Deliverable D12.8

Guidance on Developing, Using and Modifying a Requirements Management System for Waste Management Programmes with their Different Systems (WMP-RMS)

May 2024

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement N°847593.

Document information

Project Acronym	**EURAD**
Project Title	**European Joint Programme on Radioactive Waste Management**
Project Type	**European Joint Programme (EJP)**
EC grant agreement No.	**847593**
Project starting / end date	**1 June 2019 – 30 May 2024**
Work Package No.	**12**
Work Package Title	**Guidance**
Work Package Acronym	**Guidance**
Deliverable No.	**D12.8**
Deliverable Title	**Guidance on Developing, Using and Modifying a Requirements Management System for Waste Management Programmes with their Different Systems**
Lead Beneficiary	**Andra**
Contractual Delivery Date	**30 May 2024**
Actual Delivery Date	**30 May 2024**
Type	**Report**
Dissemination level	**PU**
Authors	**Piet Zuidema (Zuidema Consult GmbH)**

To be cited as:

Zuidema P. (2024): Guidance on Developing, Using and Modifying a Requirements Management System for Waste Management Programmes with their Different Systems. Final version as of 30. May 2024 of deliverable D12.8 of the HORIZON 2020 project EURAD. EC Grant agreement no: 847593.

Disclaimer

Acknowledgement

This document is a deliverable of the European Joint Programme on Radioactive Waste Management (EURAD). EURAD has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 847593.

Status of deliverable		
	By	Date
Delivered (Lead Beneficiary)	Piet Zuidema	21 February 2024
Verified (WP Leader)	Jiri Faltejsek	25 February 2024
Reviewed (Reviewer)	Barbara Pastina	28 February 2024
Reviewed (Reviewer)	Tara Beattie	30 April 2024
Approved (PMO)	Paul Carbol	10 June 2024
Submitted to EC (Coordinator)	Andra	14 June 2024

Executive summary

Within Work Package 12 (Guidance) of EURAD (European Joint Programme on Radioactive Waste Management), activities aim to develop a comprehensive suite of instructional guidance documents that can be used by EU Member States with radioactive waste management programmes, regardless of their phase or level of advancement with implementation of geological disposal.

In the course of EURAD, based on a review made by Work Package 12, it was decided to develop guidance on requirements management, because requirements management is recognised to be a very important activity for implementing waste management programmes / systems. The work package board together with the editorial board agreed to develop such guidance in a process with active involvement of end-users through a number of workshops and a training event. With this process, three documents were developed:

- a guidance document for generic waste management systems (EURAD 2024a),
- a document describing in more detail the development of requirements for disposal systems with a discussion of the post-closure safety case and its interaction with requirements management (EURAD 2024b),
- a guidance document for specific waste management programmes and with their different system systems, taking the stepwise implementation of these systems into account (this document).

As all three documents are ‘stand-alone’ documents and each of them describing the same methodology (‘the way of thinking’) but each of them looking from a slightly different angel, there is some overlap between them on the more basic issues related to the requirements management methodology.

This document has the purpose to provide guidance on requirements management, on the structure of requirement management systems and on developing, refining, using and modifying a requirements management system with the main emphasis on an overall waste management programme with its individual waste management systems (e.g., facilities / equipment for waste characterisation, facilities / equipment for waste treatment and packaging, handling / transportation equipment, interim storage facilities, disposal facilities). In contrast to the other two guidance documents, this guidance puts more emphasis on the stepwise development of a waste management programme and its systems, and explicitly takes into account that not all the systems mentioned above are implemented at once, in many cases at least some of the waste management systems (e.g. some of the disposal facilities) come much later in the programme. This means that also the development, evolution and refinement of the requirements will occur in a stepwise manner, and this is addressed in this document at a practical level. This document is mainly for those that are at an early stage of requirements management.

Table of content

List of figures

1. Introduction

1.1 Aims of the guidance document

In **EURAD** (European Joint Programme on Radioactive Waste Management) it was decided to develop guidance on requirements management, because requirements management is recognised to be a very important activity for implementing waste management programmes / systems. It was agreed to develop this guidance in a process with **active involvement of end-users** through a number of workshops and a training event. In this process, it was decided to develop t**hree documents**:

- a guidance document for generic waste management systems (EURAD 2024a),
- a document describing in more detail the development of requirements for disposal systems with a discussion of the post-closure safety case and its interaction with requirements management (EURAD 2024b),
- a guidance document for waste management programmes, taking the stepwise implementation of their systems into account (this document).

As all the three documents are **'stand-alone' documents**, each of them describing the same methodology ('the way of thinking') but each of them looking from a slightly different angel, there is some overlap between them on the more basic issues related to the requirements management methodology.

The development of these documents profited very much from the lively interactions during the workshops and the training event and from the feedback through reviews of the draft versions of the reports. As discussions continue in applying requirements management in waste management programmes, most likely these documents will see some further updates – thus, they should for the time being seen as **'living documents'**.

This guidance document **'Guidance on Developing, Using and Modifying a Requirements Management System for Waste Management Programmes with their Different Systems** – WMP-RMS' has the purpose to provide guidance on requirements management, on the structure of requirement management systems and on developing, refining, using and modifying a requirements management system with the main emphasis on an overall waste management programme with its individual waste management systems (e.g., facilities / equipment for waste characterisation, facilities / equipment for waste treatment and packaging, handling / transportation equipment, interim storage facilities, disposal facilities). In contrast to the other two guidance documents, this guidance puts quite some emphasis on the **stepwise development of a waste management programme** and its systems, and explicitly takes into account that not all the systems mentioned above are implemented at once, in many cases at least **some of the waste management systems** (e.g. some of the disposal facilities) **come much later** in the waste management programme. This means that also the development, evolution and refinement of requirements will occur in a stepwise manner, and this is addressed in this document at a practical level. This document is mainly for those that are at an early stage of requirements management.

This guidance document should allow the reader to become familiar with the key characteristics of requirements management independent of for what in detail the reader wants to use the requirements management system. Thus, the guide **describes the basic thoughts** to be made to get started with implementing and using a requirements management system (RMS) and concentrates mainly on methodological aspects with a focus on waste management systems; sometimes also providing more details. The guide is **not related to any specific waste management application**.

This guidance document should be applicable for programmes with **small and large inventorie**s with low and high complexity. The guidance document can be applied to **individual components / facilities** of the overall waste management programme or for the **whole waste management programme** – at least in the beginning when not all details are yet addressed.

This guidance document can also be applied in programmes where **implementation is already advanced without having used a requirements management system** but where such a system is planned to be used for managing future activities.

The **target audience** of the guidance document is mainly organisations that have the responsibility to develop a waste management system. However, the guideline also considers the needs of all other stakeholders. Thus, the document could also be of interest to all other stakeholders (regulator and the technical support organisation, responsible government agencies, research entities, etc.).

The document '**Guidance on Developing, Using and Modifying a Requirements Management System for Generic Waste Management Systems – G-RMS**' (EURAD 2024a) is for a generic system. The main aim of that guidance document is to provide an introduction and **make the reader familiar with the key characteristics of requirements management** independent of what the reader wants to use the requirements management system for. Therefore, it contains no details about any system. As an appendix, it also contains a **literature review on requirements management** prepared by Peter Ormai as a stand-alone document.

The document '**Developing, using and modifying a requirements management system for implementing a disposal system**' – DS-RMS (EURAD 2024a) has also the purpose to provide guidance on requirements management, on the structure of requirement management systems and on developing, using and modifying a requirements management system but its focus are **disposal systems,** mainly deep geological repositories (DGR). The document puts much emphasis on the **interaction between requirements management and the post-closure safety case**. This document is mainly for programmes working on disposal.

As the waste management programmes in the different countries differ to some extent, also the Requirements Management Systems (RMS) will differ. Thus, each programme will need its own requirements management system (newly developed or adapted from an existing requirements management system). Correspondingly, the main emphasis of the guidance document is on **methodological aspects** *('the way of thinking')* when developing and using a requirements management system and not on the details of the waste management system and on the requirements and other information as such. Thus, the guidance document mainly describes the **issues to be thought about** and not on the content – the content shown is **only for illustration**. The text is pretty dense; to better understand the text, look early enough also at the figures (**bold, underlined**).

As requirements management is a **mature methodology** that is broadly used in many applications, and as a lot of literature exists on requirements management methodology, this guide takes advantage of this **broad body of knowledge and experience**. Thus, the text in this guide relies wherever possible on the literature of that community (see reference list) that has been consulted (in limited depth) when preparing this text.

1.2 Content and structure of the guidance document

This guidance document summarises on a limited number of pages the key elements of requirements management. After the introduction explaining the aims of the guidance document (chapter 1.1) and the content and structure (chapter 1.2, this chapter), the **key characteristics of requirements** are described (chapter 2.1). Next, the two elements of **requirement management** are described: the **process** to develop, refine and use the requirements and the **system** that provides the structure for the process (chapter 2.2). This is followed by a short discussion about the **roles and responsibilities** (chapter 2.3). Finally, the practicalities of the **development and refinement of requirements** is described in somewhat more detail (chapter 2.4).

2. Aims and description of requirements management and of using a requirements management system

2.1 Key characteristics of requirements management

A brief characterisation of waste management programmes in this chapter shows that the implementation of waste management programmes is best done by using a systems engineering approach where requirements management plays a very important role. The corresponding key issues are summarised below:

- Waste disposal programmes are systems that consist of many elements that need to properly interact to make them function properly. Furthermore, most systems of waste disposal programmes have several life cycle stages that need to be considered. The implementation of waste disposal programmes is thus a process that covers a broad range of interrelated issues involving a range of disciplines; the implementation of waste disposal programmes is thus a challenging process. Based on the positive experiences in many other complex projects (e.g., aerospace, aviation, communication, computer, energy (nuclear, other), defence, software development, etc.), systems engineering is considered to be a key element for being successful with such complex projects. Systems engineering can be defined as follows (quote from INCOSE, 2015):

  'Systems engineering is an interdisciplinary approach and means to enable the realisation of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, and then proceeding with design synthesis and system validation while considering the complete problem: operations, cost and schedule, performance, training and support, test, manufacturing, and disposal[1]. Systems engineering integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation.'

  There is broad agreement in **systems engineering** (see e.g., INCOSE, 2015; NASA, 2020) that **requirements management** is an important element to support the implementation process of complex (interrelated) systems such as waste management programmes.

- Good practice in system engineering and requirements management **requires to work systematically in a structured manner with discipline** according to the rules defined by the **requirements management methodology** ('the way of thinking'). The methodology proposed here is one option, other options exist – thus, the methodology described in this document can be modified if wanted.

- A systematic approach in requirements management is essential, as it **ensures several important benefits**, e.g.:

  – Ensuring **completeness and consistency** of the information needed, and of the decisions to be taken in the stepwise approach of developing, implementing and using a product.

