of ConOps:)

This sub-clause describes any assumptions or constraints applicable to the changes and new features identified in this clause. This should include all assumptions and constraints that will affect users during development and operation of the new or modified system. An assumption is a condition that is taken to be true. An example of an assumption is that the system workload will double over the next two years, thus a new system with higher performance is required. A constraint is an externally imposed limitation placed on the new or modified system or the processes used to develop or modify the system. Examples of constraints include external interface requirements, and limits on schedule and budget.

*Note (cont'd): In section 6 of the ConOps, the operational scenarios have to be described. At first glance, these seem close to what the E-OCVM calls cases. We should investigate this more.*

(p.14, Operational Scenarios:)

A scenario is a step-by-step description of how the proposed system should operate and interact with its users and its external interfaces under a given set of circumstances. Scenarios should be described in a manner that will allow readers to walk through them and gain an understanding of how all the various parts of the proposed system function and interact. The scenarios tie together all parts of the system, the users, and other entities by describing how they interact. Scenarios may also be used to describe what the system should not do.



Scenarios should be organised into clauses and sub-clauses, each describing an operational sequence that illustrates the roles of the system, its interactions with users, and interactions with other systems. Operational scenarios should be described for all operational modes and all classes of users identified for the proposed system. Each scenario should include events, actions, stimuli, information, and interactions as appropriate to provide a comprehensive understanding of the operational aspects of the proposed system. Prototypes, storyboards, and other media, such as video or hypermedia presentations, may be used to provide part of this information.

In most cases, it will be necessary to develop several variations of each scenario, including one for normal operation, one for stress load handling, one for exception handling, one for degraded mode operation, etc.

Scenarios play several important roles. The first is to bind together all of the individual parts of a system into a comprehensible whole. Scenarios help the readers of a ConOps document understand how all the pieces interact to provide operational capabilities. The second role of scenarios is to provide readers with operational details for the proposed system; this enables them to understand the users' roles, how the system should operate, and the various operational features to be provided.

Scenarios can also support the development of simulation models that help in the definition and allocation of derived requirements, identification, and preparation of prototypes to address key issues.

In addition, scenarios can serve as the basis for the first draft of the users' manual, and as the basis for developing acceptance test plans. The scenarios are also useful for the buyer and the developer to verify that the system design will satisfy the users' needs and expectations. Scenarios can be presented in several different ways. One approach is to specify scenarios for each major processing function of the proposed system. Using this approach, this clause would contain one sub-clause for each process. Each sub-clause would then contain several more lower-level sub-clauses, one for each scenario supported by that process. An alternative approach is to develop thread-based scenarios, where each scenario follows one type of transaction type through the proposed system. In this case, each sub-clause would contain one scenario for each interaction type, plus scenarios for degraded, stress loaded, and back-up modes of operation. Other alternatives include following the information flow through the



system for each user capability, following the control flows, or focusing on the objects and events in the system.

Scenarios are an important component of a ConOps, and should therefore receive substantial emphasis. The number of scenarios and level of detail specified will be proportional to the perceived risk and the criticality of the project.

*Note (cont'd): Sections 7 and 8 of the ConOps document (summary of impact and analysis of proposed system) should contain information that is presumably gathered during the E-OCVM phases. Therefore, the ConOps could maybe become the linking document in which all information is gathered and maintained. After each step of the E-OCVM it is updated with latest information. At the end of the Phase V3, it should then contain the best possible description of the system as it should be built. As such, it can be given to industry (phase V4) as a background document to explain the requirements (or as 'THE requirements document'?).*

(p. 15, summary of impacts, section 7 of ConOps:)

This clause describes the operational impacts of the proposed system on the users, the developers, and the support and maintenance organizations. It also describes the temporary impacts on users, buyers, developers, and the support and maintenance organizations during the period of time when the new system is being developed, installed, or trained on.

This information is provided in order to allow all affected organizations to prepare for the changes that will be brought about by the new system and to allow for planning of the impacts on the buyer agency or agencies, user groups, and the support maintenance organizations during the development of, and transition to the new system.

(p.16, analysis of the proposed system, section 8 of ConOps:)

This clause provides an analysis of the benefits, limitations, advantages, disadvantages, and alternatives and trade-offs considered for the proposed system.

Note Yves: The authors of the ConOps guide have added an appendix to show that the ConOps guide closely meets the requirements for such a document in the IEEE 12207 standard, which will be reviewed later. Some overlap in the description of the IEEE 12207 and the description of IEEE 1362 may therefore occur.



### **B.3 IEEE Std 1233-1998 (Guide for System Requirements Specifications)**

(p.4) A System Requirements Specification (SyRS) has traditionally been viewed as a document that communicates the requirements of the customer to the technical community who will specify and build the system. The collection of requirements that constitutes the specification and its representation acts as the bridge between the two groups and must be understandable by both the customer and the technical community.

(p.4) The SyRS presents the results of the definition of need, the operational concept, and the system analysis tasks. As such, it is a description of what the system's customers expect it to do for them, the system's expected environment, the system's usage profile, its performance parameters, and its expected quality and effectiveness.

....

The presentation of the SyRS should take a form that is appropriate for its intended use. This can be a paper document, models, prototypes, other non-paper document representations, or any combination.

(p.5) Purpose:

The purpose of the SyRS is to provide a 'black-box' description of what the system should do, in terms of the system's interactions or interfaces with its external environment. The SyRS should completely describe all inputs, outputs, and required relationships between inputs and outputs. The SyRS organizes and communicates requirements to the customer and the technical community.

(p.6) Intended use:

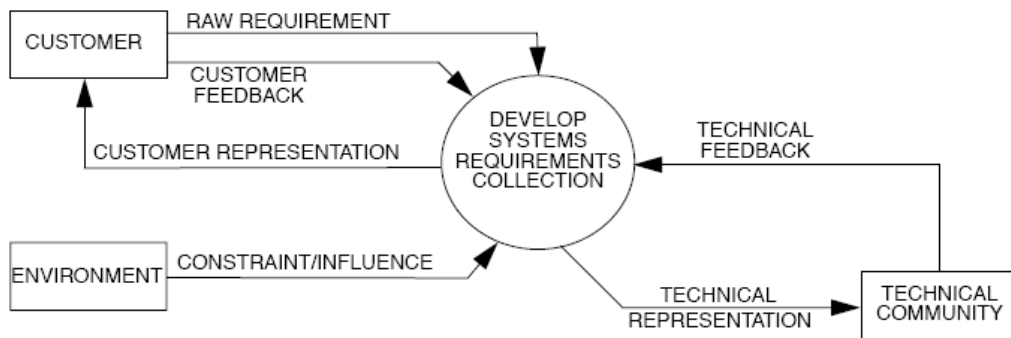
The recommended uses of the SyRS, which vary as the development cycle progresses, are as follows:

- a) During systems design, requirements are allocated to subsystems, hardware, software, operations, and other major components of the system.
- b) The SyRS is utilized in constructing the resulting system. The SyRS is also used to write appropriate system verification plans. If the system contains hardware and software, then the hardware test plan and software test plan are also generated from the system requirements.
- c) During the implementation phase, test procedures will be defined from the SyRS.
- d) During the validation process, validation procedures based on the SyRS are used to provide the customer with a basis for acceptance of the system.

*Note: The SyRS seems therefore a document for phases V4 and V5 and not in the scope of the OCVM. However, the process of getting to a SyRS, as described in this standard, seems to contain many valuable lessons which may be appropriate to the study at hand.*

(p.7) SyRS development process overview:

This clause provides an overview of the steps in the SyRS development process. The system requirements development process, in general, interfaces with three external agents - the customer, the environment, and the technical community. Each of the external agents is described in the text below. Figure 1 shows the interactions among the various agents necessary to develop an SyRS.



**Figure 1 – Context for developing an SyRS**

(p.7/8) Raw requirements:

Prior to the SyRS process the customer has an idea for a system, for a process improvement, or for a problem to be solved. At this point, any initial concept for a system may be imprecise and unstructured. Requirements will often be intermingled with ideas and suggestions for potential designs. These raw requirements are often expressed in initiating documents similar to the following:

- a) Concept of operations. This type of document focuses on the goals, objectives, and general desired capabilities of the potential system without indicating how the system will be implemented to actually achieve the goals.
- b) System concept. This type of document includes concept of operations information, but will also include a preliminary interface design for the system and other explicit requirements.

....

*Note: Description of these documents useable for main document?*



(p.11) 6.1 Definition of a well-formed requirement

As previously defined, a well-formed requirement is a statement of system functionality (a capability) that can be validated, that must be met or possessed by a system to solve a customer problem or to achieve a customer objective, and that is qualified by measurable conditions and bounded by constraints.

This definition helps in the classification of general customer requirements. Requirements can be taken from customer needs and can be derived from technical analysis. The definition provides a means for distinguishing between requirements as capabilities and the attributes of those requirements (conditions and constraints). Constraints may be functional or non-functional. An example of a non-functional constraint might be that the system is to be painted a particular shade of blue solely for non-required decorative purposes.

This guide recommends that system implementation process requirements, such as mandating a particular design methodology, not be included in a SyRS. Process requirements should be captured in other system controlling technical documentation such as quality plans, contracts, or statements of work.

(p.14) 6.4 Pitfalls

Some pitfalls to avoid when building well-formed requirements are as follows:

- a) Design and implementation. There is a tendency on the part of analysts and customers who are defining requirements to include design and implementation decisions along with the requirements statements. Such information may still be important. In this case, the information should be documented and communicated in some other form of documentation in order to aid in design and implementation.
- b) Over-specification.
  - 1) Requirements that express an exact commercial system set or a system that can be bought rather than made (these are not an expression of what the system should do);
  - 2) Requirements that state tolerances for items deep within the conceptual system (frequently stated as error requirements at very low levels);
  - 3) Requirements that implement solutions (requirements state a need).
- c) Over-constrained. Requirements with unnecessary constraints. (For example, if a system must be able to run on rechargeable batteries, a derived requirement might be that the time to recharge should be less than 3 h. If this time is too restrictive and a 12 h recharge time is sufficient, potential solutions are eliminated.)
- d) Unbounded.
  - 1) Requirements making relative statements. (These requirements cannot be verified and may only need to be restated. For example, the requirement to ‘minimize noise’ may be restated as ‘noise levels should not exceed...’)



- 2) Requirements that are open-ended (frequently stated as ‘including, but not limited to...’ or lists ending in ‘etc.’).
- 3) Requirements making subjective or vague statements (frequently contain terms such as ‘user friendly,’ ‘quick response time,’ or ‘cost effective’).
- e) Assumptions.
  - 1) Requirements based on undocumented assumptions. (The assumption should be documented as well as the requirement.)
  - 2) Requirements based on the assumption that a particular standard or system undergoing development will reach completion. (The assumption and an alternative requirement should be documented.)

