Environment Project FIS5-1999-00355 wp2_ade043_v10

CEMSIS:
Cost Effective Modernisation of Systems Important to Safety

Requirements Engineering Best Practice Guide for Refurbishment

V1.0

3/3/2004

Authors

Sofia Guerra, Adelard

Revisions

v0.1a	Draft for comment	14/12/03
v1.0	Final version	3/4/04

Circulation

Unrestricted

Contents

1 Introduction

This document is a Best Practice Guide on the development of safety requirements for SIS refurbishment. It aims to provide practical assistance for establishing the safety requirements for SIS refurbishment. It presents a simple requirements engineering lifecycle for a refurbishment project and it provides practical general guidance covering the lifecycle phases of the requirements engineering process, as described in [2]. It also contains general principles and goals that should be considered during the requirements lifecycle and links those goals to specific phases of the requirements engineering process.

Examples of techniques and methods that can be applied at each lifecycle phase can be found in [1] and are not included in this document.

1.1 Overview and structure of the guide

Capture of requirements is a difficult yet crucial part of SIS refurbishment. The requirements to be captured are safety, application and system requirements including those arising from interfaces. This includes both the elicitation of requirements from available documentation and methods for identifying new and existing (not documented but part of tacit knowledge) requirements.

Additional requirements to be identified are those not already captured by analysis of the existing documentation. These will be identified by a stakeholder viewpoint analysis to ensure that the requirements of all stakeholders (supplier, designer, client, regulator, operations, maintenance etc.) are recognised and fulfilled. In addition, other sources of new requirements such as change of the control system's environment, for example new regulatory requirements, new interfaces to plant and operators and new technologies, will need to be considered.

This Best Practice Guide has three main components:

- A requirements engineering process (described in Section 2).
- A set of stakeholder and viewpoints (described in Section 3).
- A claim-based view (described in Section 4).

The requirements engineering process describes the activities and aims of the phases of the requirements process for modernisation. The process is based on the "classical" requirements engineering process (see Appendix A), but it is modified and expanded in order to take into consideration the specific characteristics of a SIS modernisation project. Section 2 describes the requirements engineering process for modernisation projects, where each phase of the process is described using a common template that covers the main activities of each of the phases, and suggests checklists to be considered during the process.

The set of stakeholders or viewpoints guide the activities of the requirements process, to increase the likelihood of achieving a complete requirements specification, where requirements are not left unspecified because some of the stakeholders were not consulted during requirements elicitation. Stakeholders drive all the activities and they have to be taken into consideration during the requirements engineering process in order to make sure the set of requirements is as complete as possible. Section 3 describes the main stakeholders that have to be consulted for a Cemsis project.

The claim-based viewpoint of Requirements Capture for Refurbishment describes the properties we would like to see in the requirements and their specification and provides a clear link to safety justification framework. The claims are linked to different phases of the process, so that the developers know what has to be achieved and demonstrated in each of the phases of the process. This is described in Section 4 and provides a link to the Cemsis Safety Justification Framework. The claim and process views are complementary, and the template description of the phases of the process provides a link to the correspondent supported claims.

Section 5 provides some comments on cost-effectiveness aspects. They describe some of advantages offered by the requirement process proposed, so that the user can assess the interests of adopting this approach, in comparison with current practices.

We conclude in Section 6.

Appendix A describes the classical requirements engineering process, its phases, activities and their aims. It gives a rational for the phases of the classical requirements engineering process. Although specific projects vary in the implementation of the process phases, the general activities and their principles are constant. When using the requirements process for a specific application, the generic features will have to be combined with the particulars of the application.

Appendix B defines what are functional and non-functional requirements, as well as constraints, which need to be included in the specification of the SIS.

In Appendix C the claim-based view of the guide is shown in a graphical format.

Appendix E shows an example of the content of a user requirements specification.

Finally, Appendix F relates the requirements process with the IEC1508 safety lifecycle.

1.2 Scope of the guidance

The guide provides guidance on eliciting, analysing and documenting the requirements for a refurbishment project. Satisfaction of the requirements is not covered in this guide.

The aim of the guide is to present the good practices that have evolved in the requirements engineering community and adapt these in a way that is appropriate to the refurbishment of SIS. Specific techniques and methodologies that can be applied in each phase of the process are not covered in this guide and are discussed in [1].

The guidance is illustrated in the public domain example [3].

1.3 Target audience

This guide is intended to be of use to the full range of stakeholders of a given refurbishment process, but mainly is to be of use for those responsible for the requirements elicitation and documentation. It provides engineers with a base set of material which may be used to assist with requirements process activities.

While the guide provides guidance and rationale for the requirements engineering process and explains goals to be achieved at each stage, readers familiar with this could focus the attention on the summary tables of Section 2 (and their links) and only read other parts of the guide if needed.

1.4 Other documents and resources

The Best Practice Guide follows the deliverables D2.1 [1] and D2.2 [2]of Cemsis:

- D2.1 reviews tools and techniques for requirements capture and analysis. Industrial practices and techniques are described and classified according to their suitability for each of the phases of the Cemsis requirements engineering process.
- D2.2 establishes the background and rationale for the Best Practice Guide and lays the principles, activities and goals for the Best Practice. This are further operationalised in this document, taking into account the experience of the project partners with the case studies.

Other requirements engineering resources can be found in Section 7.

2 Requirements engineering process for modernisation

2.1 Overall approach

The requirements engineering process for modernisation is based on a generic requirements engineering process described in Appendix A, modified to take into account the specific case of modernisation, as described below. It is consistent with the Cemsis modernisation process, as seen in Figure 1.

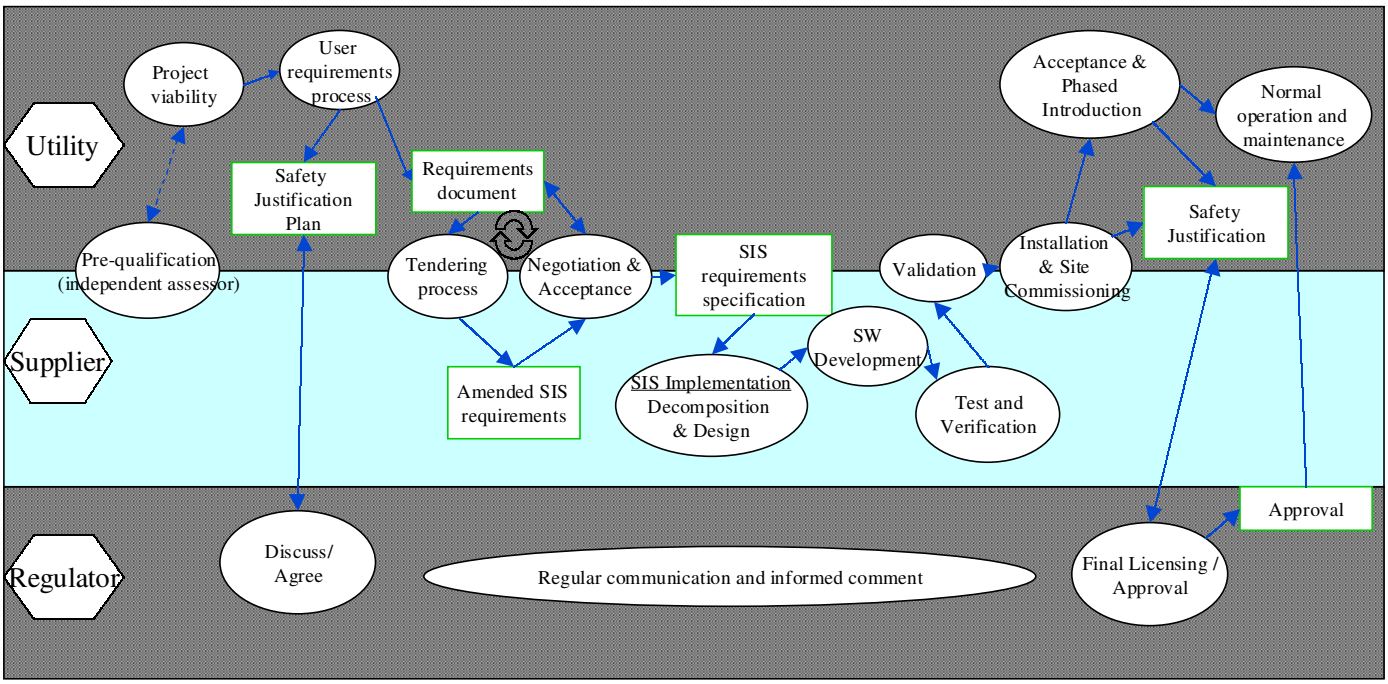


Figure 1: Cemsis modernisation process

The Cemsis process does not expand the requirements process, but includes one node that encompasses most of the requirements activities. The requirements process for modernisation has to take contractual issues into consideration contractual issues. In particular, the management of requirements documents between the utility and the supplier. In this section we also adapt the classical requirement engineering process to a set of activities that handle these contractual issues. In Figure 2 we show the Cemsis requirements engineering process.

In the case of refurbishment projects, the system being developed has to be integrated into an existing and working environment. Even in the case when the refurbished system is to keep exactly the same functionality (and hence will have the same requirements) of the existing system, the requirements of the existing system have to be gathered and a complete requirements specification is essential for the successful completion of the project. This requirements specification is typically a collection of pre-existing documents that might be extended (in the case where the existing documentation is not appropriate or sufficient) to fully specify the existing system. A full understanding of the existing functionality and of the plant design basis is essential to ensure that the system will satisfy the user's needs and achieve exactly the desired behaviour.

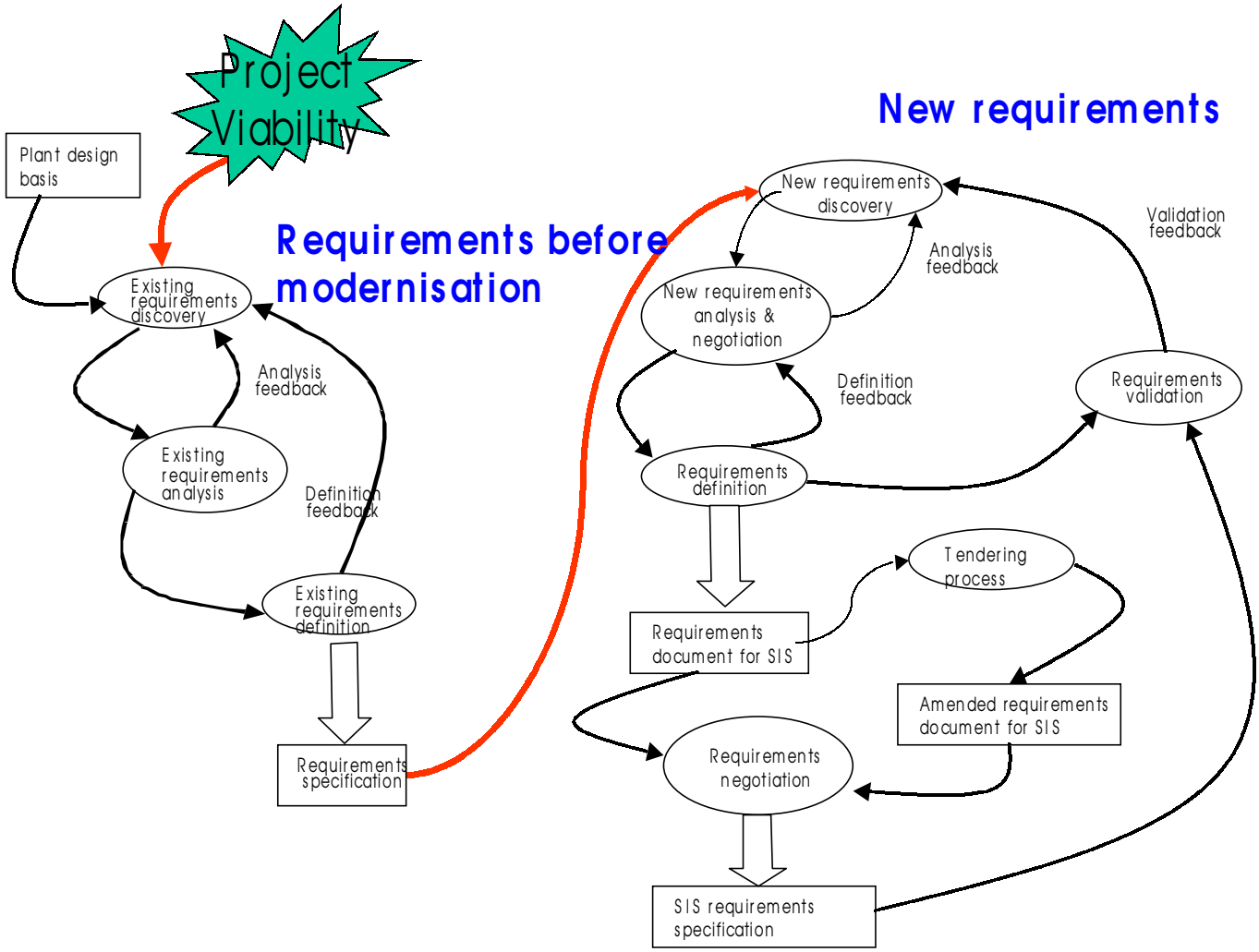


Figure 2: Requirements process for modernisation

In addition, if some of the planned functionality or features of the new system differ from those of the existing system, the new requirements have to be gathered and integrated with those of the existing system.

