Digital Design Professional

Foundation Level

Handbook

Kim Lauenroth, David Gilbert,
Michael Kemper, Norbert Seyff,
Melanie Stade, and Marcus Trapp

2.0.0 | January 31, 2024

Acknowledgements

Our special thanks go to Bernd Aschauer, Michael Burmester, Nikola Eger, Thomas Geis, Saskia Hehl, Thomas Immich, Jens Kawelke, Karsten Lehn, Rolf Molich, Knut Polkehn, Kurt Schneider, Lars Sonnabend, Hans-Jörg Steffe, Stefan Tilkov, Marcus Winteroll, Denis Woyke, and many others for their valuable feedback on the contents of the first version and second version of the handbook.

In particular, we would like to thank Martin Glinz for his great commitment and continuous support and many fruitful discussions for the success of the updated handbook on DDP V2.0.

The German version of this handbook was published on September 20, 2023, by the Council of IREB e.V. upon recommendation by Martin Glinz.

We thank everyone for their involvement.

Scope of this handbook

This handbook introduces the profession of Digital Design based on the syllabus for the Digital Design Professional (DDP) at foundation level. It complements the syllabus and addresses three groups of readers:

- Students who want to learn about Digital Design and who take the certification exam can use this handbook as a companion book to training courses offered by training providers, as well as for self-study and individual preparation for the certification exam.
- Practitioners who want to learn about Digital Design can use this handbook as a reference guide to set up a process for building digital products and to find appropriate tools and templates for their daily work.
- Training providers who offer training for the DDP foundation level can use this handbook as a supplement to the syllabus to develop their training materials or as a study text for the participants in their training courses.

This handbook provides a link between the syllabus, which lists and explains the learning objectives, and relevant literature on Digital Design. The structure of the handbook matches the structure of the syllabus.

The authors and IREB e.V. have invested a significant amount of time and effort in preparing, reviewing, and publishing this handbook. We hope that you enjoy studying it. If you find any errors or have suggestions for improvement, please contact us at ddp@ireb.org.

Version History

Version	Date	Comment
1.0.0	2021-06-01	First version of the DDP handbook
2.0.0	2024-01-31	Second completely revised version of the DDP handbook

Table of contents

1 Motivation for Digital Design

This chapter discusses the motivation for Digital Design as a profession for designing digital solutions and explains why the new profession is needed to design holistic digital solutions.

Furthermore, the chapter presents the Digital Design Professional (DDP) as a new education program for Digital Design. In this handbook, we use the abbreviation DDP for Digital Design Professional.

1.1 A Profession for the holistic design of digital solutions

EO 1.1	Justify the need for a dedicated design profession for building successful digital solutions (L2)

A prerequisite for understanding the need for a new profession for designing digital solutions is an understanding of the increasing importance of digital technologies.

1.1.1 Three stages of the use of digital technologies

The development of digital technology is changing the nature of digital solutions and can be characterized by the following stages (cf. [Bloo2018]):

- *Data digitalization* is the use of digital technology to solve problems with digital data that had previously been solved with non-digital data.
- *Process digitalization* is the use of digital technology to create solutions and business processes that are not feasible with non-digital means.
- *Digital transformation* is the use of digital technologies to create solutions that have an impact on people and society by changing people's habits and lives through digital means (transformation of ecosystems).

The early days of digital technologies were all about storing and processing data. Large businesses and organizations such as banks and insurance companies used computers to store and manage customer, account, and contract data. Thus, digitized data replaced the data previously kept on paper.

With the increasing spread of digital technologies in the private sector, the functions of these technologies were now no longer limited only to companies. A major step in this development was the worldwide spread of the Internet. Its broad availability created completely new opportunities to implement existing solutions and business processes based on digital technologies. Clear examples of process digitalization are online banking and online shopping. At this stage, existing analog processes are partially or completely replaced by digital processes based on digital technologies. Colloquially, the first two stages are often summarized under the term *digitalization*.

With the increasing availability and acceptance of digital technologies, opportunities arose beyond digitalization to realize innovative and new processes and solutions that would not have been possible before and that have the potential to change people and society. Clear examples of this development are social networks and their influence on our communication behavior. At this stage, it is no longer merely the case that existing solutions and processes based on digital technologies are replaced, rather that processes and solutions based on digital technologies are changed. This stage is therefore also referred to as *digital transformation*.

The stages described show that digital technologies are evolving from a tool for realization to a design tool that enables the creation of new and innovative solutions. Therefore, in the context of Digital Design, the term digital solution is used to emphasize this holistic approach to designing a solution, rather than preferring one of the three stages. The term *digital solution* is defined as follows:

> Digital solution: A socio-technical system that solves a real-world problem with digital means.

The understanding that a socio-technical system is a system that spans software, hardware, and people, as well as organizational aspects and the environment, follows the understanding of systems from general systems theory. This means that Digital Design is about shaping technical (digital) systems and about shaping socio-technical systems (the digital solution) with digital material (see chapter 3). The combination of these perspectives is an essential aspect of the holistic approach to Digital Design.

1.1.2 Expansion of the design scope with each stage

The three stages are not fully separable, that is, a digital solution cannot always be clearly assigned to one stage. Likewise, the three stages should not be used as a rating or evaluation. The intention and the design scope that arise at a stage are more important than the classification of a solution into a stage.

In the stages of data and process digitalization, data, solutions, and processes that already exist and have been understood are realized in digital technologies. Consequently, in such projects, there are clients who know what they need and stakeholders who know their needs and what problems a digital solution should solve.

The challenge lies in capturing and understanding what needs to be built, and this challenge is overcome through successful *requirements engineering*: identifying the right stakeholders, eliciting requirements from stakeholders, and consolidating, documenting, validating, and managing those requirements. With the requirements as a basis, a digital solution can be realized systematically and efficiently. Such approaches to building a digital solution are referred to as *requirements-driven* approaches.

In this situation, the design scope relates primarily to the implementation of the solution, for example, to the design of a good and usable user interface or to the mapping of the processes in the solution. Specialized roles and disciplines have developed for this (e.g., interaction design and UX design).

However, with the stage of process digitalization, the degree of innovation and thus the design scope also increase considerably. A new process (e.g., buying a book) based on a digital solution is different from the original process (e.g., visiting a bookstore is completely eliminated). At the digital transformation stage, the aspect of the innovative and the new comes even more to the fore, that is, there are no immediate role models and a rather vague understanding of the intended solution on the part of the clients and stakeholders. The methods and tools of requirements engineering therefore often do not go far enough in such cases.

In these situations, the question of what to build cannot be resolved merely by surveying stakeholders. A *design-driven* approach to building a digital solution is necessary: *Digital Design aims to* explore what could be done to design and shape target images for digital solutions together with clients and stakeholders, and finally, to prescribe the form, function, and quality of the digital solution.

In this situation, the design scope is extended to the entire solution and its environment. It is no longer just about mapping what already exists in digital technologies, but about designing something new based on digital technologies.

1.1.3 Digital Design as a profession for the design of digital solutions

The increasing freedom for design and the accompanying innovative character of digitalization and digital transformation require a changed approach: requirements-driven approaches must be complemented by design-driven development of digital solutions that integrates the potential and limitations of new technical possibilities into the design process at an early stage.

Digital Design [Bitk2017] is a profession that addresses this change and aims to clearly state and strengthen the skills required to holistically design and build digital solutions [LBGH2018]. Digital Design is defined as follows:

> Digital Design: The creative and holistic design of digital solutions.

Digital designers are people who have competence in Digital Design. Digital Design means understanding digital as a shapeable material (see chapter 3). This understanding goes beyond a pure technical understanding of digital technology, with the aim of combining design skills and technical skills similar to an understanding promoted by industrial design and building architecture.

Digital Design connects creative design with the design and building of digital solutions by taking a holistic view of the technical possibilities of digital material, the economic aspects, and the current or future needs of people. This understanding of creative design emphasizes that Digital Design goes beyond the prevailing technical understanding of design in software development, which primarily serves to design a solution based on given requirements.

The term *design* has multiple meanings and can be used to refer to both design as an *activity* and design as a *result*. Digital Design, with the idea of understanding digital as a shapeable material, understands design in all its facets, that is, both design and technical, artistic, creative, and explorative facets. For a good balance between readability and precise wording, in this handbook, we distinguish between design as an activity and design as a result. When we speak of design as an activity, we use the term design as a noun and (to) design as a verb. When we speak of the results of the design activity, we use either the term design as a generic term for these results or their specific names (e.g., prototype, design concept, or visual design), unless the meaning is clear from the context.

Digital Design shapes new digital solutions and optimizes existing ones by:

- *Designing the goals, benefits, and means of a digital solution together:* this reflects the holistic view of the solution and system and the ability to cooperate with all other activity areas
- *Designing on both the small and the large scale:* whereby, the large scale refers to the perspective of a digital solution at the solution and system level, and the small scale refers to the design of the elements of a digital solution in detail
- *Designing perceivable and underlying aspects of a digital solution together:* this refers to the fact that designing the perceivable form, function, and quality of a digital solution requires a profound understanding of the underlying form, function, and quality that enable the perceivable aspects
- *Designing material and immaterial aspects of a digital solution:* this refers to the fact that a digital solution often consists not only of software but also of physical parts

Digital Design means taking responsibility for the holistic design (in the sense of the actual activity and result of the design) of a digital solution and managing the building process (see chapter 3) of a digital solution from a design perspective. This also includes designing and optimizing the design as part of the building process as well as intensive cooperation with all other activities of the building process.

We present the competence areas of Digital Design in detail in chapter 2.

1.1.4 Ten principles of good Digital Design

In general, a profession is defined by its methods, techniques and values. To understand a profession, it is also important to understand the attitude it is supposed to represent. The ten principles of good Digital Design presented in the Digital Design Manifesto [LBGH2018] define the principles that guide the values of Digital Design and the fundamental attitude of people working in the field of Digital Design.

These ten principles are:

- P1 – Good Digital Design is useful and usable.
- P2 – Good Digital Design is elegant and aesthetic.
- P3 – Good Digital Design is evolutionary.
- P4 – Good Digital Design is exploratory.
- P5 – Good Digital Design focuses on the person as a whole.
- P6 – Good Digital Design anticipates the effects of its results.
- P7 – Good Digital Design respects data protection and data security.
- P8 – Good Digital Design is sustainable and creates sustainability.
- P9 – Good Digital Design appreciates analog and digital means equally.
- P10 – Good Digital Design uses digital means only where this is necessary.

It is important to understand that the ten principles are not a checklist. Instead, they describe an attitude towards digital material and the design of good digital solutions. They always refer to good Digital Design and thus to both the design process and the result. This is important because we think that the process and the result are inseparable.

1.2 Digital Design is not a role, but a profession

EO 1.2	Know that Digital Design is a profession and not a role in the building process (L1)

The difference between a role and a profession is important for understanding Digital Design:

- A role is a position that a person can take in a given situation. A role is defined by its tasks, rights, duties, and responsibilities.
- A profession is an occupation that requires specialized education.

The holistic design of digital solutions is an activity that is comparable in its breadth and skill sets to the professions of architecture in the construction industry and the design of products in industrial design. It is therefore only logical that Digital Design should be understood as a profession and not just as a role in a process model.

During the building process for a digital solution, digital designers can work in various roles that are related to the activity area design (e.g., product owner, business analyst, requirements engineer, usability engineer). However, due to the broad scope of Digital Design, certain roles will require additional specialized training and also collaboration with other experts to achieve good Digital Design.

1.3 The Digital Design Professional as an introduction to the profession of Digital Design

The Digital Design Professional (DDP) is an entry-level training program for the profession of Digital Design. Accordingly, a DDP is a person who is considered competent in the field of Digital Design. This foundation level handbook aims to provide a broad overview of Digital

Design and an introduction to the holistic design of digital solutions in terms of the ten principles (see above).

The core advantage of the DDP certification is that it provides a holistic understanding of digital solutions and a broad understanding of the building process for digital solutions. Furthermore, the DDP certification provides a broad competence in the design of digital solutions, including the necessary material and cross-cutting competencies. This broad knowledge from the DDP education especially supports experts with specialized training to enable them to better grasp and structure a digital solution in its entirety, and to integrate their personal strengths into the entire building process for a digital solution more easily.

In addition, the DDP program is useful for other professions and roles in the context of digital solution development to enable a better understanding of the design of digital solutions.

2 The three competence areas in Digital Design

In this chapter, we provide an overview of the competence areas in Digital Design. Furthermore, the chapter shows how you can assess your own competencies with regard to Digital Design.

The Digital Design competency profile is described as a Pi-shaped profile [Bitk2017] and is inspired by the Greek letter **π** as a symbol. The left leg of the Pi represents design competence, the right leg represents understanding digital as a material (material competence). The upper part of the Pi represents the cross-cutting competencies that enable interdisciplinary work in Digital Design.

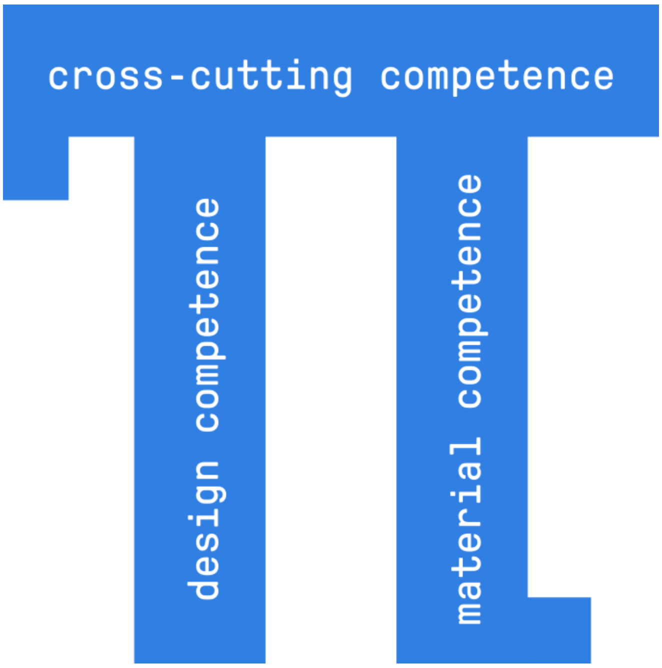

Figure 2.1 - Digital Design competence profile

The combination of design and material competence is an important factor in many design disciplines. For example, building architecture requires competencies in building materials (e.g., wood, concrete, steel) and industrial design requires competencies in the materials from which products are made (e.g., plastic, glass, metal). Material competence supports the ability to design with these materials and also supports working effectively with the appropriate disciplines on an equal footing.

2.1 Design competence

EO 2.1	Explain design competencies as part of the pi-shaped profile of Digital Design (L2)

In Digital Design, design competence is understood as the capability to design. Design is often misunderstood as merely designing the external form of an object. However, design goes much further and is understood as "an intervention in the environment that leads to its intentional change" (cf. [ErMa2008]).

At the core of this understanding of design competence is the claim that design is a creative and holistic activity, ranging from the creation of a new idea to the actual transformation of the environment.

This claim of being holistic is strengthened by the fact that design is understood as a service that designers perform for the benefit of their clients and other stakeholders (e.g., the client's customers, users of a system) (cf. [NeSt2014]).

Design competence is specified more precisely from this perspective as follows (cf. [ErMa2008]):

- Analyze and understand the need for change together with all relevant stakeholders.
- Understand and formulate the problem or goal for the change together with all relevant stakeholders.
- Design and evaluate an appropriate change to ensure that the problems defined are solved or the goals formulated are achieved.
- Accompany the realization of the change and make the change effective.

Applied to Digital Design, this means that the change is achieved through the design and evaluation as well as the realization of a digital solution. Chapters 7 to 11 look at a wide variety of aspects of design competence—in particular, designing a solution at different levels, working with prototypes, and integrating design work into the building process.

However, this understanding of design competence does not mean that design is the only competence required to realize change. Rather, the opposite is the case. Above all, design competence also means understanding that many other competencies are required to achieve environmental change. Design competence forms the content bracket with a view of the change and its realization and needs many other competencies to be successful (cf. [NeSt2014]). The cross-cutting competencies in Digital Design create the necessary conditions for effective collaboration with other disciplines (see section 2.3).

2.2 Material competence

EO 2.2	Explain material competencies as part of the pi-shaped profile of Digital Design (L2)

In Digital Design, material competence refers to digital technologies. In terms of Digital Design, digital technologies cover a broad spectrum of perceivable capabilities (e.g., end devices and user interfaces), but also hidden skills (e.g., storing and processing large amounts of data). Digital technologies include software, hardware, and other technologies and materials related to digitization (e.g., QR codes, RFID tags).

Material competence means knowledge of the skills, limitations, and prerequisites of digital technologies as well as knowledge of the constraints for their use in a digital solution and of the effects caused by their use. The constraints include, in particular, economic issues (e.g., acquisition, usage, or production costs) and legal issues (e.g., licensing conditions). The effects of their use include, in particular, social issues (e.g., data privacy and data security), as well as sustainability issues (e.g., energy consumption, ecological footprint). Furthermore, material competence also includes the awareness of keeping your own technical knowledge continuously up to date in order to be able to distinguish technical hypes from substantial technical advances.

Material competence does not necessarily include the ability to realize solutions with these materials (e.g., programming in a programming language). However, acquiring realization competence in a technology is one possible way to acquire knowledge about the limitations and skills of that technology. With regard to innovative technologies, this path can be useful to experience the skills and limitations of a technology for yourself. In principle, many technologies today offer a wide range of documentation and information sources for acquiring the necessary knowledge yourself. We introduce the fundamentals of designing with digital material in chapter 3. In addition, chapter 6 provides an overview of important digital technologies as a foundation for building up material competence.

2.3 Cross-cutting competence

EO 2.3	Explain cross-cutting competencies as part of the pi-shaped profile of Digital Design (L2)

Cross-cutting competence is required in Digital Design to enable an understanding of other disciplines and interdisciplinary collaboration with all disciplines that need to interact to build and holistically design a digital solution. The concrete form of this cross-cutting competence depends in particular on the targeted digital solution and its context. In general, cross-cutting competencies can be divided into three dimensions:

- *Project dimension:* planning of the building process for a digital solution, management of the building process for a digital solution, including all activities, time, and budget
- *Product dimension:* competencies in the industry or field of expertise of the digital solution, competencies for developing a short-term and long-term strategy for the development of the intended digital solution (market, customer segments, human factors, business model)
- *Social dimension*: management of stakeholder expectations and needs as well as the cognitive process of stakeholders, finding the right people and skills for the activity at hand (diversity in the team, skills, needs, fears)

For a DDP at foundation level, the following cross-cutting competencies are relevant:

- Understanding of the building process of digital solutions and of the interaction of the relevant disciplines (project dimension, see chapter 3 and chapter 5)
- Understanding of the integration of Digital Design in different process models (project dimension, chapter 12)
- Awareness of the relevance of teamwork and the social dimension of the building process (social dimension, chapter 13)

Other examples of cross-cutting competencies beyond the foundation level include:

- Understanding of different digital business models (e.g., pay-per-use) as part of the product dimension
- Understanding of human factors of digital solutions as part of the social dimension
- Understanding of change processes/change management as part of the social dimension

Basically, in Digital Design, it is important to be willing and motivated to learn relevant cross-cutting competencies.

3 Fundamentals of designing digital solutions with digital material

In this chapter, we explain the fundamentals of designing with digital materials. First, we give a fundamental view of digital material in section 3.1. We then specify this view in more detail with regard to digital solutions in section 3.2 in order to obtain a working model for designing digital solutions in section 3.3. Finally, in section 3.4, we use this working model to explain Digital Design's attitude towards digital technologies as shapeable material.

This step-by-step introduction is intended to support beginners in particular in gaining initial access to digital technologies as a basis for designing digital solutions.

3.1 Understanding digital as a material for data structures, data flows, and data transformation

EO 3.1	Explain digital as a material for creating data structures, data flow, and data transformation (L2)

Looking at the different stages of the use of digital technology (see section 1.1.1) and the wide range of technological possibilities, we are confronted with an overwhelming mass of options and capabilities (cf. [Kell2016]). Knowing and understanding all of these technologies seems like an almost impossible task. This is probably also true for the concrete technologies, since a very wide range of technologies already exists, new technologies are developed every day, and thus the number of possibilities also grows every day.

Nevertheless, the core potential of digital technologies can be understood very well independently of any knowledge of specific technologies. At the core of digital technologies is the term digital, which can be defined as follows:

> Digital (noun): The structure, flow, and transformation of binary data.

Following this definition, digital is about the structure, flow, and transformation of binary coded data. In this definition, structure means the structure and thus the contents of the data stored in a system. The flow of data means the transport of data within a system as well as between systems. Transformation is the generic term for all forms of calculation and modification of data in a system.

For example, an online banking solution can be understood as storing customers' bank accounts as data structures in the banking system. The flow of data occurs, for example, when customers display their bank account using the bank's app (the account data flows from the server to the app and is displayed to the customers). Similarly, transfers generate a flow of data (the transfer is recorded by the customers in the app and flows to the banking system). Finally, the transfer is executed by the banking system. This execution can be understood as a transformation of data, as account balances are changed and new entries are created on the accounts.

This abstract example works without concrete technologies; for example, there is no definition of how the banking system works or on which end device the banking app runs. Concrete digital technologies provide various means to realize systems that enable the structure, flow, and transformation of data. Depending on the specific use case, the structures, the data flows, or the transformations, as well as the technologies that realize them, can be very simple or very complicated. These concrete means (e.g., an app, a network, or a data center) can be understood as material to realize a solution.

As a first step, it is important to realize that at an abstract level, a digital solution can be understood and described exclusively in terms of data structures, data flows, and data transformations along different systems.

3.2 The FFQ model for structuring digital solutions

EO 3.2	Explain thinking in terms of form, function, and quality at a perceivable and underlying level as a model for digital solutions and digital technology (L2)

In the following, we make the definition of digital more precise for practical design work in order to obtain concrete aspects of a digital solution that can be designed. For this purpose, we present the *FFQ model*, which is based on established concepts of physical product design[1]: form, function, and quality (FFQ). The FFQ model (cf. [Laue2019]) structures digital solutions as systems along three pillars at two levels.