**B.4 ESA standard ECSS-E-40 - Software**

*Note: In general, this standard assumes that a system specification at a higher level is already available before software is addressed. Therefore this standard is less applicable to the scope of this research. Nevertheless, the standard contains some descriptions and observations that may be useful to this research. These specific parts of the standard are copied below.*

(p.23) 4.2.4 Software requirements and architecture engineering process

The software requirements and architecture engineering process consists of the elaboration of the technical specification (TS), which is the supplier’s response to the requirements baseline. This process can start in parallel or after the elaboration of the requirements baseline. The software product tree is defined by this process. The technical specification contains a precise and coherent definition of functions and performances for all levels of the software to be developed. The preliminary interface control document (ICD) is generated by this process.

During the software requirements and architecture engineering process, the result of all significant trade-offs, feasibility analyses, make-or-buy decisions and supporting technical assessments are documented in a design justification file (DJF).

The software requirements and architecture engineering process is completed by the preliminary design review (PDR). The input to the PDR is the technical specification, preliminary ICD and the DJF. The software architectural design is reviewed at the PDR.



*Note: This use of a Design Justification File (DJF) could be very useful in phases V2 and V3 of the E-OCVM to maintain a history of decisions, and their rationale, during the prototyping and experiment activities. Together with the final set of requirements at the end of V3, this DJF could be a deliverable to V4 to smoothen this transition.*

(p. 31-32) Chapter 5: Requirements

Some requirements depend on the nature of the software. This is explicitly mentioned in the requirements. They include:

...

- Man-machine interface

Software projects that include the development of a significant interactive direct interface to a human user or operator lead to the involvement of the specialized software engineering and human factor disciplines covering this field.

The reason for the special sub-clauses is that it is not feasible to specify or design modern MMI technology (e.g. graphical user interfaces, and multi-layered choice menus) using conventional software engineering documentation methods. The non-linear and multi-dimensional nature of modern MMI cannot be described adequately using only two-dimensional documents that by nature are linear in structure. This is very similar to other systems with significant human factor requirements, such as cars, airplanes, and buildings. In those cases a mock-up or model is implemented during the requirements and architecture engineering process. An analogous approach in software engineering applies for software with extensive human interaction requirements.

For any MMI development a scenario--driven requirements analysis can be performed. The term MMI used in the corresponding sub-clauses also includes customization of COTS supplied MMI.

(p. 43-45)

The software requirements and architecture engineering process consists of the following activities:

- Software requirements analysis;
- Software architectural design.

...

[Software requirements analysis is...] establishment and documentation of software requirements. The supplier shall establish and document software requirements, including the software quality requirements, as part of the technical specification.

...



[Software architectural design is...] transform the requirements for the software item into an architecture that describes its top-level structure and identifies the software components, ensuring that all the requirements for the software item are allocated to its software components and later refined to facilitate detailed design.

## **B.5 IEEE 15288 on systems engineering**

### **B.5.1 Stakeholder Requirements Definition Process**

#### Purpose of the Stakeholder Requirements Definition Process

The purpose of the Stakeholder Requirements Definition Process is to define the requirements for a system that can provide the services needed by users and other stakeholders in a defined environment.

It identifies stakeholders, or stakeholder classes, involved with the system throughout its life cycle, and their needs and desires. It analyzes and transforms these into a common set of stakeholder requirements that express the intended interaction the system will have with its operational environment and that are the reference against which each resulting operational service is validated in order to confirm that the system fulfils needs.

#### Stakeholder Requirements Definition Process Outcomes

As a result of the successful implementation of the Stakeholder Requirements Definition Process:

- a) The required characteristics and context of use of services are specified.
- b) The constraints on a system solution are defined.
- c) Traceability of stakeholder requirements to stakeholders and their needs is achieved.
- d) The basis for defining the system requirements is described.
- e) The basis for validating the conformance of the services is defined.
- f) A basis for negotiating and agreeing to supply a service or product is provided.

#### Stakeholder Requirements Definition Process Activities

The project shall implement the following activities in accordance with applicable organization policies and procedures with respect to the Stakeholder Requirements Definition Process.

- a) Identify the individual stakeholders or stakeholder classes who have a legitimate interest in the system throughout its life cycle.



NOTE: This includes, but is not limited to, users, supporters, developers, producers, trainers, maintainers, disposers, acquirer and supplier organizations, regulatory bodies and members of society. Where direct communication is not practicable, e.g. consumer products and services, representatives or designated proxy stakeholders are selected, e.g. marketing.

- b) Elicit stakeholder requirements.

NOTE: Stakeholder requirements are expressed in terms of the needs, wants, desires, expectations and perceived constraints of identified stakeholders. They are expressed in terms of a model that may be textual or formal, that concentrates on system purpose and behaviour, and that is described in the context of the operational environment and conditions. A product quality model, such as found in ISO/IEC 9126, is useful for aiding this activity. Stakeholder requirements include the needs and requirements imposed by society, the constraints imposed by an acquiring organization and the capabilities and limiting characteristics of operator staff. Exclude unjustified constraints on a solution. It is useful to cite sources, including solicitation documents or agreements, and, where possible, their justification and rationale, and the assumptions of stakeholders and the value they place on the satisfaction of their requirements. For key stakeholder needs, the measures of effectiveness are defined so that operational performance can be measured and assessed.

- c) Define the constraints on a system solution that are unavoidable consequences of existing agreements, management decisions and technical decisions.

NOTE: These may result from 1) instances or areas of stakeholder-defined solution 2) implementation decisions made at higher levels of system hierarchical structure 3) required use of defined enabling systems, resources and staff.

- d) Define a representative set of activity sequences to identify all required services that correspond to anticipated operational and support scenarios and environments.

NOTE: Scenarios are used to analyze the operation of the system in its intended environment in order and to identify requirements that may not have been formally specified by any of the stakeholders, e.g. legal, regulatory and social obligations. The context of use of the system is identified and analyzed. Include in the context analysis the activities that users perform to achieve system objectives, the relevant characteristics of the end-users of the system (e.g. expected training, degree of fatigue), the physical environment (e.g. available light,



temperature) and any equipment to be used (e.g. protective or communication equipment). The social and organisational influences on users that could affect system use or constrain its design are analyzed when applicable.

e) Identify the interaction between users and the system.

NOTE: Usability requirements are determined, establishing, as a minimum, the most effective, efficient, and reliable human performance and human-system interaction. Where possible, applicable standards, e.g. ISO 9241, and accepted professional practices are used in order to define:

- 1) Physical, mental, and learned capabilities;
- 2) Work place, environment and facilities, including other equipment in the context of use;
- 3) Normal, unusual, and emergency conditions;
- 4) Operator and user recruitment, training and culture.

f) Specify health, safety, security, environment and other stakeholder requirements and functions that relate to critical qualities.

NOTE: Identify safety risk and, if warranted, specify requirements and functions to provide safety. This includes risks associated with methods of operations and support, health and safety, threats to property and environmental influences. Use applicable standards, e.g. IEC 61508, and accepted professional practices. Identify security risk and, if warranted, specify all applicable areas of system security, including physical, procedural, communications, computers, programs, data and emissions. Identify functions that could impact the security of the system, including access and damage to protected personnel, properties and information, compromise of sensitive information, and denial of approved access to property and information. Specify the required security functions, including mitigation and containment, referencing applicable standards and accepted professional practices where mandatory or relevant.

g) Analyse the complete set of elicited requirements.

NOTE: Analysis includes identifying and prioritising the conflicting, missing, incomplete, ambiguous, inconsistent, incongruous or unverifiable requirements.

h) Resolve requirements problems.

NOTE: This includes requirements that cannot be realized or are impractical to achieve.



- i) Feed back the analyzed requirements to applicable stakeholders to ensure that the needs and expectations have been adequately captured and expressed.

NOTE: Explain and obtain agreement to the proposals to resolve conflicting, impractical and unrealisable stakeholder requirements.

- j) Establish with stakeholders that their requirements are expressed correctly.

NOTE: This includes confirming that stakeholder requirements are comprehensible to originators and that the resolution of conflict in the requirements has not corrupted or compromised stakeholder intentions.

- k) Record the stakeholder requirements in a form suitable for requirements management through the life cycle and beyond.

NOTE: These records establish the stakeholder requirements baseline, and retain changes of need and their origin throughout the system life cycle. They are the basis for traceability to the system requirements and form a source of knowledge for requirements for subsequent systems.

- l) Maintain stakeholder requirements traceability to the sources of stakeholder need.

NOTE: The stakeholder requirements are reviewed at key decision times in the life cycle to ensure that account is taken of any changes of need.

### **B.5.2 Requirements Analysis Process**

#### Purpose of the Requirements Analysis Process

The purpose of the Requirements Analysis Process is to transform the stakeholder, requirement-driven view of desired services into a technical view of a required product that could deliver those services.

This process builds a representation of a future system that will meet stakeholder requirements and that, as far as constraints permit, does not imply any specific implementation. It results in measurable system requirements that specify, from the developer's perspective, what characteristics it is to possess and with what magnitude in order to satisfy stakeholder requirements.



### Requirements Analysis Process Outcomes

As a result of the successful implementation of the Requirements Analysis Process:

- a) The required characteristics, attributes, and functional and performance requirements for a product solution are specified.
- b) Constraints that will affect the architectural design of a system and the means to realize it are specified.
- c) The integrity and traceability of system requirements to stakeholder requirements is achieved.
- d) A basis for verifying that the system requirements are satisfied is defined.

### Requirements Analysis Process Activities

The project shall implement the following activities in accordance with applicable organization policies and procedures with respect to the Requirements Analysis Process.

- a) Define the functional boundary of the system in terms of the behaviour and properties to be provided.

NOTE: This includes the system's stimuli and its responses to user and environment behaviour, and an analysis and description of the required interactions between the system and its environment in terms of interface constraints, such as mechanical, electrical, mass, thermal, data, and procedural flows. This establishes the expected system behaviour, expressed in quantitative terms, at its boundary.

- b) Define each function that the system is required to perform, how well the system, including its operators, is required to perform that function, the conditions under which the system is to be capable of performing the function, the conditions under which the system is to commence performing that function and the conditions under which the system is to cease performing that function.

NOTE: Conditions for the performance of functions may incorporate reference to states and required modes of operation of the system. System requirements depend heavily on abstract representations of proposed system characteristics and may employ multiple modelling techniques and perspectives to give a sufficiently complete description of the desired system requirements.

- c) Define necessary implementation constraints that are introduced by stakeholder requirements or are unavoidable solution limitations.



NOTE: This includes the implementation decisions that are allocated from design at higher levels in the structure of the system.

- d) Define technical and quality in use measures that enable the assessment of technical achievement.

NOTE: This includes defining critical performance parameters associated with each effectiveness measure identified in the stakeholder requirements. The critical performance measures are analyzed and reviewed to ensure stakeholder requirements are met and to ensure identification of project cost, schedule or performance risk associated with any non-compliance. ISO/IEC 15939 provides a process to identify, define and use appropriate measures. ISO/IEC 9126 may provide relevant quality measures.

- e) Specify system requirements and functions, as justified by risk identification or criticality of the system, that relate to critical qualities, such as health, safety, security, reliability, availability and supportability.