Therefore, the standard requirements process takes place twice:

- Initially, the requirements process takes place to gather the requirements and their documentation before modernisation (description of the system before modernisation) and to establish the plant design basis, i.e. the existing requirements of the working system that is being modernised.
- Finally, if new requirements exist, they too need to be elicited and combined with the existing requirements. The integration of the new requirements with the existing requirements, with any potential inconsistencies solved, will be the basis for the refurbishment project.

We note that the fact that the process is described as having two iterations of a classical requirements engineering process does not mean that one should discard what has been previously done for the operational system, but rather that the requirements specification is incrementally developed in two steps. It is important to consider whether the existing documents are complete and include all the information needed for the refurbishment.

The refurbishment process starts by analysing the documentation of the existing system to assess whether these documents are complete and unambiguous, or whether there is a need for further elicitation (e.g. enacted maintenance procedures that were not documented). When the existing

documentation is appropriate, we recommend its inclusion in the collection of documents that forms the requirements specification. However, one cannot assume that the existing documents are enough, but their assessment (and completion when necessary) is needed before moving on to the next stage of the process.

The first iteration of the process, i.e. the lifecycle for the existing requirements, has as an objective the development of a requirements specification. This requirements specification describes those requirements on the existing system (that is being replaced) that will be one of the inputs for the second iteration. As we have said before, this requirements specification is a collection of documents. It might include several pre-existing documents, which might be completed with further information if needed.

New and existing requirements will then be combined in the second iteration of the process. In this second iteration, the requirements specification (assembled and produced during the first iteration) is extended and modified to include the new requirements, and inconsistencies or redundancies corrected. This will form the requirements specification for the modernisation process. These two steps of the lifecycle support an incremental development of the requirements specification for the refurbishment.

However, before the requirements process actually starts, a project viability study is needed in order to decide whether the modernisation process is viable and will go ahead. The results and documents produced during the viability study will subsequently inform the other phases of the requirements process.

This process is further refined by taking into consideration the contractual arrangements of the Cemsis modernisation process (see Figure 1). When a decision is taken to modernise a SIS, the utility defines a set of requirements that will be passed to the suppliers: the *requirements document for SIS*. In principle, potential suppliers will tender to develop the system in question and they will comment and propose amendments to the specification developed by the utility. Utility and the chosen supplier will then agree a set of final requirements, the accepted *SIS specification*, which is a description of the system to be developed. The regulator is often (not always) invited to give an opinion.

Once the requirements process is sufficiently complete, the design phase starts. The design phase transforms the requirements specification into design that will be the base of the system. Design determines the components and devices needed for the product to be built.

In most projects, the requirements process is not limited to the front-end of the lifecycle. The different phases of the development process increase the stakeholders awareness of the stipulations made in the requirements specification, and more requirements are added, deleted or modified. However, late modifications of the requirements specification tend to be costly and they should be minimised. In the case of Cemsis this is even more the case, as there are several safety implications of late requirements changes that are costly. Once a requirements specification is approved, it can be subject to changes, but through a strict quality assurance plan that ensures that all required verification and validation and independent assessment activities are repeated.

In Figure 2, circles correspond to activities and squares to documents produced. Each of the phases of the requirements process is explained in the following sections. Note that the process shown here is very general. Each of these phases can be divided into more specific phases, depending on the practices of the organisations undertaking the refurbishment. The specific content of each specification, and in particular in the (user) requirements document for SIS, vary among partners of the project and from organisation to organisation. However, the activities described here are generally performed, and this process can be further developed and extended to fit the needs of specific users.

Discussion of requirements engineering methods and techniques is not included in this section and is contained in [1].

2.2 Structure of the section

This section discusses each of the phases of the requirements engineering process, where Section 2.3 covers the project viability, Section 2.4 the existing requirements part of the process (i.e. the first iteration of the process) and Section 2.5 the new requirements (second iteration of the process).

Finally, Section 2.6 covers requirements validation and Section 2.7 requirements management. Table 1 links sections of this document with the phases of the process where they are described.

Each of the phases is described using a common template as follows:

- Aims – describe the aims of the phase
- Brief Description – brief description of the phase
- Input – input (mainly documents) for the phase
- Output- what will be produced during the phase
- Main activities- main activities to be performed in this phase
- Goals - It lists the goals to be achieved and evidence to be developed during the phase. Goals can then be claims on the requirements process and this item establishes the relationship with the claim-based view.

Process phases	**Section number**
Project viability	Section 2.3
Existing requirements	Section 2.4
Existing requirements identification	Section 2.4.1
Existing requirements analysis	Section 2.4.2
Existing requirements definition	Section 2.4.3
New requirements	Section 2.5
New requirements identification	Section 2.5.1
New requirements analysis	Section 2.5.2
New requirements definition	Section 2.5.3
Requirements tendering, negotiation and analysis	Section 2.5.4
Requirements validation	Section 2.6
Requirements management	Section 2.7

Table 1: Summary of the section

2.3 Project viability

The project viability is a preliminary study to decide whether to go ahead with the project. This study considers a series of factors that influence the feasibility of the refurbishment. The main stakeholders involved in the project will gather enough facts to ensure that the project has a worthwhile objective, is possible to achieve (in terms of cost, safety and market capabilities) and has the commitment from the stakeholders. However, often in practice there is no choice; the renovation is dictated by a variety of reasons, typically one of the following:

- *Obsolete equipment.* The equipment may be obsolete, this can be partly addressed by maintaining a large spares inventory, but there can be problems in maintaining knowledge and expertise in the equipment (no supplier support, training and retention of maintenance staff, etc.).
- *Physical deterioration of the equipment.* The equipment might have a finite “shelf-life” so long-term maintenance may be infeasible even if spares are available.

- *Unsatisfactory performance of the existing equipment.* The equipment may not be adequately reliable, either due to design flaws or maintenance problems. This does not necessarily mean the equipment is unsafe (e.g. it could fail safe), but it could have economic consequences by decreasing production output.

- *Maintenance difficulties.* E.g. there may be commercial pressures to reduce maintenance costs, e.g. increase maintenance intervals, reduce maintenance staff levels and availability (e.g. day-shift only). This may be difficult to achieve with the existing system. In addition, there may be problems regarding the availability of personnel with know-how on the old technology.

- *Regulatory change.* Regulators may impose new safety requirements that involve changes that are difficult to implement on the existing system.

Other reasons that might influence the refurbishment decision include

- *Plant operational changes*. Plant operating regimes might change to improve operation performance. This might require more sophisticated logic to maintain safety.

- *Changes in the operating environment.* This could include changes in sensors or more generally a change in the environment/external systems that interact with the SIS.

When the project viability study has been completed, a preliminary set of requirements would have been collected. These requirements will be the basis for the requirements engineering process described below. They will be the basis for the information assembled in the first requirements identification and will be refined when the new requirements elicitation is being performed.

In addition, as the project viability study will have to consider market capabilities, the utility will have a clear idea of possible suppliers for the system being modernised. This both limits and informs the tender phase of the modernisation process.

Table 2 summarises the project viability phase.

Aims	The project viability study is a preliminary analysis to decide whether the project is viable and whether its cost makes it worthwhile (in cases where the project is not mandatory), i.e. to decide whether to go ahead with the project.
Brief description	In this study the main stakeholders involved consider a series of factors that influence the feasibility of the project, and gather enough facts to ensure that the project has a worthwhile objective, is achievable (in terms of cost, safety and market capabilities) and has their own commitment. However, often in practice there is no choice; the renovation is unavoidably dictated by a variety or reasons including reliability of installed old equipment, spare part situation, maintenance or requirements from the licensing authority.
Input	The following are inputs to the study and need to be considered before the decision is taken. • Strategic plan • Risk analysis • Safety review of the system to be replaced • Cost estimation • Market capabilities to supply products • Budget constraints • Plant level requirements

	• Implementation constraints • Plant revenue/cost benefits
Output	The following are outputs generated during this study: • identification of boundaries and scope • identification of constraints • identification of purpose - clear statement of the aims of the project/refurbishment • preliminary cost estimation • identification of stakeholders • preliminary set of requirements
Main activities	1. Clear Statement of the aims of the refurbishment. 2. Stakeholders identification. In order to decide whether to go ahead with the project, the main stakeholders have to be identified. In Section 3 we list the stakeholders usually involved in a refurbishment project. The main stakeholders typically involved in the project viability study include users, managers (contribute with the business advantages of the project), developers (who have a say in the feasibility of the project). 3. Stakeholders have to go through the following points to reach a decision: • Purpose: what is the purpose of the project? • Advantage: what advantages does it provide? • Reasonable: is the project effort greater than the advantage? • Feasibility: can the product achieve the advantage? • Achievable: does the organisation have the skills needed? • Safety: can the product satisfy the required safety level? • Technology: are there technologies available for the development of the product? • Option: is there an option not to go ahead or is the project inevitable? 4. Definition of the scope of the work and its boundaries. 5. Constraints identification. Constraints are global requirements and they apply to the entire system. Examples of constraints are the platform where the product is going to run, standards it has to comply with, amount of money or effort spent on the work itself, or other applications it has to interface with. 6. Preliminary cost estimation.

Table 2: Project viability summary

2.4 Existing system, i.e. requirements before renovation/modernisation

2.4.1 Requirements identification

The identification of the existing requirements consists of collecting information about the existing system and its environment and the identification of the plant design basis.

The existing system requirements are stated in plant documentation, such as the Final Safety Analysis Report (FSAR) or the Plant Safety Case (PSC). Requirements may, however, be immersed and scattered over a large number of paper documents, which might be missing or poorly maintained. Therefore, it may be hard for the system designer, vendor and the licensing authority to comprehend the full set of requirements [10]. In this phase of the process it is necessary to overcome these potential problems and collect a complete survey of all existing functions and interfaces of the system to be replaced, and this collected information will be basis for its functional specification and architecture.

Below we suggest a checklist for the relevant information to be collected at this stage. This checklist is organised in two types of information:

- Information about the existing system.
- Information about the system's environment.

As the documentation might not be complete or up to date, it is important to liaise with operation and maintenance staff to validate and complete the information collected in the documents assembled. It is important to have clear information about operating procedures as enacted, as these often differ from their documented versions.

Table 3 describes the identification of the existing requirements.

Aims	Collecting information about existing system and its environment.
Brief description	Requirements identification or elicitation is the process through which the stakeholders discover, review, articulate and understand users' needs and domain knowledge about the system. Some of the existing system requirements are stated in plant documentation, such as the plant safety case. Requirements may be distributed over a large number of documents, which might be missing or poorly maintained. In this phase we need to overcome these potential problems and collect a complete survey of all existing functions and interfaces of the system to be replaced. Such shortcomings are also to be considered and addressed in the analysis (Section 2.4.2). Requirements identification and analysis are interdependent activities, where the analysis of requirements identified might result in the need for further elicitation, or elaboration.
Input	1. Project viability results. In particular, • The project viability determined the context and boundaries of the work, the purpose and constraints that apply to any solution. These guide the elicitation (and analysis) of further requirements. These should also be reviewed and possibly refined during elicitation and analysis. • The project viability determined the main stakeholders of the project. The stakeholders will be interviewed to get an understanding of the project. 2. Documentation about existing system • plant description • drawings (logic diagrams, interface specification) • safety requirements (traceable to plant safety analysis) • design rationale for safety system • information on logic systems • field experience

	• operations and maintenance procedures
Output	Requirements of existing system.
Main activities	1. Assemble information about existing system. • plant interface documentation • drawings (logic diagrams, interface specification) • safety requirements (traceable to plant safety analysis) • design rationale for safety system • information on logic systems • field experience • operations and maintenance procedures • documentation on equipment and components • existing safety claims from the plant safety case 2. Establish system environment. • plant descriptions and schematics • plant safety report • states of the neighbouring systems • systems maintenance, periodic testing and operation procedures • operational safety and problem reports 3. Consider existing information and interview existing staff (including operating and maintenance). The documentation might not be complete or up to date. It is therefore important to talk to operation and maintenance staff to validate and complete the information collected. It is important to have clear information about operating procedures as enacted, as these often differ from their documented versions.
Goals	Configuration consistency (Section 4.1) Completeness (§5 of Section 4.3.1)

Table 3: Summary of existing requirements identification

2.4.2 Requirements analysis

The information assembled in the requirements discovery has to be analysed in order to:

- *Check currency with respect to actual system.* Is the information/documentation consistent with the system that is being used? Are all the documents/information consistent?
- *Look for gaps in documents.* Do the documents cover all required aspects? Are there any undocumented aspects of the actual system?
- *Correctness.* Is the system correct, i.e. is it doing what it should do in terms of safety, operation and non-functional aspects?

Requirements analysis and discovery are intertwining processes: requirements are analysed while discovered, and the analysis informs the discovery of further requirements. Gaps in the information collected are made evident by the analysis performed of the information collected.
Table 4 summarises this phase of the process.