The three pillars of a solution are:

- *Form*: the elements of a system and the relationships between the elements that make up the system's structure
- *Function*: the capabilities of an element, a combination of elements, or the capabilities of the system as a whole
- *Quality*: the degree to which defined quality characteristics are met by an element, a relationship between elements, or a capability of a system

The two levels of a solution are:

- *Perceivable level:* form, function, and quality that can be perceived or directly experienced by stakeholders
- *Underlying level:* form, function, and quality that are hidden from stakeholder perception and enable the perceivable level

With the help of the three pillars and two levels, we can structure the diverse aspects of a digital solution and thus make them easier to communicate, document, and access in collaboration with stakeholders. In chapters 8, 9, and 10, we look at these aspects in more detail.

---

[1] For an overview of the meaning and use of the terms form, function, and quality in design, please refer to [ErMa2008].

The separation between the perceivable and underlying level is an essential aspect of understanding the capabilities of digital (as a shapeable material) and digital technologies. A perceivable capability of a digital solution (e.g., surfing the Internet) can be realized with a wide variety of underlying communication technologies (e.g., WLAN or mobile communications). We provide more details on this understanding in section 3.4.

Table 3.1 - The model of form, function, and quality at the perceivable and underlying level (FFQ model)

	Form	Function	Quality
Perceivable	▪ Elements of a system that perceivably process data ▪ Perceivable relationships between elements and users ▪ Structure of interactions between elements and users ▪ Perceivable relationships between elements of the system ▪ Perceivable relationships between elements and the environment	Perceivable capabilities of the elements resulting from the interaction between: ▪ Elements and users ▪ Different elements ▪ Elements and the environment	▪ Qualities of perceivable form, i.e., elements, relationships, or interactions ▪ Qualities of the perceivable function, i.e., the perceivable capabilities
Underlying	▪ Elements of a system hidden from stakeholders ▪ Hidden relationships between the elements of a system ▪ The technical structure of the data stored in the elements	Hidden capabilities of the elements resulting from the interaction between: ▪ Different elements ▪ Elements and the environment	▪ Qualities of the underlying form, i.e., the elements, the relationships, and the data structures ▪ Qualities of the underlying function, i.e., the hidden capabilities

The practical benefits for understanding digital (as a shapeable material) and digital technology become clear when the model is applied to examples. In the following, we revisit the online banking example from section 3.1 and explain it using the FFQ model.

The banking server, together with the customers' banking apps and the servers of other banks, can be understood as a system and thus as a form. This form is underlying because the structure of this system cannot be perceived. The user interface of the banking app for displaying account data and entering transfers represents a perceivable form. Entering a money transfer is a perceivable function offered to customers via the banking app.

However, the execution of the money transfer on the banking server is an underlying function, as its execution is not perceivable. Only the effects (change of the account balance, new postings) can be seen through the user interface as a perceivable form. A perceivable quality in this example is the speed of execution of a transfer. An underlying quality is, for example, the strength of encryption of the communication between the banking app and the server.

This example shows that making a digital solution more specific by thinking in terms of form, function, and quality at two levels is helpful and makes it easier to structure and design the complexity in a first step[2].

For the practical design of a digital solution, section 5.2 presents a more detailed structuring that divides a digital solution into three levels.

3.3 Design in the tradeoff between goals and constraints

EO 3.3	Explain the general structure of design work for digital solutions in relation to goals and constraints (L2)

In section 0, we explained that Digital Design is all about creating intentional change through a digital solution. Two aspects can be distinguished for the design work. On the one hand, the desired change in the sense of a goal, and, on the other hand, the solution actually designed in the sense of a realized change. Both goal and solution are formulated and developed together with relevant stakeholders. In principle, there are always constraints in design work in the sense of laws, standards, or other restrictions that have to be taken into account when designing a solution (cf. [NeSt2014]).

To design a solution, both the goal and the constraints to be considered must be known, that is, the design of a solution must consider the tradeoff between goals and constraints (see Figure 1.3).

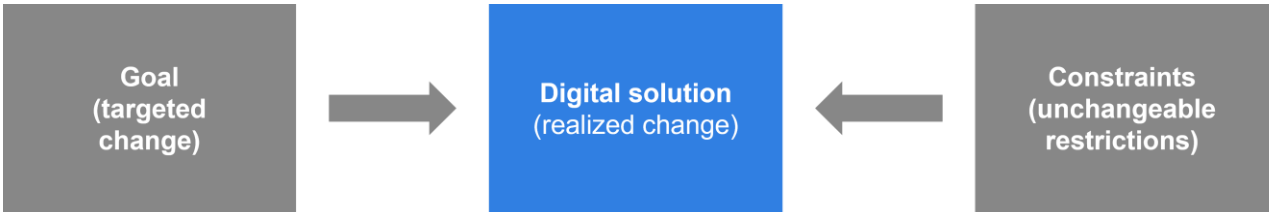

Figure 1.3 - Design work must consider the tradeoff between goals and constraints

To approach the design of a digital solution with digital technologies, the FFQ model from section 3.2 is integrated into Figure 1.3. Combining the three pillars and the two levels, we get a basic working model for understanding design work for digital solutions (see Figure 3.2). Performing design work on digital solutions means designing an appropriate form, function,

[2] There are also a number of frameworks for the structured handling of quality with various advantages and disadvantages (cf. [GSBF2023]). However, this topic goes beyond the foundation level and is not discussed further here.

and quality, at the perceivable and underlying level, taking into account relevant constraints to realize the goal (the desired change).

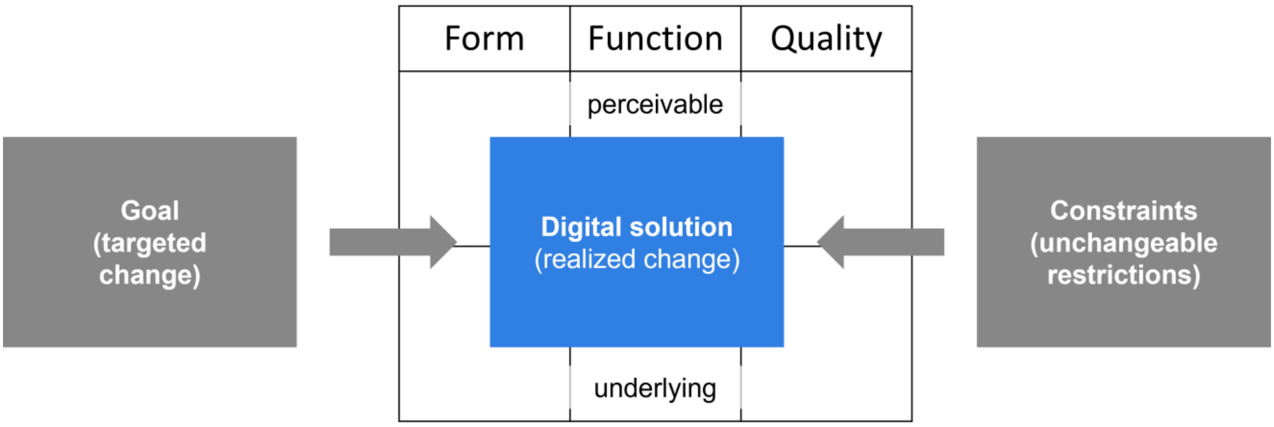

Figure 3.2 – Working model for the design of digital solutions

From this working model, we can derive immediately that design work is inseparably linked to competencies in dealing with requirements, because the systematic analysis and elicitation of goals and constraints is a core domain of requirements work (e.g., in requirements engineering [GLSB2022] or usability engineering [CPUX2022]).

This working model targets the content aspects of design work and not the process. A typical misunderstanding in this context is the assumption of a top-down process in design, that is, as a first step, goals and constraints are defined in order to design a suitable solution on this basis (cf. [NeSt2014]). This is not the case!

We address the topic of design work as a process further in chapter 7.

3.4 Understanding digital technology as shapeable material

EO 3.4	Explain the mindset of understanding digital technologies as material for designing digital solutions (L2)

The working model, and in particular the FFQ model from the previous section, is an important means for understanding digital (as a shapeable material) and digital technology capabilities at an abstract level.

With regard to technology, requirements-driven approaches (see section 1.1.2) can be understood such that a digital solution is designed abstractly based on the FFQ model (or other models) in order to identify and use suitable technologies for realization in a subsequent step.

In contrast, understanding digital as a shapeable material means that knowledge about the capabilities and limitations of existing technology must be systematically considered during the initial design of a solution. In this way, the capabilities and potential of technology can be used at an early stage to design the desired vision or goal.

In terms of Digital Design, this attitude means systematically acquiring knowledge about the capabilities and limitations of technology. This is the only way to apply this knowledge at an early stage in the building process.

The model of form, function, and quality from the previous chapter can also help in this view to enable you to look at, understand, and acquire knowledge about technologies.

Let us look at simple examples of this as well. Currently, machine learning technology is being touted in many circles as the essential digital technology of the future. There are various methods and algorithms for solving problems with the help of machine learning (e.g., image recognition, language processing, or knowledge aggregation). From the perspective of the FFQ model, machine learning is an underlying function technology—that is, machine learning provides new ways to process, analyze, and transform data. These capabilities can be used to perform multiple tasks in digital solutions and to provide innovative features and capabilities to users. The first examples of this are translation tools for texts and spoken language.

To understand digital technology as a shapeable material, it is essential to look at and experience the technologies themselves. The easiest way to do this is to use the technologies in solutions that have already been realized and also that have already been realized. When exploring technologies, you should always look at them from the perspective of form, function, and quality to not only experience them in a structured way, but to also systematically understand their capabilities.

A deep understanding of digital material, that is, technology and technological developments, offers important benefits from a Digital Design perspective:

- *Avoidance of unrealizable solutions:* A well-grounded understanding of available technologies prevents the definition of unrealizable goals and constraints, or of unachievable form, function, and quality in a digital solution.
- *Inspiration for novel solutions:* Innovative technologies offer capabilities that may enable novel aspects of a digital solution or a completely new digital solution. Examples are new applications based on the application of artificial intelligence (AI).
- *Substantial communication with software experts:* Often, specialists are needed to design and develop parts of a digital solution. Knowledge of the respective technology domains enables communication with these experts. As an example, knowledge about the existence, the general use, and the functionality of user interface programming libraries helps in such communication.
- *Communication at the same level with experts for physical products:* If the digital solution incorporates a physical product, knowledge of the technology enables important aspects to be discussed and agreed with the relevant professionals. This is especially important if the digital solution involves special software and hardware parts. For instance, if the digital solution running on a mobile device requires very high performance from the processor, a discussion and alignment on the maximum current rating of the battery available becomes important.

- *Communication at the same level with suppliers or partners:* A deep understanding of the technologies involved allows better collaboration with suppliers or partners who realize parts of a digital solution. This is applicable for vendors of hardware as well as software parts. For example, knowledge about the availability and general function of video compression codecs or white label hardware is important to be able to subcontract such product parts.

4 Fundamentals of the building process

This chapter introduces the fundamentals of the building process for digital solutions. To this end, section 4.1 presents the essential stakeholder roles in the building process. Next, section 4.2 presents the activity areas that make up a building process. Finally, section 4.3 looks at the basic flow of the building process for digital solutions.

4.1 Important stakeholder roles of the building process

EO 4.1	Explain the different core stakeholder roles in the building process and their relationship to each other (L2)

The term *stakeholder* is an important generic term for all persons who influence or have an influence on a solution. The term is defined as follows [Glin2020]:

> Stakeholder: A person or organization who influences a system's requirements or who is impacted by that system.

The definition formulates a very broad understanding of stakeholders in the sense that it is about people or organizations that can influence a system or are influenced by the system. In this definition, system should not be understood as limited to technical systems, but refers instead to systems in general, that is, in particular also to socio-technical systems (digital solutions). This broad understanding underscores the holistic perspective of Digital Design in the sense that when building a solution, attention should be paid not only to stakeholders in the immediate environment of a solution, but there should be a systematic examination of which individuals and organizations influence or are influenced by a system.

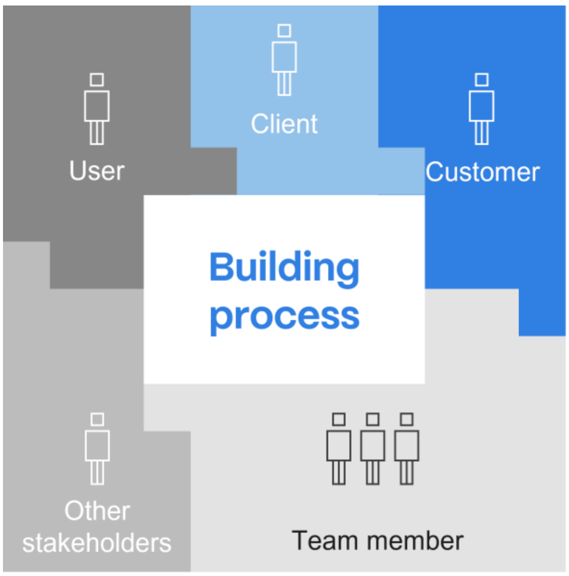

Figure 4.1 - Important stakeholder roles in the building process

There are many approaches and tools for identifying and classifying stakeholders (cf. [IIBA2022], and [Alex2005]). For the foundation level, the stakeholder roles, client, customer, and user are important with regard to the building process in order to systematically consider the immediate stakeholders. Another important stakeholder role is the building team member as an executing role. Finally, this section concludes by looking at the relationships between the roles.

4.1.1 Client as stakeholder role

The client role is defined as follows:

> Client: A person or organization who orders a system or a solution to be built.

The client orders the building of a solution and thus initiates the building process. The term order need not necessarily be understood as an order in the sense of a business relationship between two companies. When a building process is executed within an organization (e.g., for an in-house solution), the client is part of that organization, as are many other stakeholders.

Naming the client clearly is useful in both situations, since the client, as the initiator of the building process, is relevant for understanding the necessity of the digital solution as well as the goals or the problem to be addressed. Developing these themes is an essential part of design competence and the foundation for any design process (see section 0).

4.1.2 Customer as stakeholder role

The customer role is defined as follows:

> Customer: A person or organization who receives a system, a product, or a service.

The term *receives* includes both buying a solution and obtaining it for free. The definition is very broad to cover various situations. Basically, customers get added value from the digital solution. Typical situations are:

- Customers receive a system without additional services. For example, the customer buys office software.
- Customers receive a product that is embedded in the digital solution. For example: buying a games console that allows you to purchase games via the Internet or buying a book from an online store.
- Customers receive a service that the digital solution provides. For example, a customer can use the digital solution to book a hotel room.

Beyond its direct customers, a digital solution may also have indirect customers. This is the case, for example, when customers employ a digital solution to improve non-digital services that they provide to their customers (e.g., a physician's practice that offers their patients an online appointment booking service).

4.1.3 User as stakeholder role

The user role is defined as follows:

> User: A person who uses the functionality provided by a system.

This definition refers primarily to human actors who work directly with the digital system and use its functions. In addition to human users, digital systems can also be used by animals, for example (e.g., in digitized farming).

In the context of digital solutions, understanding the difference between the customer and the user[3] is essential to get a clear perspective on the socio-technical system (customer) and the technical system (user). Customers receive added value from the solution and do not necessarily have to be users of a system for this—that is, they do not have to interact directly with the system.

In the context of digital solutions, it is often assumed that the user and the customer are one and the same person. For example, a person that orders a hotel room on the website of the hotel is a customer of the hotel and at the same time the user of the website. In contrast, parents who buy an audio book subscription for their children are customers of the solution whereas the children are the users.

This implicit assumption limits the solution space of digital solutions unnecessarily because it assumes that the added value of a digital solution is created only through direct interaction with the solution. A good digital solution can also create value with indirect interaction. Consider the hotel example again: the customer in the hotel could also call reception and speak to a hotel employee, who in turn interacts with the website to book the room as a user. The added value for the customer is the same in both cases but the interaction is completely different. For good Digital Design, it is important to separate these perspectives to identify valuable non-digital aspects of a digital solution that can be enhanced or supported by digital means.

[3] In this handbook, the term user is used.

4.1.4 Team member as stakeholder role

The *building team member* role is defined as follows:

> Building team member: A person who performs one or more activities of the building process.

The building team member (short form: team member) is a person who participates in building the solution, and thus performs at least one of the activities (see section 4.2). This definition is very broad to cover the wide range of roles and tasks required to build a digital solution.

This broad understanding expresses that despite the necessary specializations and activity areas, all people active in the building process form a team, regardless of the concrete role and process organization. This team understanding is an essential success factor for a digital solution and is especially important for achieving a high-quality solution.

4.1.5 Interaction between the roles in the building process

For a holistic perspective of the building team during the building process and of the digital solution, the stakeholder roles each provide a special perspective:

- *Client*: Stakeholder role that orders a digital solution to address a certain customer and user segment. The client role makes initial assumptions about what added value customers want and is responsible to its stakeholders (e.g., owners or supervisors) for the overall success of the digital solution.
- *Customer*: Stakeholder role that obtains/uses a digital solution to create value for themselves and/or others. This stakeholder role has expectations for value as well as the entire customer experience.
- *User*: Stakeholder role that uses or employs a solution under its own motivation or under external motivation. The user role has expectations about how to use the solution as well as about the user experience.

Depending on the perspective, it is often not easy or clear to decide which person holds which role. The client of the digital solution, for example, can see themselves as a customer of the supplier and certainly has added value from the product/service commissioned.

In Digital Design, to ensure a holistic design, it is important to look at the solution from the perspective of each role during the building process. Stakeholders in the client role indicate why building a solution is important, what goals are to be achieved with the build, or what problems are to be solved. Stakeholders in the customer role provide input on the added value they expect from the digital solution (e.g., functionality, quality) and what a good customer experience (e.g., purchase process, maintenance, service) should look like. Stakeholders in the user role also provide impulses for the digital solution in terms of a good user experience (e.g., usability, identification with the product/service)—especially if they do not use the digital solution under their own motivation but are "motivated" from outside.

Typically, the requirements, desires, and needs of these three roles are not free of contradictions and can therefore usually not be fully implemented. The task of Digital Design is therefore to work out meaningful and sustainable compromises together with all relevant stakeholders in order to design a solution that is acceptable from the point of view of all stakeholders.

Stakeholders in the client role are ultimately responsible to their stakeholders for the overall success of the digital solution. Therefore, they have to decide how the digital solution will be realized and how, for example, the customer experience and the user experience will be implemented. For holistic design, the task of Digital Design here is to advise the client meaningfully through good compromises, in order to design a solution that is in the interests of the users, customers, and clients.

For a successful building process, it is important from the building team's perspective that client, customer, and user roles are clearly defined and understood by the appropriate individuals and organizations. The knowledge, perspective, and responsibilities of each role are important for a functioning process and for collaboration (see chapter 13).

4.2 Activities of the building process

EO 4.2	Understand the activity areas in the building process for digital solutions with their work products (L2)

In general, a *process* is defined as follows:

> Process: A set of interrelated activities performed in a given order to process information or materials.

In the following, we introduce a number of terms to describe the building process. These terms refer to activity areas and possible work results of the activity areas. The activity areas must not be confused with roles within a project structure. Roles can be defined from these activity areas but depend on the particular process model or project situation.

Existing disciplines for developing elements of a digital solution (e.g., software engineering, industrial design, usability engineering, product management, software testing) can be aligned with one or more of the activity areas presented.

For good Digital Design, these existing disciplines need to be embraced, and a general understanding of the building process helps you to work with a variety of existing disciplines. In this section, we describe the building process for a digital solution in a schematic way. We distinguish three core activity areas (design, construction, and realization) and two cross-cutting activity areas of a building process (management and evaluation).

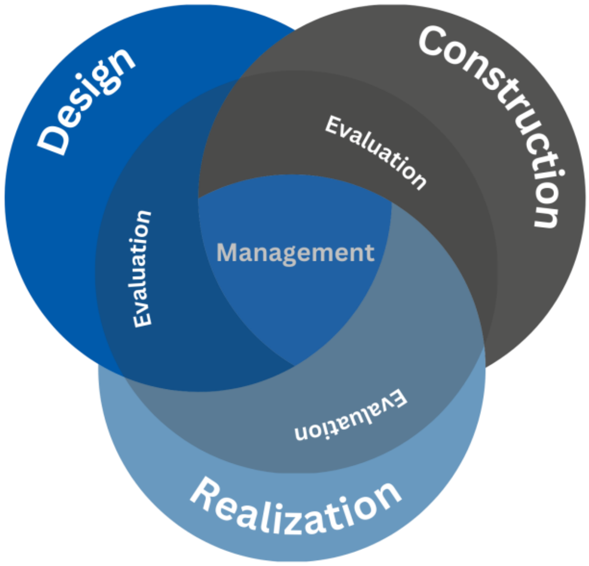

Figure 4.2 - Activities of the building process

The basic understanding of the activity areas is important to understand the different activities of the building process and to understand the integration of Digital Design into the building process. However, the basic understanding of the activity areas is not sufficient to enable you to structure or perform a building process. We consider the structuring of the building process in chapter 5. As an introduction to the execution of building processes, chapter 12 explains how Digital Design integrates with various process models and how these process models can be used to execute a building process.

4.2.1 Core activities of the building process

4.2.1.1 Design of a digital solution

We have already learned about design competence in the sense of the capability to design in chapter 2. This section focuses on the activity area of design as part of the building process. An important factor in understanding design as an activity is that design can be understood as both a result and an activity:

> Design: A plan or drawing produced to show how something will look, function, or be structured before it is made. | The activity of creating a design.

The first part of the definition uses the term design as a result or concept. This result is defined as a plan or drawing that is produced to show how something will look, function, or be structured before it is realized. The second part of the definition refers to the activity of creating this result, that is, design in the sense of designing something.

Thus, design as an activity also means imagining a desired change and designing it with the help of design concepts (see below), evaluating the designs, and accompanying their implementation.

Talking about a *desired and intended change* is a rather abstract and far-reaching formulation. Design literature often uses *design problems* or *design goals* as alternative terms. However, we prefer *change* to *problems* or *goals* because this describes what design is really about: envisioning change and making it happen.