NOTE: This includes analysis and definition of safety considerations, including those relating to methods of operation and maintenance, environmental influences and personnel injury. It also includes each safety related function and its associated safety integrity, expressed in terms of the necessary risk reduction, is specified and allocated to designated safety-related systems. Applicable standards are used concerning functional safety, e.g. IEC 61508, and environmental protection, e.g. ISO 14001. Analyze security considerations including those related to compromise and protection of sensitive information, data and material. The security-related risks are defined, including, but not limited to, administrative, personnel, physical, computer, communication, network, emission and environment factors using, as appropriate, applicable security standards.

- f) Analyse the integrity of the system requirements to ensure that each requirement, pairs of requirements or sets of requirements possess overall integrity.

NOTE: Each system requirement statement is checked to establish that it is unique, complete, unambiguous, consistent with all other requirements, implementable and verifiable. Deficiencies, conflicts and weaknesses are identified and resolved within the complete set of system requirements. The resulting system requirements are analyzed to confirm that they are complete, consistent, achievable (given current technologies or knowledge of technological advances) and expressed at an appropriate level of detail. Confirmation is made that they are a



necessary and sufficient response to stakeholder requirements and a necessary and sufficient input to other processes, in particular architectural design.

- g) Demonstrate traceability between the system requirements and the stakeholder requirements.

NOTE: Maintain mutual traceability between the system requirements and the stakeholder requirements, i.e. all achievable stakeholder requirements are met by one or more system requirements, and all system requirements meet or contribute to meeting at least one stakeholder requirement. The system requirements are held in an appropriate data repository that permits traceability to stakeholder needs and architectural design.

- h) Maintain throughout the system life cycle the set of system requirements together with the associated rationale, decisions and assumptions.

## **B.6 ESA ECSS-E-10 on systems engineering**

### **B.6.1 Requirement Engineering**

#### Objectives of requirement engineering

Requirement engineering shall ensure the following:

- a) Proper interpretation of the customer needs and constraints concerning technical requirements for a product that satisfies the customer needs, produced, consolidated and agreed with the customer.

NOTE: This can be done in interaction with the customer.

- b) Generation, control and maintenance of a coherent and appropriate set of system and lower level specifications.
- c) Full traceability of the requirements within the set of specifications stated in b. above, down to final verification close-out.

#### Requirement engineering elements

##### Overview

As specified in 4.4.2.2 to 4.4.2.6, for the purpose of performing requirement engineering, all requirements are

- a) justified,
- b) sorted by type,
- c) possessing precise attributes,
- d) assessed in terms of risk index,



e) traceable to the next upper level requirements.

NOTE: See ECSS--E--10 Part 6 for details.

#### Requirement justification

Space system requirements shall be justified.

NOTE: The objective is to avoid requirements without rationale or object, and to enable evaluation of requirement impact in case of non-conformity or requirement evolution.

#### Requirement classification

Space system requirements shall be identified, sorted and grouped on the basis of requirement classes in relationship to their objectives and sources.

NOTE: The classification [shown in Figure 4] is a guideline to structure the specification with the objective of facilitating checking of the completeness of the scope covered by the requirements.

#### Requirement attributes

In order to facilitate the system engineering process, in particular the verification activities, each requirement shall, as a minimum, have the attributes making it:

- a) Traceable, e.g. either w.r.t. a higher-level requirement, an imposed constraint (e.g. an applicable standard), or an accepted lower level constraint.
- b) Unique, and associated with an identifier (for example a document and paragraph number).
- c) Single, and not a combination of several requirements.
- d) Verifiable, using clearly identified verification methods.
- e) Unambiguous (e.g. no 'to be determined', no 'should').
- f) Referenced to other requirements (with applicable document and paragraph identification).
- g) Associated with a specific title.

#### Requirement criticality assessment

As each requirement differs in importance for the system development and operation in terms of impact on cost, schedule and risk, and, in order, to support the system engineering process vis-à-vis risk and cost minimisation:

- a) critical requirements shall be identified;
- b) the sensitivity of the system to the critical requirements identified in a. above (i.e. the impact on implementation aspects due to modification of critical requirements) shall be evaluated;
- c) an indicator of risk-severity and priority should be associated with each requirement identified in a. above, to support the requirement engineering activities.



NOTE: For severity and criticality of a requirement see ECSS-E-10 Part 6.

### Requirement traceability

#### Overview

Traceability is the ability to identify the relationship between:

- a) requirements (e.g. a higher level requirement, an imposed constraint, applicable standard, or an accepted lower level constraint);
- b) a decision and the affected requirements (e.g. within trade-off loops);
- c) a requirement and its source (e.g. in the mission statement);
- d) a verification result (e.g. test result) and the related requirement to be verified.

#### Process

The requirement engineering shall ensure implementation of the traceability, as defined in the previous paragraph.

### Requirement engineering process

#### Overview

The basic activities in the requirement engineering process are:

- a) requirement capture,
- b) requirement analysis and validation,
- c) requirement allocation,
- d) requirement maintenance.

#### Requirement capture

- a) The customer's requirements (i.e. needs and objectives) shall be captured in order to achieve the following.
  - 1) The understanding of the customer's expectations and agreement with the customer on a refined requirement baseline.
  - 2) A synthesis of responsive products and services to the requirements.
- b) To comply with a) 1) above, potentially incomplete, ambiguous or contradictory requirements shall be identified and resolved with the customer representatives.

#### Requirement analysis and validation

Requirement analysis and validation shall include the following:

- a) Assessment of different situations of the system during its life cycle from manufacturing to disposal with associated combinations of environmental conditions as well as the number of occurrences and their duration.



- b) Identification of design, implementation, and the statutory and regulatory requirements specific to the product, not customer specified.
- c) Identification of constraints (e.g. limits of applicability) related to each specific requirement.
- d) Identification of operational scenarios of the system in all its modes.
- e) Checking that the final set of requirements in the system and lower level specifications, is individually and globally consistent and non redundant (internal validation).
- f) Gaining the acceptance by the customer of the final set of requirements in the system and lower level specifications (external validation).

#### Requirement allocation

- a) The system requirements shall be allocated to the lower level products on the basis of system analyses and design and then to the requirements included in the specifications of these products.
- b) The allocation process shall be iteratively carried out in parallel with functional, architectural, and design analyses.
- c) The level of the requirements complexity (e.g. toward a single parameter requirement) shall be decreased by the requirement flow-down in order to achieve measurable detail or go, no-go criteria.
- d) Existing specific requirements (e.g. some design or procurement constraints specified by the specification of the relevant product) shall be allocated to the applicable level.

#### Requirement maintenance

- a) At all levels, requirements shall be maintained during the entire life cycle of the system down to final verification close-out.
- b) Any change in requirements shall be processed and recorded in order to guarantee full visibility and traceability of the current requirement baseline.

### **B.6.2 Documentation**

#### Mission description document (DRD)

The mission description document defines the objectives, operation profile, major system events and capabilities, contingencies and performance standards to be met by one system concept that aims at satisfying a specified mission statement.

#### System concept report (DRD)

The system concept report initially describes the principal technical characteristics of alternative concepts, relating to performance, interfaces and risks, and later addresses the selected concepts.

#### Functional specification (DRD)

The functional specification is a document that establishes the intended purpose of a product, its associated constraints and the environment, the operational and performances features for each phase of life cycle, and the permissible flexibility.

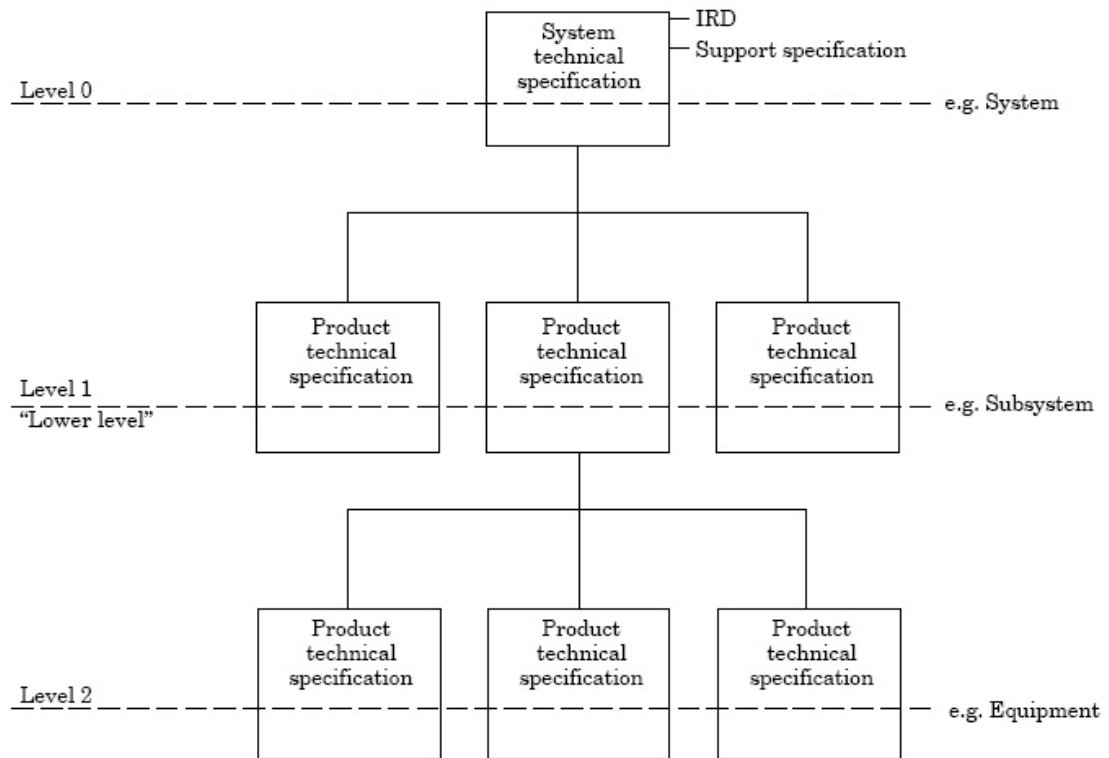


Figure 5-1: Sample Specifications Tree

Technical specification (DRD)

The technical specification is a document that contains of a product, the specified technical requirements and the exact value for the performance. It defines all the necessary and sufficient attributes of a product, with the requirements for verification.

Interface requirement document (DRD)

The interface requirement document defines the requirements for the interfaces between related items.

Requirement traceability matrix (DRD)

The requirement traceability matrix defines the relationships between the requirements of a deliverable product defined in its technical specification and the apportioned requirements of its lower level elements.

Requirement justification file (DRD)

The requirement justification file is a generic title referring to all documentation which records, describes the needs and the associated constraints resulting from the different trade-offs and relevant environment constraints and demonstrates how the requirements of the system technical specification can satisfy the need. This document is established at the upper level.