Aims	Establish a correct, complete and consistent set of requirements for the

	existent system.
Brief description	The requirements collected during the identification are integrated and analysed. Usually, this will result in the identification of missing requirements, inconsistencies and requirements conflicts. Requirements identification and analysis are intertwined processes: during identification it is necessary to carry out some analysis, and analysis informs the identification process. While requirements are being elicited, some problems might be detected and analysis is immediately carried out.
Input	1. From the Project Viability (Section 2.3): • The project viability has determined the context and boundaries of the work, the purpose and constraints that apply to any solution. These guide the analysis of the requirements. Requirements should also be reviewed and possibly refined during elicitation and analysis. • The project viability has determined the main stakeholders of the project, who are interviewed to get an understanding of the project. 2. From the Requirements Identification (Section 2.4.1) • Requirements identified in the existing requirements identification phase.
Output	Complete, correct and consistent requirements of the existing system.
Main activities	• Check currency with respect to real system. Is the information/documentation consistent with the system that is being used? Are all the documents/information consistent? • Look for gaps in the document. Do the documents cover all required aspects? Are there any undocumented aspects of the actual system? • Correctness. Is the system correct, i.e. is it doing what it should do in terms of safety, operation and non-functional aspects? • Evidence of the quality of the documentation. How much to trust the existing documentation? Is the level of detail adequate?
Goals	Currency (§1 of Section 4.3.1) Consistency (§2 of Section 4.3.1) Correctness (§3 of Section 4.3.1) Precision (§4 of Section 4.3.1) Completeness (§5 of Section 4.3.1)

Table 4: Summary of requirements analysis

2.4.3 Requirements definition

A requirements document (or set of documents) that clearly and precisely described the existing system is assembled, which we call the *requirements specification*. This document includes the information collected and analysed in the previous phases of the process (project viability, requirements identification and requirements analysis). This phase of the process is not independent from the requirements discovery and analysis, but it takes place while requirements are being identified and analysed.

The documents collected in the requirements identification phase will be part of this requirements specification. The documents should be kept in the original form if possible, as modifications will

require additional verification that might introduce errors. However, analysis of the requirements might identify information relevant to the refurbishment that has not been documented (or is missing from the documents assembled), requirements that are unclear, ambiguous or inconsistent. Hence, as this document will be a basis for the replacement of the system, in general we need to complete or complement the existing documentation.

Table 5 summarises the requirements definition phase of the first iteration of the process.

Aims	Development of a requirements specification that is a complete, consistent and clear description of the existing system and its environment.
Brief description	Writing a specification is not an activity independent from the activities that surround it. The requirements definition takes place during the discovery and analysis. The specification is built rather than written all at once—the specification is assembled during the lifecycle and incrementally developed. Documentation should be kept in the original form if possible. It should only be modified if the existing documentation is unclear, insufficient or out of date.
Input	Same as input for requirements identification (Section 2.4.1) and for requirements analysis (Section 2.4.2).
Output	Document (or set of documents) that clearly and precisely records each of the requirements of the existing system.
Main activities	1. Incremental collection of existing documents. 2. Completion of the description if existing documents are insufficient.
Goals	Currency (§1 of Section 4.3.1) Consistency (§2 of Section 4.3.1) Correctness (§3 of Section 4.3.1) Precision (§4 of Section 4.3.1) Completeness (§5 of Section 4.3.1)

Table 5: Summary of existing requirements definition

2.5 New requirements

2.5.1 Requirements discovery

When the modernisation process consists of replacing a system by an equivalent system, it is important to fully understand the system being replaced so that a compatible system is put in place. Therefore, the first part of the requirements process (described in Section 2.4) is always performed.

In most replacement cases, however, there are differences between the existing system and the new system that is going to replace it. Either because an old analogue system is replaced by a programmable digital device, or because new regulations are in place with which the existing system does not comply, the new requirements—those that differ from the existing ones in the specification document(s)—need to be identified and documented.

Table 6 describes the discovery of new requirements phase of the process. It lists possible reasons for new requirements or sources for new requirements that should be considered for elicitation. It also lists some possible techniques and methods that might be used to support the elicitation.

Aims	Elicitation of all additional requirements (if applicable).
Brief description	Requirements elicitation is the process through which the stakeholders discover, review, articulate and understand users' needs and domain

	knowledge on the system and development activities. There is a variety of possible sources for new requirements that need to be considered to achieve completeness of the requirements identification and consequently of the requirements specification. We note that the requirements should be complete in the sense that they provide all the information necessary for the user to carry out the next process step (no missing parts). As the aim is to develop a specification that will be developed by the supplier, here completeness does not imply the description of every aspect of the system to be developed, but enough information for the supplier to develop their suggested specification.
Input	1. Project viability results. In particular, • The project viability determined the context and boundaries of the work, the purpose and constraints that apply to any solution. These guide the elicitation (and analysis) of further requirements. These should also be reviewed and possibly refined during elicitation and analysis. • The project viability determined the main stakeholders of the project. These stakeholders will be interviewed to get an understanding of the project. 2. Hazard and risk analyses. Safety requirements to avoid and minimise their consequences arise from risk and hazard analyses.
Output	New requirements for the refurbishment.
Main activities	1. Look at the following sources or reasons for new requirements: • new regulations • new technology • operational needs (user interface), e.g. might be a choice between entering a parameter with a button or new computer keyboard. The use of a button might be safer because it is less error prone. • maintenance of hardware: obsolescence of hardware. The consequences of hardware obsolescence might be reduced if several parts are bought, or by terms of standardisation of components or sensors. • maintenance of software, e.g. the interface with new devices or a new operating system could be the source of new requirements. • new functionality, e.g. the change of actual application, or modernisation of another part of the plant • modifiability: tools, documentation • transition requirements, i.e. requirements related to the replacement of the old by the new system. For example, time limitations, which may limit the amount of testing done, or the need to run the new and legacy systems in parallel for a limited period of time. • safety e.g. due to new functionality the system went from SIL2 to SIL3, or due to combination of two components of initial different SILs • availability • reliability • results from field experience • human factors analysis - e.g. impact on safety, usability • security- e.g. accountability of actions, unauthorised access 2. Consult the stakeholders to gather all the different viewpoints and requirements. Stakeholders should have been identified in the project

	viability phase. A list of likely stakeholders can be found in Section 3.1. 3. The following techniques and methods can be used to support the requirements elicitation (and the requirements analysis, as described in Section 2.5.2): • Hazop • Hazan • design basis accident analysis • probabilistic safety analysis • fault tree analysis • fault hazard analysis • brainstorming • checklists • interviews
Goals	Completeness (§1 of Section 4.3.2)

Table 6: Summary of new requirements discovery

2.5.2 Requirements analysis

The analysis of the utility requirements gathered during both the previous iteration of the process (existing requirements) and in this phase (new requirements) have to be analysed. The requirements specification of the existing system is extended into a user requirements specification (called the *requirements document for SIS* in Figure 2). This analysis will consider the integrated set of existing and new requirements, check the integrated set for consistency and remove redundant requirements (see list of activities in Table 7).

In addition, each individual requirement needs to be analysed. To begin with, the need for a requirement should be analysed, as some requirements might be proposed that do not contribute to the goals set for the project. The requirements should also be assessed in order to ensure that they are feasible in terms of the resources available and safety targets. In addition, it is necessary to check their consistency and completeness insofar as they are free from contradictions and omissions of services or constraints.

We note that completeness here means that all the necessary information is provided for the suppliers' bidding. Completeness is dependent upon the author's intent, and in this case the aim is to develop a basis for the tendering phase. The resulting specification may not fully describe the new system and thus may not be inappropriate as the basis for design. Nevertheless, it is desirable to leave aspects unspecified for the supplier to complete as they review and extend this specification. The content of the specification at this stage and the level of incompleteness vary from organisation to organisation.

It is important to be careful with ambiguity and resulting misunderstandings. A requirement should be unambiguous in the sense that different users would give the same interpretation to the requirement. This includes both lack of detail and ambiguous language, giving rise to possibly different interpretations of the expressed requirement.

In order to know whether the system has successfully implemented the tasks, it is necessary to have a fit criterion to measure the achievement of the goal. The fit criterion is a quantified goal that the solution has to meet. A fit criterion is a precise, quantified, testable statement of the requirement such that it is possible to test if the solution matches the original requirement (and assess requirements satisfiability). For functional requirements, fit criteria specify how to know that the system has successfully completed the required action. For non-functional requirements (which are behavioural properties that the product must have), fit criteria quantify the resulting behaviour or quality of the product. For example, the fit criterion for usability could be the amount of learning time or training required to be able to use the system.

Examples of analysis techniques are:

- Prototypes, where a rudimentary version of the product (or part of the product) is developed. The legacy system that is being refurbished might be used as a prototype. Prototypes demonstrate the requirements, helping the developers and users to understand the consequences of the stipulations made.
- Testing. Acceptance tests include: acceptance user requirements and accepted system requirements
- Scenarios, where "stories" are developed that show the steps needed to complete a particular activity of the product.

Summary of the requirements analyses can be seen in Table 7.

Aims	To establish an agreed set of sound, correct and complete requirements.
Brief description	The analysis and negotiation involve several activities whose aims are to discover problems with the requirements and achieve an agreement among stakeholders and their different viewpoints. While requirements analysis usually results in the identification of missing requirements or inconsistencies, negotiation will result in the agreement on a sub-set of requirements. There might be more desirable requirements than can be implemented so decision needs to be made about leaving out requirements and modifying requirements to result in a safer lower-cost system. The discovery phase usually needs to be re-entered to solve some of the problems found during the analysis or to find alternative requirements during negotiation
Input	• Existing requirements specification (developed as described in Section 2.4.3) • Requirements identified in the new requirements discovery (Section 2.5.1) • Results from requirements analysis
Output	Complete, correct and consistent requirements of the refurbishment system.
Main activities	1. Check each requirement for: • Relevance of purpose: are all the requirements needed? • Feasibility: are the requirements viable within the constraints? • Consistency: do we have any conflicting requirements? • Completeness: a requirement should be complete in the sense that it provides all the program-specific information necessary for the user to carry out the next process step (no missing parts). Completeness is dependent upon the author's intent. However, it is virtually impossible to make all the correct requirements the first time. The real challenge is how to decide what kinds and levels of incompleteness the developer can live with. It is important to determine stopping conditions in the pursuit of complete requirements specifications, and this should be agreed with the suppliers tendering to develop the system. • lack of ambiguity: are requirements ambiguous? • Measurable: how to know whether the goal has been met? • Abstraction and level of detail. The detail of the requirements needs to be examined to assess whether the requirements are

	sufficiently abstract to avoid implementation bias and clutter, though not so abstract that they can not be analysed. 2. Integration of existing requirements with new requirements 3. Analysis of the new and existing requirements for consistency. Resolve any inconsistencies detected. 4. Removal of requirements of the old system that are not relevant for the new system. 5. Removal of redundant requirements. 6. Negotiation of different requirements alternatives considering • cost • safety properties • priorities of requirements
Goals	Completeness (§1 of Section 4.3.2) Feasibility of safety justification (§2 of Section 4.3.2) Consistency (§3 of Section 4.3.2) Precision (§4 of Section 4.3.2) Relevance of extra function (§5 of Section 4.3.2) Usability for intended users (§6 of Section 4.3.2)

Table 7: Summary of new requirements analysis

2.5.3 Requirements definition

When a consistent and complete requirements set has been collected and analysed, the utility extends the previous set of documents with the definition of the new user requirements, i.e. the *requirements document for SIS*. The detail of these documents varies depending on user practices. Specimen content for a user requirement specification can be seen in Appendix E. However, independently of how much detail they include, these documents are the basis of the tendering process, and they will be sent to the suppliers tendering for the project.

Table 8 summarises the requirements definition phase.

Aims	Development of a requirements specification that is a complete, consistent and clear description of the requirements on the refurbishment system (and its environment).
Brief description	The requirements identification and analysis result in a set of goals, functions and constraints that are collected into the specification of the system. The specification documents the agreed requirements. When requirements are being gathered, sometimes they are not fully formed or complete. On the other hand, a requirements specification (i.e. the documents that describe the requirements) is the basis for the development. Therefore, the initial description of the requirements might have to be expanded and refined in order to produce a complete, accurate and sufficiently detailed requirements specification that can be understood by all the people involved. Hence, the discovery and analysis phase might be re-entered.
Input	• Existing requirements specification (developed as described in Section 2.4.3). • Requirements identified in the new requirements discovery (Section 2.5.1).

	• Results from requirements analysis (Section 2.5.2). • Specification format agreed with vendor.
Output	**Requirement document for SIS:** Document (or set of documents) that clearly and precisely records the minimum set the requirements needed for the tendering phase of the refurbishment. Diagrammatic presentation and requirements animation might help the vendor to understand the requirements (e.g. fault sequence)
Main activities	1. Finish requirements specification. The requirements definition takes place mainly during the discovery and analysis. The specification is built rather than written all at once. However, the specification needs to be finished at this stage. The requirements document shall include: • precise definition of system boundaries, that is of the system interface to the plant and its man-machine interface • constraints placed on the system. Constraints are global requirements, i.e. they apply to the entire system. • specification of function and non-functional properties (e.g. dependability, security, performance) required for the system • precise definition of the plant input that must be monitored by the SIS and SIS outputs to the plant that must be controlled and of their ranges • requirements regarding operation and maintenance • SIS specific deployment requirements • results of any failure analysis performed on the above requirements 2. Hold a final meeting with the stakeholders to agree the requirements. 3. Maintain the documentation in a known, retrievable local to help future maintenance and potential refurbishment.
Goals	Precision (§4 of Section 4.3.2) Usability for intended users (§6 of Section 4.3.2)

Table 8: Summary of new requirements definition

2.5.4 Requirements tendering, negotiation and acceptance

The suppliers tendering for this work suggest and negotiate amendments that must be accepted by the utility. This process will result in the *amended requirements document for SIS* from Figure 2.