To design the desired change, the design process creates concepts as work products that describe the digital solution intended to achieve the desired change. These concepts are called *design concepts* and are defined as follows:

> Design concept: A description of the design of a digital solution, of a digital system, or of an element of a digital solution.

Design concept is a general term and can be considered as a building plan for the digital solution at different abstraction levels, which represents all relevant components required for the planned change of the environment. Thereby, design concepts can include the following contents:

- Representation of processes and workflows (e.g., business processes or interaction sequences)
- Representation of technical elements (e.g., devices that make up a solution)
- Representation of software components (e.g., user interfaces, function)

The important thing is that a digital solution is more than software. This means that the term design concept also includes the description of devices that are designed specifically for the digital solution at hand. Therefore, the creation of design concepts may require the involvement of various disciplines (e.g., interaction design, industrial design, and service design). Furthermore, depending on the type of solution, competencies in different areas of requirements work (e.g., business analysis, requirements engineering, or usability engineering) are needed to understand the goals and constraints.

The responsibility of design does not end with the design concept. Design also includes two important evaluation aspects:

- *Formulating a change:* This means ensuring in the design that the change is desirable for all relevant stakeholders. This aspect is about involving all relevant stakeholders to ensure their acceptance of the defined change.
- *Creating a design concept for a digital solution that will create that change:* This means ensuring in the design that the design concept has the potential to achieve the intended change and that the digital solution realized will actually produce the intended change. This aspect has two dimensions.

In the *first dimension*, the design concept itself is the focus of evaluation, that is, the design concept must be evaluated to determine whether it is capable of enabling the intended change. This aspect requires the involvement of stakeholders to perform the evaluation. Besides the design concept, working with prototypes is an important tool for validating certain aspects of a design concept. Chapter 7 looks at the use of prototypes as tools in more detail.

The *second dimension* concerns the proper construction and realization of the digital solution according to the design concept. This requires close cooperation with the activity areas of construction and realization. We discuss details of this cooperation in section 4.3 after introducing these terms.

Thus, the two dimensions revolve around the following questions: Are we building the right digital solution? Versus: Are we building the digital solution right?

4.2.1.2 Construction of a digtal solution

Construction is an activity area that deals with the technical details of a digital solution to prepare its realization. We define construction as follows:

> Construction: The creation of the realization concept of a digital solution that will create the desired transformation.

This understanding of construction has two aspects:

- First: the creation of the realization concept of the digital solution
- Second: the evaluation that the digital solution described by this concept will create the change targeted by the design activity

Realization concept is a generic term for all types of work products created during construction and is defined as follows:

> Realization concept: A description of the technical realization of a digital solution.

The realization concept must use real technology and has to deal with all technical details that are necessary to realize the digital solution and its elements. The realization concept may include the following parts:

- Organizational structures (e.g., departments, process specifications)
- Physical structure (e.g., physical components and materials)
- Technical structure (e.g., microprocessor and board) of dedicated devices
- Software structure (e.g., software components)
- Realization technology (e.g., programming languages, frameworks, technical sensors)
- Technical infrastructure (e.g., the definition of suitable data centers)

This broad range clearly shows that constructing a digital solution may require the involvement of different experts (e.g., organizational experts, requirements engineering, software engineering, industrial design, and production engineering) and that realization concepts have different instances (e.g., software architecture concepts, physical building plans, electronic layouts, or even organizational plans). Within the foundation level, construction is viewed from a technical perspective, as cooperation with professionals to build entire organizations is understood as a more advanced topic.

Of course, the realization concept and the design concept depend on each other. We discuss the relationship between the two concepts in section 4.3.2, when we discuss the interaction between the activity areas of design and construction.

The evaluation in construction is about the realization concept. Like the design concept, the evaluation of the realization concept has two dimensions.

First, the realization concept must describe the necessary technical capabilities to create the desired change. This especially includes the aspect that the defined technologies must achieve certain qualities (e.g., reliability of the digital solution).

Secondly, the construction must ensure that the realization concepts defined are actually implemented. Here, close cooperation between the activity areas construction and realization is necessary.

4.2.1.3 Realization of a digital solution

The activity area realization deals with the factual implementation of the digital solution and is defined as follows:

> Realization: The implementation of the digital solution according to the defined design concepts and realization concepts.

As with construction, this understanding of realization has two aspects:

- First: the implementation of the digital solution according to the design and realization concepts
- Second: ensuring that the digital solution implemented creates the intended change targeted by the design and construction activities

The realization of a digital solution is by no means a trivial endeavor. As with construction, realizing a digital solution may require the involvement of various experts. For example, digital solutions with dedicated hardware components (e.g., smart speakers) require not only competencies in software development, but also competencies in the realization and production of the hardware.

Evaluation during realization is fundamental to every building process. Realization must ensure that the digital solution is implemented according to the design concept and the realization concept so that the digital solution creates the desired change. This evaluation is performed in interaction between the activity areas design and construction.

4.2.2 Cross-cutting activity areas

4.2.2.1 Management of the building process

The process of building a digital solution is so complex that it requires a separate activity area for management. In section 2.3, we learned about the understanding of management as an essential cross-cutting competence for Digital Design. Management is defined as follows:

> Management: Leading the building process
> in cooperation with all other activities.

We distinguish between three dimensions in the management of the building process:

- *Management of the project dimension*: planning and coordination of activities, time, and budget
- *Management of the product dimension*: development of a short-term and long-term strategy for the development of the digital solution as a product
- *Management of the social dimension:* managing stakeholder expectations, managing the stakeholder insight process, getting the right people and skills for the activity at hand

To capture and structure the work of management, the following work product is defined:

> Management concept: A description of the management approach for
> the building process.

Management concept is a generic term for all types of concepts that describe the procedure for management and thus the work in the building process. Examples of management concepts can be procedure models for describing the work process, role and task descriptions for structuring the building team, or templates for creating work results (e.g., design or realization concepts). Management concepts are important for describing the procedure for working in the building process transparently for the client, the building team, and all relevant stakeholders.

In addition to the management concept, the management of the building process requires means of organizing the work in the building process:

- *Work items* are used to define and organize work to be done as part of the building process. Work items can have different scopes (e.g., creation of a design concept, design of a function, or realization of a function). Concrete characteristics for work items are typically defined by the procedure model of the building process. Examples of concrete manifestations of work items are tasks, user stories, or epics. Similarly, strategy papers or project assignments can be understood as work items.
- *Management tools* support the organization, planning, and visualization of management tasks. Such tools help, for example, with the creation and visualization of project plans, milestone plans, Kanban boards, or backlogs.

In this handbook, we present selected perspectives of the project dimension and the social dimension as cross-cutting competencies. Product management goes beyond a foundation level and is not explicitly part of this handbook.

4.2.2.2 Evaluation of the digital solution

Evaluation as an activity area considers the quality of work products created in the building process. The digital solution realized is also considered a work product. We define evaluation as follows:

> Evaluation: A systematic process for determining the value, quality, or appropriateness of something.

In the building process, evaluation determines whether a digital solution or work product used to create a digital solution actually has the qualities and properties it should have according to the design concepts and stakeholder needs.

This means that in the building process, evaluation is always linked to a work product and therefore to one of the core activity areas. We therefore consider evaluation as an inseparable part of the core activity areas. The respective perspective on evaluation is therefore always described as part of the core activity areas (see section 4.2.1). However, the independent definition of evaluation is intended to emphasize the importance of quality and continuous evaluation as an attitude and to make clear the general applicability of evaluation in the building process.

To capture and structure the work of the evaluation, we define our own work product:

> Evaluation concept: A description of the evaluation approach for a work product.

Evaluation concept is a generic term for all types of concepts that define the procedure for evaluation during a building process (e.g., test concepts, test plans, or test cases in software quality assurance, cf. [ISTQB2023]). In contrast to concepts created as part of the core activity areas (see section 4.2.1), the evaluation concept does not describe the digital solution. An evaluation concept describes the approach to the evaluation of a certain work product of the building process. The reason for this is that the evaluation should be made explicit.

4.3 Basic flow of the building process and interaction of the activity areas

EO 4.3	Describe the fundamental flow of the building process and the interaction of the activities (L1)

In the following, we first explain the basic flow of the building process based on the activity areas. Then we discuss the interaction of the activity areas during the process.

4.3.1 Basic flow of the building process

The core activity areas presented are often misunderstood as process steps: those involved in design create the design concept, from which those involved in construction create a realization concept, which in turn is implemented by those involved in realization. However, such an approach has been considered impractical for most projects since the beginning of software development (see [Royce1970]).

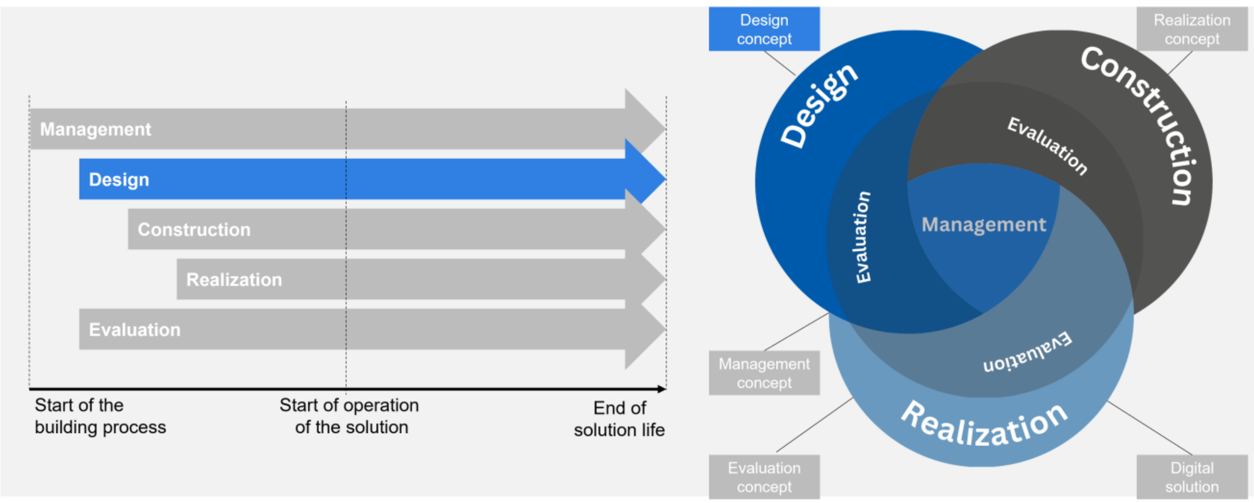

Figure 3.4 – Ratio and duration of the activities of the building process

A more realistic view is to understand the building process as ongoing activities that start staggered and are then performed together until the end of the solution's life. Figure 3.4 shows the activities of the building process in the form of a Venn diagram and the sequence as a simplified timeline.

Basically, we can state that every building process starts with management to organize the work. Then, some design work takes place to get an idea of the solution. Next, construction begins in order to gain an understanding of the technical implementation. This is a prerequisite for starting the realization of the solution. When the design activity starts, the evaluation work also starts and continuously accompanies the entire building process.

All activities of the building process run in parallel and end only with the end of life of the solution. In section 0, we specify the building process in more detail and divide it into three essential steps.

Organizations should be aware that the quality of the building process can have a significant impact on the quality of the digital solution and the digital system. Standardized and continuously improving processes [Demi2000] allow quality to be planned.

This also means that the creation of a high-quality digital solution is a joint and holistic task of the building process: all activity areas must work together to deliver a high-quality digital system that realizes the digital solution. Quality control and management as well as the continuous evaluation of the artifacts created must be central aspects of the underlying process. This is expressed by the Venn diagram in Figure 3.4.

4.3.2 Interaction of the activity areas during the building process

In the following, we briefly outline the interaction between design, construction, and realization to create a fundamental awareness of the importance of collaboration.

Interaction between design and construction

Understanding the difference between design and construction is very important for understanding the building process for a digital solution. Both activity areas work at a conceptual level and often use the same languages for communication and documentation (e.g., diagrams or technical drawings).

The most important difference between design and construction is the perspective. Design considers the intentional change of the world through a digital solution with the goal of understanding and designing the intended change. Construction looks at the inside of a digital solution and is concerned with defining and designing the technical implementation of the desired change.

For a design to be feasible, it is important that it can be realized with technologies available. People making design decisions must therefore have enough technical expertise to assess the technical feasibility of their decisions, or they must recognize when they need to consult engineering expertise from construction.

The cooperation between design and construction is not only about limits and restrictions—design and construction decisions may also create additional possibilities. If, for example, a technology is selected during construction that enables additional functionality that was not thought of in the design, it may be possible to incorporate new capabilities into the design of the digital solution.

Interaction between design and realization

Close interaction between the activity areas of design and realization is important, as digital solutions have an inherent complexity due to the software part. The implementation (programming) of software is seen as an intellectual challenge, since even the simplest programs can push the human mind to its limits (cf. [Glas2006], [Wein1971]). Important questions about the details of a software implementation often arise only during the actual implementation of the software, because the act of programming forces us to think in logical precise structures of the programming language. This identifies ambiguities and open issues in the design concepts that then need to be clarified through design.

For a solution or part of a solution to be realized, there must be a complete understanding of the solution so that it can be implemented accordingly. Elaborating this complete understanding is a laborious undertaking, as a great many details usually have to be worked out in order to serve as a template for realization. Late design decisions are an important means of reducing complexity and facilitating collaboration between design and implementation. Late design decisions mean, above all, that design decisions are made as late in the process as possible, precisely when the decisions and associated designs are needed for realization. This allows attention to be focused on the relevant parts of the solution.

Interaction between construction and realization

Close interaction between construction and realization is important—analogously to the interaction between design and realization—in order to manage the complexity of software in a meaningful way. During realization, ambiguities and open questions often arise in the realization concepts, which then have to be clarified by construction.

Important technical decisions for a solution have to be made before the actual realization starts. Analogous to design, collaboration between construction and realization can also benefit from the idea of delayed construction decisions. The benefit of these delayed construction decisions is twofold: first, there is a better understanding of the details of the respective function (e.g., data structures or interfaces to other components), which leads to a better-informed decision on the component; second, in the course of the building process, new open source components may emerge that were not yet available during the planning of the digital solution. Such a situation is not uncommon, since the software community works continuously on developing its technologies further.

Interaction between design, construction, and realization and management

The previous three subsections showed various pairs of interaction between design, construction, and realization. These overlaps focus mainly on the content details of the design, construction, and realization and evaluation of the digital solution.

The management of the building process requires cooperation between all three activity areas, as the competencies of all three areas are required to create coherent planning for the building process.

The definition and alignment of the details of the building process are by no means top-down activities from design to construction to realization. Such an approach typically leads to suboptimal results. For the development of software in particular, the agile development movement has shown that an iterative and incremental management approach is much more suitable (cf. [Meye2014]).

An iterative and incremental approach requires close cooperation between experts of all three activity areas, since each activity area provides certain input for another activity.

The bad news, however, is that there are mutual dependencies between the activity areas and that these complex dependencies make it impossible to define a general planning approach for the building process for a digital solution.

In the following, we illustrate these dependencies using two examples.

- The development environment (the environment for implementing and testing an element of a digital solution) can be viewed as a pure realization aspect. Modern development environments, however, provide important features that are also useful for design and construction, such as tools for modeling data structures. Certain development environments especially support rapid prototyping, that is, detailed design aspects of a digital solution can be defined during the realization and immediately validated and improved together with stakeholders.
- The definition of time schedules is often driven by realization costs and time to market. Such an approach typically neglects the effort for design and construction, including the evaluation of the design and realization concepts. A better effort estimation and budget allocation can be achieved by considering design and construction explicitly in the budget. In project situations with a fixed budget in particular, the design and construction competencies are of great importance in obtaining the best possible digital solution for the available budget (cf. [McCo2014]).

5 Structuring building processes from a Digital Design perspective

Building a digital solution is a complex undertaking in which multiple aspects must be considered. To understand Digital Design, it is important to understand the difference between digital solutions and physical products.

The main characteristic of the process of designing a physical product is the separation of design from manufacture: the creative act of determining and defining the form and function of a product takes place in advance of the physical act of making (realizing) the product, which consists purely of repeated, often automated, replication (cf. [Nobl1996]).

Every process model that works with implicit assumptions of mass production processes is of limited use for building a digital solution. The design of a digital solution is an ongoing process during the whole building process. Building digital solutions therefore requires process models that provide guidelines for integrating design into the entire building process.

To get clear perspectives on the building process, different models can be applied for structuring. In this chapter, we present three models that provide a good basis for a foundation level for systematically structuring the building process and perspectives on a digital solution:

- *The model of the design perspectives people, business, and technology* to understand the factors influencing the building process and the solution
- *The level model for digital solutions for* structuring a digital solution
- The model of the three essential steps of the building process

5.1 The Model of the design perspectives people, business, and technology

EO 5.1	Explain business, people, and technology as related design perspectives on a digital solution (L2)

There are three overlapping perspectives that are essential to the holistic design of a successful solution (see Figure 1.5, [Brow2009]):

- *People:* Is the solution desirable and attractive to people?
- *Business:* Is the solution economically viable?
- *Technology:* Is the solution technically feasible or realizable?

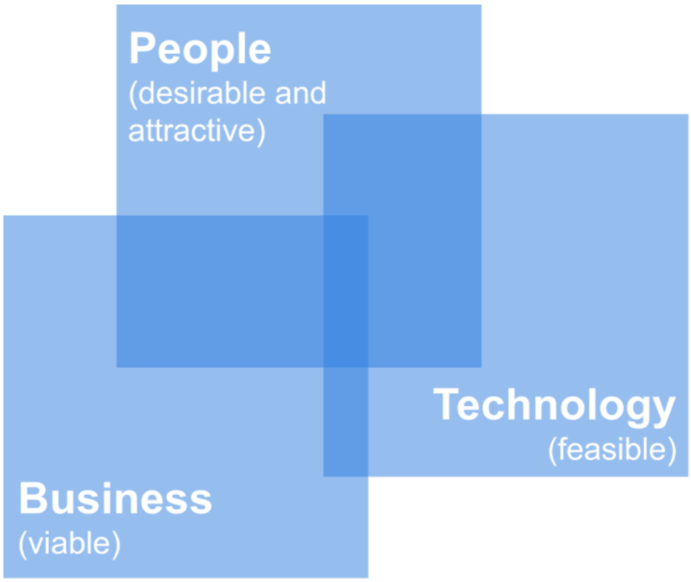

Figure 1.5 – The three design perspectives of a solution

These three design perspectives address essential aspects of a solution and, according to [Brow2009], must be meaningfully integrated and harmonized in order to design a successful and, above all, innovative solution.

When designing a digital solution, these three perspectives are useful for two reasons. First, they help you to systematically view and analyze a solution based on the three perspectives. Second, the three perspectives address important stakeholder groups and their culture. Understanding stakeholder groups and their culture is important in order to communicate appropriately with stakeholder groups in their respective perspectives and to help stakeholder groups understand the perspectives of other stakeholder groups.

In the remainder of the handbook, we use these three perspectives to define the fundamental aspects of a digital solution (see chapters 8 to 10) and demonstrate the use of these aspects for a holistic design in the building process (see chapter 11).

5.1.1 People perspective

The perspective of the people on the solution, or more specifically, the customers and users, is called the people perspective.

Relevant core aspects are the value proposition (added values) of the solution, customer experience, the user experience with the solution, and the usability of a solution.

This broader people perspective is of great importance in contexts without immediate users and in innovative contexts. Digital solutions without direct customers (e.g., control systems for energy networks) also generate added value that can be clearly defined. Innovative solutions usually not only deliver immediate benefits, but also change their environment or entire societies. For holistic design, in addition to the immediate value and benefit view, this broader view is also necessary to consider the importance of a solution to society as a whole.

In addition to these issues driven by the immediate value/benefit of the solution to people, more abstract issues can also be considered in this perspective. Examples can be the meaningfulness of a solution (Is the solution needed at all?) or questions of morality and ethics (Should this solution be allowed to exist?).

For the foundation level, it is important that these topics are part of the people perspective. A more in-depth look at these topics goes beyond the foundation level.

5.1.2 Business perspective

The *business perspective* is, in the first instance, the client's perspective on the solution. The focus is on the economic viability of the solution.

Relevant core aspects are business models (e.g., costs of the solution, profits of the solution), the process of value creation (How does the profit arise?), as well as market and customer segments (Who is interested in the solution?).

The business perspective is also relevant for digital solutions that are not designed to generate profits in the sense of making money. Examples of such solutions are public sector solutions or in-house solutions (e.g., software for employee management). Development and operation must also be financed for these types of solutions. For a holistic design, therefore, the question of economic viability must also be considered for this type of solution.

In addition to these primarily economically driven core aspects, the topic of sustainability in the broadest sense can also be counted as part of the business perspective. The sustainability of a solution has a significant impact on the environment and the society that uses this solution. Therefore, sustainability also contributes to the long-term economic viability of a solution. The topic of sustainability through digital solutions (creating sustainability through design) and the design of sustainable digital solutions (sustainability in design) is an important requirement in Digital Design (see section 1.1.4 and cf. [BCDE2015]). Nevertheless, due to its complexity, this topic goes far beyond the foundation level and can therefore only be considered as an example in this handbook.

5.1.3 Technology perspective

The *technology perspective* is the perspective of implementation on the solution, or more concretely, the languages of development and realization. The focus is on technologies and machines.

Relevant core aspects are the elements, components, or building blocks (of which a solution is composed), functions of the solution, data (which is stored in the solution), programming languages, technical systems, and the operation of these technical systems. It is important that this perspective firstly considers the basic feasibility (Can it be realized at all?). On the other hand, it is also a question of whether an organization (e.g., a team) has the necessary capabilities to implement a solution with the given technology.

In addition to these topics that focus on immediate implementation, new and innovative technologies are also part of this perspective. In the case of digital solutions in particular, technological innovations are constantly emerging that promise new opportunities (e.g., mobile Internet, blockchain technology, or machine learning).