  – Early **detection of wrong and/or missing information**.

  – Development of a **common understanding** of all the persons working in a waste management programme and supporting the **structured interaction** between them.

  – Providing **transparency** at each stage of the project ('*why, what, when, by whom, for whom, how, influenced by whom, …*'), with transparency helping to maintain an overview, and thus **supporting daily management**,

[1] In the terminology used in this document, this corresponds to managing the 'end-of-life'.

- Providing easy access to the **currently accepted 'oversight' information** as it replaces numerous individual documents by one system and thus increases efficiency,
- Providing **traceability now and in future**. Traceability is needed to **manage refinements and changes** that are **the rule and not the exception** for long-lasting projects such as waste management projects. Traceability allows to identify the features that need to be changed to cope with the refinements and changes needed. Traceability is also needed to investigate the overall effects of suggested / needed refinements and changes before their actual implementation; this allows to make some adaptions to the proposed refinements / changes, if needed.

  Traceability is also important for keeping a record of important decisions made as part of **knowledge management for future generations** to understand the *'know why'*, the *'know what'* and the '*know how'* these decisions were taken.
- Requirements management is a prerequisite for periodically assessing the **performance and the implementation feasibility** of the proposed waste management system and to assess the importance of remaining **uncertainties and risks**.
- Supporting the **setting of priorities** (incl. distinguishing between important and urgent issues),
- Providing the means to **identify the needed capabilities** (either internally within the waste management programme or through support by 'external' service providers) for the successful implementation of the waste management system,
- Requirements management also provides a proper basis for **estimating development effort and cost**.
- etc.

- The definition of systems engineering, and the brief discussion of requirements management above indicates that the activities needed when planning, implementing and using waste management systems are essentially the same activities as performed in system engineering and requirements management. Therefore, it is **recommended and justified** for waste management organisations (the implementers) to develop and **use a requirements management system** as it has been done by the waste management organisations with advanced HLW programmes, as using a requirements management system does not require additional activities but it supports the systematic way of working with all the resulting benefits mentioned above.

  For developing this guidance document, the information available mainly from the 'International Council on Systems Engineering – INCOSE', see e.g., INCOSE, 2022a), but also from other literature is used, taking the experience made in several waste management organisations into account.

- Requirements management supports **effective leadership and efficient management** (as defined e.g., by Drucker, 2001) in **product development and implementation** (as defined e.g. in INCOSE 2022b) by addressing the following two issues:
  - *'Do the right things'* (a key element of leadership) and develop *'the right products'* and implement them at the *'right time'*.

    In other words, doing *the right things* starts with defining *'why'* is '*what'* wanted by *'when'*. The *'why'* consists of the high-level goals, needs and expectations on the product as defined by the external stakeholders that initiate the development of the project. The *'what'* results from decomposing / breaking down the *'why'* into more detailed and tangible requirements.
  - *'Do the things right'* (a key element of management) to arrive at *'the right product design'* with the *'product being implemented right'*.

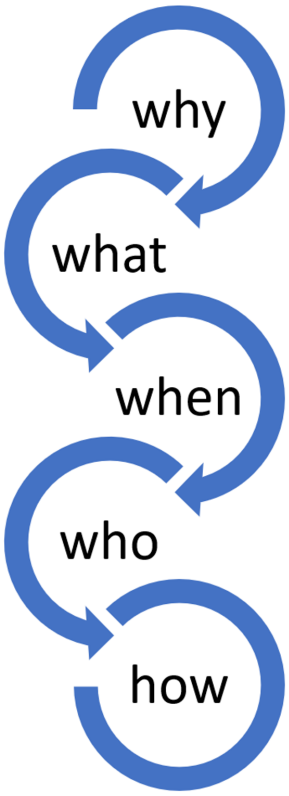
    In other words, *doing the things right* consists of specifying *'who'* (the needed products) must be implemented *'how'* to fulfil the *'what'*
- The *'why'*, *'what', 'when', 'who'* and *'how'* are the **cornerstones** of the clearly defined and **systematic requirements management process** organised and documented in a requirements management system, as briefly described below:
  - The **'why'** gives the reasons for the products[2] by defining the goals of the implemented products, the needs the products have to fulfil and the expectations on the products.
  - For developing the products, the goals, needs and expectations have to be decomposed / broken down into more tangible descriptions on **'what'** the products have to fulfil by **'when'**.
  - The products and their development, the implementation, the use and the 'end-of-life'l consist of many sub-products, including temporary support products – the **'who'** that are needed by 'when' to fulfil the 'what'. To maintain an overview on all the products needed, they are put as 'black boxes' into a so-called 'functional' architecture that captures the products and their interactions that need to be implemented to fulfil the goals, expectations and needs.
  - Finally, for each product, a specification has to be developed that describes **'how'** the product should look like (its key characteristics) and 'how' it should be implemented to fulfil the 'what'.

  The sequence of addressing new issues mentioned above is briefly summarised in **Fig. 1**. This description is also the basis for the structure of requirements management system shown in **Fig. 2** (for an **early-stage programme** (or a programme with some elements being at an early-stage) with only broad concepts being defined and some of the information just being based on plausible assumptions).

  In case of early-stage programmes, after having **developed concepts** for the systems / elements of the waste management programme that need sooner or later to be implemented, these concepts can be used to **manage / define the interfaces** between these systems / elements (e.g. by defining waste acceptance criteria) or with the systems outside of the waste management programme (e.g. the supply-chain). Then, the work can **concentrate** on those systems / elements that should be developed into a more advanced stage within the planning horizon; work on the other systems / elements can wait until they get closer to the planning horizon and need attention.

[2] Instead of the term 'product' also the terms 'element', 'system-element', 'system' / 'system of interest' (e.g. waste management system or disposal system) are used.

The scheme in Fig. 2 will thus evolve into a more complex scheme as shown in **Fig. 3** (for an **early-stage programme** with only one or several systems / elements being developed (and – if sufficient progress has been made – are implemented and used); or for an **advanced programme** with all elements being defined, most of them based on solid investigations, and some elements already being implemented and used). The **stepwise development** of the 'early-stage' scheme in Fig. 1 into a scheme with systems / elements being at a more advanced stage is illustrated with the scheme in **Fig. 4**.



The 'why' gives the reasons for developing / implementing a waste management system by defining the 'goals', 'expectations', 'needs' to be fulfilled

The 'what' provides direct input for developing / implementing a waste management system and its elements by decomposing the 'why' into more detailed and tangible requirements

The 'when' defines at which stage of implementing the waste management programme the 'what' must be fulfilled

The 'who' defines the element of the waste management system that must fulfil the allocated 'what'

The 'how' defines the design of the element of the waste management system such that all the allocated 'what' are fulfilled

Figure 1 The cornerstones of requirements management used in a systematic and structured team-effort to develop and implement waste management systems

Furthermore, it is important to recognise that **constraints** exist (e.g. the waste to be managed, existing waste management facilities, high-level decisions taken, etc.), that need to be considered. For such constraints, the **order of addressing the cornerstones changes**: *'who'* has to fulfil *'what'* (and *'why'*) by '*when'* and *'how'.* Thus, the scheme in Fig. 1 is for constraints not directly applicable.

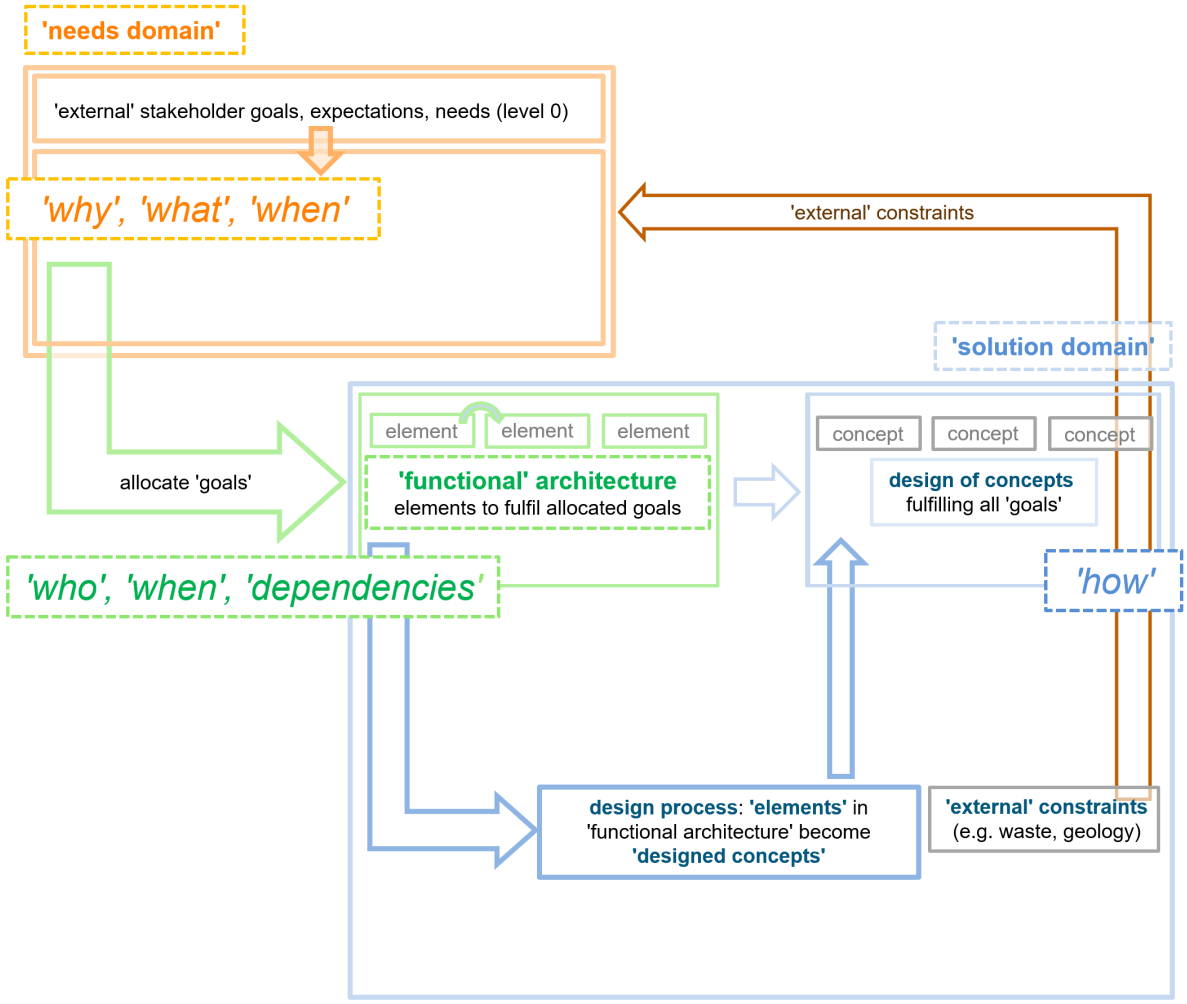


Figure 2 A scheme with the domains of the requirements management system (the 'needs domain' – describing the 'why', 'what', 'when'; the 'functional architecture' – capturing the 'who', 'when' and dependencies between elements; the 'solution domain' – defining the 'how') and the flow of information and the workflow (arrows).