The requirements documents are then subjected to analysis and negotiation between the utility and the supplier. The requirements are individually assessed to the criteria described in Section 2.5.2. In addition, further assessment is carried out, as described in Table 9.

The result of this analysis and negotiation between user and supplier is the definition of accepted user/supplier requirements specification, or *SIS requirements specification*. This specification is the basis for the work, and it should be analysed in order to ensure that is a reasonable contract and design foundation for the project.

Aims	To agree a set of sound, correct and complete requirements between the supplier and the utility.

Brief description	The requirements specification is used as the basis for the tendering process. From this specification, the supplier develops the *amended requirements specification for SIS*. In this phase user and supplier negotiate this amended document in order to agree the specification that will be used as the specification of the SIS being modernised. The different alternatives are negotiating considering: • cost analysis • safety properties • priorities of requirements • technical feasibility
Input	Amended requirements document for SIS (developed by the supplier)
Output	SIS requirement specification
Main activities	1. Analyse requirements according to: • relevance of purpose • feasibility in terms of resources available, safety targets and other constraints • consistency • completeness • lack of ambiguity • measurability 2. Amended specification is assessed according to: • compatibility of supplier requirements with the user requirements • affordability and cost benefit of different options • feasibility • acceptability • prioritisation of requirements • contradictory requirements (which can be negotiated)
Goals	Completeness (§1 of Section 4.3.2) Feasibility of safety justification (§2 of Section 4.3.2) Consistency (§3 of Section 4.3.2) Precision (§4 of Section 4.3.2) Relevance of extra function(s) (§5 of Section 4.3.2) Usability for intended users (§6 of Section 4.3.2)

Table 9: Summary of requirements negotiation and acceptance

2.6 Requirements validation

After the requirements specification has been written, validation is the process of determining whether the requirements define the system that the stakeholders want. This includes assuring the completeness, consistency and correctness of the set of requirements. If requirements validation is inadequate, errors will propagate throughout the lifecycle to system design and implementation. The costs to fix are higher if errors are found later in the lifecycle. Appendix A.5 discusses requirements validation in more detail. Table 10 gives a summary of this phase.

Aims	After the requirements specification has been written, determine whether the requirements specification defines the system that the stakeholders

	want.
Brief description	Requirements validation and analysis have much in common. Here the distinction is that: • Analysis does not assume that the requirements are completely and correctly documented, nor that the language used is formal and structured. • Validation checks the final draft of the document for completeness, ambiguity and correctness.
Input	SIS requirements specification
Output	Revised SIS requirements specification (if needed)
Main activities	To demonstrate the validity of the safety requirements it is necessary to show that: 1. A necessary and sufficient set of system-level safety requirements exist, which describe the functionality and performance required of the computer system in order to support a safe system. 2. The system-level safety requirements are derived from a hazard and risk analysis of the environment in which the system is required to operate. 3. The failure modes that must be detected and mitigated in order to control the hazard rates have been identified for all of the following: associated systems, system-system interactions, equipment, pre-existing software and all user-system interactions. 4. The failure modes identified include generic failures relevant to the safety-related application, e.g. security threats, loss of communications, and loss of power. 5. Hazard analysis has been carried out identifying all hazardous failure modes. 6. Safety requirements embody the results of the hazard and risk analyses. Common techniques for requirements validation include prototyping, formal requirements inspection (meetings involving main stakeholders).
Goals	Requirements validity (Section 4.3)

Table 10: Summary of requirements validation

2.7 Requirements management

Requirements management consists of planning and controlling of the all phases of the requirements engineering process. It concerns the control of requirements information and, in particular, its integrity during the life of the system with respect to changes of the system and its environment. It includes organisation, traceability, analysis and visualisation.

The management of the changes to the system requirements includes control of the impact these changes have on:

- other requirements and their documentation
- the design and implementation of the specification

The former is concerned with the collection, verification and assessment of changes; the latter with the consequences of the changes in the system as a whole.

To manage requirements we need to keep *requirements traceability.* Requirements traceability assists the understanding of the relationships between requirements, their sources, design and implementation of a system. A requirement is traceable if we can identify the parts of the system that exist because of that requirement. Traceability management applies to the entire development lifecycle from project initiation through operation and maintenance.

For more on requirements management see Appendix A.6. Table 11 gives a summary of the activities involved in requirements management.

Aims	Management and planning of all requirements activities to achieve a product of quality.
Main activities	1. Management of changes of the requirements and their impact on • other requirements and their documentation • the design and implementation of the specification 2. Keep requirements traceability - relationships between requirements, its sources, design and implementation [8]. 1. Define policies for requirements management—the procedures and standards that should be followed to achieve the requirements management goals should be defined and documented as part of the quality management system. This should include: • objectives for the requirements management process and their rationale • standards for requirements specification documents • change management and control policies • requirements review and validation policies • traceability policies 2. Define traceability policies—define what information on dependencies between requirements should be maintained and how this information should be used and managed. Different types of traceability information might be kept 1. Requirements-source traceability: link requirements with their sources 2. Requirements-rationale traceability: link requirements with their rationale 3. Requirements-requirements traceability: link requirements with other dependent requirements 4. Requirements-architecture traceability: link requirements with the sub-system where they are implemented 5. Requirements-design traceability: link requirements with components which are used to implement the requirements 6. Requirements-interface traceability: link requirements with the interfaces of external systems which are used in the provision of the requirements The traceability information might be kept using a different number of methods: 1. traceability tables that record the dependencies

	between requirements or between requirements and design components or plant hazards 2. traceability lists, which are simplified form of traceability tables 3. databases can be used to maintain the requirements. This makes it easier to manage large number of requirements and to reuse the requirements when this system is modernised.
Goals	Requirements traceability (Section 4.2) Configuration consistency (Section 4.1)

Table 11: Summary of requirements management

3 Stakeholders and viewpoints

3.1 Stakeholders

A stakeholder is anyone with an interest in or involvement with the product. It includes people that use the product, that build the product, and whose knowledge is needed to build the product. Examples of stakeholders include users, sponsors, testers, business analysts, technology experts, system designers, legal experts, marketing experts and domain experts.

It is important to identify all the stakeholders for a given project. If we fail to identify some of them, we might miss some of the requirements. In this case, several modifications will be necessary later in the development process to accommodate the requirements of the stakeholders not originally included.

In order to identify the stakeholders, it is useful to look at candidate categories, and from them derive the relevant stakeholders. The following categories of stakeholders are adapted from [4].

- Client: who's paying for the work
- Customer: who's buying
- Users: end users
- Management: directors, project managers or any other type of managers if they have any interest in the work.
- Developers: everyone who is involved in the development, including designers, programmers, testers, requirements engineers.
- Domain experts: sources of subject matter knowledge.
- Technical experts: experts on subjects related to non-functional requirements.
- Inspectors. Include safety inspectors, auditors, regulators, license authorities and safety departments.
- Legal: lawyers consulted for legal requirements, and applicable standards.
- Regulators, licensors and certifiers

Other classes of stakeholders that we might want to consider are:

- Opposition: people who oppose the product
- Professional bodies

- Government
- Special interest groups.
- Adjacent systems: adjacent systems, defined by the scope analysis discuss earlier, are the systems that directly interface with the product. These systems are likely to impose requirements or constraints.

For each stakeholder it is necessary to identify:

- Stakeholder identification (e.g. role/job title, person name, organisation name).
- Knowledge needed by the project.
- Degree of influence for that stakeholder/knowledge combination.

3.2 Viewpoints

Different stakeholders have different perspectives or views on the system that they are trying to describe. These perspectives are often partial or incomplete descriptions, as they tend to reflect the particular roles or responsibilities of each stakeholder. The agent or stakeholder combined with the view of the agent is called viewpoint [5].

Organising requirements in different viewpoints helps to structure the elicitation. It also helps the prioritisation and management of requirements [8].

Different stakeholders might have common goals and be involved with similar parts of the system. However, in the case of the different goals it is natural that some of their requirements are contradictory. The following table shows how we might aggregate viewpoints according to goals of the stakeholders.

Primary goal	*Example stakeholders*
Safety and licensing	Licensors/regulators, safety departments, engineering
System design and development	Engineering department, supply chain (system integrator, component supplier)
Operation and maintenance	End users—operations department, Maintenance, designer, client

Table 12: Example stakeholders and their primary goal

Requirements elicitation techniques such as those listed in Table 6 should be used to capture the viewpoints and requirements of the different stakeholders involved in the project.

4 Claim-based view

The requirements engineering lifecycle for a refurbishment project should be influenced by the objectives and goals that are to be achieved. Stating clear goals for the results of the requirements lifecycle gives the freedom to develop the process in different ways and accommodating different techniques, but keeping in mind what is ultimately to be achieved.

If a goal-based view is adopted, it is necessary to provide evidence that the goals are achieved. Goals become claims and the Claim-Argument-Evidence framework can be developed to show how they are

progressively satisfied: explicit *claims* are stated, convincing *arguments* to justify the claims are met are presented, and adequate *evidence* to support the arguments is described. So we have

- goals used to design the requirements process
- claims to be made about the requirements to support the safety justification

This Claim-Argument-Evidence approach [11][12] provides a clear link with the WP1 justification framework. The Cemsis safety justification framework is based on the notion of "claim" and provides a structure to justify these claims. Claims correspond to system dependability properties and are inferred from subclaims at different levels: subclaims on the system requirements properties, on the architecture, on the design and on the operation. Subclaims at the system requirement level are those to be considered here by the requirement elicitation process.

The Claim-Argument-Evidence framework simultaneously provides guidance for the requirements process and a justification for its completeness. The claim-based view is provided graphically in Appendix C. In this structure, the top-level claim is divided into three overall objectives (sub-claims), similar to what is done in [13]:

- **Requirements integrity.** To ensure that the arguments and evidence are available to show that the requirements are derived from relevant sources and traceable both to the sources and to design and implementation. Requirements integrity can be divided into *configuration consistency* and *requirements traceability:*
    - **Configuration consistency.** To ensure that the arguments and evidence are at all times derived from: a known set of supporting documentation and a known set of software products and descriptions that have been used in the production of that version. Examples of items under configuration consistency include: plant description, drawings, safety requirements, design rationale for safety system, user manuals and operating instructions and results of hazard analysis.
    - **Requirements traceability.** To ensure that arguments and evidence are available which show that all requirements (and in particular safety requirements) can be traced to their source (in particular the plant safety analysis), to the design, are satisfied in the implementation of the software, and that other functions implemented in the software do not adversely affect safety.
- **Requirements validity.** To ensure that arguments and evidence are available which show that the requirements correctly state what is necessary and sufficient to achieve adequate safety and the desired functionality, in the system context.
- **Requirements satisfaction.** To ensure that arguments and evidence are available, in which there is sufficient confidence, which show that the SIS satisfies its safety requirements.

As discussed in Section 2, for refurbishment projects, we divide the requirements process and specification in two parts: the existing requirements and the new requirements. Similarly, the goal for the validity of the requirements is divided in two parts: validity of existing requirements (including in the *existing requirements specification*) and validity of new requirements.

Claims for the existing requirements specification are listed below and further developed in Section 4.3.1.

- Completeness. The specification should be a complete description of the current SIS, system environment, plant process model, plant safety analysis, plant and SIS performance reports and plant design basis.
- Precision. The specification should be precise and non-ambiguous. If needed, existing documents should be modified and clarification provided on syntax and semantics, possibly developing a glossary of terms.

- Currency with respect to the real plant and systems. Existing documents should be assessed and updated if necessary.
- Consistency: absence of contradiction.

Claims for the new requirements specifications are listed below (for both the user *requirements document for SIS* and the *SIS requirements specification*). They correspond to properties of the description of the SIS, as described in D3.2/3 and are further discussed in Section 4.3.2.

- Completeness with respect to the features of SIS that are significant for safety
- Accuracy, so as to reduce the uncertainty margins between the real features of the SIS and the corresponding description of values acceptable in the safety justification
- Precision (no ambiguity), so as to preclude any diverging interpretations by intended readers
- Consistency
- Correctness with respect to the real features of the SIS, i.e. the description corresponds to its how the SIS behaves.
- Clarity for intended readers – usability. Usability suitable for the SIS engineering process; clarity for intended users. Understandable for those who need to read the document. Can it be used to produce another artefact? Accounts for the people who are going to use it. Suitability for the engineering process.
- Relevance of extra functions
- Feasibility of safety justification

Since user requirements specifications vary widely in terms of detail and levels of completeness among the Cemsis project members, the properties of the requirements specification should be defined with respect to the agreed requirement specification content.

Note that validation of the top level System Requirements, which include the safety requirements, is only part of the demonstration of system safety (safety case). In particular, the methods one can use to validate top level system requirements (simulation, modelling, prototyping, experience) are methods that produce evidence to support arguments and subclaims at the plant–system interface level in the safety justification framework.

The three objectives are discussed in the following subsections.

4.1 Requirements integrity—Configuration consistency

The objective of configuration consistency is to ensure that the arguments and evidence are at all times derived from:

- a known set of supporting documents
- a known set of software products and descriptions that have been used in the production of that version

Candidates for configuration consistency typically include but may not be limited to:

1. Plant description.
2. Drawings (logic diagrams, interface specifications).
3. Safety requirements (traceable to plant safety analysis).
4. Design rationale for safety system.
5. Information on logic systems.

6. All user manuals and other operating instructions.
7. Intermediate design descriptions, either in natural language or formal or semi-formal notations.
8. The results of hazard analysis undertaken on the system and software.
9. Requirements' traceability records (where these are kept separately from the source code).