This broader technology perspective is important for considering the latest trends when designing a solution. In this way, the significance of new developments for a solution can be critically evaluated and a decision can be made as to whether an innovative technology is useful for a solution or not.

5.2 The level model for digital solutions

EO 5.2 Explain solution, system, and element as three essential abstraction levels of a digital solution (L2)

The level model (see Figure 2.5) divides a digital solution into three successive abstraction levels: *solution level*, *system level,* and *element level*. The levels help to structurally capture the diverse details of a digital solution and prepare them for different target groups and purposes.

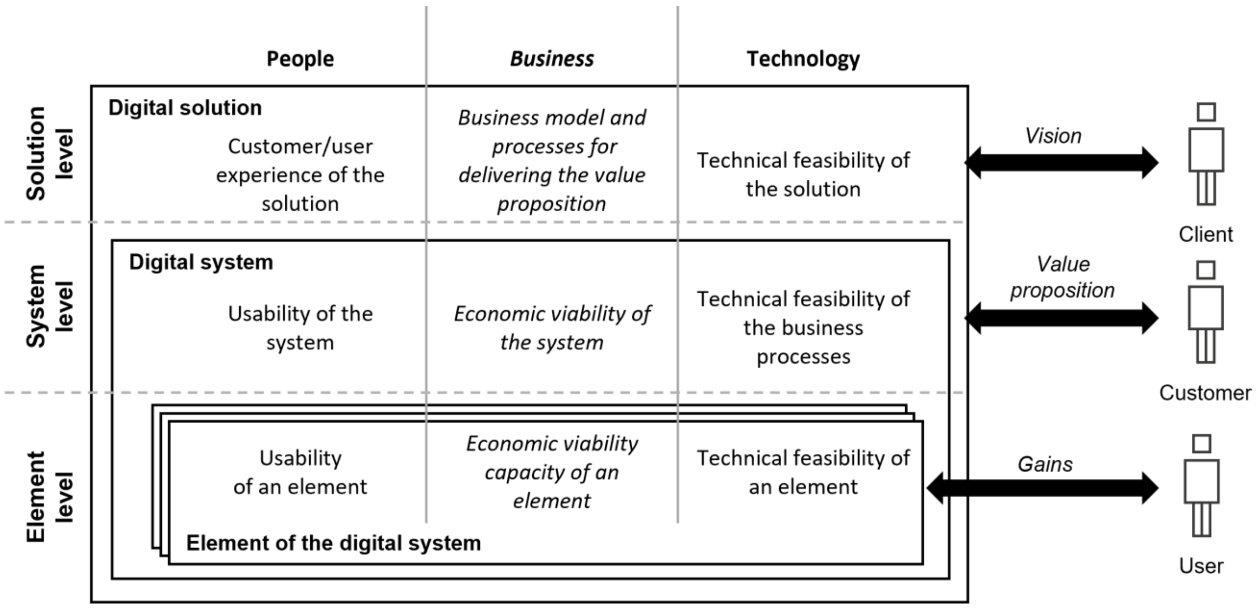

Figure 2.5 - The three levels of a solution with the primary stakeholder roles

In the following, we present the three levels, starting with the solution level.

5.2.1 Solution level

At the solution level, the focus is on the change that the client wants to bring about with the solution (also referred to as the vision), that is, in particular, the value proposition that the client wants to deliver to the customers and users of the digital solution.

All three design perspectives must be taken into account: the people perspective is about the customer and user experience that the solution should provide. The business perspective is about the business model and the business processes through which the solution should realize the intended value proposition. In the technology perspective, you must assess whether the solution is feasible with the available technology and the costs envisaged in the business model.

It is important to abstract as much as possible from the technical implementation of the solution at the solution level and focus on the client's vision and value proposition for customers and users.

5.2.2 System level

The *system level* views the technical system in a higher-level perspective. The primary target group for this level is the customer, for whom the technical system should deliver a value proposition. Secondary target groups are the client and the user.

At the system level, the focus is on the technical realization (technology perspective, see above) of the added value for the customer in a usable way (people perspective, see above). Key issues are the structure of the system (users, devices, software, hardware, communication technology, and existing system) and the appropriate generation of added value for the customer by a system.

In addition, at the system level, we have the question of economically viable implementation from the client's point of view. At the system level, with the technical implementation (technology perspective), significant technology decisions are made that are associated with corresponding costs (business perspective). The costs of technology include costs for licenses (e.g., for libraries), for operation (e.g., for data centers), or for the use of existing systems (e.g., transaction fees for payments on the network). Furthermore, personnel costs for development and operation are also among the costs that must be considered when designing a solution at the system level.

The third target group—that of the user—is relevant to the design at the system level in the sense that it defines the user groups that interact directly with the system. We have already considered the separation between customer and user as stakeholder roles in section 4.1. With the design at the system level, an important design decision is processed in this respect: whether and how customers of the solution (solution level) are integrated into the technical system as users.

An essential factor in understanding the interplay between solution and system level is the fact that system-level design is closely intertwined with solution-level design. At the solution level, the added value that must be delivered by the system is formulated. The business model and the business processes that are technically realized at the system level are also addressed at the solution level. Holistic design means that solution and system levels are designed together and in alternation. In the case of digital solutions in particular, innovative technologies and system structures can have a significant impact on added value and the business model.

5.2.3 Element level

The *element level* captures the individual proprietary elements of the technical system of a digital solution. The primary target group for the element level is the users, the secondary target groups are the client and the customer.

At the element level, the focus is on the details required for the technical realization of the elements from the user's point of view. Essential topics are the structure of the user interfaces and technical interfaces, functionalities of the elements, and the data required for the functionalities. These topics concern both the technical feasibility (technology perspective) and the usability (human perspective) of the solution.

In addition, at the element level, we have the perspective of the technical interfaces to other elements that are required to implement the functionalities of an element (technology perspective).

As part of the design of a solution at the element level, all relevant decisions regarding the structure of an element and interaction with users are taken. These decisions have a significant impact on the scope and functionality of a solution and thus on the cost of development (business perspective). Even though the system level is where the major technology decisions and associated costs are decided, the element level has a significant impact on the cost of developing a solution.

An essential factor in understanding the interplay between system and element levels is the aspect of the concrete implementation of an element and the impact of the implementation on the system in terms of economic viability and usability of the solution for the user. Holistic design means that the element and system levels are designed together and meaningfully alternate. Innovative forms or functions of an element, for example, can have a significant impact on the success of a solution.

In contrast to the interaction between solution and system level, the level of detail and complexity is a decisive factor in the interaction between system and element level. The degree of detail of the element level is much greater compared to the system level. Therefore, the elaboration of an element in all its details is an extensive process that needs to be structured in a meaningful way in order to provide the right details for the design work at the higher levels.

5.3 The model of the three essential steps of the building process

EO 5.3	Explain the three steps of the building process for digital solutions (L2)

The previous models addressed the digital solution with its structure and help to enable a better understanding of the digital solution itself. Understanding the building process of a digital solution is also a challenge. In literature, there are various process models and frameworks for this with different approaches. To understand Digital Design, it is important to have a basic understanding of the process structure of a building process that is independent of process models and frameworks.

For this basic understanding of the building process, the model of the three essential steps helps. It divides the building process into the following three steps: *scoping*, *concept work,* and *development and operation*.

This model is primarily intended as a learning model and helps in two ways:

- On the one hand, the model provides a basic understanding of the steps that every building process must fundamentally go through and the goals, challenges, and ways of working that are relevant in each step. It is important to note that the model says nothing about the length or intensity of the work in any one step.
- On the other hand, the model shows how to basically jump back and forth between the individual steps to get an initial understanding of the importance of iterative work in the building process.

We introduce all three steps below. For this purpose, we present the essential goals and tasks of the individual steps and the work products created in the process.

We use the three essential steps at various points throughout the handbook to look at Digital Design along all the steps. Furthermore, in chapter 12, we present selected frameworks for the building process from a Digital Design perspective and from a three-step perspective.

With regard to the building of a solution, the three steps can be applied both to the building of a digital solution as a whole and to subareas (e.g., subsystems, subproblems, or subtasks) of a solution. For the readability of the following explanations, reference is always made to the digital solution.

5.3.1 Scoping

The goal of the scoping step is to achieve a clear and common understanding of the assignment with the client and among all relevant stakeholders and to make a decision for or against starting the concept work based on this understanding. This involves working with the client and the relevant stakeholders to develop a common understanding of the need for change (What is the reason for starting a new building process?), the target image for the digital solution (What do we want to achieve?), and the client's available resources. This task requires skills in requirements elicitation in particular. Depending on the context and type of solution sought, skills in design and construction are required.

Scoping can be accomplished quickly if the client and relevant stakeholders are clear about the goal and scope of the planned solution and understand both. However, it can also become a project in its own right if there is no clear understanding of the goal or of the digital solution, or if the organizational situation and clarification of the mission is very complex, which can be the case in large companies/organizations, for example.

Digital Design brief as a work result of scoping and work order for the concept work

In order to adequately record the results of the scoping, it is useful to create a Digital Design brief (abbreviation: design brief) as a work product:

> Digital Design brief: The description of the context, vision, scope, and general terms for building a digital solution.

Table 5.1 – Exemplary structure of a Digital Design brief

Section	Content
1. Context of the project	Presentation of the initial situation of the planned project, including motivation for the need for change, potential customers and users, potential further stakeholders, related solutions, and potential competitors
2. Vision	Presentation of the initial vision for the planned solution as part of the target picture for the planned change
3. Solution space	Presentation of the potential solution space for the planned solution (e.g., possible technologies and conceivable functions)
4. Constraints	Presentation of the constraints for the planned project (e.g., resources, budgets, schedules)

The work on the design brief and the results recorded in the design brief form the basis on which a client must decide whether the need for change and the target image offer sufficient potential to address the concept work in the building process. If the client decides to start the concept work, the design brief becomes the work item for the concept work. An example structure for a design brief is shown in Table 5.1.

Scoping means risk assessment

At the end of the scoping step, there is a risk assessment for or against the next step of the building process (concept work, see section 5.3.2). With this risk-oriented attitude, the specific approach can be tailored to the client's particular situation, risk appetite, previous knowledge and understanding, and available resources.

The conscious decision to start or not to start concept work is important, as this is the next step towards realization and takes additional resources from the organization. Therefore, the scoping and a design brief consider not only the content aspects of a solution (e.g., the vision), but also the client's important constraints in terms of resources, budget, and timelines.

On the one hand, the constraints are important to support the risk assessment on the part of the client. On the other hand, the constraints are important to define the design freedom in the further course of the building process. For example, the available budget influences the size of the development team and thus the feasible scope of the solution. External deadlines can further define the time frame in which the building process must be completed in order for the solution to be available according to schedule.

5.3.2 Concept work

The goal of the concept work is to develop a sufficient understanding of the solution and the underlying technical system among all relevant stakeholders based on the Digital Design brief. Based on this understanding, a decision can be made, together with the client, whether or not to take the risk of implementing the solution. This task requires a mix of competence in requirements work, design competence in the sense of Digital Design, and technical competence in the sense of construction.

Analogous to the scoping step, the risk assessment is an essential goal of the concept work. The specific approach to the concept work and in particular, the level of detail of the design concepts, must be defined depending on the willingness to take risks, the available resources, and other factors. For example, if the scope of a digital solution is small and well understood, a very short phase of concept work may be sufficient. If, on the other hand, many aspects of a solution are still unclear, much more extensive concept work with evaluation is recommended.

To structure the concept work, the areas of activities of the building process (see section 4.2) and the three levels of a solution (see section 5.25.2) can be used as structuring tools. This results in the following work products:

- Design concepts (see section 4.2.1.1) for documentation and coordination of the creative design at the three levels solution, system, and element
- Realization concepts (see section 4.2.1.2) for documentation and coordination of the technical design of solution, system, and elements
- Evaluation concepts (see section 4.2.2.2) for documenting and executing the evaluation of the design and realization concepts

Digital Design is responsible for the elaboration of the design concepts. The specific process for development and the extent (level of detail, content) and the composition of the concepts (structure, templates) depend on many factors and in particular, on the client's willingness to take risks (see above). In order to provide an initial orientation with regard to concept work in Digital Design, we present simple structures of chapter headings for design concepts below for all three levels of a digital solution. Due to the focus on Digital Design, we do not present templates for realization concepts. It is important for your general understanding to note that realization concepts are typically required for all three levels.

Exemplary structure of a solution design concept

A solution design concept can be used to represent a digital solution at the solution level from a higher-level perspective (see section 0). Table 2.5 shows an exemplary structure for a solution design concept based on the working model for the design of digital solutions. We present further details on possible content in chapter 8 when we introduce fundamental aspects for designing at the solution level.

Table 2.5 – Exemplary structure of a solution design concept

Section	Content
1. Vision	Description of the intended change to be achieved by the solution
2. Context	Description of the context of the solution at a glance
2.1 Customer segments	Description of the customer segments relevant for the solution
2.2 User groups	Description of the user groups relevant for the solution
2.3 Further stakeholders	Description of other stakeholders relevant for the solution
3. Business model	Description of the business model of the solution with value proposition and value creation architecture to deliver the value proposition
4. Business process	Description of the business process through which the solution delivers the value proposition
5. Quality requirements	Description of the quality requirements for the solution
6. Constraints	Description of the constraints that must be taken into account when building the solution

Exemplary structure of a system design concept

A system design concept can be used to represent the structure of the technical system for realizing a solution from a higher-level perspective (see section 5.2.2). Table 5.3 shows an exemplary structure for a system design concept based on the working model for the design of digital solutions. We present further details on possible content in chapter 9 when we introduce fundamental aspects for designing at the system level.

Table 5.3 - Exemplary structure of a system design concept

Section	Content
1. Introduction	Textual introduction to the system design concept
2. Goals	Description of the goals to be achieved by the realization of the system
3. Form	Description of the form of the system at a glance
3.1 User types	Description of the user types using the system
3.2 Existing elements	Description of unchangeable elements (e.g., objects, hardware, or software systems) whose existence is assumed for the operation of the system
3.3 Own elements	Description of the elements of the system that must be realized for the operation of the system
4. Function (scenarios)	Description of the functions of the system to achieve the goals
5. Quality requirements	Description of the quality requirements to be fulfilled by the system
6. Constraints	Presentation of the constraints to be observed when building the system

Exemplary structure of an element design concept

With an element design concept, a separate element of the technical system can be represented in detail. Table 5.4 shows an exemplary structure for an element design concept based on the working model for the design of digital solutions. We provide further details on possible content in chapter 10.

Table 5.4 – Exemplary structure of an element design concept

Section	Content
1. Introduction	Textual introduction to the element design concept
2. Goals	Description of the goals to be achieved by the realization of the element
3. Form	Overview of the form of the element
3.1 User interfaces	Description of the user interfaces to be implemented for the element
3.2 Data structures	Description of the data stored in the element
3.3 Technical interfaces	Presentation of the technical interfaces of the element to other existing elements
Function	Overview of the function of the element
4.1 Use cases	Description of the use cases realized by the element to achieve the defined goals
4.2 Technical functions	Description of the technical functions realized by the element to achieve the defined goals
5. Quality requirements	Description of the quality requirements to be fulfilled by the element
6. Constraints	Description of the constraints that must be observed during the building of the element

Organization of the concept work

The structures presented above are intended to provide a first impression of a possible design concept structure and, in particular, to serve beginners as a starting point for their own concept work. In chapter 7, we look at concept work as part of design work and discuss it in detail.

To organize the concept work and for an informed decision for or against starting the realization, the model of the three levels (see section 5.25.2) and the three perspectives (see section 5.15.1) can be used as guidelines.

The economic viability, feasibility, and attractiveness of the solution should be clarified at the end of the concept work. On the one hand, this requires extensive work on a solution design concept for the economic viability and attractiveness of the solution. On the other hand, work is required on the system design concept to ensure the technical feasibility and attractiveness of the solution. Elaboration of details in terms of element design concepts should occur when this contributes to clarification in the three perspectives. Otherwise, work on element design concepts in terms of the idea can be postponed for later design decisions (see section 4.3).

Essential for successful concept work is the inclusion of all relevant skills. These include business analysis for working on the business model and business processes at the solution level, UX and interaction design for working on user groups and user interaction, and requirements engineering for working on requirements and constraints.

There are different philosophies and approaches for the concrete design of the concept work. We discuss selected examples in chapter 12, where we consider selected process models from a Digital Design perspective.

5.3.3 Development and operation

The goal of development and operation is the actual implementation of the digital solution based on the design and realization concepts and with it, the achievement of the planned change as the goal of the design work (see section 0).

The development and operation of a digital solution are considered together in one step, as digital solutions are usually constantly evolving (e.g., new functions are added, existing obsolete functions are removed). This means that the operation of a digital solution and further development during ongoing operation should be considered from an early stage of the building process. This is not an original Digital Design topic, but it has an impact on the design of a digital solution.

Development and operation can be divided into four phases. We present these four phases briefly below.

Phase 1: preparation of the development

In the preparation phase, all necessary measures must be taken to start with the development of the first version of the digital solution. This includes, among other things, setting up the development organization with personnel, technical infrastructure, and further resources for realization.

Furthermore, all relevant content-related aspects that are required for the start of development must be clarified—in particular, the target image for the first version of the digital solution. A large part of the content work here takes place at the element level and can be documented and evaluated based on corresponding element design concepts.

From a Digital Design perspective, development preparation is an important phase for making critical changes with reasonable effort. In the preparation phase, no solution is implemented yet, so changes to the solution affect only the concepts that have already been created. As soon as the realization of the first version is running, this changes significantly.

The level of detail, structure, and timing of the work depend on the process model used. For economically efficient development, the preparation is a critical phase, because here, in a certain sense, the worklist for the development team is compiled, which then has to be worked through and continuously replenished during the initial development.

This aspect of the business perspective in the preparation phase must not be underestimated, as the efficiency of a development team depends to a large extent on a good workload. Every hour a development team is unproductive, that is, waiting for work, wastes time and budget for development, as this time in which no work takes place cannot subsequently be made up.

Second phase: development of the initial release

In the second phase, the development of the first version of the digital solution starts. At the end of this phase, operation starts.

The work in this phase is characterized by intensive content-related cooperation between all persons involved. The necessary details for the realization are worked out in design and realization concepts and evaluated and realized based on these concepts. The realization must also be evaluated appropriately.

Furthermore, the necessary work for the realization of the first version and beyond must be continuously processed to ensure an economically efficient utilization of development resources. Management is responsible for maintaining an adequate worklist. Nevertheless, from a Digital Design perspective, this is a major challenge, as work must be sensibly divided between developing and elaborating new content. If you neglect the worklist, you run the risk of running out of work. If you neglect the realization, this can lead to problems in the detail or to the slowing down of the development.

From a Digital Design perspective, when development starts, this changes the scope for critical changes. As soon as a solution is realized in parts, changes usually mean immediate adjustments to the elements already realized. This circumstance is often used as an argument for elaborating out the design concepts as comprehensively as possible before starting development. However, practice shows that such a comprehensive elaboration is not always purposeful.

Third phase: further development during operation

In the third phase, the solution is in operation and is maintained or further developed. As soon as an initial version of a digital solution is implemented and in operation, the focus of the building process shifts significantly.

The first part of this change is that the digital solution requires care and maintenance during operation. Users may report bugs that require fixing or may request additional functionality. This maintenance and optimization effort is often underestimated and creates a conflict between maintenance and the further development of a digital solution.

The second part of this change is that every decision on the further development of a digital solution must take into account that there is already an existing solution. From a design perspective, this shift creates opportunities and risks. The changes also include obtaining feedback from real users or customers in real settings. This feedback can be used to understand and further improve a digital solution. On the other hand, the risk is that users or customers need to adapt to modification of the digital solution. In addition, the further development of a digital solution must take into account the existing technical structures and constraints.

For example, modifying a digital solution will require an update of the software parts or even a replacement of existing devices. Such an update or replacement may require substantial effort and planning depending on the type of digital solution concerned. Another challenge is created by updates and modifications of existing technologies that are used to build or operate a digital solution. For example, if the operating system of a smartphone is updated, modification may be necessary in order to keep the digital solution operational.

Fourth phase: end of life of a solution

In the final phase, the end of life of a solution is prepared and the solution is decommissioned. Digital solutions usually cannot be "simply" switched off when they are integrated into a larger system. The concrete procedure for decommissioning a solution depends on many circumstances. In principle, decommissioning must be planned sensibly so that customers and users can switch to appropriate alternatives if necessary.

The end of life of a solution often includes final realization work to transfer important data to a new solution. Furthermore, laws usually require data to be archived for a certain period of time (e.g., tax laws) and then verifiably deleted after defined deadlines have expired.

From a Digital Design perspective, early consideration of the decommissioning of a solution can be an important tool and can, for example, promote the economical use of data. This aspect is not only relevant for solutions with a long service life (e.g., operational systems), but also for solutions that function as an intermediate step (e.g., an MVP in lean startup, see section 12.1.6).

5.4 The interplay of design perspectives, levels, activity areas, and building process steps

EO 5.4	Explain the interplay between the design perspectives, the abstraction levels, the activity areas, and the steps of the building process (L2)

To conclude this chapter, we consider the interplay of the three models presented for structuring the building process (section 5.15.1 to 0) and the activity areas from chapter 4 in order to obtain an integrated picture of the building process and the digital solution from a Digital Design perspective.