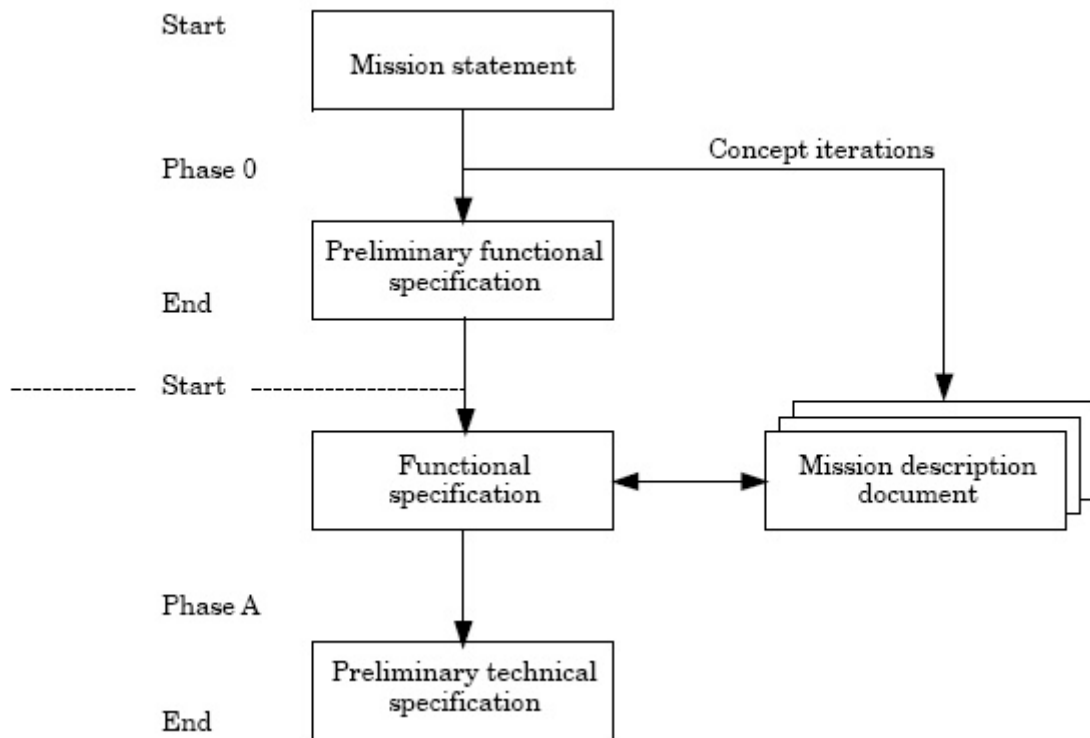


Figure 5-2: Relationships between Key Documents

### B.6.3 Requirements and Recommendations for the Wording

#### General format

- Technical requirements should be stated in performance or ‘what-is-necessary’ terms, as opposed to telling a supplier ‘how to’ perform a task, unless the exact steps in performance of the task are essential to ensure the proper functioning of the product.
- Technical requirements should be expressed in a positive way, as a complete sentence (with a verb and a noun).

#### Required verbal form

- The verbal form ‘shall’ shall be used whenever a provision is a requirement.
- The verbal form ‘should’ shall be used whenever a provision is a recommendation.
- The verbal form ‘may’ shall be used whenever a provision is a permission.



- d) The verbal form ‘can’ shall be used to indicate possibility or capability.

Format restrictions

List of terms that shall not be used in a TS requirement

- ‘and/or’,
- ‘etc’,
- ‘goal’,
- ‘shall be included but not limited to’,
- ‘relevant’,
- ‘necessary’,
- ‘appropriate’,
- ‘as far as possible’,
- ‘optimise’,
- ‘minimise’,
- ‘maximise’,
- ‘typical’,
- ‘rapid’,
- ‘user-friendly’,
- ‘easy’,
- ‘sufficient’,
- ‘enough’,
- ‘suitable’,
- ‘satisfactory’,
- ‘adequate’,
- ‘quick’,
- ‘first rate’,
- ‘best possible’,
- ‘great’,
- ‘small’,
- ‘large’, and
- ‘state of the art’

**B.7 ESA ECSS-E-40 on Software Engineering**

ECSS-E-40 documents relevant to RORI-OCV are:

- ECSS-E-40 part 1B

**B.7.1 Software Requirements Analysis**

Establishment and documentation of software requirements

The supplier shall establish and document software requirements, including the software quality requirements, as part of the technical specification.

EXPECTED OUTPUT: Software requirements specification [TS; PDR].



- a) Functional and performance specifications, including hardware characteristics, and environmental conditions under which the software item executes, including budgets requirements [TS; PDR];
  - b) Software product quality requirements (see ECSS-Q-80B sub-clause 7.2) [TS; PDR];
  - c) Security specifications, including those related to factors which can compromise sensitive information [TS; PDR];
  - d) Human factors engineering (ergonomics) specifications, including those related to manual operations, human equipment interactions, constraints on personnel, and areas requiring concentrated human attention, that are sensitive to human errors and training [TS; PDR];
  - e) Data definition and database requirements [TS; PDR];
- EXPECTED OUTPUT: Interface control document [TS; PDR]
- f) Interfaces external to the software item [ICD(TS); PDR].

#### Definition of functional and performance requirements for in flight modification

When in flight modification is specified for flight software, the supplier shall perform analysis of the specific implications for the software design and validation processes and include the functional and performance requirements in the technical specification and the corresponding design in the software architectural design.

EXPECTED OUTPUT:

- a) Specifications for in flight software modifications [TS; PDR];
- b) Design for in flight modification [DDF; PDR].

#### Identification of requirement unique identifier

Each requirement shall be separately identified in order to allow for traceability.

EXPECTED OUTPUT: Requirements unique identifier [TS; PDR].

#### Definition of a software logical model

The supplier shall construct a logical model of the functional requirements of the software product.

NOTE 1: The logical model can be the result of an iterative verification process with the customer. It also supports the requirements capture, documents and formalizes the software requirements.

NOTE 2: A logical model is a representation of the technical specification, independent of the implementation, written with a formalized language and it can be possibly executable. Formal methods can be used to prove properties of the logical model itself and therefore of the technical specification. The logical model allows in particular to verify that a technical specification is complete (i.e. by checking a software requirement exists for each logical model element), and



consistent (because of the model checking). The logical model can be completed by specific feasibility analyses such as benchmarks, in order to check the technical budgets (e.g. memory size and computer throughput). In case the modelling technique allows for it, preliminary automatic code generation can be used to define the contents of the software validation test specification.

EXPECTED OUTPUT: Software logical model [TS; PDR].

#### Definition of behavioural view

The logical model shall include a behavioural view.

EXPECTED OUTPUT: Behavioural view in software logical model [TS; PDR].

#### MMI software mock-up development

a) The supplier shall develop a software mock-up to support the requirements and architecture engineering process, in accordance with customer's requirements.

EXPECTED OUTPUT: MMI specifications for software [TS; PDR].

b) The supplier shall use the mock-up to prototype the specifications of man-machine interfaces for the software, such that MMI specifications are consolidated and evaluated with respect to human factors and use.

AIM: The aim of this sub-clause includes:

- proper consideration of human factors,
- that the man-machine interface achieves an acceptable state of definition during requirements and architecture engineering activities, and
- that the technical performance of the man-machine interface is verified.

NOTE: Depending on the nature of the project, the supplier can opt to later upgrade the software mock-up of the MMI to become part of the final software product. However, unless the mock-up is later upgraded to become part of the final product tree, the mock-up need not be a formal delivery to the customer.

EXPECTED OUTPUT: Report on evaluation of MMI specifications using a software mock-up [DJF; PDR].

c) The customer shall ensure that end-users, or their representatives, participate in the MMI mock-up evaluation.

### **B.7.2 Requirements baseline (RB)**

#### RB contents at SRR

E-40B--5.2.2.1--a Functions and performance requirements of the system



- [RB; SRR]
- E--40B--5.2.2.1--c Design constraints [RB; SRR]
- E--40B--5.2.2.1--d Identification of lower level software engineering standards [RB; SRR]
- E--40B--5.2.2.1--e Verification and validation product requirements [RB; SRR]
- E--40B--5.2.2.2 Overall safety and reliability requirements of the software to be produced [RB; SRR]
- E--40B--5.2.2.3 MMI software mock--up requirements [RB; SRR]
- E--40B--5.2.2.4 MMI general requirements [RB; SRR]
- E--40B--5.2.3.2c System partition with definition of items [RB; SRR]
- E--40B--5.2.3.2d System configuration items list [RB; SRR]
- E--40B--5.2.4.2 Verification and validation process requirements [RB; SRR]
- E--40B--5.2.4.3--a Functional requirements for support to system and mission level validation [RB; SRR]
- E--40B--5.2.4.3--b Installation and acceptance requirements for the delivered software product at the operational and maintenance site [RB; SRR]
- E--40B--5.2.4.5 SRR milestone report [RB; SRR]
- E--40B--5.2.5.1a Software observability requirements [RB; SRR]
- E--40B--5.2.5.1b System observability requirements [RB; SRR]
- E--40B--5.2.5.1c System observability requirements [RB; SRR]
- E--40B--5.2.5.1d System observability requirements [RB; SRR]
- E--40B--5.2.5.4 System database specification (content and use) [RB; SRR]
- E--40B--5.2.5.5 Development constraints [RB; SRR]
- E--40B--5.2.5.6 Requirements for “design for reuse” [RB; SRR]
- E--40B--5.2.6.2 Software operations requirements [RB; SRR]
- E--40B--5.2.7.1 Software maintenance requirements [RB; SRR]
- E--40B--5.2.7.2 Requirements for in flight modification capabilities [RB;SRR]
- E--40B--5.3.2.6 Customer approval of requirements baseline [RB; SRR]
- E--40B--5.3.4.2--a Interface management procedures [RB; SRR]
- E--40B--5.3.4.2--b Part of configuration management requirements [RB; SRR]
- E--40B--5.3.5.1 Technical budgets and margin philosophy for the project [RB; SRR]
- Q--80B--5.6.1 Software procurement process for COTS, OTS or MOTS [RB; SRR]



Q--80B--6.2.2.1 Critical functions identification and analysis [RB; SRR]

#### IRD contents at SRR

E--40B--5.2.2.1--b Interface requirements [IRD(RB); SRR]

E--40B--5.2.3.2a Software--hardware interface requirements [IRD(RB); SRR]

E--40B--5.2.5.2 System level interface requirements [IRD(RB); SRR]

E--40B--5.2.5.3 System level data interfaces [IRD(RB); SRR]

E--40B--5.2.5.7 System level integration support products [IRD(RB); SRR]

E--40B--5.2.5.8 System level integration preparation requirements [IRD(RB); SRR]

E--40B--5.3.4.1 Interface requirements document [IRD(RB); SRR]

## **B.8 EUROCAE ED-12B on Airborne Software Engineering**

### **B.8.1 Software Requirements Process**

The software requirements process uses the outputs of the system life cycle process to develop the software high-level requirements. These high-level requirements include functional, performance, interface and safety-related requirements.