The scheme shows the elements of a programme at an early stage, where the main aim for the 'needs domain' is on the goals, needs and expectations of the 'external' stakeholders and where the 'functional architecture' only contains high-level elements and the 'solution domain' only concepts for the high-level elements. However, 'external' constraints (e.g. waste, geology, existing facilities) need alsoto be considered from the beginning on.

2.2 Requirements management: the process and the system

2.2.1 Overview

In this chapter, an overview on the process of requirements management and the platform used for that – the requirement management system – is provided.

- Requirements management has two important elements: the **process** of developing, refining and managing the information (called requirements management) and the requirements management **system** that provides a **platform** and the **structure** to organise and manage the information in a useful manner, and also stores the information in an adequate format for its further use.

  Requirements management as process of the team members of the implementer includes the following activities:

  It starts with **defining the needs** ('needs domain' in **Fig. 2** and **Fig. 3**):

  – Elicitation / extraction of the **goals**, needs and expectations of the 'external' stakeholders (the high-level 'why' – 'level 0') – at an **early stage**, defining the high-level goals is sufficient; the high-level goals often also provide broad indications about the high-level 'what' and 'when' (see **Fig. 2**)

    At a **more advanced stage**, the goals by the 'external' stakeholders are complemented with more detailed goals with some of the goals then being decomposed into sub-goals, making the 'why' more detailed ('level 1' in **Fig. 3**) that normally gives a clearer indication about the 'when'.

  – Decomposition / breaking down of these 'goals' into **functions** with the corresponding performance targets and into **(quality) characteristics** with the corresponding quality targets (the 'what') is done at a more **advanced stage** in considerable detail – this leads to more detailed new information ('levels 2a/b' and 3a/b' in **Fig. 3**), with the new information also giving more information about the 'what' and the 'when'.

    At an **early stage**, defining the high-level 'what' is done at a rather superficial level with the aim to better capture / define the 'level 0' - goals, if this is needed at all. This also applies to defining the 'when' based on the high-level goals; thus, this very limited superficial information is added to the 'level 0' - information (see **Fig. 2**)

    The distinction between **functions** and the **(quality) characteristics** defined at a more **advanced stage** ('levels 2a/b' and 3a/b' in **Fig. 3**) is made because they are somewhat differently used in the design process:

    – **Functions** are the starting point to define the more detailed and tangible requirements. They define the **key purposes of the products** to be designed (e.g. for a shaft of a DGR, transportation of material and persons is a typical function); the functions also dictate the elements needed for implementation as captured by the 'functional architecture',

    – **Characteristics** are equally important but come second in the development of the more detailed and tangible requirements. They are not related to the purpose of the products but to key characteristics that can be related to the functions (e.g. in relation to transportation through a shaft of a DGR, its safe operation and its reliability are typical characteristics) or are related to other issues (e.g., not to significantly perturb the functionality of an other element).

Then, the **elements / products / systems to fulfil the needs** are identified ('functional architecture' in **Fig. 2** and **Fig. 3**):

- This requires the identification of the elements[3] (documents, decisions etc. as result of activities / processes, and objects (incl. equipment, etc.)) that are needed to fulfil the requirements. Then, the 'when' of these elements needs to be defined. These **elements** and the dependencies between them are captured in the so-called **'functional architecture'**.

  At **an early stage**, only the **high-level systems** of the waste management programme or the **high-level elements** of a waste management system are identified. As indicated in the design process below, this can require some iterations because upfront, sometimes not all high-level systems / high-level elements are thought about. However, not many iterations will be needed as the functional architecture will not contain many elements (see **Fig. 2**).

  At a more **advanced stage**, the high-level systems of the waste management programme or the high-level elements of a waste management system are decomposed into more detailed 'elements' (see **Fig. 3**)

  In a next step, the requirements of the 'needs domain' are **allocated** to the 'elements' of the 'functional architecture' that need to fulfil them. This also allows to define the 'when' of the 'elements.' Then, the compatibility of the different requirements (incl. their 'when') with the 'element' needs to be checked; this can lead to some modifications of the requirements, adding additional 'elements', etc.

  At an **early stage**, only high-level elements exist and the number of requirements (the high-level 'why', 'what' and 'when') is very limited.

  At a **more advanced stage**, more requirements (now consisting of functions and quality characteristics with their targets) and more elements of the functional architecture must be considered; thus, more work needs to be done but the process is the same as at an early stage.

Then, the **design of the elements / products / systems** starts ('solution domain' in **Fig. 2** and **Fig. 3**) in view of their later implementation, use and 'end-of-life:

- For the **design of the 'elements'**, besides knowing the allocated requirements other important information is needed, e.g. the environmental loads and conditions acting on them (e.g. flooding of surface facilities). This allows then the design of the different 'elements'.

  For most 'elements', the different **stages of their life cycle** (planning (with different levels of detail), implementing, using, managing the 'end-of-life') need to be explicitly addressed in the **design / planning process**.

  At an **early stage**, this process is done at a 'high-level' – for the high-level 'elements' **broad concepts** are developed based on broad assumptions on the loads and conditions.

  At a **more advanced stage**, first a pre-design / pre-planning is made that defines e.g. for objects the loads and conditions, the types of material to be used and the rough dimensions. With this, the **'design input requirements'** ('level 4a') for the elements are defined.

  Based on this input, the **'design output specification'** ('level 4b') is developed with a design of the element that fulfils the allocated goals, functions / characteristics with their targets. This results in **product specifications** and **production specifications**.

[3] Often also called products, systems, (system) elements; sometimes also 'means'

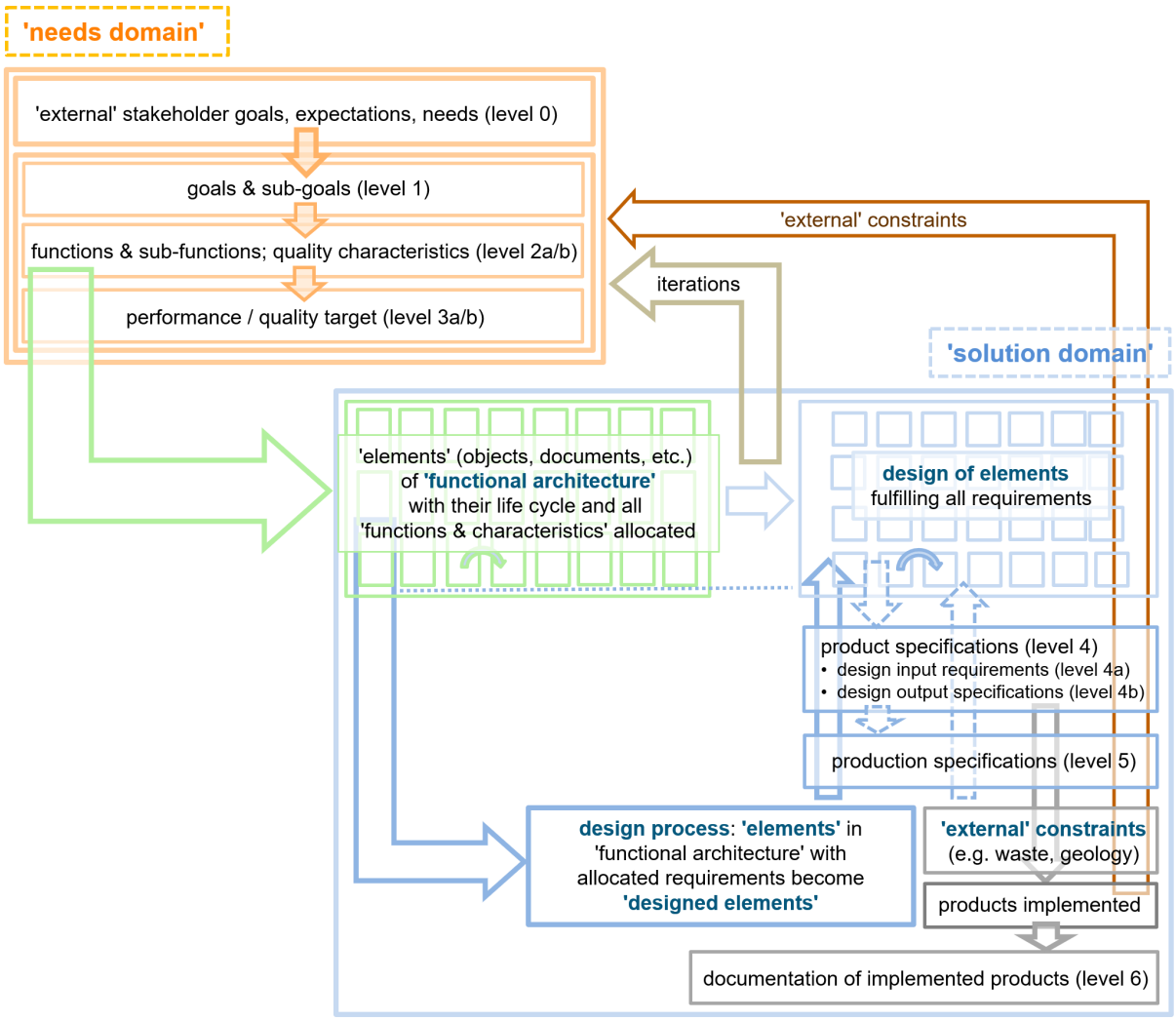


Figure 3 A scheme with the domains of the requirements management system ('needs domain', 'functional architecture', 'solution domain') and the flow of information and the workflow for a programme at an advanced stage.

At a more advanced stage, the main aim of the 'needs domain' is not only on defining the goals, needs and expectations of the 'external' stakeholders but also on the development of the functions / characteristics with their targets, and where the 'functional architecture' now contains also the detailed elements and the 'solution domain' now consists of product and production specifications for the detailed elements. The constraints contain besides the external elements (waste, geology, ..) now also the already implemented products.

- The **requirements management system** provides the structure (domains, relations, information flow, …) to store the information derived by the requirements management process described above ('needs domain', 'functional architecture', 'solution domain'), and to manage this information (with the help of attributes, links, …). The more simplified approach at an early stage as described above, has a less detailed requirements management system (see **Fig. 2**); with the programme getting more advanced, the scheme gets more detailed (see **Fig. 3**).
- In its simplest form the system can be a set of (written) instructions using predefined templates, but for complex projects there are digital software solutions that help represent and work with the requirements management system.