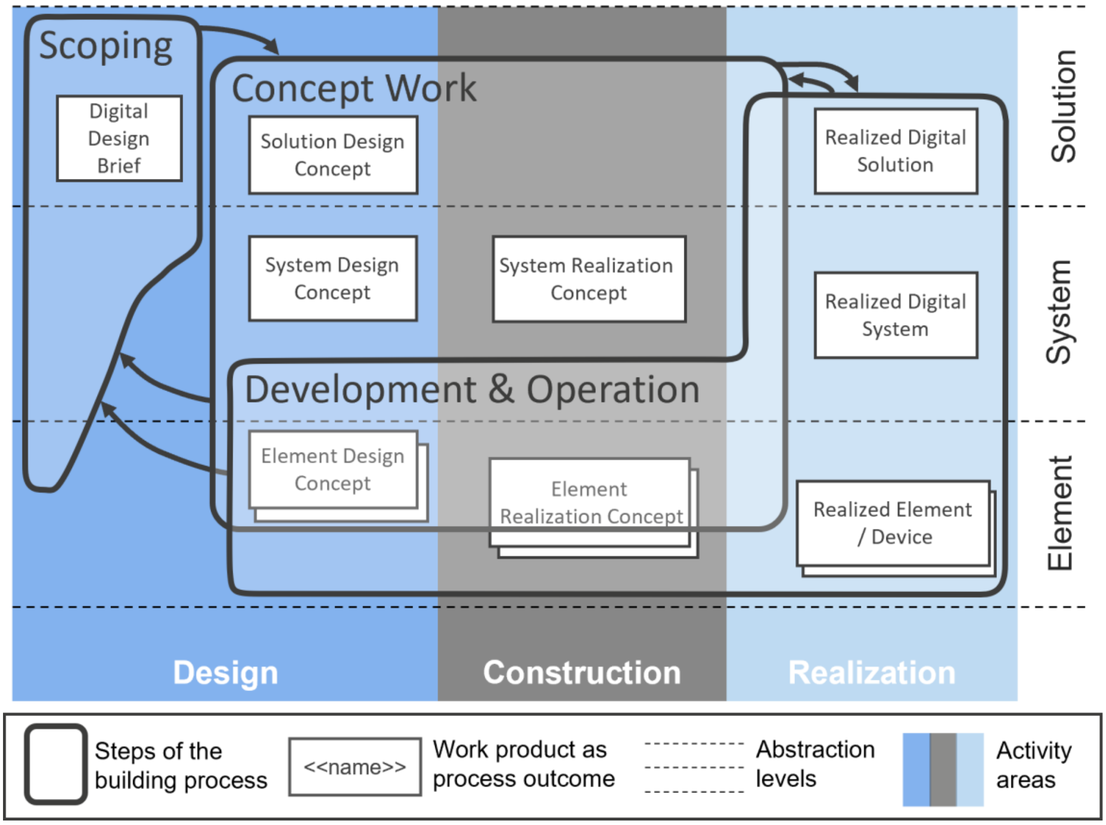

Figure 3.5 – Structuring of the building process and the digital solution

Figure 3.5 illustrates the interplay between the three core activities (design, construction, realization), the three essential steps in the building process with their results, and the three levels of abstraction. A more comprehensive representation as a big picture can be found in the appendix.

This overview is intended to help newcomers to Digital Design to structure the complexity of the process and the complexity of the solution being designed, making it more manageable. By combining these models, it is possible to structure tasks and responsibilities in the building process (see section 5.4.1), the complexity of a digital solution as a whole (see section 5.4.2), and the focus on specific aspects of a digital solution during the building process (see section 5.4.3).

5.4.1 Structuring tasks and responsibilities in the building process

The three steps of the building process (section 0) in interaction with the activity areas (section 4.2) and the three design perspectives (section 5.1) address the complexity of the process by structuring the timing of a construction process and the responsibility of each activity area to build a solution.

Table 5.5 - Tasks and responsibilities in the building process

<table>
<tr><th></th><th colspan="2">Scoping</th><th colspan="2">Concept work</th><th colspan="2">Development and operation</th></tr>
<tr><td>Management</td><td colspan="6">Management of the process along the project, product, and social dimensions

Responsible for the economic execution of the building process as a whole and compliance with the economic constraints

Responsible for definition of and compliance with management concepts</td></tr>
<tr><td>Design</td><td colspan="3">Responsible for context, vision, and solution space in terms of acceptance and viability

Coordination with construction to ensure technical feasibility</td><td colspan="3">Responsible for design concepts in terms of acceptability and viability and compliance with constraints from a design perspective

Coordination with design/realization to ensure technical feasibility

Continuous updating and more detailed specification of the design concepts, clarification of open questions</td></tr>
<tr><td>Construction</td><td colspan="3">Responsible for context, vision, and solution space in terms of feasibility

Coordination with design in terms of acceptance and viability of the solution</td><td colspan="3">Responsible for realization concepts in terms of feasibility and compliance with technical constraints

Coordination with design in terms of acceptance and viability of the solution

Continuous updating and more detailed specification of the realization concepts, clarification of open questions</td></tr>
<tr><td>Realization</td><td colspan="3">Not actively involved,
possibly as part of early prototypes</td><td colspan="3">Responsible for realization of the solution according to design and realization concepts</td></tr>
<tr><td rowspan="2">Evaluation</td><td colspan="2">Evaluation of whether the vision in the design brief is viable, acceptable, and feasible</td><td colspan="2">Evaluation of whether design and realization concepts are viable, acceptable, and feasible</td><td colspan="2">Evaluation of whether the solution is viable, acceptable, and feasible</td></tr>
<tr><td colspan="6">Responsible for definition of and compliance with the evaluation concepts</td></tr>
</table>

For each of the three steps of the building process, the activity areas can be used as the basis for determining which activity areas are relevant and what responsibility an activity area assumes in relation to the building process (see Table 5.5

Table 5.5). At the foundation level, the focus is on the design activity area, the evaluation of design concepts, and related activities.

For an overview of the entire building process and the concrete implementation of the building process, there is a wide range of frameworks in literature (e.g., design thinking, scrum, lean startup) with various processes and techniques. In chapter 12, we use the steps and activities of the building process to introduce and explain different frameworks for the building process from a Digital Design perspective.

5.4.2 Mastering the complexity of a solution

The three design perspectives (people, business, technology, see section 5.15.1) and the level model of a solution (solution level, system level, element level, see section 5.25.2) address the structural complexity of a digital solution by structuring the digital solution into three perspectives and three levels of abstraction. At each level of abstraction, the three perspectives play an important role, and a different issue is raised by the perspectives at each level.

First, the levels of a solution can be viewed through the lens of the three design perspectives. This approach provides a better understanding of how each level of a solution affects the three design perspectives. This view is taken in sections 8.2, 9.2, and 10.2 for the three levels.

Table 5.6 - Levels of a solution and design perspectives in interaction

	People Section 11.2.1	Business Section 11.2.2	Technology Section 11.2.3
Solution Section 8.2	What value proposition does the solution offer the customer or user?	What is the business model and business process of the solution?	What potential does technology offer for the solution? Is the solution feasible with given technologies?
System Section 9.2	What benefits must the system deliver to realize the value proposition?	How does the system realize the business model?	What technologies are needed to implement the system? Is the system feasible with given technologies?
Element Section 10.2	How does the element realize the benefits of the system in interaction with the user?	How does the element support the business model?	What technologies are needed to realize the element? Is the element feasible with given technologies?

Secondly, the three design perspectives can be viewed from the perspective of the three levels. By taking this approach, we gain a better understanding of how design work along the three perspectives is influenced by work at the levels. This view is taken for each perspective in section 11.2.

Table 5.7 shows important design issues at each level in the three design perspectives and refers to the sections in the handbook that flesh out each level with detailed aspects or look at the levels through their respective perspectives.

Similarly, the three levels of a solution can be used with the design perspectives to structure the evaluation of a solution in terms of design. The following table provides an overview of the questions along the levels and perspectives and refers to the relevant passages in the handbook that flesh out the corresponding questions. This view is taken in sections 8.3, 0, and 10.3.

Table 5.7 - Questions for evaluation along the three levels and the three design perspectives

	People	Business	Technology
Solution Section 8.3	Are customer and user groups meaningfully defined and understood? Have all relevant stakeholders been identified? Is the solution desirable and attractive to the client and relevant stakeholders, as well as the customer/user groups?	Can the solution be operated in an economically viable manner? Can the solution be realized in an economically viable manner?	Is the solution technically feasible? Does the solution exploit the potential of the existing technology?
System Section 0	Are the user types meaningfully defined and understood? Is the system desirable and attractive to the client and relevant stakeholders, as well as to users?	Can the system be operated in an economically viable manner? Can the system be realized in an economically viable manner?	Is the system technically feasible? Does the system make good use of the potential of the available technology?
Element Section 10.3	Is an element desirable and attractive to the client and relevant stakeholders, as well as to users?	Can an element be operated in an economically viable manner? Can an element be realized in an economically viable manner?	Is an element as a whole technically feasible? Does an element make good use of the technology's potential?

5.4.3 Mastering the complexity of the solution in the course of the process

Finally, the models in combination serve to focus the work on meaningful aspects and perspectives during the building process (see Figure 3.5). In particular, we can answer the question of which level of detail and intensity which level of a solution is considered in in which step of the building process in order to design a good solution. Table 5.8 provides an overview of relevance (How much does the level need to be focused?), focus (What does the level need to focus on?), and design freedom (How much can be designed?) along the levels in the three steps. It is essential that in each step, a different combination of levels is of particular relevance. Sections 8.4, 9.4, and 10.4 address and explain these classifications.

Section 11.2 continues to demonstrate the importance of an iterative approach between the three steps to holistically design a solution.

Table 5.8 – Structuring the holistic design of a digital solution based on the three levels and three design perspectives

	Scoping	Concept work	Development and operation
Solution	**High relevance** Idea for value proposition and solution in terms of vision **Full freedom of design** (Taking into account the given constraints)	**High relevance** Evaluated design for an attractive, viable, and feasible business model/business processes **High freedom of design** (Within the framework of the vision)	**Medium relevance** More detailed specification of the business model and the business processes parallel to the realization of the system **Decreasing freedom of design** As realization progresses
System	**Low relevance** Contributions to the understanding of feasibility and innovative ideas for the vision **High freedom of design** (In interaction with the solution level)	**High relevance** Evaluated design for an attractive, viable, and feasible system **High freedom of design for the system design** (In interaction with the solution level)	**High relevance** More detailed specification of the system design in parallel with the realization of the system **Decreasing freedom of design** As realization progresses
Element	**Low relevance** Contributions to the understanding of feasibility and innovative ideas for the vision **High freedom of design** (In interaction with the solution and system level)	**Medium relevance** Evaluated design for success-critical aspects of own elements **High freedom of design** for the element designs (In interaction with the solution and system level)	**High relevance** Realization-ready and evaluated design of the own elements **Medium freedom of design** (Limited by element realization concepts)

6 Overview of fundamental digital technologies

This chapter introduces important digital technologies relevant to the foundation level. The structure of the chapter follows the two levels of the FFQ model from section 3.2 and distinguishes technologies at the perceivable level and the underlying level.

As a linguistic simplification, we use the terms perceivable technology and underlying technology. When useful for understanding, we make specific reference to form, function, and quality at the two levels.

6.1 Introduction to perceivable technologies

EO 6.1	Name examples of perceivable technologies (L1)

Perceivable technologies are used to realize those parts of a digital solution that a stakeholder can perceive directly. For the foundation level, knowledge of end user devices and interaction technologies is important.

6.1.1 End user devices

Standardized end user devices, such as notebooks, tablet computers, or smartphones, are often used to realize a digital solution. These devices can generally be classified as perceivable technology. Nevertheless, these devices also offer technical features that can be assigned to the underlying technology (e.g., wireless communication technologies such as WLAN or Bluetooth).

However, Digital Design has no direct influence on the internal structure of such standardized end user devices, therefore the perceivable form is the most appropriate category to consider in this context. If standardized end user devices are used as part of a digital solution, the assumed technical capabilities—for example, screen size and resolution, communication technologies, performance of the processors, memory size—must be clearly defined in order to provide the necessary resources for the digital solution.

The following distinction helps to differentiate between rough classes of end user devices:

Stationary computers

Because of their fixed location, stationary computers can use a power socket as a power source, and the space occupied is—in principle—not limited. Therefore, this category can have the highest computing power among the classes defined here. This means that very fast central processing units (CPUs) and graphics processing units (GPUs), large working memory, high mass storage capacities, and large displays are possible.

Mobile computer

Devices that are transportable and can therefore be used in different locations. This class ranges from multi-purpose devices such as notebooks and tablet computers to smartphones. Because of their portability, these devices usually need to be powered by a (rechargeable) battery and can only have a limited size to fit in the space provided, for example, in a pocket, bag, or to be placed in a specific location. A power outlet or wired network connection can be used only in some usage scenarios. Therefore, the available computing power and memory are smaller compared to stationary computers.

Portable devices

Devices that are worn on the body or even implanted into the body (also known as wearables). Examples are activity trackers, blood glucose meters, smartwatches, or other small devices that can be carried. These devices may use only a very limited space and must run on (rechargeable) batteries only. Compared to the other two classes in this list, these devices have significantly lower computing power and the lowest available memory profile. Also, the display size is very small. Some devices do not have a display and must use other interface technologies—such as voice technology or wireless technologies connected to another device—to exchange information directly or indirectly with the user.

Embedded devices

Devices that are integrated into larger devices or objects. Examples include single-purpose devices such as card readers, fingerprint or retina scanners, or multi-purpose devices as control units in smart homes or vehicles. Like portable devices, these devices usually have limited size and capability. Similarly, the available direct interfaces to the end user are often limited or must be established via other end user devices. For example, lighting control in a smart building can be controlled via interfaces based on light switches or via a smartphone app.

This rough distinction is not without overlap; there are many examples of devices that can be assigned to more than one of the classes above. A clear assignment of devices to a class is not relevant for Digital Design. With regard to material competence, it is more important to address the question of which class of equipment is appropriate for which application and whether the performance of the equipment under consideration bears a reasonable relationship to the costs and requirements of the solution. For example, using standardized stationary or portable devices is much less expensive than developing a specialized embedded device.

6.1.2 Interaction technology

Modern interaction technology consists of a combination of complex hardware and software systems that are counted as perceivable form and function. For example, a working smartphone touch screen, which enables touch interaction, requires a complicated combination of hardware and software components in order to generate an appropriate response by the device to the user’s touch on the screen.

From the perspective of the DDP, the dynamic aspects of this user experience are most relevant, that is, the interaction flow enabled by the combination of hardware and software of the interaction technology.

Below, we give several examples to provide an overview of interaction technologies currently available.

Graphical user interfaces (GUI)

A widespread type of user interface is the graphical user interface (GUI). When using a GUI, the user navigates—usually within a window—through menus on a screen and selects (*menu*) items or icons with a cursor (or other navigation device) or via the keyboard. This kind of interaction through mouse and keyboard is indirect, as the item to be selected on the screen cannot be directly clicked on the screen with the finger, for example. Alternatively, a user can also interact directly, for example, with the finger (*touch operation*) or with an input pen. One advantage of a GUI is that it shows all available commands of the interface through menus, icons, and other graphical elements. The user finds commands by exploring these items and does not have to remember any specific commands.

Command line interfaces (CLI)

Before the existence of GUIs, command line interfaces (CLI) were a very common way of interacting with computers. In a CLI, the computer displays a prompt on the screen and waits for the user to enter a command or instruction that will cause the computer to perform a function. The user must type this command into the CLI manually. The user generally has to remember possible commands and their options. Usually, there is only a limited possibility to explore these commands via additional help commands or options, if these are available at all. A CLI can still be activated in many modern operating systems, such as Android, Mac OS, Linux, or Windows to solve specific tasks. Today, these interfaces play an important role in batch processing, for IT administration, or for use by computer programming experts in general.

Voice user interface (VUI)

A more complex audio interface is a voice interface or voice user interface (VUI), which enables voice output and voice input. Voice output usually works via a text-to-speech (TTS) system, which allows written (digital) text to be transformed into a voice output. The input part of a voice user interface is usually a speech-to-text system that uses speech recognition to identify the content of the spoken words. Such input systems have already been in use in call centers for a long time for forwarding customers to the relevant employees.

Modern applications are the voice assistants (e.g., Google Assistant, Apple Siri, or Microsoft Cortana) in smartphones or smart speakers. In these application examples, voice input and output have been combined with an artificial intelligence system to provide intelligent support.

Mixed reality interfaces (VR, AR, MR)

Milgram and Kishino [MiKi1994] categorize visual displays in what is referred to as the virtuality continuum (see Figure 1.6). At one end of this continuum is the real environment, that is, reality, and at the other end is a completely virtual environment in the sense of a virtual reality. Between these ends a combination of virtual elements and reality, which is called *mixed reality (MR),* is presented to users. One example of MR is the concept of *augmented reality (AR)*, in which the real world or images of the real world are enhanced virtually with artificial elements. In an *augmented virtuality (AV)*, elements of the real world are added to a virtual environment.

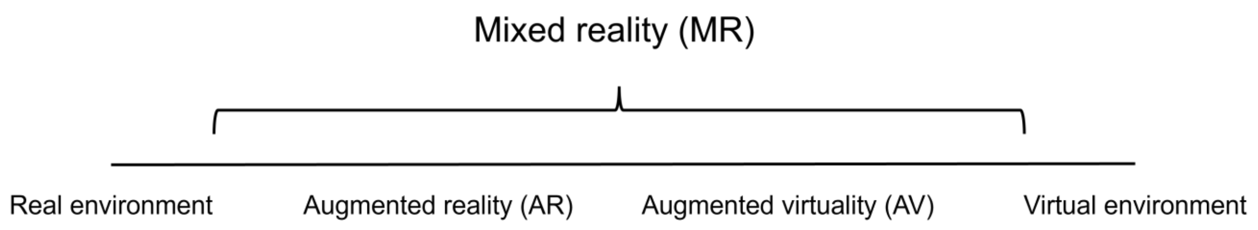

Figure 1.6 – The virtuality continuum according to Milgram and Kishino [MiKi1994]

There are currently many user interface technologies that can be divided into the categories of virtual environment and augmented reality.

6.2 Introduction to underlying technologies

EO 6.2 Name examples of underlying technologies (L1)

Underlying technology is beyond the parts of a digital solution that a user has direct contact with. Nevertheless, the underlying technology can have a significant influence on the solution or system design and therefore have a considerable, even if indirect, impact on the user experience. A DDP needs to have at least a basic understanding of underlying technology to identify parts of the design that may be hard to realize.

For the foundation level, technologies for programming, technologies for operations, and communication technologies are important.

6.2.1 Programming technology

Programming technologies determine a significant part of the underlying form, function, and quality of a digital solution. Beginners often intuitively associate programming technology with programming languages. This often leads to the misunderstanding that writing program code is the central activity in the development of software. This is mainly due to the fact that entering program codes in development environments is the most important visible activity of software developers. However, the central activity in the development of software is rather the creation of a suitable structure in the chosen programming technology, which actually realizes the desired form, function, and quality. This competence belongs to the activity areas construction and realization and is not part of the competence profile of a DDP at foundation level.

For the foundation level, the understanding of the perceivable and underlying form, function, and quality already introduced (see chapter 3) is sufficient as a foundation for communicating the desired digital solution to experts in construction and realization.

However, programming technology goes beyond programming languages. Programming technologies offer a wide range of features relevant to the design of digital solutions.

The following programming technologies are relevant for the foundation level:

- Technologies for storing large amounts of data (e.g., databases)
- Technologies for software frameworks (e.g., technologies for machine learning or business process implementation)
- API technologies for the use of external services and systems (e.g., route planners or payment service providers)

In addition to the technical capabilities of the technologies, it is important for Digital Design to recognize that there is often a cost associated with using technologies. These costs include license costs as well as usage costs. For example, many external services are billed based on usage (e.g., payment service providers often charge a percentage fee). These costs must be considered in the design and selection of technology as part of the business model.

Furthermore, the selection of a technology is associated with dependence on the provider. Providers can change functions or even terms of use and thus influence their own solution.

6.2.2 Technologies for operating software

Technologies for operating software are an important part of the infrastructure for building a digital solution. It consists of computer hardware, with processors, memory, and data storage as the typical components, and operating systems. Computer hardware is mostly built in large volumes as a standardized commodity.

Specialized hardware (e.g., for data encryption) is used when certain quality requirements (in particular, speed and security) cannot be achieved with standard hardware.

Operating systems are required to manage the computer hardware, provide basic software services such as organizing data storage with a file system, and also to provide an environment for running application software.

Hardware and its operating systems can be provided in several ways:

- Part of a standard device (e.g., a smartphone or white label components)
- Part of a custom-made device (e.g., a DIY smart home controller)
- Local server (e.g., a desktop computer)
- Remote server (e.g., in a data center)
- Service on-demand over the Internet (e.g., cloud computing)

As with programming technology, operating technology is also associated with costs; the level of costs differs, for example, depending on the performance and availability of technology. From a Digital Design perspective, it is therefore important to know the wide range of technologies and their costs in order to take this knowledge into account at an early stage when designing a solution.

6.2.3 Digital communication technology

A core feature of the digital age is connectivity on all levels between users and devices. Most digital business models rely on the capability to offer services regardless of location. The backbone of this capability is the underlying technology that enables this digital communication.

Communication technology consists of communication hardware, such as cables, antennas, transmitter units, receivers, etc., which is operated by a stack of protocol layers realized with computer hardware and communication software. Together, they provide communication services at various levels, for example:

- Basic services such as Ethernet, WLAN, Bluetooth, and mobile, cellular telephony including 5G, radio-frequency identification (RFID), near-field communication (NFC), and infrared (e.g., for face recognition)
- Network services such as the Internet or the network that connects phones when a number is dialed
- Application services such as WWW or email

6.3 Communication technology and machine learning as examples of capabilities and limitations of digital material

EO 6.3	Explain capabilities and limitations of communication technology and machine learning as examples of digital material competence (L2)

This section uses communication technology and machine learning as examples to discuss the capabilities and limitations of digital technologies. These two examples are intended to help make the idea of digital material competence more comprehensible on the one hand, and on the other, to convey material competence for two important technologies.

6.3.1 Capabilities and limitations of communication technology

We introduced communication technology as underlying technology in section 6.2.3. From a Digital Design perspective, communication technology is a *ready-to-use* technology, meaning that this technology can be used in a variety of ways in digital solutions.

Communication technology in relation to form

In terms of form, communication technology enables the realization of a digital solution that consists of multiple elements (i.e., more than one) interacting with each other. As mentioned above, almost every digital solution consists of more than one element.

Different communication technologies allow for different forms. Communication technologies with a short range, such as Bluetooth or WLAN, enable local networks and also allow elements to be connected within a short range at no additional cost to the user.