#### Software Requirements Process Objectives

The objectives of the software requirements process are:

- a) High-level requirements are developed.
- b) Derived high-level requirements are indicated to the system safety assessment process.

#### Software Requirements Process Activities

Inputs to the software requirements process include the system requirements, the hardware interface and system architecture (if not included in the requirements) from the system life cycle process, and the Software Development Plan and the Software Requirements Standards from the software planning process. When the planned transition criteria have been satisfied, these inputs are used to develop the software high-level requirements.

The primary output of this process is the Software Requirements Data.

The software requirements process is complete when its objectives and the objectives of the integral processes associated with it are satisfied. Guidance for this process includes:

- a) The system functional and interface requirements that are allocated to software should be analyzed for ambiguities, inconsistencies and undefined conditions.



- b) Inputs to the software requirements process detected as inadequate or incorrect should be reported as feedback to the input source processes for clarification or correction.
- c) Each system requirement that is allocated to software should be specified in the high-level requirements.
- d) High-level requirements that address system requirements allocated to software to preclude system hazards should be defined.
- e) The high-level requirements should conform to the Software Requirements Standards, and be verifiable and consistent.
- f) The high-level requirements should be stated in quantitative terms with tolerances where applicable.
- g) The high-level requirements should not describe design or verification detail except for specified and justified design constraints.
- h) Each system requirement allocated to software should be traceable to one or more software high-level requirements.
- i) Each high-level requirement should be traceable to one or more system requirements, except for derived requirements.
- j) Derived high-level requirements should be provided to the system safety assessment process.

#### Software Requirements Standards

The purpose of Software Requirements Standards is to define the methods, rules and tools to be used to develop the high-level requirements. These standards should include:

- a) The methods to be used for developing software requirements, such as structured methods.
- b) Notations to be used to express requirements, such as data flow diagrams and formal specification languages.
- c) Constraints on the use of the requirement development tools.
- d) The method to be used to provide derived requirements to the system process.

#### Software Requirements Data

Software Requirements Data is a definition of the high-level requirements including the derived requirements. This data should include:

- a) Description of the allocation of system requirements to software, with attention to safety-related requirements and potential failure conditions.
- b) Functional and operational requirements under each mode of operation.
- c) Performance criteria, for example, precision and accuracy.
- d) Timing requirements and constraints.
- e) Memory size constraints.



- f) Hardware and software interfaces, for example, protocols, formats, frequency of inputs and frequency of outputs.
- g) Failure detection and safety monitoring requirements.
- h) Partitioning requirements allocated to software, how the partitioned software components interact with each other, and the software level(s) of each partition.

## **B.9 EUROCAE ED-79 on Airborne Systems Engineering**

### **B.9.1 Requirements Capture**

Requirements, together with related hazards, provide the common basis for the supporting processes. Because the hazards may have different levels of importance, the allocation of requirements, through system architecture, has significant impact on the ease of substantiating system certification. The top level process in the aircraft development cycle includes the identification of aircraft functions and the requirements associated with these functions. The aircraft functions, including functional interfaces and corresponding safety requirements, form the basis for establishing the system architecture. Selection of the architecture establishes additional requirements necessary to implement that architecture. At each phase of the requirements identification and allocation process (i.e., system, item and hardware/software) both additional detail for existing requirements and new derived requirements are identified. Choices made and problems encountered during implementation are a primary source for derived requirements and may lead to identification of new system safety requirements.

#### Types of Requirements

The requirements associated with a given function define the way the function acts in its environment and include the definition of the user/machine interface. The types of requirements detailed below should be considered at various phases of the development activities (i.e., function, system, item and hardware/software). There may be requirements that address strictly business or economic issues and do not impact safety or certification requirements.

#### Safety Requirements

The safety requirements for aircraft and system-level functions include minimum performance constraints for both availability (continuity of function) and integrity (correctness of behaviour) of the function. These safety requirements should be determined by conducting a functional hazard assessment consistent with the processes in paragraph 6.1.

Safety requirements for aircraft and system functions are determined by identifying and classifying associated functional failure conditions. All functions have associated failure modes and associated aircraft effects, even if the classification is “No safety effect.” Safety related functional failure modes may have either contributory or direct effects upon aircraft safety.



Requirements that are defined to prevent failure conditions or to provide safety related functions should be traceable through the levels of development at least to the point of allocation to hardware and software. This will ensure visibility of the safety requirements at the software and hardware design level.

### Functional Requirements

Functional requirements are those necessary to obtain the desired performance of the system under the conditions specified. They are a combination of customer desires, operational constraints, regulatory restrictions, and implementation realities. These requirements define all significant aspects of the system under consideration. Regardless of the original source, all functions should be evaluated for their safety related attributes.

### Customer Requirements

Customer requirements will vary with the type of aircraft, the specific function or the type of system under consideration. Requirements may include those associated with the operator's intended payload, route system, operating practices, maintenance concepts, and desired features.

### Operational Requirements

Operational requirements define the interfaces between the flight crew and each functional system, the maintenance crew and each aircraft system, and various other aircraft support people and related functions or equipment. Actions, decisions, information requirements and timing constitute the bulk of the operational requirements. Both normal and non-normal circumstances need to be considered when defining operational requirements.

### Performance Requirements

Performance requirements define those attributes of the function or system that make it useful to the aircraft and customer. In addition to defining the type of performance expected, performance requirements include function specifics such as: accuracy, fidelity, range, resolution, speed, and response times.

### Physical and Installation Requirements

Physical and installation requirements relate the physical attributes of the system to the aircraft environment. They may include: size, mounting provisions, power, cooling, environmental restrictions, visibility, access, adjustment, handling, and storage. Production constraints may also play a role in establishing these requirements.



### Maintainability Requirements

Maintainability requirements include scheduled and unscheduled maintenance requirements and any links to specific safety-related functions. Factors such as the percent of failure detection or the percent of fault isolation may also be important. Provisions for external test equipment signals and connections should be defined in these requirements.

### Interface Requirements

Interface requirements include the physical system and item interconnections along with the relevant characteristics of the specific information communicated. The interfaces should be defined with all inputs having a source and all output destinations defined.

### Additional Certification Requirements

Additional functions, functional attributes, or implementations may be required by airworthiness regulations or may be necessary to show compliance with airworthiness regulations. Requirements of this type should be defined and agreed upon with the appropriate certification authorities.

## **B.9.2 Derived Requirements**

At each phase of the development activity, decisions are made as to how particular requirements or groups of requirements are to be met. The consequences of these design choices become requirements for the next phase of the development. Since these requirements result from the design process itself, they may not be uniquely related to a higher-level requirement and are referred to as derived requirements. Derived requirements should be examined to determine which aircraft-level function (or functions) they support so that the appropriate failure condition classification can be assigned and the requirement validated. While most such requirements will not impact the higher-level requirements, some may have implications at higher levels. Derived requirements should be reviewed at progressively higher system levels until it is determined that no further impact is propagated. For example, derived requirements may result from the decision to select a separate power supply for equipment performing a specific function. The requirements for the power supply, including the safety requirements, are derived requirements. The hazard resulting from the fault or failure of the function supported by the power supply determines the necessary development assurance level.

Derived requirements may also result from architecture choices. For example, selecting a triplex architecture for achieving a high integrity functional objective would have different consequences and different derived requirements from selection of a dual monitored architecture for achievement of the same objective. Derived requirements may result from a design decision to isolate function implementations having more severe failure condition



classifications from the malfunction or failure effects of systems having less severe failure condition classifications.

Derived requirements also include those defining the hardware-software interface. Some of these requirements may be significant at the system level. The remainder, dealing with detailed aspects of the hardware-software interface, may be handled under the guidance of ED-12B and ED-80.

Derived requirements should be captured and treated in a manner consistent with other requirements applicable at that development phase.

### **B.9.3 Validation of Requirements**

Validation of requirements and specific assumptions is the process of ensuring that the specified requirements are sufficiently correct and complete so that the product will meet applicable airworthiness requirements. Validation is a combination of objective and subjective processes. In showing compliance with JAR/FAR 25.1301 and JAR/FAR 25.1309, the validation process supports the development of requirements from functional needs and safety considerations. This development should generate a complete set of requirements. The validation process addresses each of these requirements. While the format is left to the developer's definition, a structured process should be defined in the validation plan (see paragraph 7.7.1).

Ideally from the point of view of facilitating a smooth development process, requirements should be validated before design implementation commences. However, in practice, particularly for complex and integrated systems, the necessary visibility of the whole set of consequences that flow from the requirements may not be obtainable until the system implementation itself is available and can be tested in its operational context. In consequence, validation is normally a staged process continuing through the development cycle. At each stage the validation activity provides increasing confidence in the correctness and completeness of the requirements.

The validation process at each level of the requirements hierarchy should involve all relevant technical disciplines, including the safety assessment process. Experience indicates that careful attention to requirements development and validation can identify subtle errors or omissions early in the development cycle and reduce exposure to subsequent redesign or inadequate system performance. Individual tests may simultaneously serve the purposes of verification as well as validation when the system implementation is used as part of the requirements validation process. One purpose of this activity is to check that the requirements are met by the implemented system, while a separate purpose is checking that the requirements are appropriate to the context in which the system is operating. Such dual purposes should be reflected by coordination of the verification and validation plans.



### Validation Process Objectives

Ensuring correctness and completeness of requirements are the objectives of the validation process. Errors in the definition of system requirements can arise from three primary causes: (1) ambiguity, (2) incorrect statements, or (3) incomplete statements (i.e., omissions). The validation process should adequately cover all of these potential deficiencies. Examination of requirements to ensure they are both necessary and sufficient is a key aspect of validation. A further objective of the validation process is to limit the potential for unintended functions in the system or for unintended functions to be induced in interfacing systems.

For the purpose of this document, correctness and completeness are defined as follows:

- a) Correctness of a requirement statement means the absence of ambiguity or error in its attributes.
- b) Completeness of a requirement statement means that no attributes have been omitted and that those stated are essential.

### Validation Process Model

Requirements and assumptions should be validated at each hierarchical level of requirements definition. This includes validation of requirements at the aircraft function, system and item levels as well as validation of the FHA. Generally, validation of requirements and assumptions at higher levels serves as a basis for validation at lower levels.

The relationship of validation to system development is shown in Figure 2. An expanded process model is shown in Figure 5. Inputs to the validation process may include a description of the system (including the operating environment), the system requirements, a definition of system architecture, and the development assurance level.

An overview of the requirements and assumptions validation process is outlined below. These processes may be used for validation at the various hierarchical levels. These processes, or suitable alternatives, may be used to support certification.

- a) Validation Plan:

The validation plan should define the specific methods to be used for validation of system requirements and assumptions. (Additional information on validation planning is provided in paragraph 7.7.1.)

- b) Determination of the Level of Validation: The necessary level of validation is determined by the development assurance level of the function addressed by the requirement (see paragraph 7.6).

- c) Completeness and Correctness Checks: The checks for requirements completeness and correctness may require engineering judgment, as well as analysis or test. The validity of such judgments will be established more easily if they and their supporting rationale are



recorded at the time that the related requirement is developed. (Additional information on these checks is provided in paragraphs 7.3 and 7.4.)