Evidence shall be available to show that:

1. All items under configuration are unambiguously and consistently identified.
2. Any tools used to support configuration consistency do not corrupt the configuration consistency structures.
3. Any tools used to construct or maintain configuration consistency, have been verified and validated to a level appropriate for the system.

Evidence, obtained from a change-control and configuration-management process, shall be available which shows that:

1. A detailed change-control and configuration-management process has been specified and is sufficient to meet the configuration-consistency objective described above.
2. The process has been adhered to and has been effective.

Configuration consistency needs to be taken into account in the requirements management, as described in Section 2.7.

4.2 Requirements integrity—Requirements traceability

The objective of requirements traceability is to ensure that arguments and evidence are available which show that all requirements (and in particular safety requirements) can be traced to their source, to the design, are satisfied in the implementation of the software, and that other functions implemented in the software do not adversely affect safety.

To give confidence that the traceability records are correct and complete, arguments and backing evidence shall be available to demonstrate that:

1. Traceability encompasses all pre-existing documentation related to the system being replaced/modernised.
2. Requirements are traced to its source and through to design and implementation.
3. Any tools used to support traceability do not corrupt the traceability structures.
4. Procedures or tools have been used to ensure that any loss of traceability or incorrect traceability is detected and corrected.
5. Any tools used to manage and maintain traceability, have been verified and validated.

Requirements traceability is part of the activities addressed by requirements management, as described in Section 2.7.

The requirements traceability can be decomposed in three main parts for each of which evidence needs to be available.

4.2.1 Hazard traceability

All hazards identified at each level in the design or in the software implementation are traceable to a defence (i.e. to a safety requirement for software, hardware or operation) or to a justification that no defence is necessary.

4.2.2 Traceability of requirements to its source

Every requirement is traceable to its source. The source might be among the following:

- plant hazard analysis
- requirement(s) on the existing system
- source of new requirement (as listed in the discovery of new requirements Section 2.5.1])

4.2.3 Traceability through design and code

Each requirement introduced at each level in the design can be traced to the software design and the source code.

4.3 Requirements validity

The objective is to ensure that arguments and evidence are available that shows that the requirements correctly state what is necessary and sufficient to achieve adequate safety and the desired functionality, in the system context.

The requirements validity can be divided into three parts:

- Requirements specification is valid, which refers to the requirements of the existing SIS.
- Requirements document for SIS is valid, which is the document prepared by the utility to be used by the suppliers as the basis for bidding. This document(s) includes the existing requirements and some of the new requirements required and constraints, and it will be revised and extended by the supplier.
- SIS requirements specification is valid, which is the final requirements document agreed by utility and supplier and on which the development will be based.

The validity objectives for these three specifications are discussed in the next sections. Section 4.3.1 discusses the validity of requirements specification, i.e. the validity of the existing requirements.

Section 4.3.2 discusses the validity of both the *requirements document for SIS* and *the SIS requirements specification*, i.e. the documents that include the new requirements. The validity objectives of these two only differ in the scope of completeness: while the latter needs to be as complete as possible, as it will be the basis for design and implementation, the former will only include what is needed for the supplier to develop the complete specification.

4.3.1 Validity of existing requirements—Requirements specification

1. **Currency.** The requirements should be current with respect to the plant and other relevant systems. Existing documentation should be assessed and updated if necessary.
2. **Consistency.** Absence of contradiction, assessment of consistency.
3. **Correctness.** Assess the system according to whether it is doing what it should in terms of safety, operation and non-functional aspects. Assess the documentation for correctness.
4. **Precision.** The specification should be precise and non-ambiguous. If needed, existing documents should be modified and clarification provided on syntax and semantics, possibly developing a glossary of terms.
5. **Completeness.** The specification should be a complete description of the current SIS, system environment, plant process model, plant safety analysis, plant and SIS performance results. In order to ensure completeness it is needed to:
   - assess documents with respect to gaps: Do the documents cover all the required aspects? Are there any undocumented aspects of the real system?

- all items should be considered (see Section 4.3.3)
- identify documents constituting the Plant Design Basis
- review all relevant documents

4.3.2 Validity of new requirements

1. **Completeness.** Completeness means that the specification should include all the information necessary for the user to carry out the next process step, and consequently is dependent upon the author's intent. Depending on whether we are considering the *requirement document for SIS* or the *SIS requirements specification,* the author's intent is reflected in the scope of the completeness assessment. While the latter specification needs to include all the information needed for design and implementation of the system, the scope of the former should be agreed with the suppliers tendering to develop the system.
2. **Feasibility.** Assess the feasibility of justifying the safety of the system, taking into consideration the system constraints.
3. **Consistency.** Absence of contradiction, assessment of consistency.
4. **Precision.** The specification should be precise and non-ambiguous. If needed, existing documents should be modified and clarification provided on syntax and semantics, possibly developing a glossary of terms. The safety requirements should be specified in sufficient detail and clarity to allow the design and implementation to achieve the required level of safety.
5. **Relevance of extra function.** The extra function(s) should be relevant to the intended application.
6. **Usability for intended users.** The requirements specification should be usable and understandable by the intended users.

4.3.3 Items for consideration in the requirements specification

The following items shall be included in the requirements specification:

- Boundaries: a precise definition of system boundaries.
- Functional properties: the primary functional behaviour of the SIS.
- Timing properties: the time allowed for the software to respond to given inputs or to periodic events, and/or the performance of the software in terms of transactions or messages handled per unit of time.
- Robustness: the behaviour of the software in the event of failures including failure of the interface with the plant, potential incorrect outputs.
- Accuracy: the required precision of the computed results

4.4 Requirements satisfaction

The objective is to ensure that arguments and evidence are available, in which there is sufficient confidence, to show that the SIS satisfies its safety requirements.

This is not developed further in this work, but it is an important part of WP1 and WP3.

5 Cost minimisation

The Cemsis approach to Requirements Capture for Refurbishment should help to control the costs of the overall refurbishment in variety of ways:

- By identifying requirements errors early in the lifecycle. Requirements errors found later in the lifecycle are considerably more costly to fix.

- By allowing changes to the existing system to be minimised, through integrating new requirements if desired.
- By using an incremental requirements process, where existing documents are reused, updated and extended as necessary.
- By correctly and completely documenting requirements, the next refurbishment would be based on complete specifications that would not have to be redone.

6 Conclusions

This document summarises the Cemsis approach to Requirements Capture for Refurbishment. It is the final deliverable of WP2: the Requirements Engineering For Refurbishment Best Practice Guide (D2.3). It aims to provide practical assistance for establishing the safety requirements for SIS refurbishment and it provides guidance on eliciting, analysing and documenting the safety requirements for a refurbishment project. It presents a simple requirements engineering lifecycle for a refurbishment project and it provides practical general guidance covering the lifecycle phases of the requirements engineering process. It also contains general principles and goals that should be considered during the requirements lifecycle and links those goals to specific phases of the requirements engineering process.

7 References

[1] PJ Caspall-Askew. D2.1: Review of tools and techniques for requirements capture and analysis. Cemsis document Wp2_BNFL008 v 0.1, February 2002.

[2] Sofia Guerra, PJ Caspall-Askew, RE Bloomfield, PG Bishop. D2.2: Requirements Process for Refurbishment: overall approach and rationale. Cemsis document wp2_ade009_v10, September 2002.

[3] PG Bishop and M.J.P. van der Meulen. Public domain case study. Cemsis document wp5_ade039.

[4] S. Robertson and James Robertson. Mastering the Requirements Process. Addison-Wesley, 1999.

[5] A. Finkelstein and I. Sommerville. The Viewpoints FAQ. Software Engineering Journal, 11, 1, 1996.

[6] Richard Harwell. What is a requirement. In Richard H. Thayer and Merlin Dorfman, editors, Software Requirements Engineering. IEEE, Los Alamitos, California, second edition, 1997.

[7] Michael Jackson. Software Requirements and Specifications: a lexicon of practice, principles and prejudices. Addison-Wesley, 1998.

[8] Ian Sommerville and Pete Sawyer. Requirements Engineering: a good practice guide. Wiley, 1997.

[9] Derek Fowler, Peter Cole, and Mike Wise. A Goal-Orientated, Evidenced-Based Approach to Software Assurance in Critical-Systems Applications.

[10] Pentti Haapanen, Jukka Korhonen and Urho Pulkkinen. Licensing Process for Safety Critical Software Based Systems.

[11] PG Bishop, RE Bloomfield, CCM Jones. Adelard Safety Case Development Manual, Adelard, ISBN 0 9533771 0 5, 1998

[12] J Penny, A Eaton, PG Bishop, RE Bloomfield. The Practicalities of Goal-Based Safety Regulation, Proc. Ninth Safety-critical Systems Symposium (SSS 01), Bristol, UK, 6-8 Feb, pp. 35-48, New York: Springer, ISBN: 1-85233-411-8, 2001.

[13] CAP670 Air Traffic Services Safety Requirements, Amendment 6, document SW01 "Regulatory Objectives for Software Safety Assurance in ATS Equipment", December 2002

Appendix A Classical requirements engineering process

"If it mandates that something must be accomplished, transformed, produced, or provided, it is a requirement". [6]

Requirements are capabilities needed by a user to solve a problem, or capabilities that must be met by a system to satisfy a contract, standard or specification. They establish an understanding of the user's needs, and also provide a yardstick against which implementation success is measured.

The classical requirements lifecycle has the following phases:

1. Project viability study.
2. Requirements discovery.
3. Requirements analysis and negotiation.
4. Requirements definition.
5. Requirements validation.

These phases are supported throughout by requirements management. Figure A1 shows the "classical" requirements engineering process. In the following subsections we discuss each of these phases in more detail. Afterwards we discuss how this process should be modified for refurbishment projects, and what each of the phases involves for a Cemsis project.

A.1 Project viability study

The first phase of the requirements engineering process aims at deciding whether the project is viable, and whether its cost makes it worthwhile. In summary, the decision between yes and no.

In order to answer this question, the stakeholders partake of the decision-making process should be identified. Stakeholders are the individuals, or organisations, who stand to gain or lose from the success or failure of the system. The main stakeholders, i.e. the users, customer and client are typically involved.

Other stakeholders that are typically relevant to this decision include managers (concerned with the business advantages of the project), and developers (who have a say in the feasibility of the project).

Further stakeholders are identified as the initial (core) stakeholders study the context of the work. The involvement of the complete set of stakeholders is prerequisite to the acquisition of a complete set of requirements (and viewpoints) on the system (see Section 2.1.2).

They have to go through the following points to reach the decision:

- Purpose: what is the purpose of the project?
- Advantage: what business advantages does it provide?
- Reasonable: is the project effort great than the advantage?
- Feasibility: can the product achieve the advantage?
- Achievable: does the organisation have the skills to build the product?
- Safety: can the product satisfy the required safety level?
- Technology: are there technology available for the development of the product?

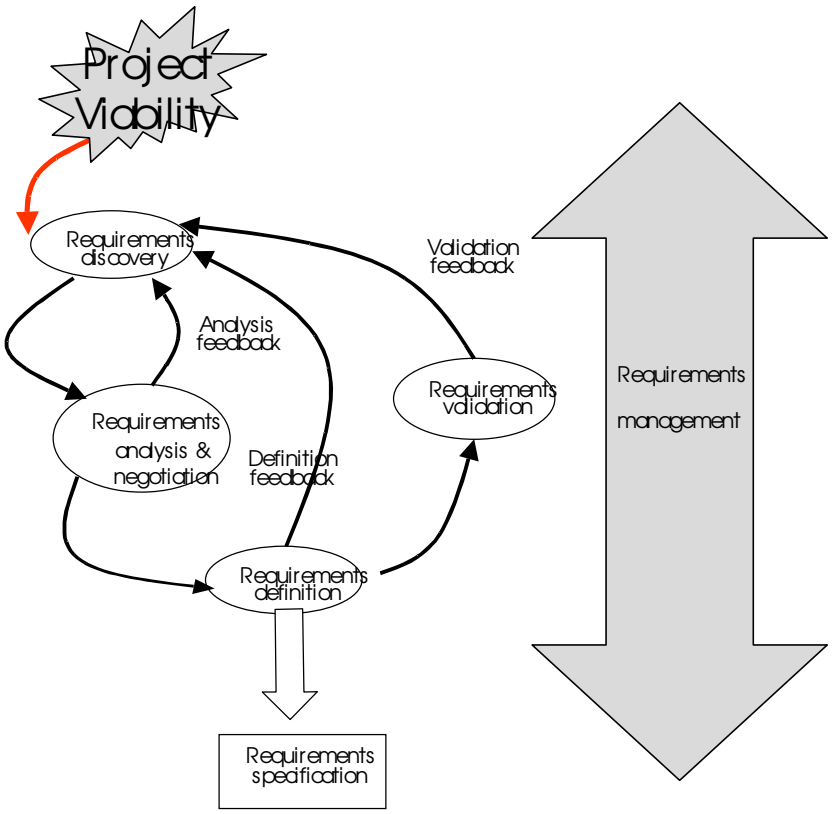


Figure A1: Classical requirements engineering process

It is also part of the project viability study to define the scope of the work, and consequently the boundaries of the product being developed. In [7] Michael Jackson distinguishes the "machine", which is the product being developed, from the "application domain". Application domain is not a generic class of applications (as the term is used in the phrase *banking domain*), but is specific to the problem in hand, and it is where the requirements are to be found. The definition of the application domain is the first step towards recognising the shape of the problem and its boundaries. A problem is characterised not only by the nature of the product built but also by the structure of the application domain. The definition of the system boundaries also helps to estimate how much work is needed before determining the requirements of the product, what is inside the requirements and what can be left outside. The budget available and time allowed are also constraints for the definition of the scope.