2.2.2 Details about the stepwise process of requirements management

Based on the information in the last chapter about the differences in requirements management between early stage and more advanced programmes, this chapter provides more details about the stepwise process of requirements management.

- The **requirements management process** needs to be clearly described, e.g. by specific workflows. It consists of the **following steps** (**Fig. 4** and **Fig. 5**), with the steps also being explicitly used / addressed in the requirements management system:
  - Step 1: obtain and clarify the **goals** (needs) of 'external' stakeholders[4] (e.g., based on their documents) for the overall waste management system under consideration,
  - Step 2: identify and assess the impact of the so-called **'constraints'** (facts: existing / planned waste, existing / planned facilities, siting options, high-level decisions taken, …) on some of the goals (step 1) and elements of the 'functional architecture' (step 3),
  - Step 3: identify the **'elements'** (products as the result of activities / processes, and objects) and their dependencies (captured by the 'functional architecture') that are needed to **achieve the goals** – in an early stage of a programme focussing mainly on the goals, as only the high-level system-elements are considered,
  - Step 4: in an early stage of the programme, **allocate the goals** to high-level system elements of 'functional architecture' (defined in step 3).
  - Step 5: find a **'solution'** for each of these high-level system elements. For high-level system elements, these 'solutions' are at an early stage normally in the format of **concepts** (high-level representation of the key characteristics of the high-level elements without any details); the level of detail provided by concepts is sufficient for a programme at an early stage.

    The simplified version of the requirements management system that only contains the goals (as 'needs', the high-level system elements (as 'functional architecture') and concepts (as 'solution') is depicted in **Fig. 2**.
  - Step 6: recognising the evolutionary nature of requirements management, **refine the information**, and correspondingly also the requirements management system (see **Fig. 3**). This includes:
    - **decomposing the goals** into sub-goals / functions / characteristics with their targets (step 6a); this also includes the more detailed consideration of constraints,
    - **refining / decomposing the 'functional architecture'** correspondingly by adding more detailed elements (sub-systems, components, etc.) and by including the stages of the life cycle of the elements (step 6b),
    - **allocating the functions / characteristics** for each stage of the life cycle (planning / licensing of element, delivering / implementing of element, using of element and 'end-of-life' of element) to the different elements of the 'functional architecture', as far as applicable (step 6c),
    - **developing a 'solution'** with a design for all elements such that all elements together fulfil all requirements. This leads to product specifications and production specifications for all elements and for all stages of their life cycle (step 6d).

[4] Discussed in chapter 2.4

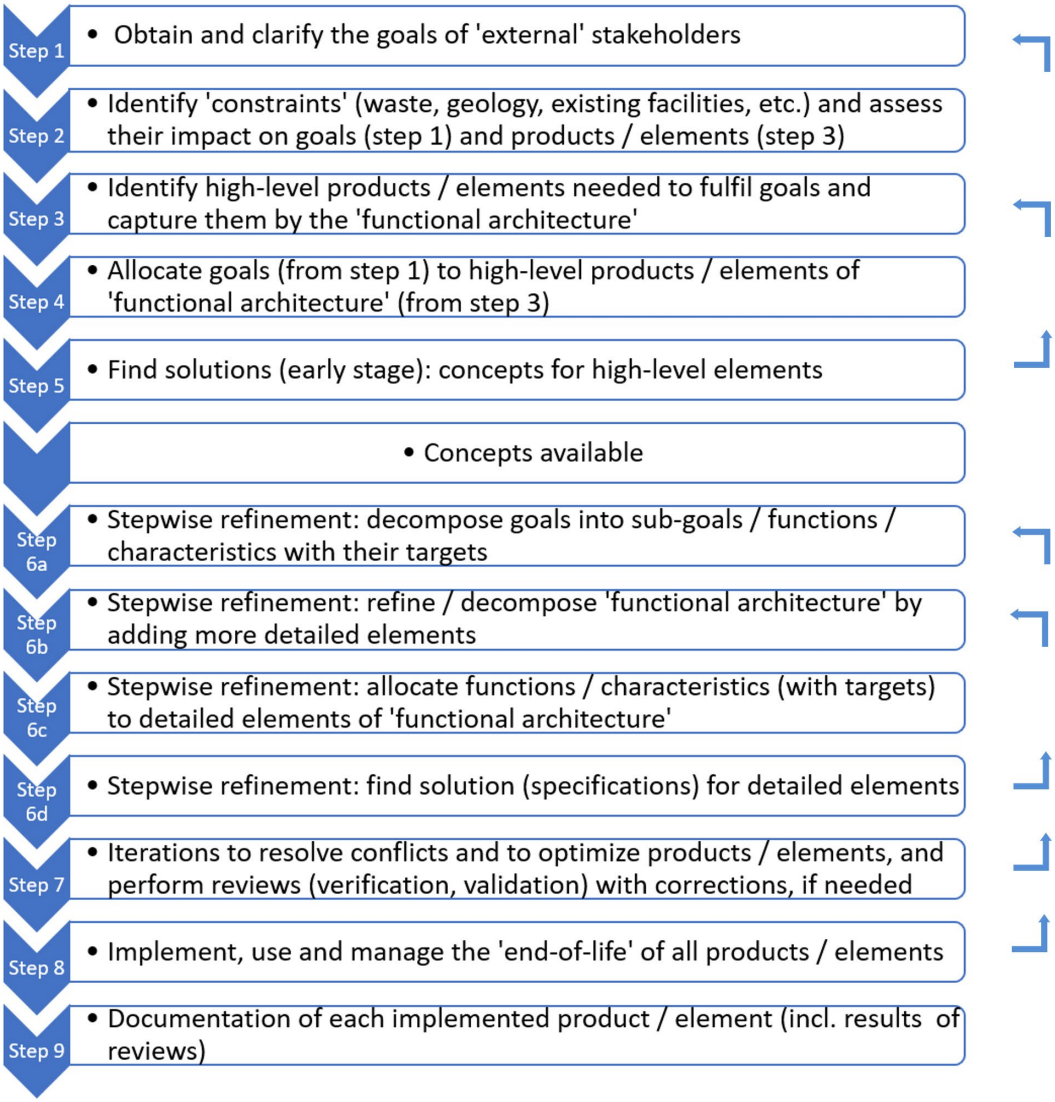


Figure 4 Overview on the different steps of requirements management to fully define the waste management system, with all the stages of the life-cycle of its elements. This process is performed in an interative manner, as indicated by the arrows

Out-going arrows indicate that an earlier step may be revisited, the in-going arrows show where the iteration could start. In principle, each iteration can start at any of the inidicated starting points earlier in the process – where to start, is based on a case-by-case assessment.

- Step 7: The process of developing information is done in an **iterative manner** (e.g., to resolve conflicts between requirements, to change issues that turn out not to be feasible, to optimise the system, etc.), that also involves **reviews at the different stages** of development:
  - **verification** (*'doing the things right'* for the individual elements), and
  - **validation** (*'implementing the right things'* for the overall system).

- Step 8: **implement, use and manage the 'end-of-life'** of all elements of the waste management system using the specifications developed; if needed, based upon updated requirements / specifications, again involving **verification** and **validation.**
- Step 9: **documentation** of each implemented element that also includes the documentation of the successful completion of the reviews (verification and validation).

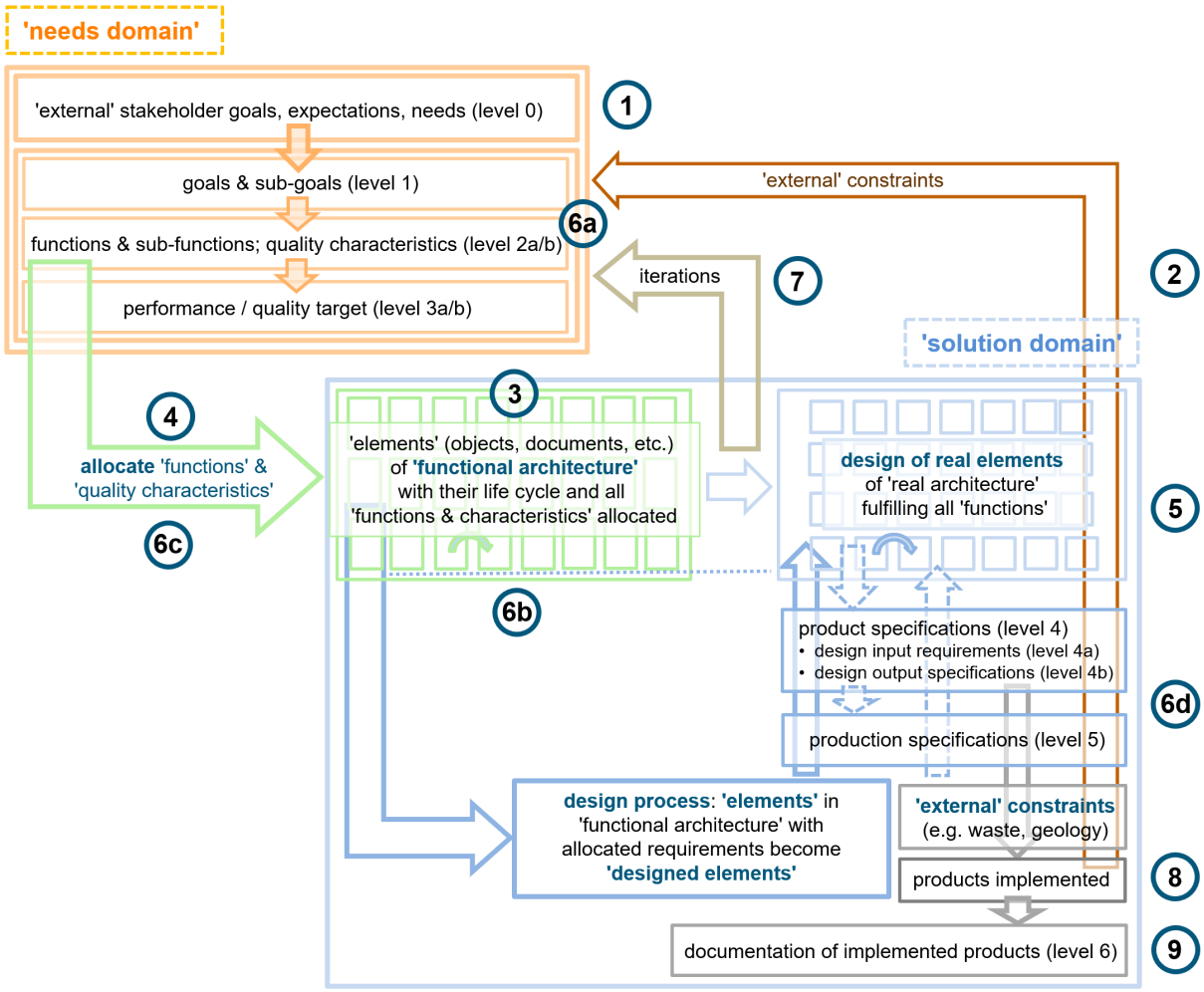


Figure 5: A scheme with the domains of the requirements management system ('needs domain', 'functional architecture', 'solution domain') and the flow of information / workflow for a programme at an advanced stage (same as **Fig. 3**).