A good example is the Bluetooth connection between a smartwatch and a smartphone for a sports app.

When it comes to communication over longer distances, the Long Range Wide Area Network (LoRaWAN) technology is available on the one hand, and commercial Internet providers on the other. Various technologies are available for the commercial Internet, mobile phone technology (e.g., LTE or 5G), and direct network connections (e.g., DSL, FTTX). Commercial Internet connections incur additional costs for a digital solution. The user of the solution must have an Internet connection. Furthermore, the provider of the digital solution must also have an Internet connection.

Although Internet connection is a mass product today, the additional costs should be taken into account in the business model, especially if the digital solution may create high-volume data transfer. A good example is a video streaming solution. The costs for the data connection for the client and the provider can be substantial, since video streaming is a data-intensive function.

Communication technology in relation to function

In terms of function, communication technology is normally invisible for the user since it transports data between elements. However, when it comes to a connection failure or to a weak connection, a well-designed digital solution can adapt itself to this situation. The concrete method of dealing with communication issues depends on the type of solution.

In general, it is possible to define functions in such a way that they can handle interrupted connections. Depending on the specific application, such solutions are more or less complex. Two examples of this situation are caching and the offline capability. With caching, a solution stores a certain amount of data in order to use this in the event of a short-term connection failure. Music or video streaming solutions are an example of this.

With offline capability, a solution is implemented in such a way that the solution can execute certain partial functions with the existing data and finishes as soon as the Internet connection is restored. Offline capability is found, for example, in email programs such as Outlook: emails on the end user device can be read and answered without Internet access. The emails are then actually sent as soon as Internet access is available again.

Communication technology in relation to quality

Communication technology influences the quality of a solution through two factors: bandwidth and latency.

The bandwidth defines the volume of data that can be transported between two elements in a given time. Different technologies have different bandwidths. For example, mobile technologies offer different bandwidths depending on the technology chosen as well as signal strength and other environmental conditions.

The bandwidth becomes especially important when larger volumes of data have to be transferred from one element to the other. A low bandwidth will reduce the speed of a function and can therefore significantly reduce the quality of such a situation. Digital solutions can basically be implemented in such a way that they can handle different bandwidths. An example of this is again video streaming solutions: depending on the bandwidth available, the resolution of the video image can be reduced or increased to use the bandwidth available to display an acceptable video image.

The latency indicates the length of time by which a communication is delayed. An illustrative example of latency is video conferencing. If the latency of the connection between two people involved in the conversation is small enough, then the communication acts as if the two people are in the same room. At higher latencies, the communication is no longer synchronous and the communication appears unnatural. Analogous to the bandwidth, different technologies offer different latencies.

6.3.2 Capabilities and limitations of machine learning

Machine learning is a category of technology that can be used to realize underlying functions of a solution. Functions based on classical algorithms are realized by explicitly defined rules and sequences of instructions. In contrast, machine learning follows the approach that a function is trained on examples with the help of training data. Training data generally consists of the input data and the associated expected results. Based on this training data, the technology learns the relationship between inputs and results and stores this knowledge in a statistical model. Based on this statistical model, the function can then analyze unknown data sets and provide a probability-assigned answer[4].

Machine learning in relation to form

Even though machine learning is a technology for realizing underlying functions, the use of machine learning through the training of the model is related to the form of a solution.

Greatly simplified, training can be realized in a dedicated training environment or within the solution in operation. A training environment is an environment that is independent of the solution in operation and on which the training of the function takes place. In the other case, the training takes place in the environment of the solution. Both approaches have advantages and disadvantages.

An advantage in the case of the training environment is that the model can be trained and tested under controlled conditions. The disadvantage of the training environment is that the training data must be prepared systematically and that the model must be transferred to the operating environment at a suitable time.

An advantage of training in the operational environment is the possibility of ongoing training based on real user feedback. The disadvantage, however, is that the model must be monitored extensively to avoid undesirable developments.

Machine learning can also be used without any training at all by using pre-trained models. For example, trained models can be acquired for defined functions or external services can be used.

[4] A clear introduction to the technology of machine learning is provided by the website https://teachablemachine.withgoogle.com

Machine learning in relation to function

Machine learning is a powerful technology that can be used to realize a wide range of functions. In relation to the function, machine learning can be divided into analytical and generative machine learning. Analytical machine learning can analyze, evaluate, and modify existing data (e.g., translation of texts into other languages). Generative methods can generate new content based on an input (e.g., generation of images based on text).

Looking at the capabilities and limitations of machine learning, it is essential that machine learning can realize functions that are very difficult to realize using classical algorithms. For the design of a solution, the decision for or against the use of machine learning is a fundamental one, since the use of machine learning sets the course for the realization and operation of a solution.

For solutions with innovative features in particular, machine learning is often misunderstood as a miracle technology and the effort to realize a working solution is underestimated. In particular, training an innovative function can require considerable effort (e.g., obtaining good training data) and also has a significant risk of failure.

Machine learning in relation to quality

With regard to the quality of a solution, it is important to note that machine learning is a technology based on statistical methods. More generally, this means that the results of such a function are always subject to uncertainty and machine learning can make mistakes. One possible approach for dealing with this circumstance is a transparent approach to probability. For example, the probability assessment can be displayed to the user so that the user can decide for themselves whether the probability is sufficiently high to trust that result.

Another aspect related to the quality of machine learning-based solutions is the training data and possible problems in the training data. Basically, a lot of attention should be paid to the selection and definition of the training data, because the quality of the machine learning depends directly on the quality of the training data.

In addition to ensuring that the content of the training data is correct, a second perspective is important. Depending on the function realized, there may be unwanted tendencies and problems in the training data, which can go as far as discrimination. Examples of this can be found on the web under the catchphrase *discrimination by machine learning*.

More generally, functions based on machine learning require special consideration in relation to evaluation in order to achieve adequate quality of the functions realized. This requires a solid base of training data and focused evaluation and validation of results. This aspect is a further argument for the early involvement of appropriate experts in the building process.

7 Fundamentals of design work

In this chapter, we look at the essential fundamentals of design work. First, we consider the design work itself. We then present essential tools for design work in the form of design concepts and prototypes. To conclude this chapter, we look at the use of prototypes in different disciplines to support interdisciplinary work with prototypes.

7.1 Thinking models for design work

EO 7.1	Explain the fundamental thinking models of design work (L2)

Learning to design is like learning any other complex skill: it requires education and training [Cros2006]. To this end, design education relies on models of thinking about design processes and a description of design work. These models are important for developing your own understanding of the functionality of design (cf. [Dors2003]). A design model can be used to gain a basic understanding of how design works. This provides a good starting point for the foundation level.

In this section, we first present a thinking model that provides an initial understanding of the design process and allows us to discuss important aspects of any design process. Second, we present a model that allows us to discuss the nature of design work. As a third model, we introduce the difference between tame and wicked problems as two fundamentally different categories of problems in design work.

7.1.1 The design squiggle

The first model we look at here is the design squiggle by Damien Newman ([Newm2020], see Figure 1.7). This thinking model shows that a design process is typically a chaotic and iterative activity that ultimately leads to a clear understanding of a particular design solution. The design squiggle consists of three phases that merge into one another:

- *Research & synthesis*: This phase is about understanding the problem space and gaining insights into users, customers, and the situation at hand. It is typically characterized by a rather unstructured journey with a lot of swooshing and uncertainty. However, at some point in time, the process leads to an understanding of the problem.
- *Concept/prototype*: Once an initial understanding of the problem has been achieved, initial concepts and prototypes can be created to explore possible solution ideas. This phase is not really linear either. Concepts or prototypes may lead to completely new insights into the problem. They may even mean that the original understanding of the problem has to be discarded completely and that the process has to start all over again.

- *Design*: At some point in time, one solution idea comes forward as the final solution. The process now becomes more linear, since the one solution has to be elaborated in all its details until it becomes the final design. It is important to understand that in this figure, the term design refers to the end result (design as a noun) and not to the activity (design as a verb) (see section 4.2.1.1).

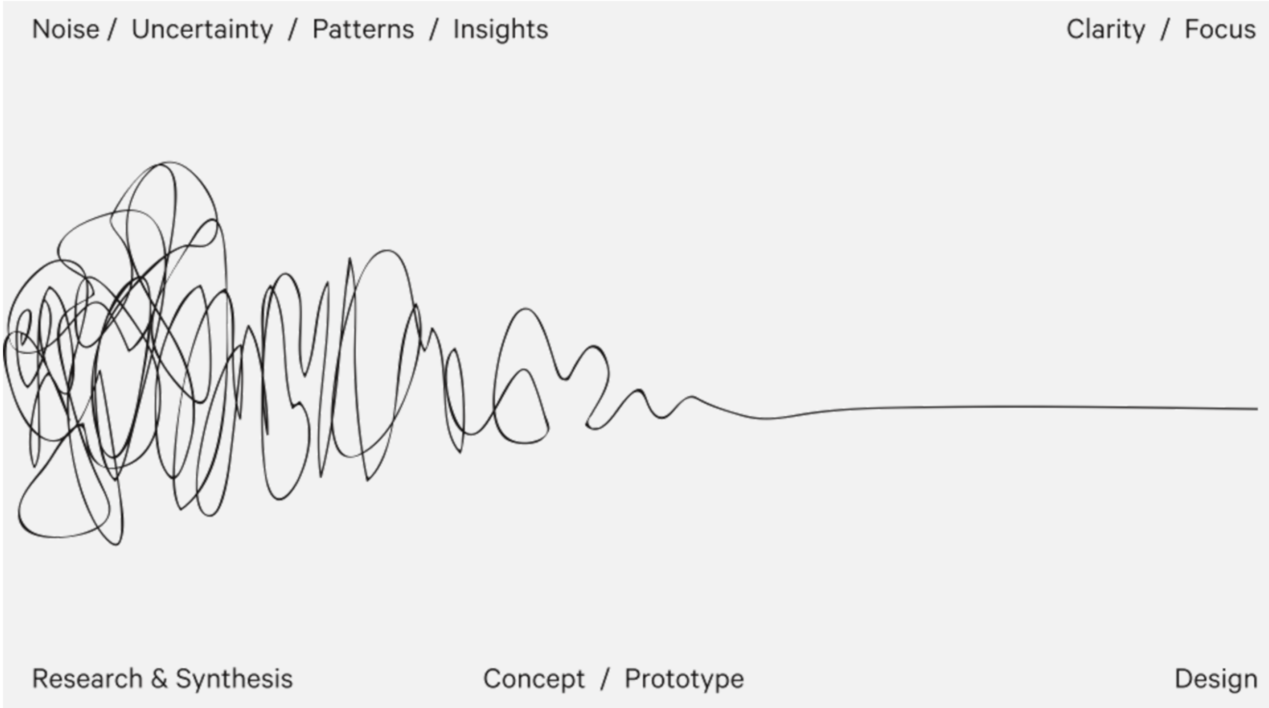

Figure 1.7 – The design squiggle [Newm2020]

The three phases of the design squiggle can be applied to the building process of a digital solution on the one hand as a whole, and on the other as a basic attitude. In terms of the building process, the phases of the design squiggle can be applied as follows:

- The *Research & synthesis* phase corresponds to the *scoping* step as it involves an understanding of the problem in its entirety.
- The *Concept/prototype* phase corresponds to the *concept work* step, since it involves developing an idea for a solution to the problem.
- The *Design* phase corresponds to the *development and operation* step, as it is involves the detailed elaboration and the factual implementation of the solution.

Furthermore, the design squiggle introduces three important core aspects of the design process that are part of the attitude in Digital Design, regardless of the process.

Attitude 1: Before working on solution ideas, the environment should first be understood

A typical beginner's mistake is to understand design only as the third phase of the design squiggle. A prerequisite for creating good design solutions is a proper understanding of the overall environment for which the solution will be designed. Special emphasis should be given to the context of use, that is, to the customer of the solution and the environment in which the solution will be used.

A profound understanding of the environment is important because many design decisions depend on the manifold details of the environment.

Attitude 2: ongoing evaluation of everything

A second beginner's mistake is to believe too quickly in your own understanding of the environment and the solution ideas that have been created. The design squiggle clearly shows that the design process is a rather chaotic one that goes back and forth between the different phases. One reason for this is that the initial understanding of the environment is seldom right and that initial solution ideas are not the best solutions. Inexperienced designers often learn this lesson when their solution ideas fail with the customer or the market.

Experienced designers have learned to tolerate the uncertainty and have made the evaluation of everything (understanding of the environment and of solution ideas) part of their attitude, always looking out for opportunities to evaluate their understanding and their design.

This does not mean that experienced designers always apply laborious evaluation methods; it only means that experienced designers are aware of the limits of their own understanding.

Attitude 3: iteration is the normal working mode

A third beginner’s mistake is to assume a linear design process. The design squiggle highlights that every design process is a rather chaotic and iterative process. Only at the end, when the solution idea is really clear, does the process become more linear.

Experienced designers have learned to tolerate this way of working and have made iteration (that is, the continuous understanding of the environment and working on solution ideas again) the normal way of working. It is only through iterations that several solution ideas can be created and evaluated to identify those solution ideas that really are promising.

This does not mean that the whole process and the whole team performs an iteration—sometimes, an iteration can also be a rather short event that takes place only in the mind of a designer.

7.1.2 The dual-mode model of design

Becoming a good designer means, in particular, developing a distinct individual personality. A first step in developing the individual personality as a designer is offered by addressing the distinction between the paradigm of rational problem-solving (objective mode) and the paradigm of reflective practice (subjective mode) from the dual-mode model of design [Dors1997].

The objective mode means looking at design work as objectively as possible, focusing on models, theories, (customer/user) research, principles, etc. The subjective mode means understanding design from your own experience (from working with design) and learning from the exchange with experienced designers.

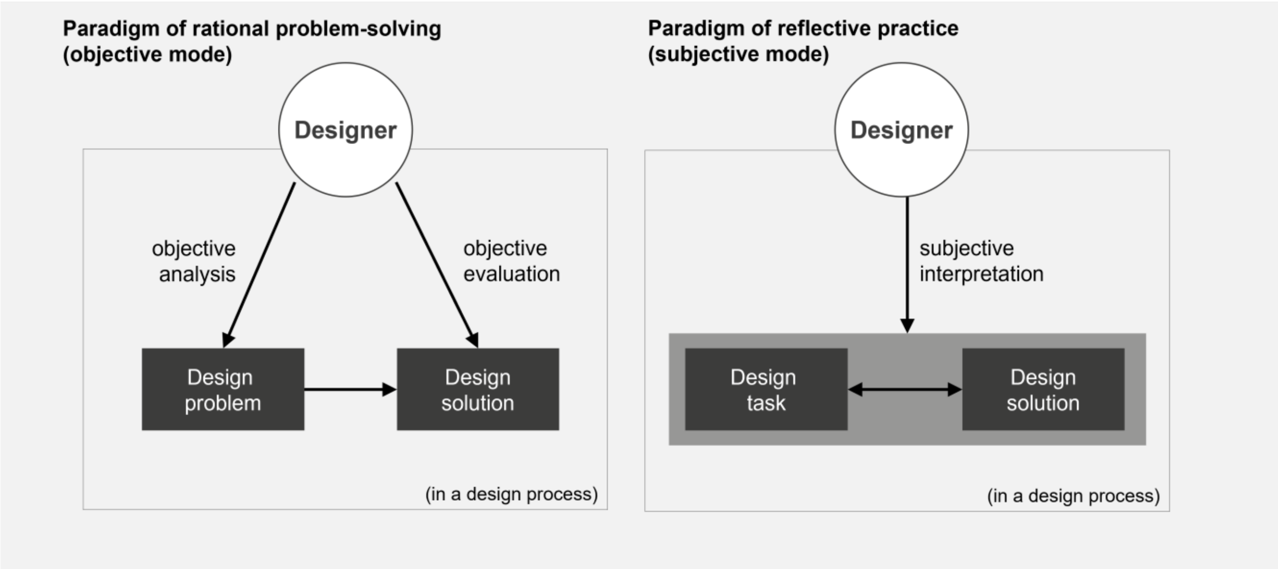

Figure 2.7 - The dual-mode model of design [Dors1997]

The dual-mode model of design [Dors1997] describes these two modes as follows (see Figure 2.7):

- *Paradigm of rational problem-solving (objective mode)*: In this mode (left part of Figure 2.7), the designer works on the given design problem as objectively as possible through rational analysis and observation with the aim of solving the problem. The goal is to conduct the process as rationally as possible and to work objectively. Above all, rational problem-solving means conducting problem analyses that are as objective as possible, rationally justifying design decisions to all relevant stakeholders, and making perceptions and interpretations explicit and negotiating with stakeholders.
- *Paradigm of reflective practice (subjective mode)*: In this mode (right part of Figure 2.7), the designer works subjectively on the whole situation to find a way to deal with the design task. The designer works on the design task (the procedure, the given problem, the given situation, and the time frame) in relation to the desired design solution. In the subjective perspective, the designer consciously incorporates existing skills, as well as their tacit knowledge and intuition, to find, evaluate, and select courses of action with stakeholders. An essential factor in this mode is the awareness that alternative perspectives on the design problem and task must be taken in order to find a meaningful approach and solution. The subjective mode is important when the design task is unclear, ill-defined, or when there are ethical or moral conflicts related to the design task.

In simple terms, both paradigms can be explained with the following example. Let us assume that in a process step of a digital solution, many mistakes are made during processing in a user interface (design problem) and the targeted design solution is to improve the user interface by providing appropriate hints and technical checks to reduce the error rate in this user interface. In the objective mode, a designer would immediately address the problem of designing proper hints and technical verifications. In the subjective mode, the designer would consider the design task as a whole. In the example, the designer has experienced in a previous situation that errors in a process are not necessarily caused by a user interface, but rather by other circumstances. Before accepting the design problem and the intended solution, the designer takes a step back in the process and analyzes the overall situation to find other causes for the errors, if there are any.

For beginners in design, this model provides three important lessons:

- Design can be approached with an objective or subjective attitude. Both modes are important, and a skilled designer must make use of the two modes and switch between them when necessary.
- Working on the design process (e.g., planning the next steps to analyze the problem or to create a prototype) is part of the reflective practice mode. There are many ways to approach a design problem. The message of the reflective practice mode is that the way to proceed in a given situation depends on subjective factors (e.g., experience, education, and intuition).
- When you are objectively stuck with a design problem or solution, the reflective practice mode can be helpful in understanding the big picture. Here, on the one hand, the process is questioned, and, on the other hand, both the given problem and the previous understanding are questioned.

7.1.3 Tame vs. wicked problems

The nature of the problem plays a central role in planning a design process. To capture the nature of a problem, the distinction between tame and wicked problems [RiWe1973] is useful to define two different initial situations of a process. A *wicked problem* can be defined as follows:

> Wicked problem: A problem that is difficult or impossible to solve because of incomplete, contradictory, and changing requirements.

The *tame problem* is the opposite of a wicked problem:

> Tame problem: A problem that is well-defined with clear and stable requirements.

Tame problems by their nature can be handled and solved using an analytical process (rational problem-solving, section 7.1.2). Useful approaches can be found in requirements-based approaches, such as those described in the CPRE [GLSB2022].

In contrast, wicked problems require a mix of reflective practice and rational problem-solving and the iterative approach outlined in the design squiggle. This is the only way to achieve a common and stable understanding of the problem with the client and all relevant stakeholders, which can then be used as the basis for designing a solution. However, even with such an approach, you cannot be sure that the solution designed will actually solve the wicked problem.

The distinction between tame and wicked problems describes two extreme categories of problems. In practice, design problems are usually a mixture of these two extremes, that is, a design problem usually has shares of tame and wicked problems. The sub-problems can then be addressed with the attitude described here.

7.2 Design concepts as a tool and result of design work

EO 7.2	Explain design concepts as a core design tool for design work (L2)

We have already introduced design concepts in terms of document templates for a work product in section 5.3.2. This section discusses important fundamentals of working with design concepts as a tool and result of design work. These fundamentals form the basis for explaining further details on working with design concepts at the various levels of a solution in the following chapters.

7.2.1 Fundamentals of design concepts

In general, concepts are ideas that are used in thought or in communication (written or verbal) and can be considered as elements of thought (cf. [MaLa2015]). In Digital Design, concept work means working mentally to create a digital solution, that is, formulating the goals of the digital solution and designing the corresponding form, function, and quality of the digital solution under the given constraints.

Design concepts (see section 4.2.1.1) can be built using a wide variety of structures, models, and templates. The concrete structure also depends on the process model and the way of working in the building process. Literature contains a wide variety of approaches and templates for building design concepts. For entry into the world of Digital Design and for a basic understanding of design concepts, concrete templates are of only limited use. Much more important to understanding design concepts are the essential aspects and perspectives that must be considered to design a digital solution. Therefore, in addition to the exemplary templates (see section 5.3.2), in chapters 8, 9, and 10, this handbook introduces fundamental aspects that structure design at the three levels of solution, system, and element for beginners in Digital Design.

Concepts can occur in a rather linear verbal/written form or in a highly structured technical form. We use the term design concept to distinguish it from concepts in other activity areas and disciplines.

In the following, we consider the benefits and limitations of working with design concepts in design work.

7.2.2 Benefit from design concepts

Mastery of complex solutions and systems

A design concept structures thoughts about the digital solution. In the following chapters 8, 9, and 10, we address this aspect by presenting different aspects of the respective levels that provide support as a basis for documents and at the same time, the structuring of thoughts about a digital solution, the corresponding system, and its elements.

This structured approach is important for understanding the system and solution as a whole and for identifying those parts of the solution/system that have particular complexity.

Communication tools and basis for construction and realization

A design concept communicates the digital solution to various stakeholders. To support this communication purpose, different types of design concepts need to be created to address the different information needs and prior knowledge of stakeholders.

A design concept serves, in particular, as the basis for the activity areas of construction and realization (see section 4.2). In contrast to requirements-driven work (e.g., from requirements or usability engineering), the concept work in Digital Design adopts the solution-driven perspective (cf. [Cros2006]) and focuses on concrete solution ideas instead of an intensive definition and analysis of requirements that lead to a solution.