Constraints are gathered and recorded at this phase. Constraints are global requirements (see Appendix B for further discussion). The main stakeholders are aware of the system constraints, but they need to be documents at this phase.

In this phase it is important to have a preliminary cost estimation of the project. For the cost estimation it is necessary to estimate the effort likely to be required. There may be several different options for performing effort estimation at this point. Estimates can be based on function point analysis, use a model developed of the application domain or any other relevant technique or method. The important point is to measure the work so that the estimates are based on evidence rather than just being guesswork. The ultimate activity of the first phase of the project is to decide whether to go ahead with the project, and this decision is based on the cost estimation.

In summary, the main activities of the project viability study are:

- Stakeholders identification.
- Identification of the boundaries and scope of the work: what should be included and is outside the scope.
- Identification of constraints.
- Purpose identification: clearly statement of the aims of the project/refurbishment
- Preliminary cost estimation.
- Stakeholders decision whether the project is worthwhile and viable.

A.2 Requirements discovery

Requirements discovery or elicitation is the process through which the stakeholders of a software system discover, review, articulate and understand users' needs, the domain knowledge and the constraints on the system and the development activity. The process of elicitation consists of interaction between the different participants of the system and recording of the information in such a way that a proposed system can be specified.

The requirements' gathering is a collaborative work among the stakeholders. Elicitation involves an understanding of psychological and sociological methods as well as system development and the application domain of the system to be built. In fact, requirements emerge from the social interactions between the system users and the analysts. Requirements acquisition is a strongly communicative, iterative and creative design activity. It might also involve the gathering and analysis of previous documentation, if a similar system to the one being developed is in place.

Requirements discovery use some of the outputs of the viability study:

- The project viability study has determined the context and boundaries of the work, the purpose and constraints that apply to any solution. These guide the elicitation (and analysis) of further requirements.
- It has determined the stakeholders involved in the project. These stakeholders will be interviewed to get an understanding of the project.

Requirements discovery should also use the results of the hazard and risk analysis. Hazard and risk analysis identifies possible hazards and risks. Safety requirements to avoid or minimise their consequences arise from these analyses.

A.3 Requirements analysis and negotiation

The objective of the analysis and negotiation is to establish an agreed set of (consistent and complete) requirements. The requirements elicited from the stakeholders need to be analysed and negotiated to arrive at a definition of the system requirements that will be included in the requirements documents. The analysis and negotiation involve several different activities whose aim is to discover problems with the requirements and achieve an agreement among the stakeholders and their different viewpoints.

Requirements elicitation and analysis are intertwining processes: during the elicitation it is necessary to carry out some analysis which, in turn, informs the elicitation process. While requirements are being elicited, some problems might be detected and analysis is immediately carried out. Although we treat elicitation and analysis separately, these two activities should be seen as depending on each other.

Some authors consider validation as part of analysis. Here, however, we distinguish analysis from validation as the former is performed with an incomplete set of requirements, while the latter has an agreed set of requirements as a starting point.

Elicitation and analysis result in a set of requirements that are subject to negotiation among the stakeholders. The multiplicity of stakeholders involved will result in different views on the system and often in conflicts. Hence, it is necessary to find and resolve these conflicts, and to negotiate in order to reach a compromise. This negotiation should consider cost analysis and safety properties of the different alternatives. Negotiation also involves prioritisation of the requirements, as in any set of requirements some are more important than others. Requirements are discussed and a compromise should be agreed. In many cases, however, some of these problems are not found until later stages of the development, when some modelling and design have been done. We should therefore not consider these activities as an ordered list, but they are intertwined and depend upon other activities of the requirements process.

A.4 Requirements definition

The analysis and negotiation result in a set of goals, functions and constraints that are collected into the specification of software system behaviour. The development of a document (or set of documents) that clearly and precisely record each of the requirements of software system is the requirements definition phase of the requirements process.

Compiling the requirements is not really separate from the other activities that surround it. It takes places while requirements are being discovered and analysed. Writing a specification is more like "building" a specification – a specification is assembled during the lifecycle and incrementally developed rather than written all at once.

When requirements are being gathered, sometimes they are not fully formed or complete. On the other hand, a requirements specification (the document(s) that describe the requirements) is the basis for the development. Therefore, the initial description of the requirements might have to be expanded and refined in order to produce a complete, accurate, with sufficient detail requirements specification that can be understood by all the parties involved. In this way, it makes sense to talk about "requirements definition task" to emphasise the main principles of documenting requirements.

System models are based on computational concepts such as objects or functions rather than application domain concepts (although application domain concepts might be used to describe interfaces). They are important bridges between the analysis and design processes. The use of documents and partial documents by different stakeholders may require a range of notations to be used.

A.5 Requirements validation

After the requirements specification has been written, validation is the process of determining whether the requirements define the system that the stakeholders want. This includes assuring the completeness, consistency and correctness of the set of requirements. If requirements validation is inadequate, errors will propagate throughout the lifecycle to system design and implementation. The costs of fixing them are higher if errors are found later in the lifecycle.

Requirements validation and requirements analysis have much in common. Here, we distinguish these two activities, as it was done in [8]. Analysis does not assume that the requirements are completely and correctly documented, and the language used might be informal or unstructured. Validation, however, checks the final draft of the document for completeness, ambiguity and correctness.

The problem with requirements validation is that there is nothing concrete or absolute to validate against. While software design and implementation can be verified against its specification, there is no way to demonstrate that the requirements specification is indisputably

correct. The validation will only increase the confidence that the specification describes the system as the stakeholders want it developed, and hence decrease the likelihood of finding requirement errors further down the lifecycle.

Demonstrating that a set of requirements meets the stakeholders' needs is often difficult if only an abstract representation of the requirements is used. Common techniques to support validation include prototyping (executable model of the requirements), simulation and development of (mathematical) models.

A.6 Requirements management

Requirements management consists of the planning and controlling of the requirements elicitation, analysis, definition and validation activities. It concerns the control of requirements information and, in particular, its integrity during the life of the system with respect to changes of the system and its environment. It is a pre-requisite for quality-oriented development. It includes organisation, traceability, analysis and visualisation.

In fact, central to the success of any requirements management process are the management of the changes to the system requirements. The management of the changes to the system requirements includes control of the impact of changes on:

- other requirements and their documentation
- the design and implementation of the specification

The former is concerned with the collection, verification and assessment of changes; the later with the consequences of the changes to the system as a whole.

In managing requirements we need to keep *requirements traceability.* Requirements traceability assists the understanding of the relationships between requirements, its sources, design and implementation of a system. A requirement is traceable if we can identify the parts of the system that exist because of that requirement, and for each part of the system we can identify the requirement that originated it. Traceability management applies to the entire development lifecycle from project initiation through operation and maintenance.

Requirements traceability plays two important roles in requirements management, in respect of:

- *Requirements change*. When requirements change, it supports the identification of other requirements that might be affected by the change. In addition, it will identify the parts of the design and code that will be affected by the change (if the development has gone that far).
- *Verification and testing.* Design and implementation can be verified against the relevant requirements. In addition, coverage achieved by testing can only be defined and guaranteed if requirements are traceable to the products of the phase concerned.

Current techniques provide support to the development of a system so that it meets the stakeholders' needs by managing traceability using a combination of manual and automated assistance. An essential element of successful traceability management, provided by currently available tools, is the ability to provide links form requirements forward to design, code, testing and implementation, and backwards from any of these activities to requirements.

Typical traceability tools work by assigning and linking unique identifiers (usually manually). This information is subsequently supported by document managers, databases or CASE tools. By establishing a unique identification system and following this scheme throughout the lifecycle, it is possible to trace these specific entities both forward and backwards. This unique identity may be linked within and across documents, from previously existing documents (e.g. requirements sources) to new documents developed during the lifecycle of the project.

Appendix B Functional, non-functional requirements and constraints

B.1 Functional requirements

Functional requirements describe the functionality of the product: the behaviour and actions it has to carry out in order to fulfil its intention.

B.2 Non-functional requirements

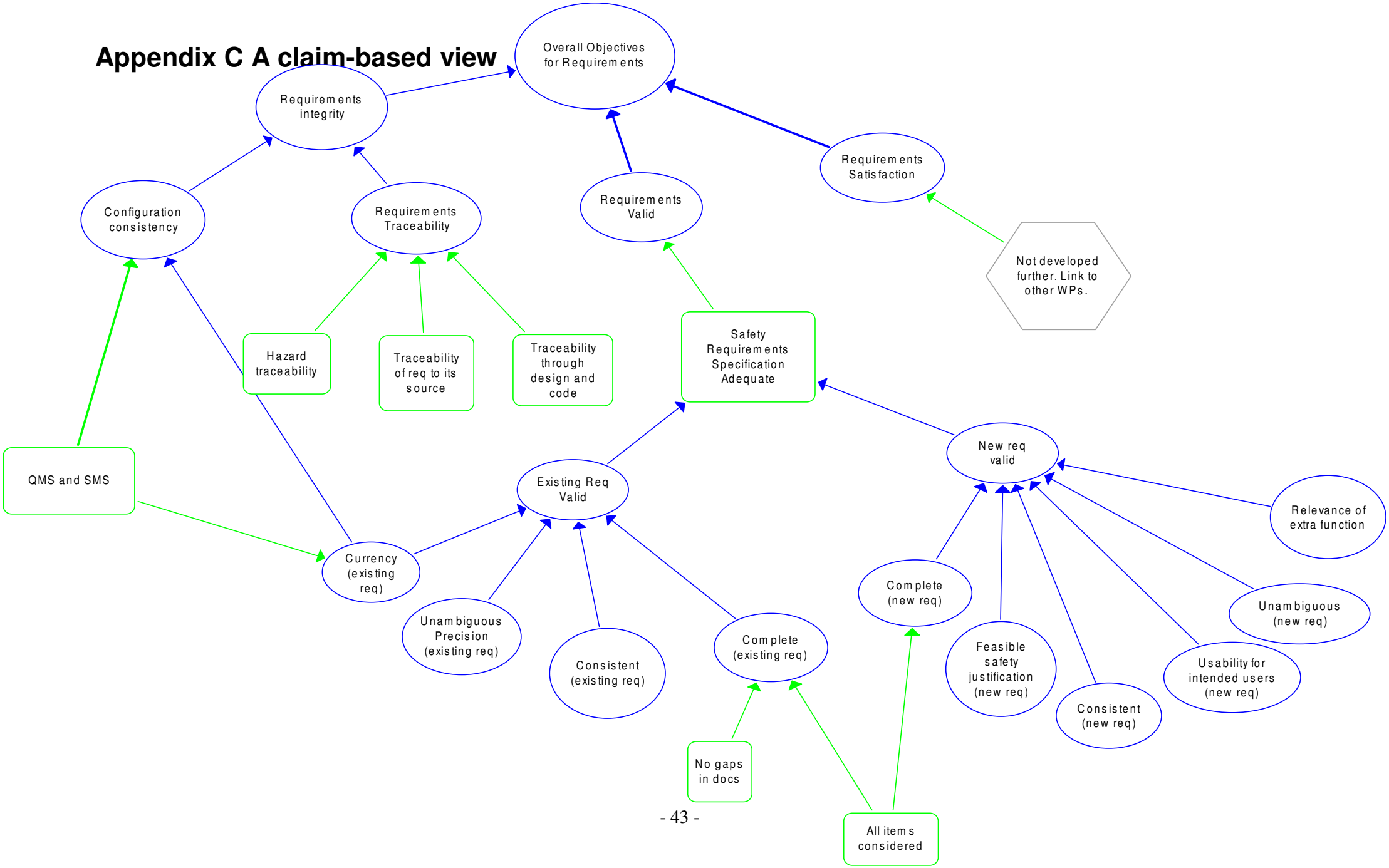
Non-functional requirements concern properties that the product must have. There are several classes of non-functional requirements. An important issue regarding non-functional requirements is that they should supply a benchmark against which their implementations can be evaluated.

- Usability: describe the appropriate levels of usability.
- Timing
- Robustness
- Reliability: reliability should be stated in probabilistic terms involving time (i.e. that a given failure rate must not be exceeded), and testing or field service experience is to be used to obtain direct evidence of requirements' satisfaction. The software must be observed to operate correctly over a time period that gives a statistical confidence of 95% that all required failure rates have been met.
- Accuracy: How close to the real value a measurement is.
- Maintainability: Requirements concerning the ability to maintain and update the system, possibly without taking it out of service.
- Security: Security requirements typically concern three types of constraints: confidentiality (data stored by the product is protected from unauthorised access and disclosure), integrity and availability (authorised users are not prevented from accessing the data).
- Legal.

B.3 Constraints

Constraints are global requirements. They apply to the entire system. They should be gathered at the early stages of the process, and they should be used to confirm the appropriateness of the other requirements. Examples of constraints are the platform where the product is going to run, standards it has to comply with, amount of money or time spent in the work or other applications it has to interface with.

Appendix C A claim-based view



Appendix D Example: Content of a user requirements specification

In this appendix we illustrate the content of a specimen user requirement specification requested by a supplier. This content was provided by Andreas Klein, from Framatome ANP NGLTK.