The scheme also contains the steps (see numbers in scheme) performed in the requirements management process as described in chapter 2.2.3 and depicted in **Fig. 4**.

2.2.3 Details about the requirement management system

In this chapter, more details about the requirements management system and its structure are provided.

- The fully populated version of the requirements management system is depicted in **Fig. 5** with the numbers in that figure corresponding to the steps mentioned in chapter 2.3.2 and in **Fig. 4**. The system contains:
  - the **'needs domain'** with all requirements (goals, functions / characteristics with their targets),
  - the **'functional architecture'** with all elements of the waste management system, with the requirements allocated to the different elements of the waste management system,
  - the '**solution domain'** with the **'product specifications**' and the **'production specifications',**
  - and at the very bottom of the 'solution domain' the **implemented products** and the **documentation** for each implemented element.

  To organise the requirements management system and to manage the information, for the different types of information, so-called levels are defined, see the **levels** in the boxes in **Fig 3** / **Fig. 5**. These levels apply to the requirements of the 'needs domain' and of the 'solution domain'.
- The **'needs domain'** contains the goals of the 'external' stakeholders ('level 0') that are complemented by sub-goals ('level 1') and these are decomposed into functions and characteristics each of them with its target ('level 3a/b' and 'level 4a/b'.)
- Next, the **'functional architecture'** needs to be defined. It has to capture all elements that are needed for planning, implementing, using and managing the 'end-of-life' of the waste management system and its elements. Then, each of the **'level 3a/b'- requirements with its target** is **allocated** to one of the elements of the 'functional architecture'.

  The development of the 'functional architecture' is thus an important task. The structure and content (generic) of the 'functional architecture' is shown in **Fig. 6**.

  The **'functional architecture'** has to address the different **life cycle stages** of the waste management system and its elements (also called themes):
  - **Using the system** (T-1). As this is the most important part of the overall system, this life cycle stage is normally the starting point and driver for the overall planning of the waste management system and for the corresponding requirements management.
  - **Implementing the system** (T-2). Before one can use the waste management system, it has first to be implemented. Thus, this life cycle stage has to be included in the overall planning and in requirements management.
  - **Planning and licensing the implementation and the use of the system** (T-3). Before implementing and using the system, its implementation, its use and its 'end-of-life' has first to be planned, corresponding decisions need to be made and the system (objects/facilities, equipment, operational procedures, etc.) have to be designed and the needed licences and permits have to be acquired. Thus, this life cycle stage has to be included in the overall planning of the waste management system and in requirements management.

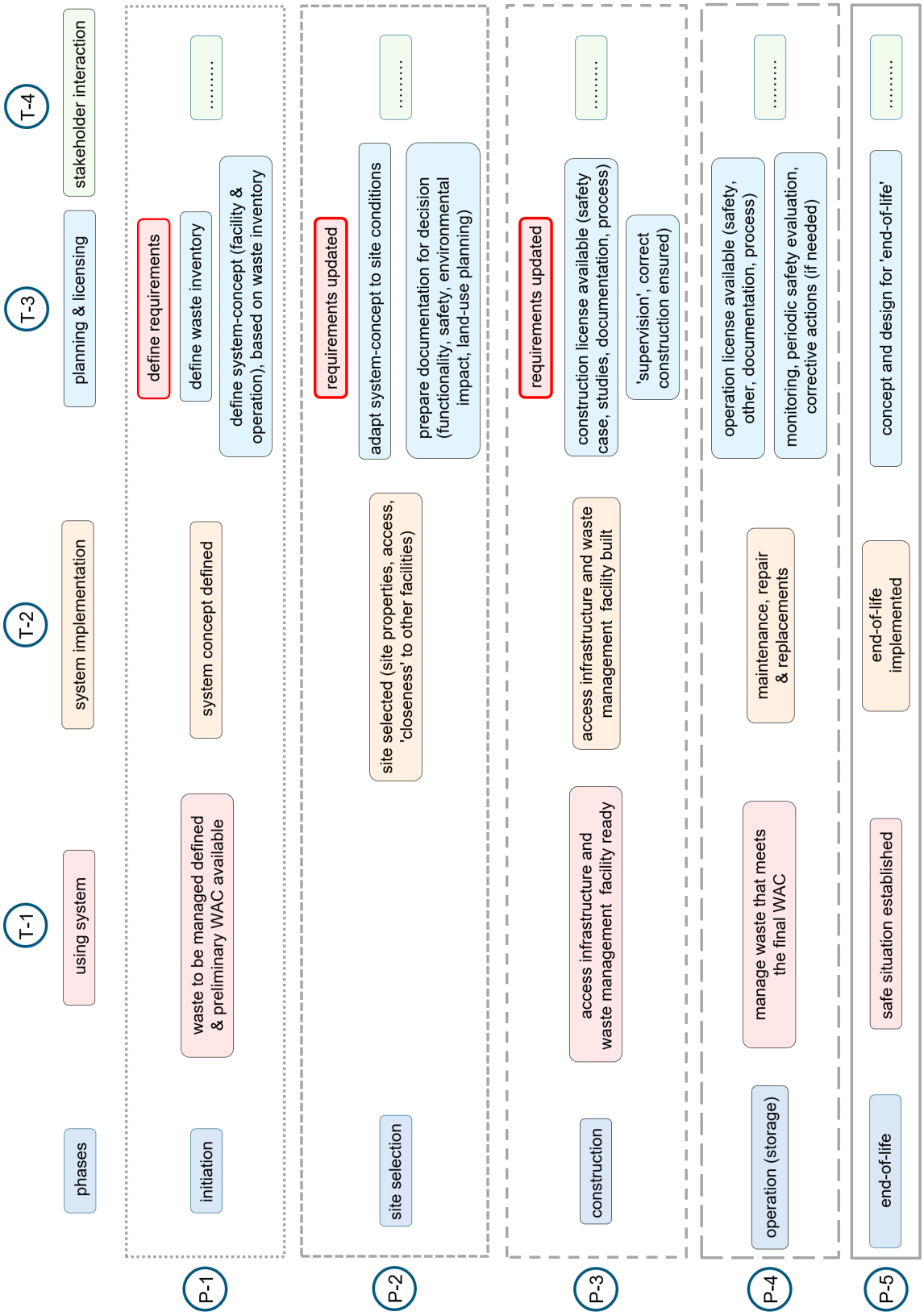


Figure 6: Elements of the 'functional architecture' of a waste management system with the delivered / implemented products / elements as coloured boxes, with these belonging to a life cycle stage / theme (T-1 to T-4) and being dealt with in a phase (P-1 to P-5).

- **Managing stakeholder interaction** (T-4): This interaction[5] (mainly with the general public, the local population at selected sites, interest groups, etc.) is essential for the full duration of the programme to inform and interact with the public and to take their views in the development of the project adequately into account to achieve the needed support. Thus, this theme has to be included in the overall planning of the waste management system and in requirements management.

Next, it is important to define the **'functional' phases** of the programme (see again **Fig. 6**). This includes (simplified, for illustration):

- programme initiation (P-1)
- site selection for the facilities needed (P-2)
- construction of the facilities needed (P-3)
- using the facilities (P-4)
- managing the 'end-of-life' of the facilities (P-5)

- In the **'solution domain'**, the requirements allocated to the elements of the waste management system are designed. This includes:
  - The development of the **'design input requirements'** ('level 4a'), where the compatibility of the allocated requirements is checked and made compatible (based on negotiations and changes, if needed), the loads and conditions acting on the element (mainly for objects) are defined (with some margins to take uncertainties into account) and a pre-design is established (mainly for objects, with the types of materials and the broad dimensions being defined)
  - This allows then to develop the **'design output specifications'** ('level 4b'), where the detailed design is developed, based on the 'design input requirements'. This leads to **'product specifications'** and **'production specifications'**.

    With the specifications for all life-cycle stages being available, **implementation** can start, followed by **use** and eventually managing the **'end-of-life'**.
- Finally, the **broad products / elements** must be delivered / implemented according to the specifications to achieve the goals and to meet the corresponding functions / characteristics with their targets. This requires that all stages of the life cycle of the products / elements are considered. The boxes for the different themes / activities in the different phases in **Fig. 6** are simplified illustrations for these products / elements for a waste management system.

[5] This does not involve the 'external' and 'internal' stakeholders discussed in chapter 2.3.

2.3 Roles and responsibilities of the different stakeholders

In this chapter, the roles and responsibilities of the different stakeholders are briefly described. Then, the work by the implementer – a team effort – is briefly summarised.

- Requirements management involves on the systemic level **different actors with different roles and responsibilities**. It includes:
  - **'External' stakeholders**:
    - having a 'problem' (expressed as goals, needs, expectations) and are strongly interested in finding a 'solution' and initiate the corresponding process, as they have a formal (co-)responsibility to ensure that a solution will be found,
    - but are not directly involved in 'developing' the solution,

    There exists a range of 'external' stakeholders (policy maker, regulator, waste generators, …), with many of them having documents that define their goals, needs and expectations.

    In case of a voluntary approach for site selection (incl. the approach by providing the municipalities with a 'veto-right'), the municipalities potentially hosting the facility are directly involved in the decision-making; thus, they are also part of the 'external' stakeholders.

    For the success of the project, it is very important to capture all the relevant goals, needs and expectations of the 'external' stakeholder, incl. the corresponding documents.
  - **'Internal' stakeholders** – the **implementer** (the 'waste management organisation'):
    - have the responsibility to find and implement a solution,
    - transform (decompose) the goals, needs, expectations of the 'external' stakeholders into tangible requirements (functions / characteristics with their targets),
    - develop and refine the 'functional architecture' and allocate the functions / characteristics and their targets to the elements of the 'functional architecture',
    - develop product specifications and production specifications (for objects, documents, decisions, …) that fulfil the design requirements,
    - ensure the implementation, the use and the 'end-of-life' management according to the product specifications and production specifications, perform the corresponding verification and validation and prepare the final documentation for each step.
  - **Other stakeholders**
    - are not directly involved in finding a solution,
    - but are involved through information and consultation and this needs to be considered in requirements management,

- To **capture the 'external' stakeholder information**, it can be useful to develop a map with all the stakeholder documents, see e.g., the simplified scheme in **Fig. 7**. This scheme shows that the needs of the external stakeholders cover a **broad range of areas**, e.g.:
  - obligation to manage the radioactive waste
  - nuclear safety
  - radiation protection
  - safeguards and security
  - environmental impact
  - land-use planning
  - working standards for all the disciplines involved
  - public interaction
  - etc.