- d) Validation of Assumptions: The process of validation of assumptions focuses on ensuring that assumptions are explicitly stated, appropriately disseminated, and justified by supporting data (see paragraph 7.5).
- e) Validation Matrix: This process includes preparation of a validation matrix (see paragraph 7.7.3) that references requirements and validation results, including, as appropriate, those for hardware/software performance, derived requirements, environmental and operational considerations, assumptions and supporting data. The source of each requirement should be identifiable. This matrix should be updated regularly during the development and included in the validation summary.
- f) Validation Summary: Data describing the process, as well as the results, can be an effective means of ensuring that communication is based on a consistent and balanced understanding of the issues significant to the system design (see paragraph 7.7.4).

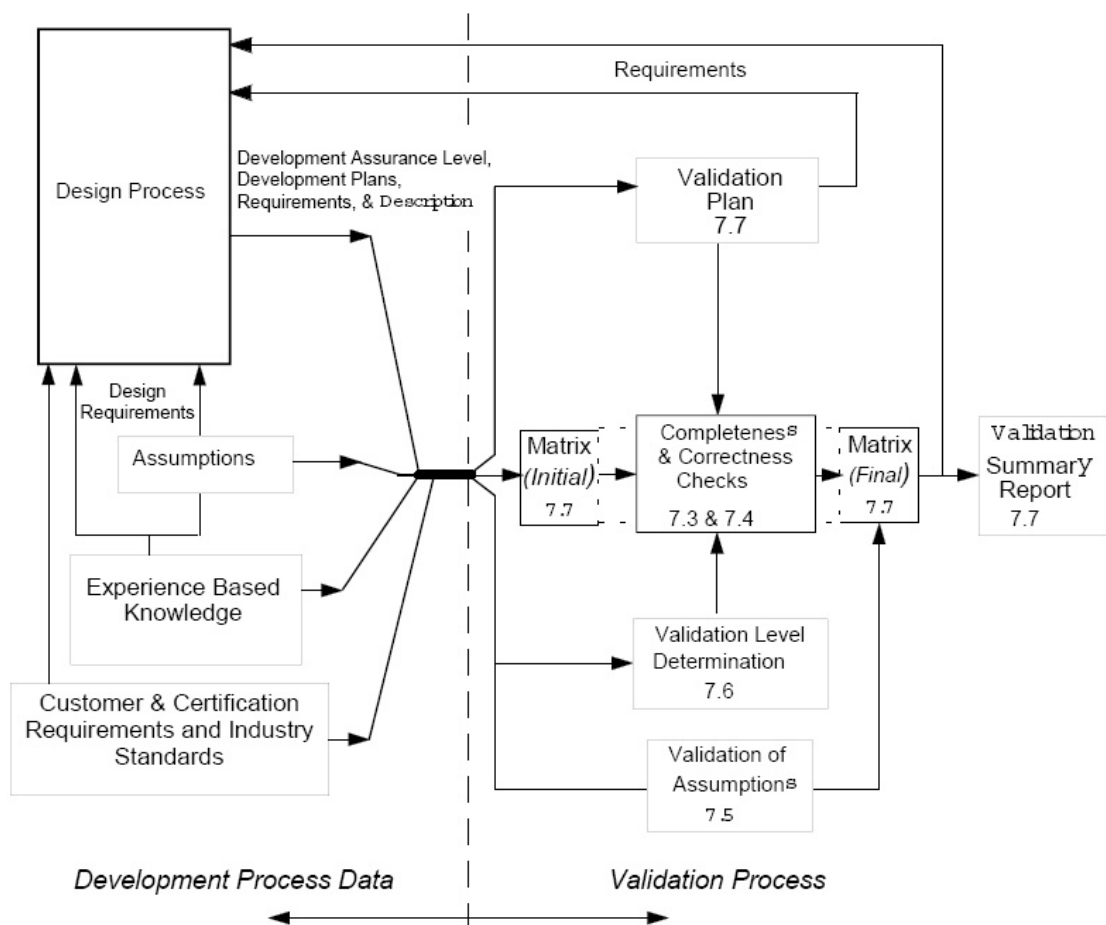


Figure 5-3: Validation Process Model



### Completeness Checks

The following is an example set of questions for assessing completeness at each hierarchical level of requirements. This list should be tailored for the specific application.

- a. Do requirements trace to identified sources?
  - (1) Intended functions — aircraft-level, system-level
  - (2) All functions, hazards, and failure condition classifications identified in FHA
  - (3) All failure conditions incorporated in PSSA
  - (4) Derived requirements — design decisions or assumptions
  - (5) Applicable regulatory standards and guidelines
  - (6) Anticipated operating environment
  - (7) Established flight operations or maintenance procedures
- b. Are constraints and assumptions adequately defined, substantiated and addressed?
  - (1) Market considerations
  - (2) Safety considerations (e.g., FHA, FMEAs, PSSAs)
  - (3) Environmental constraints
  - (4) Industry and company standards
- c. Has the system implementation been adequately specified?
  - (1) All aircraft and system functions fully allocated
  - (2) All interfaces defined — internal, external, physical, functional, human
  - (3) System architecture defined and requirements allocated to hardware and software
- d. Are prohibited behaviour characteristics explicitly stated?

### Correctness Checks

During the validation process both the correctness of failure condition classification and the correctness of the stated requirements content should be reviewed and justified. Correctness checks should be carried out at each level of the requirements hierarchy. The following questions may help assess correctness of requirements. This list should be tailored and expanded for the specific application.

- a. Are all requirements correctly stated?
  - (1) What is required (as opposed to how it should be designed)
  - (2) Unambiguous
  - (3) Statements leading to appropriate design
  - (4) Realizable and verifiable to the level of rigor appropriate to the system development assurance level
  - (5) Stated for all required environmental conditions
  - (6) Stated for degraded and normal modes
  - (7) Derived requirements are correct and supported by analysis



- (8) Source(s) of each requirement identified
- b. Are assumptions correct?
  - (1) Significant to/inherent in the requirements
  - (2) Documented
  - (3) Traced
  - (4) FHA failure condition classification assumptions confirmed
- c. Do requirements correctly reflect the safety analyses?
  - (1) Appropriate safety analyses completed correctly
  - (2) All system hazards identified and classified correctly
  - (3) Impact of unsafe design or design errors
  - (4) Reliability, availability, and fault tolerance requirements

## **B.10 EUROCAE ED-80 on Airborne Hardware Engineering**

### **B.10.1 Requirements Capture Process**

The requirements capture process identifies and records the hardware item requirements. This includes those derived requirements imposed by the proposed hardware item architecture, choice of technology, the basic and optional functionality, environmental, and performance requirements as well as the requirements imposed by the system safety assessment. This process may be iterative since additional requirements may become known during design.

#### Requirements Capture Objectives

The objectives for the requirements capture process are:

1. Requirements are identified, defined and documented. This includes allocated requirements from the PSSA and derived requirements from the hardware safety assessment.  
NOTE: Traceability of verification results to the hardware requirements is addressed in Section 6. It is desirable to establish this method of traceability during the requirement capture process.
2. Derived requirements produced are fed back to the appropriate process.
3. Requirement omissions and errors are provided to the appropriate process for resolution.

#### Requirements Capture Activities

The requirements capture activities form an iterative process which helps assure consistency of the requirements with the design implementation, the system requirements and the software requirements. Guidance for the requirements capture activities includes:

1. The system requirements allocated to the hardware item should be documented. These may include identifying requirements, such as functionality and performance, and architectural



considerations, such as segregation, Built-In- Test, testability, external interfaces, environment, test and maintenance considerations, power, and physical characteristics.

2. The safety requirements from the PSSA related to the hardware item should be identified. These may include:
  - a. Design assurance levels imposed on the functions to be implemented in the hardware.
  - b. Probabilistic requirements for malfunctions or loss of function.
  - c. Hardware architectural and functional safety attributes, such as those outlined in Section 2.3.1, selected to meet the functional allocation.
3. Design constraints due to production processes, standards, procedures, technology, design environment and design guidance should be identified.
4. Derived requirements necessary for implementation should be determined. Requirements derived from the hardware safety assessment that have safety implications should be uniquely identified.

NOTE: Derived requirements may address conditions, such as:

- a. Specific constraints to ensure that functions of a higher design assurance level can withstand anomalies of functions of a lower design assurance level as seen at the interface of the function with the lower design assurance level.
  - b. The range of data inputs considering typical and full-scale data values as well as the high and low states of bits in data words or control registers.
  - c. Power-up reset or other reset states.
  - d. Supply voltage and current demands.
  - e. Performance of time-related functions, such as filters, integrators and delays.
  - f. State machine transitions that are possible, whether they are anticipated or not.
  - g. Signal timing relationships or electrical conditions under normal and worst-case conditions.
  - h. Signal noise and cross-talk.
  - i. Signal glitches in asynchronous logic circuits.
  - j. Specific constraints to control unused functions.
5. Derived requirements should be fed back to the SSA process so that the effects on the system requirements can be assessed.
  6. The requirement data should be documented in quantitative terms, with tolerances where applicable. This does not include the description of design or verification solutions.
  7. Requirement omissions or errors discovered during this process should be provided to the system development process.
  8. The requirements, including those generated to meet the PSSA requirements, should be traceable to the next higher hierarchical level of requirements. Derived requirements should be identified and traced as far as possible through the hierarchical levels.



NOTE: System level validation of allocated hardware safety requirements may occur during the requirement capture process.

### **B.10.2 Conceptual Design Process**

The conceptual design process produces a high-level design concept that may be assessed to determine the potential for the resulting design implementation to meet the requirements. This may be accomplished using such items as functional block diagrams, design and architecture descriptions, circuit card assembly outlines, and chassis sketches.

#### Conceptual Design Objectives

The conceptual design objectives are:

1. The hardware item conceptual design is developed consistent with its requirements.
2. Derived requirements produced are fed back to the requirements capture or other appropriate processes.
3. Requirement omissions and errors are provided to the appropriate processes for resolution.

#### Conceptual Design Activities

Guidance for the conceptual design activities includes:

1. A high-level description should be generated for the hardware item. This may include:
  - a. Architectural constraints related to safety, including those necessary to address design errors and functional, component over-stress, reliability and robustness defects.
  - b. Identification of any implementation constraints on software or other system components.
2. Major components should be identified. The way they contribute to the hardware safety requirements should be determined, including the impact of unused functions.
3. Derived requirements, including the interface definition, should be fed back to the requirements capture process.
4. Requirement omissions and errors should be fed back to the appropriate process for resolution.
5. The reliability, maintenance and test features to be provided should be identified.

NOTE: Consensus between the relevant parties that the conceptual design objectives have been met is recommended. Typically, a design review is used to accomplish this consensus.

### **B.10.3 Hardware Design Standards**

#### Requirements Standards

Requirements standards may be used during the requirements capture process to define the rules, procedures, methods, guidance and criteria for developing the requirements.