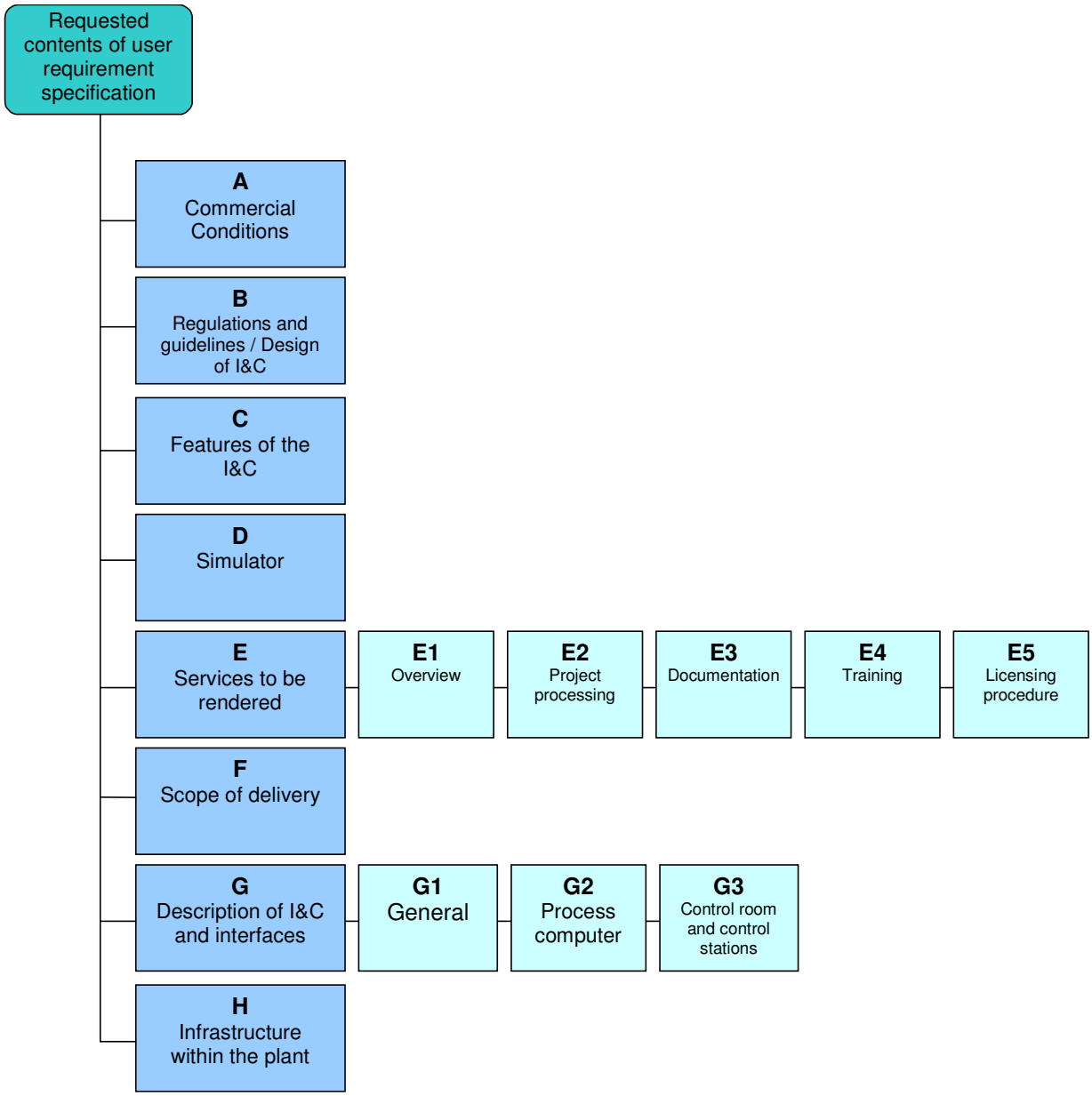


Figure E.1: Overview of requested contents of user requirement specification

Overview of necessary documentation
***M**andatory* : *Must be present in the user requirement specification*
***O**ptional* : Can *be part of the user requirement specification.*
If not, a proposal is made by the supplier

Themes	**M /O**	**Remarks**
A Commercial Conditions		
Basis of contract, deadline, award criteria (technology, delivery period, price,...), warranty, guarantees (reliability, schedule observance,...) liability	M	
Priority of documentation in case of an order, contract, invitation to tender, commenting etc.	O	
Description: how does the project proceed from the customer's viewpoint up to awarding of the contract?	M	
Project organisation on the customer's part after fina lisation of the contract	O	
Interaction between customer and supplier during project processing, milestones, splitting of work, services & supplies	O	

B Regulations and guidelines / Design of I&C		
List of applicable regulations, guidelines, customer instructions Differentiate between: - actual status - demand to be fulfilled - partial fulfilment - guideline to be followed - do customer regulations take new technology into account?	M O	
Safety concept for I&C and the plant: - probabilistic targets for I&C - presumption of failures, requirements to cope with combination of failures (e.g. single failure & maintenance), - supposed influences form outside and inside, - Defence-In-Depth concept, - demands for redundancy and diversity	 O M M O M M M	
Classification of I&C functions and equipment classification concept;	M	
Control characteristics required control accuracy	O	

C Features of the I&C		
Requirements regarding functionality:		

Themes	**M /O**	**Remarks**
Signal types Interfaces Operating/monitoring functions	M M O	
Requirements for self-monitoring malfunction diagnosis behaviour in case of malfunction electrical isolation	O	
Requirements regarding testing and calibrating as well as recurrent inspections	O	
Structural features of the I&C process-oriented structure central / de-central installation independence of redundant I&C processing, assignment to electrical systems and equipment under control	M	
Global features, such as uniformity/continuity of operating + monitoring structure possibility of interface with existing 3rd-party systems maintainability features demands on configuring tools and documentation concept data management and archiving	O	
Marking of cabinets, cables and other equipment marking system, especially signal identification: format, number of digits	O	
Specify lifetime of I&C	M	
Ambient conditions: temperature, humidity; EMC requirements	M	
Demand for reserves: for modules, in racks, in cabinets, in power supply, processor load, communication load...	O	
Reliability features: availability; defined malfunction repair times	M	
Access control: for control room equipment service and engineering equipment cabinets	O	
Required timed resolution of analogue / binary / serial data (third party systems) accuracy of signal acquisition / processing	O	

Themes	**M /O**	**Remarks**
Required response times for: drive control control system reactor protection, ESFAS malfunction recording special measuring systems ...	M	
Response times for: man-machine interface updating cycle display selection time reaction times for operator input	O	
Seismic requirements: basis for design, spectra, affected equipment	M	

D Simulator		
Description of training simulator: original manufacturer scope of simulation computer technology, cycle time software environment, models, control systems serial and parallel interfaces type of control room boards, type of control room interfaces free capacity	M	
Description of interfaces between I&C and process models in training simulator, interfaces to simulator control system, to MMI (reflected in requirements for services and services from supplier)	M	
Scope of modernisation of simulator, if necessary (as for plant? subset? process models new / to be modified)	M	
Requirements regarding: simulation accuracy and scope of I&C models (e.g. fast signals not necessary); requirements of I&C failure models	O	
Space requirements for: instructor simulator control system (new, or use existing	O	

Themes	M /O	Remarks
one?) scope of control commands standard set quantity of ICs, of Snapshots.		
Description of current development system, change requests	O	
Requirements of the I&C from the simulator viewpoint, e.g. compatibility with the process models; generability of Code for I&C models from the engineering system of the I&C; effectiveness of I&C data banks for the simulator	O	
Necessary lead time for retrofitting of the simulator before (retrofitting) the plant	O	
Available time slot for simulator retrofitting	M	
Customer's/operator's own scope when retrofitting the simulator, e.g. project coordination adjustment of the models; installation of the simulator control **...**	M	

E Services to be rendered		
E.1 Overview		
Services to be rendered: Engineering; Procurement; Installation and assembly; Dismantling (degree of dismantling); Modification/adjustment of the remaining equipment; Modification/adjustment of power supply, ventilation, structure1; Commissioning; Training; Documentation; Spare parts, spare parts strategy	M	Only for I&C retrofit
Which services will the customer provide himself? Dismantling? Cable route planning...	M	

E.2 Project processing		
Specific requirements for project processing / project organisation requirements for project phase model desired steps/structure requirements to involve the customer in review steps, approvals processing time by customer for review steps	O	
Requirements for supplier project organisation	O	

Themes	M /O	Remarks
Requirements for quality assurance and verification/validation	O	
Stipulated frame schedule for the project; Milestones	M	
Requirements for schedule and resource follow-up in the project	O	
Planned time slot for introduction of the I&C For I&C retrofit plant outage / I&C exchange specification of times available for conversion operations	M M	
Problem-Management: regulations for handling additional costs, unexpected malfunctions and problems in project running	O	

E.3 Documentation		
Model documentation (drive, measurement, operating and maintenance instructions...) Description of availability and scope of existing documentation: Paper? Computer-aided? e.g. electronic data model for process computer? Conformity with actual status in the plant? Model documentation (drive, measurement, operating and maintenance instructions...) Documentation of function of process computer and functions realised with this	M M	 For I&C retrofit
Modification of the documentation of remaining systems during modernisation red entry by supplier? complete renewal/revision? procedure? (acquisition, copying, transfer, revision)	M	For I&C retrofit, similar requirements for interface documentation between I&C of different suppliers
Are certain tools stipulated for documentation?	O	
Preparation/modification/adaptation of operating and inspection manuals, what should be done by the supplier, what will be done by the customer himself?	O	
Property status of power plant documentation; Are patent infringements to be expected when duplicating functions?	M	

E.4 Training		
Concept for personnel training: Content of training which personnel categories: operators, I&C engineers, engineering personnel, trainers, simulator staff... how many persons from each category Training Schedule Training location and language	O	
Requirements for training documentation	O	

Themes	M /O	Remarks
E.5 Licensing procedure		
Description of licensing procedure Desired form and scope of support for licensing procedure	O	

F Scope of delivery		
Exact definition of scope For I&C modernisation listing of I&C equipment to be replaced (cabinets, measuring transducers, cables, local control centres, emergency control centre, cable networks to be renewed...) Listing of remaining equipment which must be connected with the new I&C with description of the interfaces Task definition of I&C to be replaced (which must take over the new I&C): functional description, set standard quantity of transducers, drives, etc., specifying affected process systems and I&C equipment (i.e. which drive will be controlled from which cabinet, which transducer is supplied from which cabinet); description of internal interfaces between sub-systems (particularly where intersections are made between replacement packages) Description of where function relocation is permissible/desired: taking over local control stations in the control room, Description of the installation sites of the I&C to be replaced, e.g. by means of structural plans	M M M C C	
Desired extension of functions of the I&C, e.g. additional measurements, additional displays, additional /optimised interlocks, modification and optimisation of controls additional /optimised automatic controls additional process computer functions	O	For I&C retrofit
Simultaneous process engineering modifications? (new/modified systems?)	O	For I&C retrofit
Description of desired replacement strategy: Formation of work packages	M	For I&C retrofit
Requirements for parallel operation of old/new I&C (e.g. plant computer)	O	For I&C retrofit
Other modifications together with I&C modernisation in separate rooms, fire protection ventilation other structural measures (e.g. core drilling, static calculations, new cable routes, removal of problematic materials) extensions of power supply	O	For I&C retrofit
Earthing, lightning protection	M	
Electrical isolation, EMC	M	
Interfaces to I&C of other suppliers (e.g. to operation management system, to data recording computer for nuclear	O	

Themes	M /O	Remarks
parameters etc.); type of protocol redundancy requirements type and volume of data to be transferred time response (when, how much) independence requirements (e.g. fire wall)		
Control room philosophy: Description of requirements for control room and local control stations: technology, monitor or conventional personnel and staffing duties of work places remaining equipment to be taken into consideration: communication equipment, fire alarm panel, video monitoring	M	
Task definition of standby control station Task definition of local control stations	O	
Configuration of work places specific fixture wishes, e.g. large-scale screen, video image integration requirements for control desk assignments, image configuration, signalling concept... scope of conventional desk-top panels internal architectural configuration Exclusions (customer contributions)	O	
Requirements regarding signalling concept	O	

Installation / dismantling Dismantling/removal of old cabling1) Cable channels Laying of cables Supporting frame Status of remaining systems/equipment	O	

G Description of I&C and interfaces		Section G is related to I&C retrofit
G.1 General		
Description of present I&C structure	M	
Spatial layout of the existing I&C; Indication of rooms additionally available and free places (also for temporary use), cable routes; status of old cabling, bushings	M	
Characterisation of interfaces: Signal convention analogue and binary; rough standard set quantities of actuators and transducers (how many of which, per train, per process engineering /electrical engineering system, per L&C cabinet) transducer types interfaces to switch gears and actuators, to remaining	M	

Themes	M /O	Remarks
analogue systems (even if only temporary): signal convention, frequency, particular forms of impulse, time response) interfaces to remaining systems or systems from other suppliers to be connected: bus type, protocols, signal scope, required time response prescribed connection technique plant organisation: sub-distribution boards, yes/no		
Requirements for electrical isolation	M	
Power supply: present concept for I&C power supply features, design, (capacity; available reserve ranch feeders), rooms	M	
Degree of automation: Number of groups, sub-groups, partial controls, control circuits per system, per cabinet, average number of steps per automatic system	O	
Earthing and lightning protection – present concept	M	
Power consumption of the systems (per room, better still per cabinet), and capacity of ventilation systems	O	