  Furthermore, for each of the areas there may be a **hierarchical structure**, e.g.:
  - **international** agreements, guidance and standards at the highest level. The international agreements are often integrated into national legislation; international guidance and standards are sometimes also reflected / considered in national guidance and standards,
  - **national** law (also addressing at least some of the issues mentioned above),
  - national ordinances and regulations providing more details for some themes addressed in national law,
  - country-specific standards for different disciplines (also outside of the nuclear field),
  - for specific projects, based on national law, ordinances, regulations, etc., **reviews and decisions** are made that can include binding requirements for the project under development,
  - this also applies to **planning documents and decisions** made at the level of the state / **province** and/or **municipality** that may also contain binding requirements.

  In this map, also the systems interacting with the system analysed can be included and the broad approach to **manage the interfaces between these systems** can be described.

  This can include, e.g., the **waste acceptance process a**nd criteria for each high-level waste management (sub-)system (e.g., waste characterisation, waste (pre-)treatment / conditioning / packaging, handling / transportation, interim storage, disposal).

	obligation to manage radioactive waste	nuclear safety	radiation protection	safeguards security	environmental impact	land-use planning	working standards	public interaction	etc.	other systems
international agreements & standards	IAEA standards and guidance, ...	IAEA standards and guidance, ...	ICRP publics, recomms	NNP Treaty, agreements IAEA guidance	Espoo convention	agreements (FAO)	Eurocodes, etc	Aarhus convention	xxx	
national law	xxx	xxx	xxx	xxx	xxx	xxx		xxx	xxx	
national ordinances regulations	xxx	xxx	xxx	xxx	xxx	xxx		xxx	xxx	
national (regulatory) guidance, standards, codes	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	
legal & regulatory review / decisions	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	
state / municipality planning and decisions	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	
... and their impact on the internal stakeholders of system under investigation										
implementer documents	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	
interfaces with other systems										xxx

xxx: country-specific

Figure 7: Landscape of external stakeholder documents: illustation about important elements (incomplete). For more detailed explanations, see text.

- Implementing a waste management system and developing, refining and using the corresponding requirement management system is the **task of the 'internal' stakeholders** – the **implementer** (often called waste management organisation).

  The work to be done by the implementer is a **team effort** that consist of both the technical development of the products needed and the project management supporting the technical development process. The overall process consists of the following elements:

  - The **technical development** of the products is done by:
    - **Systems engineers**, responsible for the process of correctly populating the overall requirements management system with information (incl. design input requirements and design output specifications) and the correct use / application of the information. For this, the system engineers have to organise and oversee the design process and thus need to have a good understanding of the system to be developed, but they do not need to understand all the details about each product – they are 'generalists'.

      The system engineers are also responsible to **manage the requirements management process**. That includes the development of workflows and keeping them up-to-date as well as ensuring that they are applied whenever needed; thus, they ensure that the status of the information in the requirements management system is at all times clearly visible and that only 'cleared' information is being used for developing the projects.
    - **Subject matter experts**, responsible for the scientific-technological details of the different products, with also having a good understanding about the context of the products they are responsible for. They also have to be able to decompose the goals into functions / characteristics with their targets.

      Some of the subject matter experts **oversee the RDD** for the different products and provide the corresponding information, while the other subject matter experts are the **specialists for the design** and the development of the product specifications and production specifications and oversee the implementation, the use and the management of the 'end-of-life' of the product.
    - **Project management** for the work to be carried out **supports** the efficient technical development of the products and includes the management of the following issues:
      - activity lists, bar charts, network diagrams,
      - resources needed / available,
      - cost / budget available,
      - organisational framework of the project,
      - etc.

    Thus, there are **strong interdependencies between systems engineering, using and maintaining the scientific-technological basis and design, requirements management and project management** that must also be considered when setting up the requirement management system – what should be included and what not? At a high level, it is considered worthwhile to **include also** the project management issues. This also needs to be considered when organising the team; it is essential that the **team works as an integrated unit** well together without any administrative / organisational hurdles.

  - At an early stage, the **team will be very small,** with one person having several of the responsibilities mentioned above; the **team will grow** as the programme advances.

2.4 Details about developing and using the requirements management system

In this chapter, some more details are provided about stepwise development / refinement of a requirements management system as described in chapters 2.2.1 and 2.2.2.

- The **development** of the information needed for implementing the **high-level product** starts with the 'needs', the 'functional architecture' and the 'solution'. The steps in this process are depicted in **Fig. 8** – the evolution of a **single high-level product / element** through the different stages of its life cycle. **Fig. 9** shows the different issues to be addressed in the development process for the first stages (only part of Fig. 8). **Fig. 10** illustrates the flow of information and the workflow in relation to the issues shown in Fig. 9. **Fig. 11** and **Fig. 12** show in more detail the workflow with the different process steps.
- The horizontal sections in **Fig. 8** define the **different stages of the life cycle** of a system / product / element. The (sub-)stages of the life cycle are:
  - **Planning** and development:

    that **starts** at an early stage with
    - defining the goals (based on input by external stakeholders) → goals defined (initial 'needs domain'),
    - identifying / defining the high-level element(s) that have to fulfil the goals (initial 'functional architecture'),
    - developing a concept for the element(s) / product(s) → concept(s) defined with using the concepts to define and manage the interfaces between them and to other systems (initial 'solution domain'),

    and is **followed** by further **developing / refining** the domains (the 'needs', the 'functional architecture', the 'solution domain') at a more advanced stage:
    - the 'needs domain' by adding goals and sub-goals and decomposing these into functions and characteristics with their targets,
    - the 'functional architecture' by breaking down the high-level elements into more detailed elements,
    - the 'solution domain' by replacing the concepts by design input requirements and design output specifications for a system / element(s) / product(s).
  - **Delivering / implementing** the element(s) / product(s) according to the design output specifications → element(s) / product(s) implemented,
  - Prepare use of system / element(s) / product(s):
    - update of operations requirements → updated operations requirements,
    - update of operations specification → updated operations specifications.
  - **Using a system / product / element** according to updated specifications → products needed produced,

- Prepare management of 'end-of-life':
  - Update of 'end-of-life' requirements → updated operations requirements,
  - Update of 'end-of-life' specification → updated operations specifications,
- **Manage the 'end-of-life'** of system / product / element according to updated specifications → 'end-of-life' successfully managed.

stages of life-cycle	input	activities	output
planning (requirements, concepts, designs) fulfils goals			
defined goals to be fulfilled by concepts			
develop concepts according to goals			
concepts defined			
defined functions / characteristics & targets to be fulfilled by specifications			
develop design input requirements			
design input requirements defined			
specifications defined			
implementation fulfils specifications			
implement according to specifications			
product implemented			
prepare use of product that fulfils specifications			
update operations requirements			
updated operations requirements			
updated operations specifications			
use / operate according to specification			
products needed produced			
prepare end-of-life management that fulfils specifications			
update end-of-life requirements			
updated end-of-life design input requirements			
updated end-of-life specifications			
implement end-of-life specification			
end-of-life management completed			

Figure 8: Stages of the life-cycle of a product with key issues addressed in each stage of the life-cycle indicated in column-titles. The level of detail in the requirements, the functional architecture and design increases until implementation starts, followed by use and manging 'end-of-life'.

Colours: red – aim / goal of stage of life cycle; yellow – performing activities; green – design input requirements defining the basis for the detailed design; blue – design output specification; grey-blue: resulting products that fulfil all requirements.

Fig. 9 shows the different issues ('boxes') with information for the first stages of the life cycle shown in Fig. 8. The 'boxes' cover:

- in the columns:
  - the different **stages of the life cycle** – column 1,
  - the **functions** and the performance targets to be fulfilled – the main driver of the design process (input for the design in column 4) – column 2,
  - the **(quality) characteristics** and the quality targets to be fulfilled for the product to be feasible and acceptable (input for the design process in column 4) – column 3,
  - the **output from the development process** (interim results, requirements, specifications, products / elements, etc.) – column 4,
  - the **activity / process of developing the output** (in column 4), based on the input from column 2 and 3 and on products / interim results in column 4 – column 5,
  - the **interaction with society** – column 6.
- in the lines, information is given for the **different stages of the life cycle** for the topics as described in the columns above, with the stage of the life cycle advancing from one line to the next.

Fig. 10 shows the same boxes as Fig. 9, now with arrows that show the **flow of information and the corresponding workflow**:

- For the horizontally oriented scheme, red horizontal arrows **from left to right** go **from a goal** or a function / quality characteristic with their targets **as input towards an activity** / process to initiate a process that transform this input **into an output** (see next line).
- Yellow horizontal arrows **from right to left** go **from an activity** / process **towards the output** (a product or a change of the stage of the life cycle of a product),
- Yellow horizontal and inclined arrows **from left to right** are **using a product / interim result** (output from an earlier step) as **input for an activity** / process that will transform the input **into a new output** (a product or a change of the stage of a product). Sometimes, these yellow input arrows are complemented by red input arrows (see above).
- The activities / processes often involve **integration** of different sources of information and the assessment of the correct interplay of the information is very important.
- Green **vertical dotted arrows** are related to the **decomposition** of functional goals and quality characteristic goals into functions and characteristics and their targets and the decomposition of the functional architecture by replacing the concept by more detailed products / elements and with covering the full life cycle for each of the elements.
- The arrows in Fig. 10 do **not include the needed iterations** – they only depict the 'linear' part of the workflow.