Requirements standards may include methods and criteria for developing and specifying requirements, methods and criteria for validating the requirements, notations used to express the requirements, guidance on the use of requirements specification tools, and the means used to provide derived requirements to the system design process.

#### **B.10.4 Hardware Design Data**

##### Hardware Requirements

The requirements specify the functional, performance, safety, quality, maintainability, and reliability requirements for the hardware item being developed.

The requirements should include:

1. The system design and safety requirements allocated to the hardware.
2. Identification of applicable standards for the hardware.
3. Hardware functional and performance requirements, including derived requirements and stress limits for normal use.
4. Hardware reliability and quality requirements, including requirements related to failure rates, exposure times and design constraints.
5. Hardware maintenance and repair requirements throughout the hardware item service life.
6. Hardware manufacturability and assembly requirements.
7. Hardware testability requirements.
8. Hardware storage and handling requirements.
9. Installation requirements.

#### **B.11 EUROCAE ED-78A Approval Guidelines**

This appendix contains several definitions and process descriptions from the document.

##### From the introduction:

This guidance material is intended for stakeholders and approval authorities involved in the operational implementation of the provision and use of ATS supported by data communications. Stakeholders are those organizations that have a financial investment in the provision and use of ATS. Stakeholders include ATS providers, ATS equipment manufacturers, supporting service providers, such as those that provide communication and weather services, aircraft and equipment manufacturers, and operators. These stakeholders may be identified as applicants in the context of this document. Approval authorities are those organizations in control of or responsible for issuing approvals to any element of the CNS/ATM system.

...

This guidance material recommends minimum acceptable criteria for approving the provision and use of an ATS supported by data communications when approvals are required to show



compliance to civil regulations. The criteria are in the form of process objectives and guidance for evidence. As used throughout this document, evidence is data produced during the accomplishment of the process objectives. Applicants can use the evidence to show an approval authority that the objectives have been satisfied. For example, evidence may take the form of standards such as the SPR and INTEROP standards, or plans such as the approval plan, or results of verification activities such as test results. The formulation of related standards and the means to qualify related systems to those standards are discussed. ‘Approval’ denotes those activities related to ATS provider operational approval, operator operational approval, and aircraft type design approval. These separate and distinct types of approvals collectively define the conceptual ‘CNS/ATM system approval’.

Definitions:

**Operator operational approval** is the authorization granted to an operator to use the ATS, aircraft equipage, communication services procured by the operator, and related inter-networks with the ATS provider’s communication services. It is supported by information provided by the aircraft type design approval and ATS provider operational approval;

**Aircraft type design approval** is the approval granted to an aircraft manufacturer or modifier to indicate that the type design of the aircraft equipage complies with applicable airworthiness requirements. It includes information to support operator operational approval; and

**ATS provider operational approval** is granted to an ATS provider for the provision of ATS within an airspace and includes information to support the operator operational approval. It includes approval of the technical system and the communication services procured by the ATS provider.

**Operational Services and Environment Definition (OSED):**

The OSED is used as the basis for assessing and establishing operational, safety, performance, and interoperability requirements for the related CNS/ATM system. It is developed based on an operational services and environment information capture process (OSEIC) that co-ordinates the information among stakeholders. The OSEIC captures elements related to a defined CNS/ATM system, including aircraft equipage, ATS provider technical system, communication service provider systems, and procedural requirements. The OSED identifies the ATS supported by data communications and their intended operational environments and includes the operational performance expectations, functions, and selected technologies of the related CNS/ATM system. The OSED facilitates the formulation of technical and procedural requirements, based on operational expectations and needs and is updated as necessary throughout the co-ordinated



requirements determination process. The OSED captures requirements that have been derived and/or validated as being necessary for a particular operational service.

**Operational, Safety, and Performance Requirements (SPR) Standard:**

An SPR standard is used to coordinate the operational, safety, and performance objectives and allocate requirements for the different approval types. It is developed using an operational safety assessment (OSA) and an operational performance assessment (OPA) of the functions, performance expectations, and characteristics of operational environments needed to support the ATS identified in the OSED. The SPR standard identifies the objectives and allocated requirements, including the substantiation, for a specific operation. The SPR provides traceability from each requirement to its source, the services, and operating environments described in the OSED, and captures the results of the OSA and OPA. An SPR standard can be tailored to meet the needs of a particular operational implementation.

**Interoperability Requirements (INTEROP) Standard:**

An INTEROP standard is used to provide sufficient information to enable different stakeholders to develop system elements that are compatible for an operational implementation. It is developed using an interoperability assessment (IA) of selected functions and technologies needed to support the ATS identified in the OSED. An INTEROP standard identifies the technical, interface, and related functional requirements for a specific technology or a mix of technologies. The INTEROP provides traceability from each requirement to the functions it supports, the services, and the operating environments in the OSED. Similar to an SPR standard, an INTEROP standard can be tailored to meet the needs of a particular operational implementation.

**MOPS and MASPS:**

In most cases, minimum operational performance standards (MOPS) and minimum aviation system performance standards (MASPS) provide performance requirements tailored to characteristics of a specific technology. These standards can be used to assess the feasibility of a specific technology to meet the minimum operational, safety, and performance requirements defined in an SPR. These standards normally do not provide an operational performance basis. The production of MOPS and MASPS is not part of this guidance material, whereas the SPR and INTEROP standards are included.

**Operational Services and Environment Information Capture (OSEIC):**

The OSEIC captures, in a systematic and formal way, the operational objectives of the operators and the ATS providers, in terms of the implementation of ATS supported by data



communications. All potential applicants including the operators, aircraft modifiers, system integrators, equipment designers, ATS providers, communication service providers and approval authorities are involved in the OSEIC process. Cost benefit analyses may influence the coordination process to decide what will be included in the OSED. The information captured during the OSEIC process is assessed and validated by the OSA, OPA and IA and may have to be modified as a result of those assessments to produce an updated OSED.

**Operational Safety Assessment (OSA):**

The OSA includes an operational hazard assessment (OHA) and an allocation of safety objectives and requirements (ASOR). The inputs to the OSA are derived from the OSED.

**Operational Hazard Assessment (OHA):**

The OHA is a qualitative assessment of the operational hazards associated with the OSED. For the OHA, services are examined to identify and classify hazards that could adversely affect those services. Hazards are classified according to a standardized classification scheme based on hazard severity, taking into account human factors. Overall safety objectives are assigned to the identified hazards.

**Allocation of Safety Objectives and Requirements (ASOR):**

Based on the OHA results, the ASOR allocates safety objectives to organizations, develops and validates risk mitigation strategies that are shared by multiple organizations, and allocates safety requirements to those organizations. Requirements are allocated to the CNS/ATM system elements that provide the functional capability to perform the service and the stakeholders in control of or responsible for each of the elements. Understanding the interactions of the ATS, procedures, and airspace characteristics will assist in the identification of failures, errors, and/or combinations thereof that contribute significantly to the hazards identified in the OHA. The allocation may require updating based on feedback from other processes.

**Operational Performance Assessment (OPA):**

The OPA provides methods to derive or validate required communication performance type (RCP type) from the OSED, based on the RCP concept.

**Required Communication Performance (RCP):**

The RCP concept is the operational framework to express the communication performance necessary for operation within a defined airspace or to perform a specified operation. When determining airspace requirements, an RCP is determined according to safety analysis and operational needs and is specified for ATS operational service for a given region that is deemed



necessary for safe and efficient operation of the airspace. The RCP is independent of the technology used and applicable to voice and data communication.

**Interoperability Assessment (IA):**

The IA reviews the technical, functional, and interface requirements for the defined technologies and allocates the requirements to different stakeholders and subsequent approval processes. The allocation is based on the selected technologies and functions defined in the OSED. The IA is coordinated with the OSA and OPA, and safety and performance requirements that are necessary for interoperability are allocated in the IA.

**Interoperability Requirements:**

Interoperability requirements [in the scope of the IA] are the minimum technical and functional requirements that provide the basis for ensuring compatibility among the various elements of the CNS/ATM system using specific technologies.

**Co-ordinated Requirements Determination (CRD):**

Co-ordinated requirements determination establishes requirements that require co-ordination among organizations involved in the development, qualification, operation, and approval of the CNS/ATM system. It consists of the OSEIC, OSA, OPA, and the IA. The OSEIC produces the OSED, which captures service descriptions, including operational communication processes, operational performance expectations, selected technologies, and characteristics of the intended operational environments. The OSA, OPA, and IA identify, co-ordinate, allocate, and validate the operational, safety, performance and interoperability requirements, and update the OSED, as necessary. The operational, safety, and performance requirements provide the operational basis for the operational implementation and are captured in the SPR standard. The interoperability requirements provide the technological and functional basis for the operational implementation and are captured in the INTEROP standard. The requirements in the standards are allocated to each of the stakeholders in control of or responsible for an element of the CNS/ATM system.

...

This document does not provide methods to allocate human performance to the human components of the system. However, it is necessary to qualify the human for approval, and criteria for allocation are negotiated with the approval authority during approval planning. The outputs of the co-ordinated requirements determination process are an OSED, a SPR standard, and an INTEROP standard.

Co-ordinated requirements determination may be subject to external influences which are beyond the scope of this document, but which are indicated for completeness.



Generic OSA, OPA, and IA activities:

**Identification of Requirements:**

This activity involves the capture of requirements, which identify the operational, safety, performance, and interoperability attributes from the initial OSED.

**Co-ordination of Requirements:**

This activity involves coordination among the stakeholders while they perform the appropriate assessments, in order to ensure that requirements affecting operational, safety, performance, and interoperability are addressed and correctly integrated into the OSED, SPR, and INTEROP. Co-ordination facilitates the validation of the operational, safety, performance, and interoperability requirements.

**Allocation of Requirements:**

This activity allocates the operational, safety, performance, and interoperability requirements to the different stakeholders in control of or responsible for qualifying an element of the CNS/ATM system.

**Validation of Requirements:**

This activity ensures that requirements are necessary and sufficient for operational implementation. It may include analysis, simulation evaluations, prototype testing, and operational trials. The requirements provided in the SPR and INTEROP standards are validated prior to publication of the standards. The validation includes a consistency check between the requirements specified in the INTEROP standard and the SPR standard and consistency and completeness with the OSED. To support this validation, these standards are typically developed the first time an ATS or technology is implemented for operations, trials, or prototype development. The SPR and INTEROP standards are used to establish the basis for evaluating evidence produced during development and qualification in support of an operational implementation.

Validation Objectives (as part of the Qualification Process):

For qualification of a system element, the following validation objectives should be met:

- a) Requirements validation: The requirements produced at the organisational level are validated.
- b) Risk mitigation strategy validation: Risk mitigation strategies which are not shared and related architecture and design for the element of the CNS/ATM system are validated.
- c) Human machine interface requirements validation: Human machine interface requirements are validated.