G.2 Process computer		
Description of the present computer system: hardware, acquisition level	M	
Description of process computer: General description	M	
Number of conditioned analogue/binary signals to the process computer, plant/curve/bar diagrams, protocols, calculations...	M	
Description of process computer functions: protocol types, images, trend curves, accuracy, signal concept...	O	
Installation location, space separation		
Plant-specific special functions (e.g. parameters, calculations, special functions, calculations, WKP, standard set quantity); Are there any specifications for OEM / are these available?	M	
Core-computer, manufacturer, remaining lifetime, how is it integrated into the present process computer?	M	
Present interface with external computers to the process computer (example, office network, expert systems)	M	

G.3 Control room and control stations		
Task definition and equipping of the control room and of the various control stations: Layout, personnel and staffing and tasks of the personnel of the in all operational modes Description of tasks, Rough standard set quantity	M	
Task definition of standby control centre, local control stations, of other rooms affected by the conversion with	M	

Themes	M /O	Remarks
operating or monitoring equipment (expert rooms, technical support centre...)		
Detailed allocation (indicators, desk-top panels, monitors, communication equipment) and safety engineering classification Relation with process engineering/electrical systems	O	

H Infrastructure within the plant		
Which infrastructure is available within the plant (offices, containers, scaffolding, crane equipment, other space available, ventilation, power...) Conditions of use for this Does the customer provide telephone/fax/e-mail?	M	
Waste treatment and disposal, transfer stations	M	
Requirements based on Work regulations, Work approval procedure, (electronic) approval procedure working hours regulations, legal regulations	O	

Appendix E Example of checklist in order to complete a URS

1. Identify interfaces between the system and the operating environment in order to analyse hazards of the interfaces.
2. Identify interfaces between the system and the operating environment in order to analyse system level hazards.
3. Identify interfaces between the system and the operating environment in order to analyse human-machine interfaces.
4. Identify interfaces between the system and the operating environment in order to analyse hardware-software interfaces.
5. Identify interactions of subsystems within the system in order to analyse propagation of failure modes (i.e., hazards).
6. Identify interactions of hardware subsystems within the system in order to analyse incompatibilities between or at interfaces.
7. Identify interfaces between disciplines in order to interpret disciplinary specific terminology between disciplines.
8. Identify interfaces between disciplines in order to co-ordinate interdisciplinary efforts to mitigate risk.
9. Identify human-machine interfaces within the system operating parameters in order to assure that system demands do not exceed human physical or cognitive limitations. (Have we uncovered a potential fault)
10. Identify hardware-software interfaces within the system to determine compatibility of software and hardware at the interfaces. (Have we uncovered a potential fault)
11. Assess the adequacy of information exchange at hardware-software interfaces within the system.
12. Identify human-hardware-software-environment interfaces within the system to analyse the effect that the software has on the system.
13. Identify applicability of hardware failures as they affect the system.
14. Identify the effect of the environment on the system.
15. Identify the importance of design element factors in terms of their effect on the system.
16. Identify software functions that are critical to the safe operation of the system.
17. Identify safety critical elements in the software specification.
18. Discern proper and improper software controls for hardware operation.
19. Develop criteria to ensure that human limits and boundaries of operation are not exceeded.
20. Identify relationships between human operators, procedures, machines, and the environment which ensured total system operational safety.
21. Define relationships between human operators, procedures, machines, and the environment to ensure total system operational safety.
22. Evaluate workplace design to ensure that operational flows and human requirements are compatible with safe operations.
23. Develop and evaluate procedures to ensure proper operational flows and correct operator reactions to anomalies.
24. Define proper safety program activities commensurate with the development life cycle of the system.
25. Define and co-ordinate the system safety program life cycle with the system engineering life cycle.
26. Identify the safety implications associated with each life cycle phase of the system.
27. Identify the effect of the life cycle environments on the safety of the system.
28. Identify the effect of the system's life cycle on the environment.

29. Evaluate procedures to ensure proper operational flows and correct operator reactions to anomalies.
30. Determine potential human errors and their effects on the system.

Appendix F Requirements and the safety lifecycle

The requirements process (and the full development lifecycle) may need to interface to the safety lifecycle from IEC 61508. In the previous sections, for each of the phases of the requirements lifecycle, we outlined what were the main points of the safety lifecycle to be taken into consideration. In this section we draft the connection points between the safety and the requirements lifecycle, and the safety lifecycle in the context of the full development process.

The IEC61508 safety lifecycle begins with the activities called *concept* and *overall scope definition*. These activities consist of the understanding of the system application domain and boundary definition. This corresponds to some of the specific phases the project viability study and the system requirements gathering involve.

Hazard and risk analysis identifies possible hazards and their causes. Requirements gathering should be informed by hazard analysis, and it should use its results to write requirements to avoid these hazards or minimise the consequences of the accidents. Requirements discovery includes the safety requirements gathering that result from hazard and risk analysis.

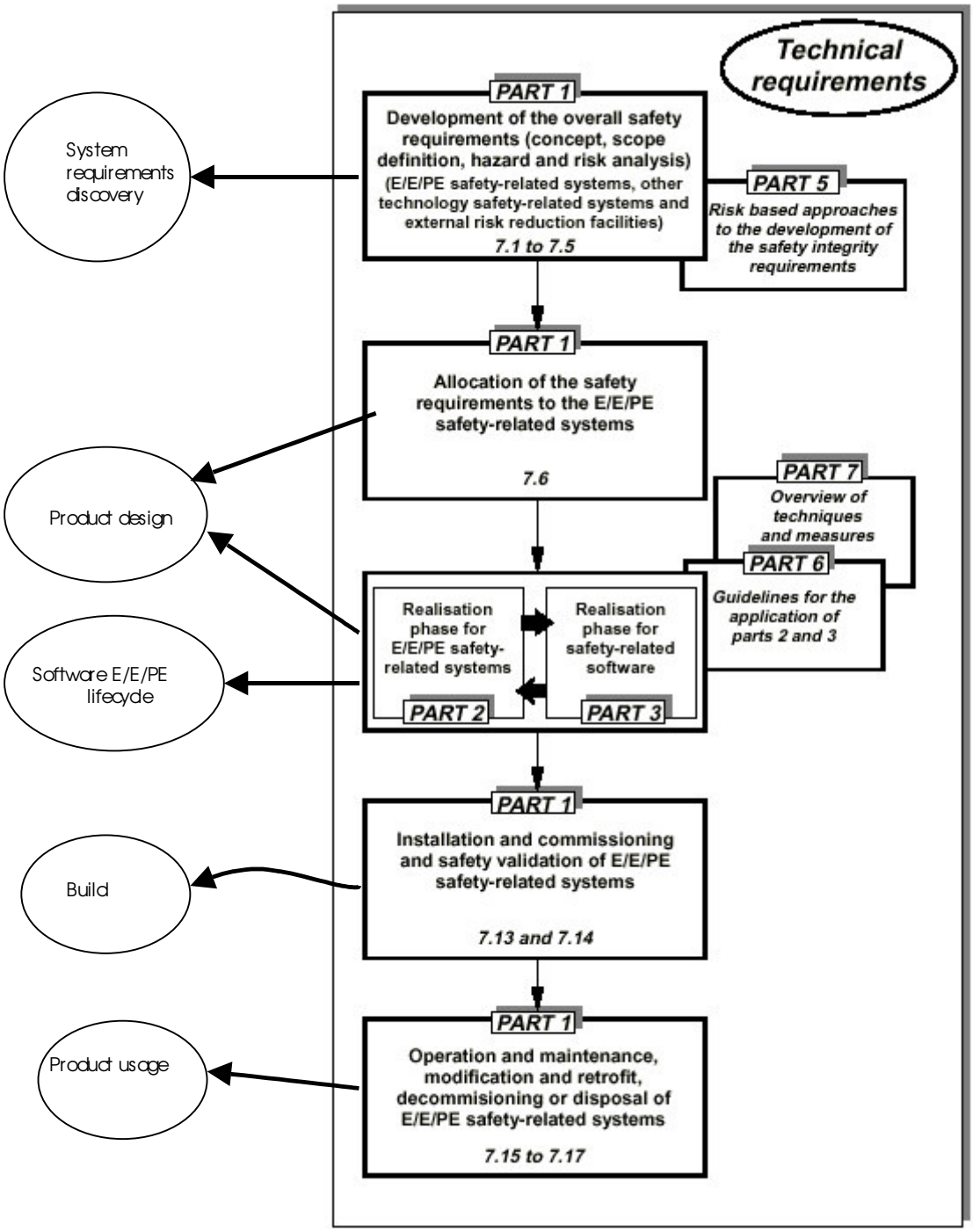
The results of hazard identification should also be used during the requirements negotiation and prioritisation of requirements.

A safety-case can be used to demonstrate the correspondence between safety requirements and functional requirements.

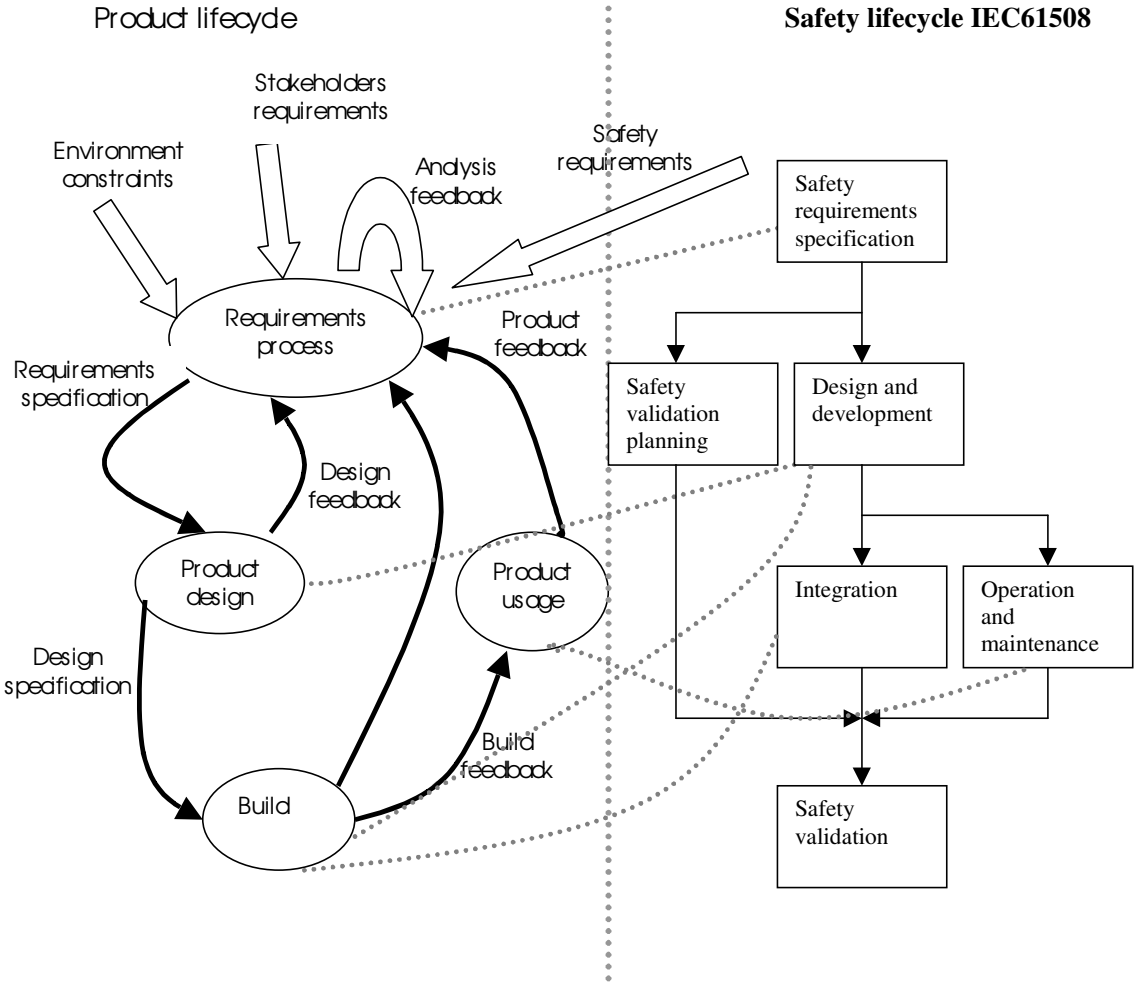
Requirements validation includes requirements safety validation.

The IEC61508 overall safety lifecycle and its relation with the requirements and process lifecycle are presented in the figure below.

As mentioned earlier, the first box of the safety lifecycle is subdivided in three activities, which are called concept, overall scope definition and hazard and risk analysis, which are part of the requirements discovery.



It is important to integrate the safety lifecycle in the product lifecycle. Each of the activities of the development process should include safety checks. The IEC61508 safety lifecycle for software and E/E/PES is easily integrated in the lifecycle proposed earlier.


Product lifecycle
Safety lifecycle IEC61508
Stakeholders requirements
Environment constraints
Analysis feedback
Safety requirements
Safety requirements specification
Requirements process
Product feedback
Requirements specification
Safety validation planning
Design and development
Design feedback
Product design
Product usage
Integration
Operation and maintenance
Design specification
Build feedback
Build
Safety validation