This does not mean that requirements are neglected in Digital Design. Where necessary, requirements are documented and used, especially quality requirements, constraints, and goals. Furthermore, the stakeholders' requirements—in particular, those from the client and the customer—are crucial input when creating design concepts but are immediately converted into an appropriate design of the solution.

External memory for building solutions and systems

A design concept serves as an external memory during the whole life cycle of a digital solution. This is because the amount of information about a typical digital solution that is created during the building process far exceeds the capacity of human memory.

To serve as external memory, all design concepts must be revised and optimized continuously. This allows the typically complicated structures of solutions/systems to be mastered with reasonable effort.

Evaluation instrument and evaluation reference

A design concept can be used to evaluate the design of the digital solution described at an early stage, for example, with regard to business cases, customer or user acceptance, or legal issues (evaluation tool for the question *Is the right solution being built*?).

Likewise, a design concept can serve as a basis for evaluating whether the digital solution realized was realized according to the defined concept (evaluation reference for the question *Was the solution built correctly*?).

7.2.3 Limits of design concepts

Concepts need ongoing maintenance

Even the best and most detailed design concept is incomplete and never truly finished. The reason for this is simple: life is too complex to be anticipated and captured completely in a concept. However, this fact is not a weakness, because concepts are a means of communication.

Many weaknesses and missing details of design concepts are identified in particular when the development of a digital solution starts. At foundation level, it should be accepted that concepts are incomplete and will be extended and revised during the whole building process.

This work is by no means a waste of time. The constant work on design concepts is an important backbone for keeping the building process under control from a design perspective since the design concepts serve as external memory (see above).

Concepts need ongoing interpretation

A high degree of unambiguity is an essential quality feature of good concepts. Nevertheless, even the clearest concepts always leave room for interpretation. Interpretation is a core feature of human communication and since concepts are communication tools, they have to be interpreted.

When working with concepts, it is therefore essential to work continuously on a common interpretation within the building team and among all relevant stakeholders, thus cultivating a common understanding of the concepts among all parties involved. Prototypes (see below) are an exemplary tool for this, as is a regular exchange of concepts among all stakeholders.

For the foundation level, the need for interpretation should always be kept in mind and attention should always be paid to possible misunderstandings and misinterpretations of concepts in order to cultivate common understanding.

Concepts are not the digital solution

Working with concepts is one of the main tasks in Digital Design. However, concepts are only a means to an end. Concepts are means that serve the building process (see section 4.2.1.1). The goal of the building process is to bring a digital solution to life in order to bring about change.

Even at foundation level, concept work must be taken seriously. Likewise, you must always remember that a good digital solution is more important than good concepts.

Sophisticated concepts can create false confidence

Good design concepts with corresponding prototypes of the digital solutions can be very impressive, especially to inexperienced people in the client role. The downside of this impressiveness can be that clients and stakeholders can get a false confidence in the success and the current state of a digital solution and become too optimistic about its potential success.

This point is not an argument for creating sloppy design concepts, but rather advice for careful handling of concepts and managing expectations about the maturity of a concept.

7.3 Prototypes as a tool for design and evaluation

EO 7.3 Explain prototypes as a core design and evaluation tool for design work (L2)

Unlike concepts, prototypes are a tool that makes the digital solution tangible. First, we take a closer look at the use of prototypes. Then, we describe how different disciplines understand prototypes to support the collaboration with these disciplines.

7.3.1 Prototype fundamentals

Prototypes are used in different disciplines with different meanings. Common to all definitions is that prototypes are built to clarify, test, or validate something and learn from the results.

In Digital Design, we define the term *prototype* as follows:

> Prototype: A preliminary, partial instance of a design solution.

The use of prototypes is the key technique that enables us to achieve several (partly overlapping) goals in design (cf. [McEl2017]):

- Explore the problem, user needs, and requirements
- Communicate solution ideas and concepts
- Evaluate and improve concepts and solution ideas
- Advocate a solution or a solution idea

In all of these cases, the creation of prototypes supports the iteration of problems, requirements, concepts, solution ideas, and solutions (see the design squiggle, section 7.1.1).

7.3.2 Use of prototypes in the building process

Prototyping has three main purposes in Digital Design:

- Clarification of ideas: A prototype can manifest the idea for a future digital solution in a format to communicate the idea to others or to test it with customers or users.
- Model of the solution (or part of the solution): A prototype can be a model for later phases or for the final, complete version of the digital solution (or part).
- Evaluation model: A prototype can be used to get early feedback on a concept by providing a working model of the expected digital solution before it is actually realized.

The use of a prototype or prototypes is also referred to as *prototyping*. As stated in [Dick2019], the term *prototype* (the object) is sometimes used when *prototyping* (the process of working on and with a prototype) is meant.

Prototypes should always be made for a specific purpose, such as communication with team members or testing with customers or users. Such tests are often more difficult to perform on a design concept basis.

The following example will illustrate this: it is hard to test the idea of the future digital solution with users when the idea is presented as a list of user stories in a spreadsheet. Such purposes require a very concrete version of the early digital solution, system, or element.

A concrete prototype can be either physically tangible or intangible. An example of a tangible prototype is an interactive mock-up that enables the customer or user to interact with parts of the planned digital solution. Examples of intangible prototypes are simple sketches or storyboards that show the (early) ideas much more concretely than an abstract text of a concept can. To emphasize the context and the planned customer or user experience, short narratives in the form of storyboards or fictional (commercial) videos may provide clarity on the digital solution envisioned.

Creating a prototype takes a certain amount of effort. However, this effort is well spent, as the feedback gathered helps you to base decisions on more information and thus make better decisions. When you explore solution ideas in many fundamentally different directions using prototypes, many of these ideas will fail but will also generate new ideas for better solutions. This means that you must be ready to create a prototype for a single purpose only and discard it afterwards. Therefore, the scope of the prototype should be selected specifically and the effort required to create it should be carefully considered.

Chapters 8, 9, and 10 present various concrete examples of the use of prototypes at the three levels of solution, system, and element.

7.4 Understanding prototypes in different disciplines

EO 7.4	Know the usage of prototypes in different disciplines (L1)

Digital solutions are often created in an interdisciplinary context, that is, various disciplines participate in the building process. Since each discipline understands and uses prototypes differently, it is useful to know the meanings of prototypes in different disciplines (cf. [McEl2017]):

- Building architects, for example, work with floor plans, airflow models for ventilation, heating and cooling, daylight simulations to optimize the incidence of light through windows, material studies, and building simulations, where users can walk through the planned building.
- Industrial designers have a long tradition in extensive use of prototypes. They use sketches (e.g., created by real or digital pencils), foam models, or models from additive manufacturing (e.g., 3D printing); they conduct material studies, make scaled

mock-ups, and create final forms as prototypes before communicating the design result to manufacturing.

- Filmmakers use storyboards and previews to visualize the flow of a movie before actually filming the scenes.
- When developing electronic devices, designers usually create industrial designs and electronic designs in parallel. The positioning of electronic components influences the industrial design. Prototypes consisting of selected and assembled electronic components are useful for studying the implications for and optimizing the industrial design.
- Interaction designers who develop user interfaces of software applications use prototypes such as sketches, wireframes, coded prototypes, and visual designs to improve a solution iteratively.
- Software architects and software developers use coded pieces of software as functional prototypes to explore feasibility, verify requirements, or study certain quality aspects of alternative software solutions.

8 Design work at the solution level

The solution level looks broadly at the entrepreneurial perspective (business perspective) and the added value that a solution can provide for customers (people perspective).

At first glance, the solution level is strongly commercially oriented and uses the language of business. For instance, we ask how something is financed or purchased. Therefore, the solution level seems to fit primarily for entrepreneurial contexts (e.g., selling solutions as products, eCommerce). At second glance, this perspective is equally useful when applied to contexts that are not primarily economically motivated (e.g., in-house solutions).

In the following, we first present fundamental aspects for designing at the solution level (section 8.1) and how these aspects interact (section 8.2). We then address the evaluation work at the solution level (section 8.3). Finally, we consider design work at the solution level during the building process along with work on solution design concepts (section 8.4).

8.1 Fundamental aspects for designing a solution

EO 8.1	Explain the key aspects of conceptual design at the solution level (L2)

At the solution level, the design can be divided into the following fundamental aspects:

- Vision for the solution
- Customer/user group(s)
- Value proposition of the solution
- Value creation architecture for realizing the value proposition
- Business processes for realizing the value proposition
- Solution quality requirements
- Solution constraints

In the following, we explain these aspects and give examples for documenting them. The examples are not relevant for the foundation level but serve as references to further literature and to make the contents of the exemplary structure in the solution design concept more specific (see section 5.3.2).

8.1.1 Vision

The vision asks what a solution should achieve and why a client should set out to realize a solution. Thus, the vision is not only a content-based means for characterizing the solution as a target image, but in particular, also a means for the organization and the entire building process.

By working on the vision, the client and the associated organization formulate a substantive target image and also a clear motivation as to why everyone involved should set out and put their energy into a project. Likewise, based on the vision, the question of technical feasibility can be discussed.

From a Digital Design perspective, working on the vision is especially important at the beginning of a new endeavor to conceptually design a shared picture of the future solution and a common motivation. The term *conceptually design* is important here, because a vision is always just an idea, it describes the planned change at an abstract level but is not itself the solution. For this reason, the vision is part of the Digital Design brief and the scoping (see section 5.3.1).

However, the vision is not only important at the beginning of the scoping of an endeavor, it remains important for the entire life cycle of a solution as a reference and orientation point. At any point in the building process, you can take the previous solution or your own understanding of the solution and compare it to the vision. This allows you to check whether the solution fits the vision and whether you are still on the right track. Therefore, the vision is part of the solution concept (see section 5.3.2) and should also be adjusted during the building process when a clearer or changed understanding of the vision has emerged.

The vision is typically documented in the form of a text. A good technique for beginners is the future press release (cf. [Ross2019]).

8.1.2 Customer and user segments

We introduced customers and users as key stakeholders in section 4.1. Customers or users use a product or make use of a service. Understanding a solution's customer and user groups is therefore an essential anchor for any digital solution, because they are the reason a solution exists in the first place. If, for example, no customer asks for the solution, then the solution is probably not needed.

From a Digital Design perspective, working on customer and user groups is important for two reasons. On the one hand, you design the possible target groups of a solution through the customer and user groups (Who wants to use the solution?) and, on the other hand, you must clearly understand the desires and needs of the customer and user groups, as well as their respective context of use, in order to formulate a good value proposition. When defining customer and user groups, note that customers and users do not necessarily have to be identical (see section 04.1)—there may be customers who are also users. However, it is equally conceivable that customers of a solution are not users and that users of a solution are not customers.

For this reason, defining customer and user groups is the first important design decision of any solution and this decision should therefore be captured in the solution design concept (see section 5.3.2). It is only by making a clear decision, one way or the other, that you can define and ultimately better understand customer and user groups. How that decision is brought about is again a matter of process. Often, for example, there is already an initial understanding of the customer and user groups because there is a related product or a previous product.

For example, personas [PrAd2006], the context of use descriptions [ISO2018], or the customer profile of the value proposition canvas [OPBS2014] can be used to document customer and user groups.

8.1.3 Value proposition

The value proposition defines what added value a solution as a whole or its components offer to the customer groups. The value proposition is the central anchor for defining the solution as a whole and is therefore captured in the solution design concept (see section 5.3.2).

From a Digital Design perspective, working on customer groups and value propositions is always an integrated activity. Without a clear understanding of the customer groups, a clear value proposition cannot be formulated. And in turn, a better understanding of the value proposition creates a clearer picture of the customer groups.

Thus, the formulation of the value proposition is also a key design decision of the solution. The value proposition characterizes to a large extent the change in the environment that the design seeks to achieve: by realizing the solution, customers should benefit from the value proposition.

To document the value proposition, the value map can be used as part of the value proposition canvas [OPBS2014].

8.1.4 Value creation architecture

The value creation architecture defines, as part of the business model, which building blocks (e.g., the system with existing and proprietary elements, see section 9.1), people involved, and organizational structures are required to realize the value proposition and how the value proposition is monetized to finance the cost of a solution or to generate revenue. The value creation architecture is thus the essential factor in designing a commercially viable solution (business perspective, see section 5.15.1). Therefore, the value creation architecture is captured as part of the business model in the solution design concept (see section 5.3.2).

The costs of a solution can also be counted as part of the value creation architecture and can be very diverse. Costs include initial development, implementation and training, operation and further development, but also personnel costs for customer support, manufacturing costs for hardware, and licensing costs for software. Regardless of the types of costs, when designing the solution, it is important to think very seriously about the funding in order to consider costs and possible revenue equally and not to focus only on the question of what the customer has to pay for the added value and how a solution can generate profit. This part of the value creation architecture, while important for many solutions, would push the perspective toward profit and economics too early. Similarly, the question of value creation architecture may also entail a question of technical feasibility. For example, implementing a value creation architecture may require costly technologies that represent a significant portion of the costs (e.g., dedicated end user devices for customers).

The value creation architecture aspect is always important for understanding the cost of a solution and making it transparent to the client. Similarly, many digital solutions are free of charge in whole or in part (e.g., social networks or search engines) because the costs of the solution are not financed directly by the customer but via other channels (e.g., through advertising). Even in such situations, the value creation architecture must be designed and understood in a meaningful way in order to design a commercially viable solution.

For example, the business model canvas [OsPi2010] can be used to document the value creation architecture as part of the business model of a solution.

8.1.5 Business process

A solution's business process defines how added value is delivered to the customer based on the value creation architecture. The design of the business process is thus the first step into the concrete world of actions and activities and is therefore recorded in the solution design concept (see section 5.3.2).

By defining the business process, the following questions in particular are answered:

- Is the business process technically feasible?
- What steps does a customer need to take to access the added value? When and how does money flow?
- Does a product have to be delivered?
- What happens when a problem occurs in the process?

For holistic design work, the business process should be viewed more broadly and we must also consider how the customer learns about the solution in the first place. In this context, literature refers to the *customer journey* (cf. [TuPA2021]). This journey can basically be divided into three parts:

- Before the purchase, everything happens that leads to the customer learning about the solution and choosing it.
- Acquisition/use looks at the part of the process after the customer has decided on the solution and then defines the actual acquisition and use of the solution.
- After the end of use (e.g., at the end of the life of a solution or even if the use is limited in time), the time after use follows. During this time, a customer can reflect on the usage, decide to use it again if necessary, or recommend the solution to others.

Modeling techniques such as BPMN (Business Process Model and Notation [OMG2014]) are suitable for documenting business processes. Customer journey maps [Angr2020] can be used to document the customer journey.

8.1.6 Quality requirements (solution)

Solution quality requirements describe qualitative aspects of the form or function of the solution. The aspect of quality requirements is important for the design of the solution, because quality requirements have a great influence on the acceptance of the solution by customers and users (people perspective), on the economic viability (business perspective), and the technical feasibility (technology perspective).

A simple example of a quality requirement for a solution is the high reliability of delivery time predictors in online retail. Reliable prediction of delivery time is positive from the customer's perspective, as the desired product is delivered on time (people perspective) and increases the likelihood that a customer will buy again from the same retailer (business perspective).

The concrete details and characteristics of the quality requirements are developed in collaboration with all relevant stakeholders. This involves not only the customers and users, but also, depending on the type of solution, industry experts, process experts, and technology and implementation experts.

Working on quality requirements at the solution level is teamwork. However, like the design of the solution, the essential responsibility for formulating good quality requirements lies as a whole with Digital Design.

For documentation and working with quality requirements at the solution level, there are whole series of practical hints and procedures, especially from business analysis [IIBA2022], requirements engineering [GLSB2022], and service design [PoLR2013]. A basic technique for documentation is the use of textual descriptions. In addition, there are a number of norms and standards that specify various aspects of quality requirements (e.g., [ISO2011]).

8.1.7 Constraints (solution)

Constraints at the solution level are often organizational or regulatory in nature and arise from the industry, context, or domain of a solution. To formulate constraints systematically, it is therefore essential to consult appropriate experts. Examples of constraints at the solution level are regulations on data protection, but also regulations on the technical realization (e.g., for the storage of data) or for the evaluation of a solution (e.g., medical devices must be certified depending on the country).

Formulating solution constraints is an essential responsibility of Digital Design. Business analysis [IIBA2022] and requirements engineering[GLSB2022] provide important input for documenting the constraints on a solution. Textual descriptions, for example, are a good choice.

8.2 Use of aspects in interaction when designing the solution

EO 8.2	Apply conceptual design to create design concepts at the solution level (L3)

To use the aspects for design work at solution level, it is important to understand that each of the above aspects is equally important for solution-level design. For the conceptual design work at the solution level, this means that the aspects must be considered and designed in interaction.

What is important for good design thought at the solution level is an awareness of the interrelationships of the aspects (see Figure 1.8):

- The vision defines why the client is setting out the solution and what they want to achieve with it. Based on the vision, the following can be defined: the form, function as a means and path, as well as initial quality requirements and constraints.

- The value proposition as part of the function specifies, from the customer's point of view, what added value the solution should offer and thus makes what the client wants to achieve more specific.
- The customer and user groups make the understanding about the people to whom value is to be provided and who are involved in the value creation architecture more specific.
- With an understanding of the customer groups, you can define how the value proposition can be embedded in a value creation architecture to design a commercially viable and technically feasible solution.
- Quality requirements for the solution can be used to formulate important qualitative properties that are necessary for the success of the solution.
- With knowledge of the constraints, value proposition, value creation architecture, and customer/user groups, the business process can be clearly formulated. With this knowledge, the vision can then be questioned again and the design of the solution can be scrutinized in terms of iteration.

At the end of this list is again the vision. This highlights the iterative nature of the design work at the solution level.

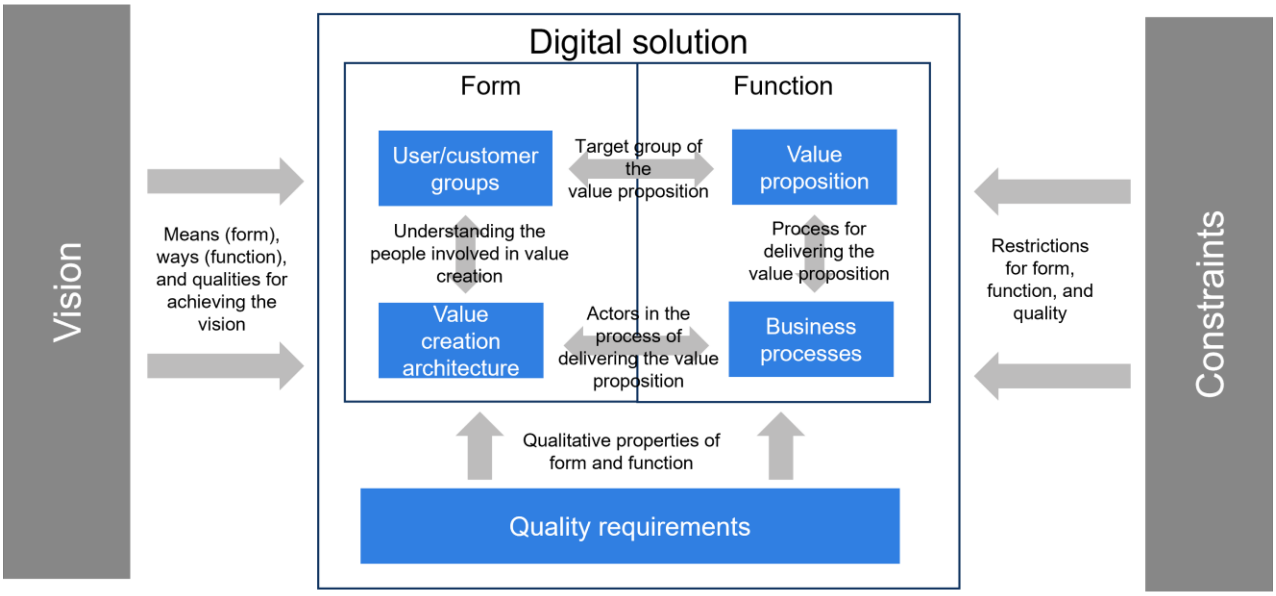

Figure 1.8 - Aspects for designing the solution in interaction

To apply the aspects in design, the vision is the starting point of the design work. Once there is an initial understanding of the vision, the design work is an iterative process that uses the interaction described to approach a good solution-level design in terms of the three design perspectives (see section 5.15.1).

For example, a better understanding of the value proposition leads to a better understanding of customer groups and an initial idea of implementation in a value creation architecture and business processes. This knowledge can be used to improve the definition of the value proposition and the process starts again. In this way, a better overall design of the solution is elaborated.

With regard to the question of technical feasibility, however, care should be taken to ensure that the details of technical implementation are a system-level issue. At the solution level, only the question of technical feasibility should be asked and considered as part of the design work.

8.3 Evaluation of the design work at the solution level

EO 8.3 Explain the essential questions for evaluation work at the solution level (L2)

In addition to the development of designs, the evaluation of the design work is also an essential part of the design work at the solution level.

For beginners in Digital Design, the three design perspectives (section 5.15.1) provide a helpful structure for evaluation. In the following, we use the three perspectives to consider essential issues for evaluation at the solution level. In this context, *essential* means that the questions characterize fundamental topics of evaluation in order to provide a broad overview of the spectrum of evaluation of design work at foundation level. We do not present specific techniques here because the selection and application of evaluation techniques are beyond the foundation level.

The following note is important for learners: the compactness of the presentation should not be misunderstood to mean that the individual questions are easy to work through. Rather the opposite is the case—behind all questions lie extensive areas of knowledge that must be consulted for the evaluation depending on the type of solution.

Evaluation with a view to the people perspective

Based on the people perspective, the design work can be evaluated using the following key questions:

- *Are customer and user groups meaningfully defined and understood?* The evaluation of customer and user groups ensures that the solution is implemented for the right customers/users and that the requirements of the customer/user groups are taken into account during the process.
- *Have all relevant stakeholders been identified?* The evaluation of the stakeholders identified ensures that all relevant stakeholders have been considered.
- *Is the solution desirable and attractive to the client and relevant stakeholders?* The evaluation of the solution designed from the perspective of the client and the stakeholders ensures that the solution has value for these groups and is thus accepted.
- *Is the solution desirable and attractive to the customer/user groups?* Analogous to the client and other stakeholders, the evaluation of the solution designed from the perspective of the client and users ensures that the solution is desirable and attractive to this group. In literature, this aspect is also often subsumed under the term of a good customer experience.