1	2	3	4	5	6
stages of life-cycle	functions (different levels)	quality characteristics (different levels)	needed products / needed stage of products	activities to develop needed products / needed stage of product (production specifications)	interaction with society
defined goals to be fulfilled by concepts	high-level functional goal of facility / process	high-level characteristics goals of facility / process	facility / process concept fulfilling high-level functional / quality characteristics goals	develop facility / process concept such that it can fulfil its goals	inform public about facility / process
develop	high-level functional goals to be met		high-level elements needed to fulfil high-level functional goals	develop broad ideas about how to fulfil high-level functional goals	
develop		high-level quality characterisitcs to be fulfilled	high-level elements needed to fulfil high-level quality characteristics goals	develop broad ideas about how to fulfil high-level characteristic goals	
develop			functional' architecture consisting of high-level system elements	review proposed elements and consolidate needed elements and identify dependencies	
example facility / process			concept fulfilling all high-level goals	develop conceptual design for high-level elements of functional architecture	
defined functions / characteristics & targets to be fulfilled by specifications	functions / targets of facility / process	quality characteristics / targets of facility / process	facility / process basic design fulfilling the functions / characteristics	develop basic design of facility / process to fulfil functions / characteristics and corresponding targets	
develop	functions needed and corresponding targets		facility / process basic design fulfilling the needed functions	develop basic design of facility / process to fulfil functions needed and corresponding targets	
develop		characteristics needed and corresponding targets	facility / process basic design fulfilling the needed characteristics	develop basic design of facility / process to fulfil characteristics needed and corresponding targets	
design input requirements defined			design input requirements for facility / process for all stages of life cycle (planning, construction, use, end-of-life)	review information and assess compatibility and - in case of contradictions - find solution. compile in-situ condtions and loads, etc.	
product specifi-cation defined			product specification	perform design and develop product specification	
production specifi-cation defined			production specification	develop production process (incl. verification and validation)	
implementation fulfils specifications			facility / process implemented	activities related to implementation	
	hiring suitable contractor		tendering documents	prepare tendering	
			contractor selected; contract signed	manage tendering process	

Figure 9: Issues addressed ('boxes' in figure) in the stepwise development, illustrated for the first stages of the life-cycle. For more explanations, see text.

Colours of boxes: see explanation for Fig. 8

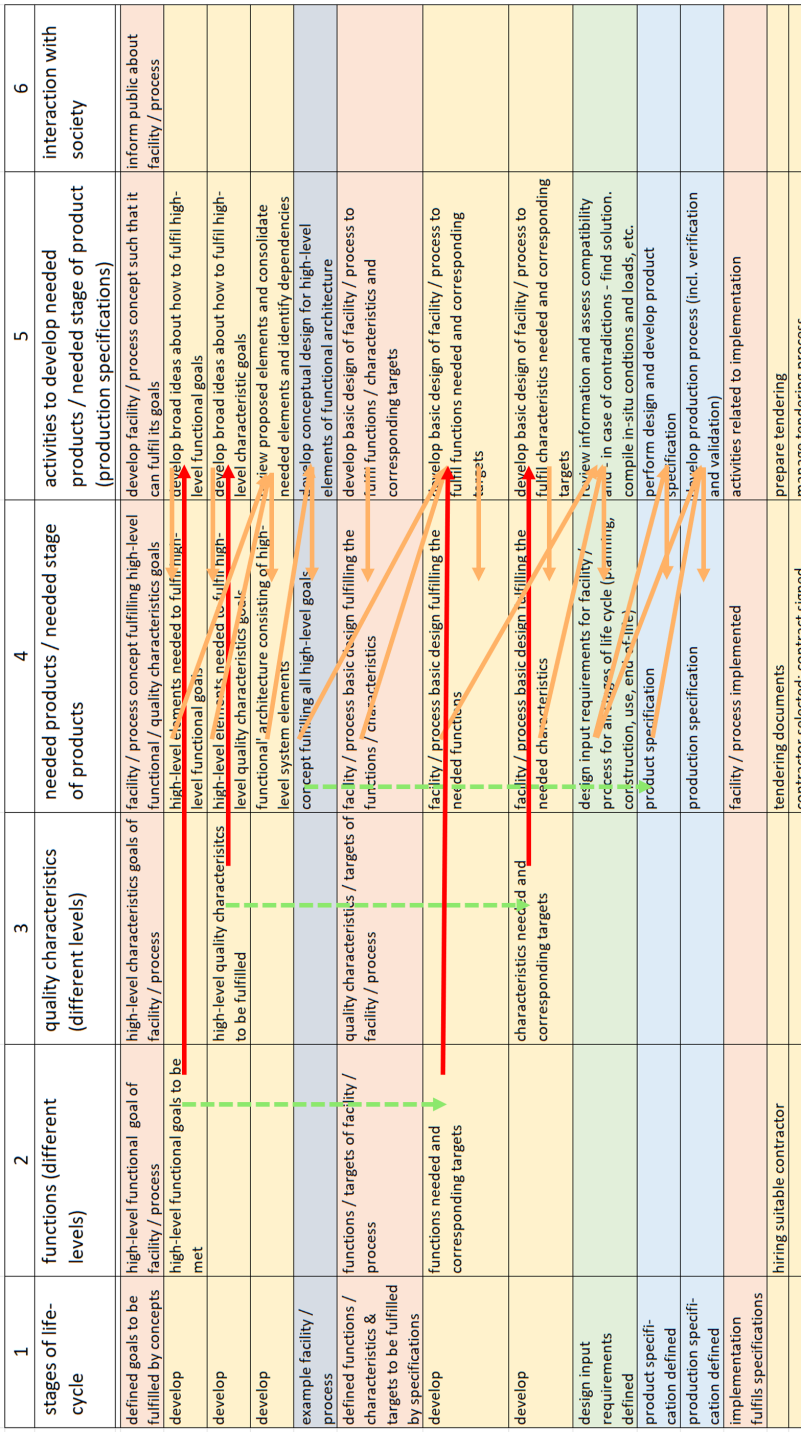


Figure 10: Issues addressed ('boxes' in figure) in the stepwise development, illustrated for the first stages of the life-cycle. The arrows show the workflow, as explained in the text.

For 'horizontal' table, arrows from left to right provide input for an activity that transforms the input into an output (yellow arrows from right to left). Red arrows are for goals and functions / characterisitcs, yellow arrows are related to using / producing (interim) results; green arrows are related to decomposition.

Colours of boxes: see explanations for Fig. 8.

Fig 11 and **Fig 13** are **more abstracted versions** of **Fig. 10** that show all stages of the life cycle for an element. **Fig 11** is for an early-stage programme with the focus being on the development of concepts and **Fig 12** for a more advanced programme with the focus of implementing an element, its use and its eventual end-of-life. Going from the top of the schemes in these figures downwards, the following activities / steps take place, with **Fig. 11** stopping at step 5.

- Step 1: **decomposition of the goals** in functions and characteristics with their targets. For an early-stage programme (Fig. 11), this is limited to derive functional and characteristic goals,
- Step 2: develop **design input** based on functions and their performance targets. For an early-stage programme (Fig. 11), this is limited to develop broad conceptual **design ideas**,
- Step 3: develop **design inpu**t based on characteristics and their quality targets. For an early-stage programme (Fig. 11), this is limited to develop broad conceptual **design ideas**,
- Step 4: identify / define **element** in functional architecture that has to fulfil the **allocated functions** and **characteristics** with their **targets**. For an early-stage programme (Fig. 11), this is limited to the identification of **high-level elements** with the **allocated functional and characteristic-related goals,**
- Step 5: develop '**design input requirements'**. For an early-stage programme (Fig. 11), this is limited to develop **conceptual designs** for the high-level elements,
- Step 6: at an advanced stage, develop **'design output specifications'** (product specification and production specification), this then allows implementation and use.
- Step 7: **implement element** according to product specification and production specification and document the successful implementation (incl. verification and validation),
- Step 8: **use element** according to the (updated) specification,
- Step 9: **manage 'end-of-life'** of element and bring elements into safe situation according to the (updated) specification.

Fig. 12 illustrates with a scheme the detailed information flow in the box 'activity' in **Fig. 11** and **Fig 13**.

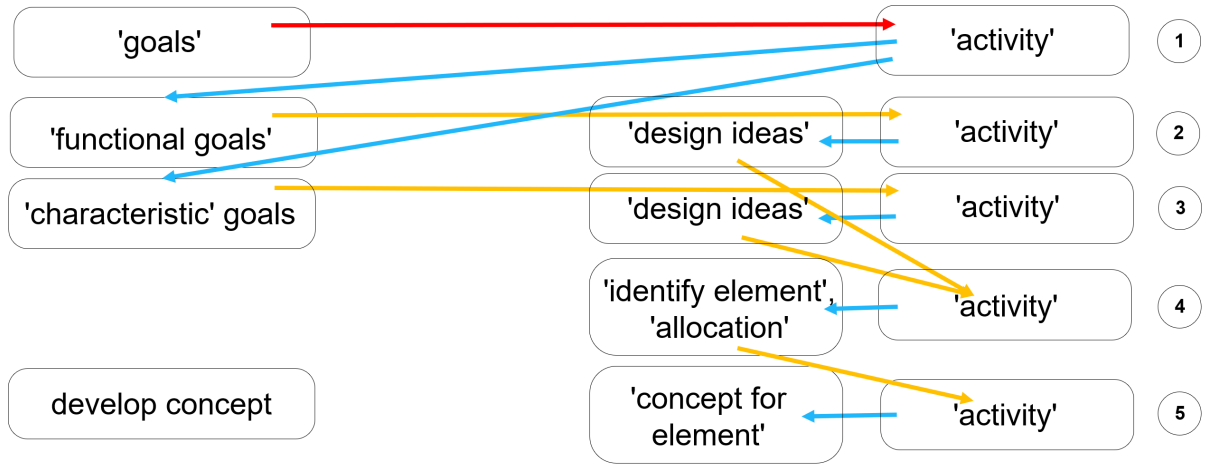


Figure 11: Scheme illustrating the process of developing the concept of a system / system element for an early-stage programme.
The numbered steps at the right side are explained in the text. For the arrows and their colours, see Fig. 13.

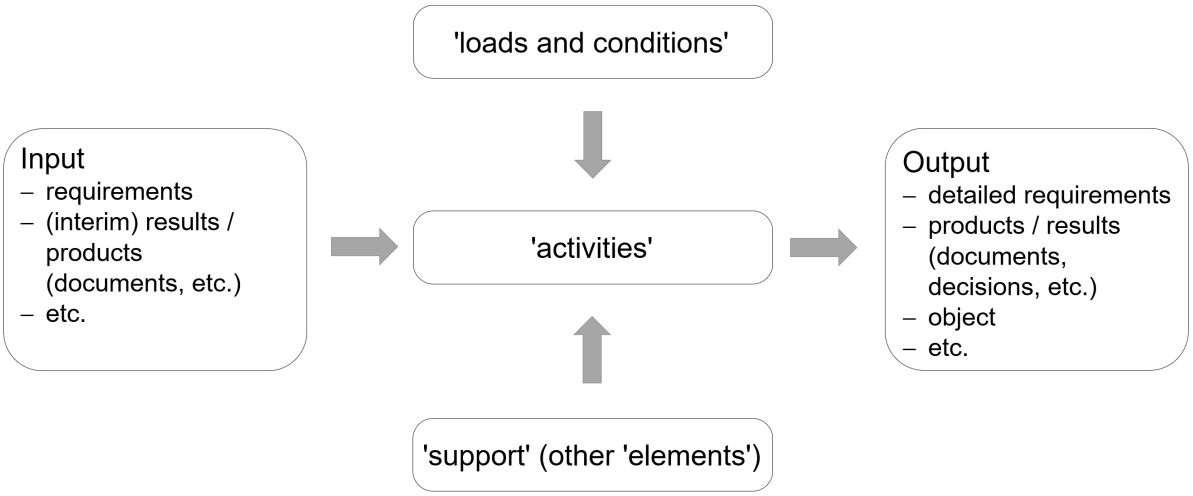


Figure 12: Scheme illustrating the information flow in the box 'activity' of Fig. 11 and Fig. 13.
The items mentioned are for illustration only – for more explanation, see the text, e.g. related to 'level 4a'.