Figures:

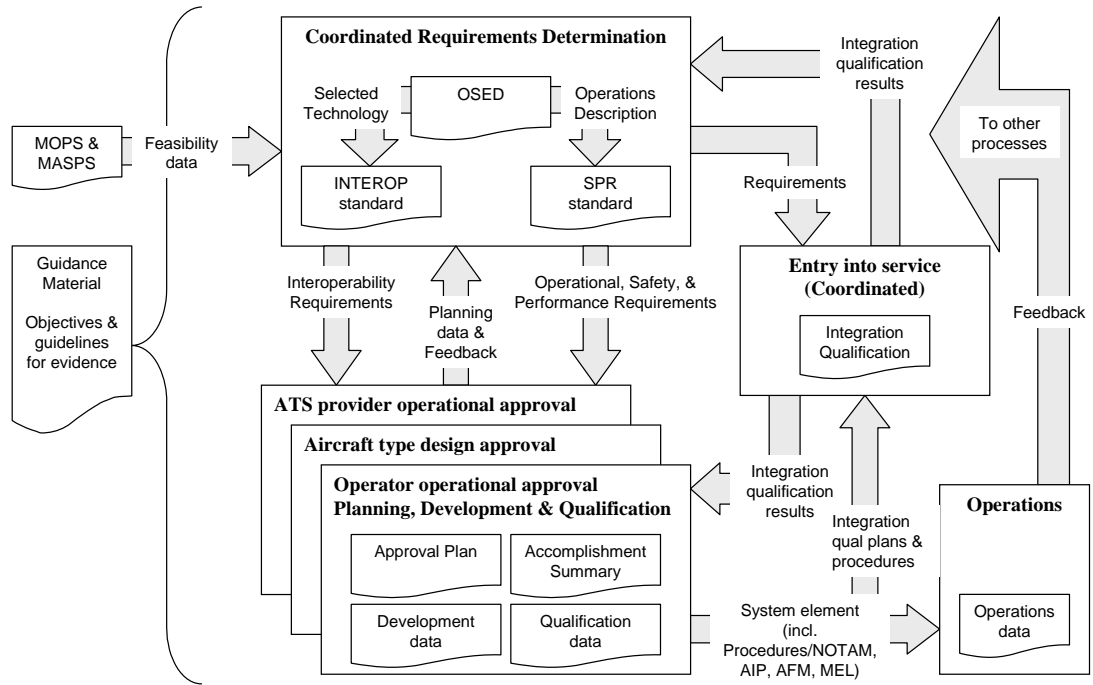


Figure 5-4: Relationship of Guidance Material to Standards and Evidence

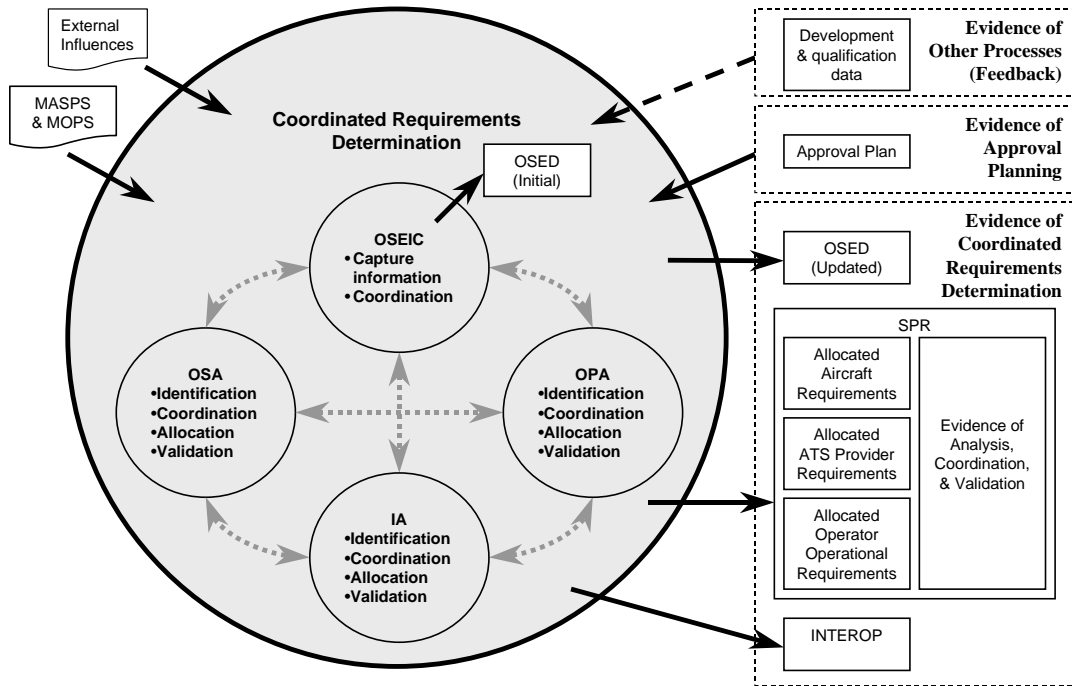


Figure 5-5: Overview of Co-ordinated Requirements Determination

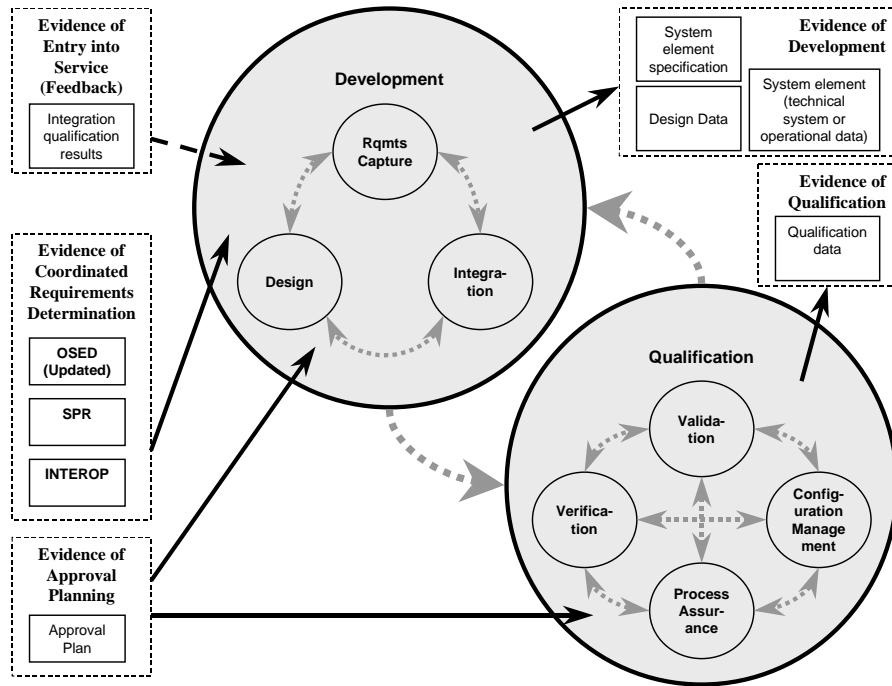


Figure 5-6: Development and Qualification of a System Element

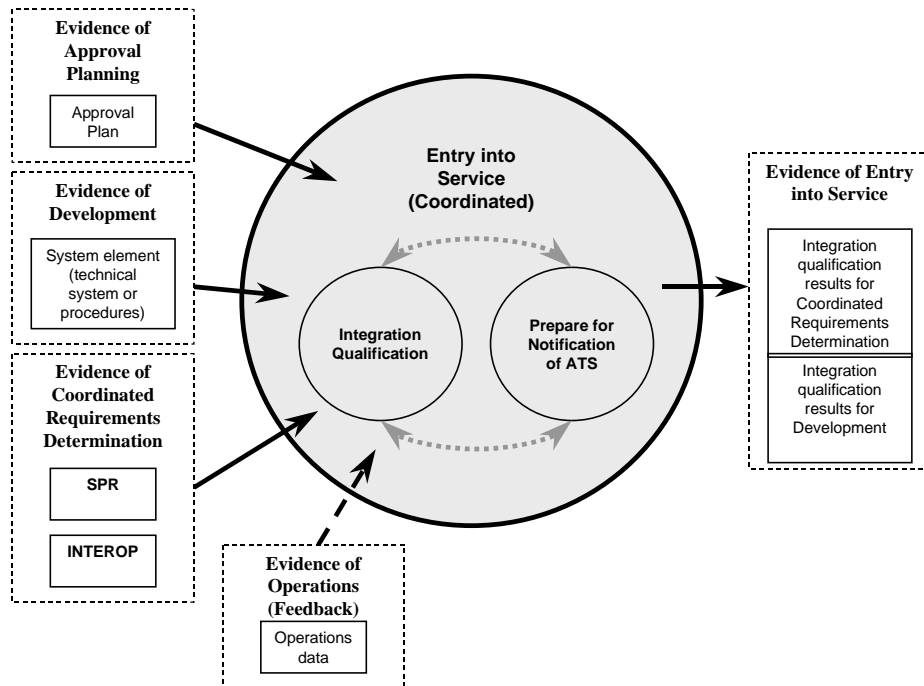


Figure 5-7: Entry-into-Service Process

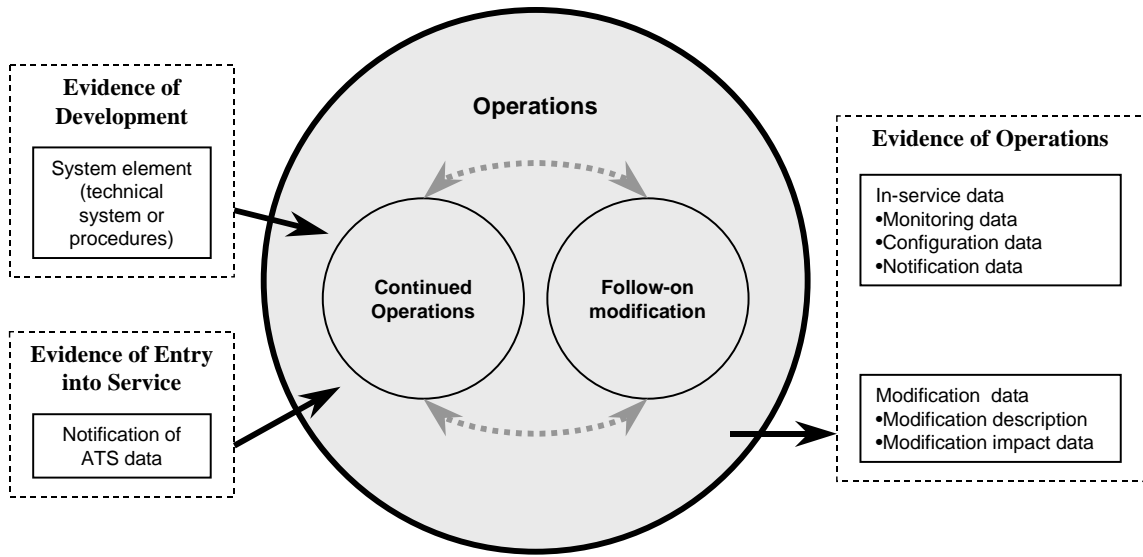


Figure 5-8: Operations Process