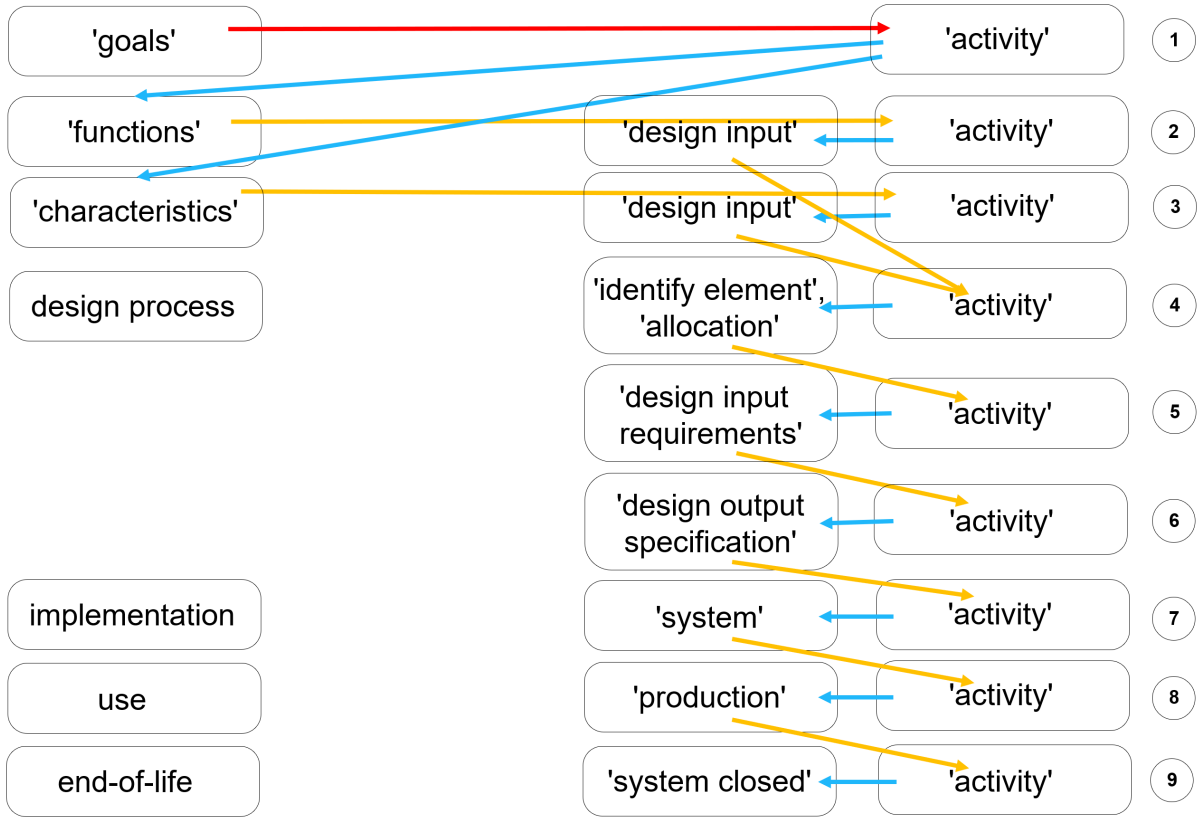


Figure 13: Scheme illustrating the process of implementing the different stages of the life-cycle of a system element.

The red arrow illustrates the use of the input provided by the 'external' stakeholders (their goals). This input is the starting point of the development of the system. The blue arrows point to the output of an activity that is based on the input of the 'external stakeholders (only for the first step) and on an input in the format of an interim result (yellow arrow).

The numbered steps at the right side of the scheme are explained in the text

References

Drucker P. (2001): The Essential Drucker – Management Challenges for the 21st Century, 2001.

EURAD (2024a): Developing, Using and Modifying Requirements Management System for a Generic System' (G-RMS). 2024

EURAD (2024b): Developing, Using and Modifying a Requirements Management System for Implementing a Disposal System - DS-RMS, 2024

International Council on Systems Engineering – INCOSE (2015): Systems Engineering Handbook – A Guide for System Life Cycle Processes and Activities, Forth Edition, INCOSE-TP-2003-002-04 – 2015, 2015.

International Council on Systems Engineering – INCOSE (2022a): Needs and Requirements Manual – Needs, Requirements, Verification, Validation Across the Lifecycle, INCOSE-TP-2021-002-01, Rev 1.1, 2022.

International Council on Systems Engineering – INCOSE (2022b): Guide to Verification and Validation, INCOSE-TP-2021-004-01, Version/Revision: 1.0, May 2022.

National Aeronautics and Space Administration – NASA (2020): NASA Systems Engineering Handbook, 2020

Appendix A. Glossary

architecture	the (functional) architecture identifies all elements of the system of interest that are needed to perform the functions and show the (quality) characteristics of the system of interest as specified by the requirements of the 'needs domain'. The (functional) architecture defines the interfaces between these elements and their hierarchical structure. The functional architecture is used to document the allocation of the requirements of the 'needs domain' to these elements. The elements act as black boxes (no design defined yet).
(quality) characteristic	is related to a requirement ((quality) characteristic requirement); it includes characteristics such as reliability, availability, repairability, etc.
constraint	a requirement that limits the solution space beyond what is necessary for meeting the given functional requirements and (quality) characteristic requirements; constraints are often externally imposed.
element	element (of the waste management programme / system) applies to all issues used to implement the system of interest. The elements provide the 'means' to implement the system of interest.
end-of-life	last stage of the life cycle after the stage of using a product has come to an end.
external entity	is outside of the system of interest but interacts with the system of interest.
function	a task, action, activity or behaviour that must be performed to achieve a desired outcome.
goal	instead of goal (for a high-level requirement) also the terms 'objective' or 'principle' are sometimes used.
level 0	needs, expectation and goals (high-level requirements) expressed by the external stakeholders to be fulfilled by the system of interest.
level 1	goals for the different phases / different stages of the life cycle derived by the internal stakeholders to fulfil the 'level 0' requirements.
level 2a	functional requirements derived from decomposing the 'level 1' requirements.
level 2b	(quality) characteristic requirements derived from decomposing the 'level 1' requirements.
level 3a	performance target related to specific functional requirement.
level 3b	(quality) target related to specific (quality) characteristic requirement.
level 4a	design input requirements.
level 4b	design output specifications.
level 5	production specifications.
level 6	documentation of implemented product.
living document	document to be updated when needed.
loads and conditions	acting on a system element; is considered in the development of both the design input requirements and design output specifications.

means	objects, activities with their deliverables / products / decisions, and other measures with their achievements / situations (e.g. stability) that form the elements of the functional architecture.
needs domain	defines the 'why', 'what', 'when'. It contains all the 'level 0 to level 3' – requirements.
outside world	elements outside of the system of interest but potentially relevant for the system of interest.
problem	needs, expectations and goals of the 'external' stakeholders.
product	can be an object (building, equipment, etc.), a document, a contract, a decision, etc. Instead of the term 'product' also the terms 'element', 'system', 'system element' are used.
service provider	supports the 'internal stakeholder' with implementing the system of interest. The (external) service provider is available on the market.
solution domain	contains all the system elements that make up the (total) system; the system elements make up the 'means' (with the objects, activities with their deliverables and other measures with their achievements) to achieve the 'level 0'-goals of the 'needs domain' The 'solutions domain' defines the 'who, 'with whom' (dependencies), 'when, and 'how'.
stakeholder	'external' stakeholder: is not involved in the development of the system of interest, but has a strong interest in its implementation and has the corresponding needs, expectations and goals (sometimes summarised as the 'problem' statement). 'internal' stakeholder: has the task to implement the system of interest.
stage	(or status) defines where in the life cycle an element of the waste management system is; the stage / status can be: 'initial thoughts / planning', 'production / construction / building',' using the system / system element / product', 'decommission / dismantle / close'.
system	items fulfilling the defined requirements, consists normally out of several elements. Sometimes, also the term 'system of interest' is used.
the way of thinking	described by the methodology to be applied.
V-model	verification of 'having done the things right' and validation of 'having done the right things' are in the literature sometimes represented as the 'V-model', where each verification-step and each validation-step is linked to the corresponding requirement as defined at the outset of the implementation process.
validation	validation includes the evaluation whether 'the right things have been done'; thus, it is evaluated whether the needs, expectations and goals of the (external) stakeholders are met; validation applies to the whole system of interest or to its sub-systems.
verification	verification includes the evaluation whether 'the things have been done right'; thus, it is evaluated whether all requirements are fulfilled; verification applies to sub-systems, components, etc (only part of the system of interest); however, all sub-systems, components must undergo verification.

voluntarism	describes an approach of site selection where municipalities must volunteer to be considered as a municipality that potentially will host a facility. The approach by providing a 'veto-right' to municipalities falls in the same category.
waste management programme:	contains all elements (processes, facilities) that are needed to manage the waste 'from cradle to grave'. It consists of several waste management systems related to collection of raw waste, waste characterisation, waste treatment / solidification / packaging, handling / transportation of waste, interim storage of waste, waste disposal.
why, what, when, who and how:	the cornerstones of the requirements management process – the 'why' captures the 'needs', 'expectations' and goals of the 'external stakeholders', the 'what' defines the functional requirements and the (quality) characteristic requirements and their targets, the 'when' defines the phase when then 'what' needs to be achieved, the 'who' defines the element (as part of the functional architecture) that has to fulfil the allocated requirements and the 'how' is defined by the 'design input requirements' and the 'design output specification' (together: the 'product specification') and the 'production specification'.

Appendix B. Abbreviations used

DGR	Deep Geological Repository. Normally used for the disposal of SF, HLW and LL-ILW
DS-RMS	Document 'Developing, Using and Modifying a Requirements Management System for Implementing a Disposal System' (EURAD document)
EURAD	European Joint Programme on Radioactive Waste Management
G-RMS	Guidance document 'Guidance on Developing, Using and Modifying a Generic Requirements Management System' (EURAD document)
INCOSE	International Council for System Engineering
P-x	Phase-x
RDD	Research, Development, Demonstration
RMS	Requirements Management System
T-x	Theme x or life-cycle stage x
WMP-RMS	Guidance document 'Guidance on Developing, Using and Modifying a Requirements Management System for Waste Management Programmes with their Different Systems' (this document)