Evaluation with a view to the business perspective

Based on the business perspective, the design work can be evaluated against the following key questions:

- *Can the solution be operated in an economically viable manner?* Evaluating the economic operability of the solution designed ensures that the facets of the business model designed are viable and questions whether the business model of the solution works as a whole.
- *Can the solution be realized in an economically viable manner?* The evaluation of the economic feasibility of the solution designed questions whether the solution designed can be realized based on the available resources, as well as under consideration of the defined constraints. This issue becomes particularly important when the realization of a solution generates costs that go beyond the technical realization of the system. These can be, for example, costs for training, sales, or marketing.

Evaluation with a view to the technology perspective

Based on the technology perspective, the design work can be evaluated against the following key questions:

- *Is the solution technically feasible?* The evaluation of the technical feasibility of the solution as a whole is only possible to a limited extent, since major technology decisions are only made at the system and element level. Nevertheless, questioning technical feasibility within a narrow scope is helpful in examining the question of feasibility in innovative solution ideas.
- *Does the solution exploit the potential of the existing technology?* The question of potential utilization questions whether the solution designed sensibly exploits the potential of existing technologies. Again, while this question is primarily considered at the system and element level, at the solution level, you can also question whether innovative technologies offer additional opportunities for designing a solution.

8.4 Design work at the solution level during the building process

EO 8.4	Explain the design work at the solution level along the three steps of the building process (L2)

8.4.1 Scoping at the solution level

The solution level is highly relevant in the context of scoping. At the beginning of the scoping, there is an idea that something needs to be done. Together with the client and all relevant stakeholders, this initial solution idea must be understood and made more specific in order to make it tangible in terms of a possible solution space in the Digital Design brief (see section 5.3.1). The main question is whether the client and the relevant stakeholders see sufficient potential in the potential solution idea and its value proposition (described by the vision) to want to take the first step towards realization. A positive answer to this question leads to the start of the concept work.

Answering this question requires a proper understanding of the initial situation. However, this does not mean that a concrete solution idea in the sense of a business model and a detailed value proposition should already be formulated as part of the scoping. The intention of scoping is essentially to understand an idea for value proposition and solution in terms of an initial vision and with it, the reason for action as comprehensively as possible. Based on this idea, you can then decide whether the first step towards the planned solution makes sense. From a Digital Design perspective, therefore, a lot of energy should be invested in understanding the context of the planned solution and, with it, the vision. In addition to working on the Digital Design brief, prototypes for the vision can also be created to gain a better understanding of the vision and to fully evaluate the vision as a whole.

At the solution level, you are at the beginning of the building process during the scoping and therefore have full design freedom to define the vision under the given constraints. Value proposition, customer and user groups, business model, and business process are sketched out in different variants at best during scoping in order to fathom the potential solution space and to obtain the most comprehensive understanding of potential solutions and their advantages or disadvantages.

At the end of the scoping, the Digital Design brief thus contains a clearly defined, comprehensively understood, and accepted initial idea for the vision, in order to record the target image of the planned solution and the reason why the client should take the first step towards realization.

8.4.2 Concept work at the solution level

The solution level is highly relevant in concept work. The goal is to use the Digital Design brief as a basis for developing a sufficient understanding of the planned solution for the client, relevant stakeholders, and the building team so that an informed decision can be made for or against starting the implementation. This understanding is then captured in the solution design concept.

For concept work, the solution level plays an essential role, as this is where the key decisions are made about the design of an attractive, economically viable, and technically feasible solution. However, the question of the technical feasibility of the solution is decided at the system level and must be considered there through appropriate design work at the system level (see section 9.4.2).

From a Digital Design perspective, there is a great deal of design freedom at the solution level during concept work within the framework of the vision, as the vision from the Digital Design brief has defined the essential direction. Nevertheless, the vision formulated in the scoping is not fully established during the concept work. It is very likely that working on the design of the solution and working with prototypes will lead to a deeper understanding of the vision and will require the vision to be revised.

The results of the concept work at the solution level are recorded in a solution design concept. We introduced a template in terms of an example structure in section 5.3.2. The level of detail of the solution design concept is essentially based on the complexity and the handling of potential risks in terms of attractiveness and economic viability. The more critically these perspectives are considered in the context of evaluation activities, the more comprehensively and precisely the solution design concept should be elaborated. This is because the evaluation of the solution designed is an important means to support the decision for or against the start of the realization in a meaningful way.

8.4.3 Development and operation at the solution level

During development and operation, the focus is on developing the solution or the system for realizing the solution in an initial version, putting it into operation, and then continuously developing it until the end of its life.

In contrast to the concept work, the solution level has a medium relevance during development and operation. The freedom of design is greatly reduced, since the solution defined is to be realized and adjustments to the solution become more and more complex with increasing realization. This is explained below in relation to the solution level along the four phases of this step:

- During the *preparation of the development*, the goal is to work out the solution and with it, the system and its own elements to such an extent that the development of the first version can actually begin. In this phase, no system has yet been developed and comprehensive changes to the solution level and the solution design concept are basically conceivable and possible with comparatively little effort. If no development has been started yet, in the worst case, work already completed on the system design concept and the element design concepts will have to be revised.
- As soon as the *development of the first version of* a solution is started, the solution level is continuously made more specific analogous to the preparation of the development. The goal of this phase is to get a first version up and running. From a Digital Design perspective, the decision space for the solution layer changes massively with the start of this phase, as the costs for potential changes increase significantly due to parts of the solution that have already been realized. In the worst case, if the solution is changed, large parts of the system already developed have to be modified and completely redeveloped.
- The *launch of the first version* is a key milestone for the solution level, as the change imagined now becomes a real change. Through feedback from customers and experience with the solution, an intensive phase of optimization and improvement often follows the launch of the first version. Here, holistic design in the sense of Digital Design means that the launch of the solution is not understood as the end of Digital Design, but as a new beginning, because only now does the intentional change as the goal of design (see section 0) begin to take effect. However, significant changes to the solution should not be made during operation and understood as further development but should instead be handled in a parallel building process (see section 11.3).

- At first glance, Digital Design ends at the *end of the solution's life*. When the old solution is replaced by a new one, however, there are often still activities to be done to bring the customers along to the new solution or to convince them of the new solution. Meaningful and effective business processes (possibly with references to a new solution replacing the old one) are an important factor for professional organization and should be designed with a high-quality customer experience (customer journey) in mind.

9 Design work at the system level

The system level considers the technical implementation of the solution holistically, that is, how are the vision, the value proposition, the business model, and the business process appropriately realized by a technical system for the customer and for the client.

In the following, we first introduce fundamental aspects for designing at the system level (section 9.1) and how these aspects interact (section 9.2). We then address the evaluation work at the system level (section 0) and finally, we consider the design work at the system level during the building process together with the work on system design concepts (section 9.4).

9.1 Fundamental aspects of designing at the system level

EO 9.1	Explain the key aspects of conceptual design at the system level (L2)

The system level can be divided into the following aspects for design purposes:

- System level goals
- User types of the system
- Existing elements
- Own elements
- Scenarios
- Quality requirements for the system
- Constraints for the system

In relation to the model of form, function, and quality (see section 3.1) user types, existing elements (e.g., existing systems and objects), and own elements create the form of the system. The scenarios describe the function, and the quality requirements describe the quality. Goals and constraints are outside the FFQ model but are important means in the design.

In the following, we explain these aspects and give examples for documenting them. The examples are not relevant for the foundation level, but serve as references to further literature and to make the contents of the exemplary structure in the system design concept more specific (see section 5.3.2).

9.1.1 Goals (of the system)

For the system-level design, the overall goal of the system is to realize the solution and implement the vision. This general goal must be made more specific and refined accordingly in order to obtain a concrete idea, based on the vision, of which goals the technical system should fulfill to realize the solution.

Templates such as SMART [Wake2003] or goal models [GLSB2022] can be used to document goals for the system.

9.1.2 User types of the system

Defining user types is the first important step in designing the system. User types determine who can use a system and interact with the system's elements.

Even the simplest systems usually have more than one user type. The starting point for user types of a system are the customer groups and user groups defined in the solution design concept (see section 8.1.2) and for whom the system should realize the value proposition. Other candidates for user types can be found in the solution's business processes and in the customer journey.

With regard to user types, it is equally important to note that not every person who plays a role in the business process of the solution is necessarily a user type (see section 04.1). In addition to user types, the question of how users access the system is also relevant, because access to the system is also used to design the process of how a person becomes a user and thus part of the system (for example, by registering as a user).

Access to the system almost always differs according to the user types and the question of which user type can interact with which element in which way. In business applications, for example, it is often the case that different types of users are allowed to perform different functions or view different data. This question is considered at the level of the individual elements and is therefore taken up again there.

Short textual descriptions are typically used to document user types.

9.1.3 Existing elements

Existing elements are hardware or software systems and other objects whose existence is assumed as an element to realize the solution.

Examples of existing elements may include:

- *Hardware systems:* end devices for using the solution (e.g., smartphones or PCs), server hardware for operating a solution
- *Software systems*: operating systems, browsers (for web applications), existing services (e.g., payment services)
- *Other objects:* objects with QR codes, machines, buildings, rooms, furniture

In addition to the hardware and software systems as part of the design, other objects also play an important role in the holistic design. By looking at existing objects, we define and design the real context of a system and thus of the digital solution.

When designing the system, however, it is not only a matter of technical questions of realization (e.g., should the system run in the browser or not?), but also of questions of economic efficiency, openness, and sustainability.

Questions of economic efficiency relate, among other things, to operating and manufacturing costs. Operating systems generate costs for licenses and must be maintained and serviced accordingly (updates). The argument for cloud operation (see chapter 6) merely shifts the cost issue to another position, because cloud systems are typically billed according to use. The form of billing is part of the holistic perspective and the economics of the business model (see chapter 11).

The question of openness in terms of the accessibility of a solution for customers and users is largely defined by the required operating systems, especially for the end devices. Openness is only a purely technical issue at first glance, even though competence in digital material plays a major role here.

A good example is the question of technical platforms for smartphones (Apple vs. Android). By choosing one of these platforms, a whole group of potential customers is immediately excluded. However, it is not only the platform that plays a role—the version of the operating system supported is also decisive for the openness of a solution. The newer an operating system has to be, the smaller the number of potential customers becomes. At the same time, however, the platforms and operating systems influence the costs of realization and thus the business model. Realizing an app for different platforms and operating system versions inevitably drives up development costs, as experts must be available for the various platforms and the different platforms/versions must be maintained and tested.

Closely related to openness is the question of sustainability. Supporting older operating systems is an important contribution to sustainability in terms of resource conservation. Potential customers would then not have to buy a new device with a new operating system if they were interested in the solution and could use the existing device for longer. Even if at first glance this seems to be only a commercial issue, the question of social sustainability must not be ignored here either, because the purchase of new devices for participation in digital solutions is a critical factor with regard to the inclusion or exclusion of people based on their financial possibilities.

Textual descriptions and models for the structure (e.g., component models) are suitable for documenting existing elements.

9.1.4 Own elements

Own elements are the elements of the system that must actually be implemented to realize the solution. The plural is important here, because even the simplest systems usually consist of more than one element. When designing a digital solution, it is important to look specifically at your own elements to get a clear picture of the scope of the system and also the effort required to implement it.

A pragmatic approach to designing your own elements is to orient them to the real world, that is, to end devices used, user types, and data used. Specifically, in a first draft, you can define a separate element for each user type and associated end device and draw the boundaries so that the required data for the functions are present in the element. In addition, there may be elements that work in the background, that is, server elements for data processing, storage, etc.

The design of own elements should be oriented very closely to the technical structure of the system. This technically oriented design approach has advantages for the holistic design of a solution. On the one hand, the technical reference allows seamless collaboration with realization. For example, software architects think in terms of technically realizable units and their own elements are then virtually the technical unit that must actually be realized and operated.

Furthermore, the technically oriented design supports the work organization in the further development, because technical elements are often a good starting point for the division of labor and the team structure. A technically oriented design leads to clearly defined interfaces that enable independent development and testing. Finally, a technically oriented design has the important advantage that the concepts of the system are as close as possible to the technical reality. This is of great importance for understanding the system.

Conversely, a technically oriented design also means that, in terms of Digital Design, you have to work on the design in a team with experts in technical implementation right from the start. Specifically, for example, experts in software architecture are needed for the software portion of a system, and experts in industrial or product design are needed for the device portions of a proprietary device.

Textual descriptions and models for the structure (e.g., component models) are useful for documenting a system's own elements.

9.1.5 Scenarios (function of the system)

The system-level aspects introduced so far concern the form of the system. The work on the design of the system also includes the function of the system.

The form of the system can be sensibly and completely designed, described, and thus represented using the previously mentioned aspects. In the case of function, this becomes problematic to impossible, since the function of a system is composed of the totality of the functions of its elements.

Scenarios are a good compromise for system-level design. Scenarios refer to exemplary descriptions of the function in terms of processes or short stories. Scenarios also carry the risk of becoming very extensive and detailed. However, due to the exemplary nature of the scenarios, there is no need to be comprehensive, so there is much less risk of the description becoming overly detailed.

Textual descriptions in the sense of continuous text or in the sense of numbered steps can be used to document scenarios.

9.1.6 Quality requirements (for the system)

Quality requirements describe qualitative aspects of the form or function of the system. The aspect of quality requirements is important for the design of the system because quality requirements have a great impact on technical decisions (e.g., response time, performance, availability).

The concrete details and characteristics of the quality requirements are developed in collaboration with all relevant stakeholders. This involves not only customers and users, but also technology and implementation experts.

Working on quality requirements at the system level is teamwork. However, like the design of the system, the essential responsibility for formulating good quality requirements lies with Digital Design, since an essential component of the quality of a solution as a whole is defined by the desired qualities of the system.

There are a whole series of practical tips and procedures for documenting and working with quality requirements, especially from requirements engineering [CPRE2020]. A basic technique for documentation is the use of textual descriptions. In addition, there are a number of norms and standards that specify various aspects of quality requirements (cf. e.g., [ISO2011] or [ISO2019]).

9.1.7 Constraints (for the system)

Constraints refer to (almost) unchangeable specifications that the system must fulfill. Constraints are often technical, organizational, or legal in nature.

Analogous to quality requirements, system-level constraints are a major factor in design decisions and in the success of a system. If, for example, significant legal constraints are not considered in the system design, this can lead to a system not being allowed to go into operation for legal reasons.

Analogous to quality requirements, the formulation of constraints is a significant responsibility of Digital Design. Requirements engineering [CPRE2020] provides important information on the documentation of constraints. Textual descriptions, for example, are a good choice.

9.2 Using the aspects in interaction when designing the system

EO 9.2	Apply conceptual design to create design concepts at the system level (L3)

The description of the system-level aspects shows that the design work at the system level is much more technical than the work at the solution level. For a complete design at the system level, all aspects are equally relevant.

What is important for good design thought at the system level is an awareness of the interrelationships of the aspects (see Figure 1.9):

- Goals are used to formulate what the system should achieve. Based on the goals, the following can be defined: form, function as a means and path, and initial quality requirements and constraints.
- The formulation of scenarios defines the way in which the goals of the system will be achieved. Quality requirements for the scenarios can capture important qualitative properties of the system.
- The form defines the possible actors in the scenarios and the relationships between the building blocks of the system with their planned interaction relationships.
- Working on form and function creates a better understanding of the goals.

- Quality requirements define qualities of form or function that are necessary for the fulfillment of the goals or the constraints.
- Constraints define unchanging specifications that the form, function, and quality of the system must meet.

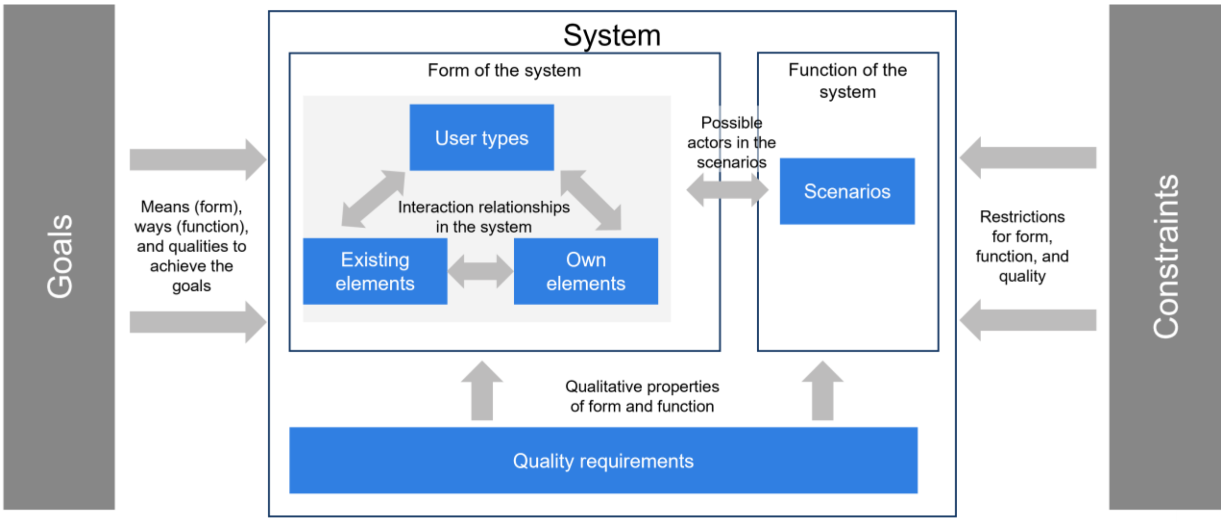

Figure 1.9 - Aspects for designing the system in interaction

The structure of the aspects and especially the formulation of goals suggests a top-down process, that is, formulate goals, capture the constraints, and then design an appropriate form, function, and qualities. This top-down process is rarely useful in practice.

To start work on the design of a system, the goals of the system are the first topic. Without a goal statement and an initial answer to the question *What does the system want to achieve*?, no meaningful design can start. The basis for target setting is the value proposition at the solution level.

After the goals, we move on to form, function, and quality. For beginners, the path via the function is usually easier, that is, it is necessary to consider how a system and with what quality requirements the goals can be achieved. Scenarios can then be used to develop ideas for the form of the system (Which actors play a role in the scenarios and how do they interact?). Furthermore, corresponding qualities of the form can also be defined. The scenarios then also usually yield ideas or potential sources for constraints.

With an initial idea of form, function, and quality, the work starts all over again. Typically, when working on form and function, new ideas lead to goals, a better understanding of qualities, and so an iterative thought process that moves back and forth between goals, form, function, and quality, as well as constraints, then starts. Just as useful in this process can be jumping back to the solution level and diving into the details of the elements. This interaction is highlighted in chapter 11.

9.3 Evaluation on the design work at the system level

EO 9.3 Explain the essential questions for evaluation work at the system level (L2)

Analogous to the description of the evaluation at the solution level, we describe essential questions of the evaluation work at the system level below. We do not describe specific techniques as they go beyond the foundation level.

Evaluation with a view to the people perspective

Based on the people perspective, the design work can be evaluated using the following key questions:

- *Are the user types meaningfully defined and understood?* The evaluation of the user types designed questions whether the customer/user groups defined are meaningfully represented and whether their interaction possibilities are adequately defined in the system.
- *Is the system desirable and attractive to the client and relevant stakeholders?* Analogous to the solution level, it is important to question whether the system designed is desirable and attractive from the perspective of the client and relevant stakeholders in order to ensure acceptance of the system. At the system level, the system is evaluated as a whole, that is, the evaluation refers to the form, functions, and quality of the system, as well as the type of technical implementation. Details of the elements (e.g., user interfaces) are evaluated at the element level.
- *Is the system desirable and attractive to users?* Analogous to the client and relevant stakeholders, evaluation of the system from the user's perspective is also important to ensure acceptance of the system. Again, the form, function, and quality of the system and the technical implementation designed should be evaluated from the user's point of view. In literature, this is also referred to as the evaluation of the user experience (cf. [CPUX2022]).

Evaluation with a view to the business perspective

Based on the business perspective, the design work can be evaluated against the following key questions:

- *Can the system be operated in an economically viable manner?* At the system level, the evaluation of viability questions whether the system can be operated economically from a technical perspective with the given resources and constraints, and whether the solution's business model can cover the costs of operation. The costs of operation include not only hardware costs, but also costs for software and for personnel to operate the solution (e.g., administration, maintenance, and service for hardware and software).

- *Can the system be realized in an economically viable manner?* Analogous to the solution level, the economic feasibility of the system designed should also be evaluated to ensure the economic viability of the solution as a whole. At this point, it is important to separate the system and element levels. At the system level, the focus is on the economic feasibility of the system as a whole under the given resources and constraints (e.g., realization of a data center or technical infrastructure).

Evaluation with a view to the technology perspective

Based on the technology perspective, the design work can be evaluated against the following key questions:

- *Is the system technically feasible?* At the system level, the evaluation of technical feasibility questions whether the designed form, function, and quality of the system are technically feasible. This question is particularly exciting when a digital solution moves into an innovative field in which the building team has little or no experience. This may be the case when a completely new technology is used or when the technology is new to the building team.
- *Does the system make good use of the potential of the available technology?* At the system level, you should evaluate whether the potential of existing technologies is adequately exploited by the system designed. On the one hand, this evaluation can be used to question whether the technologies selected and their capabilities are being used sensibly or whether, if necessary, alternative technologies offer more potential for the realization of the system.

9.4 Design work at the system level during the building process

EO 9.4	Explain the design work at the system level along the three steps of the building process (L2)

9.4.1 Scoping at the system level

The goal of scoping is to work out the initial idea for the client's vision and to decide whether to take a first step towards the solution in the sense of a conceptual design—or not. Therefore, the system level tends to have low relevance during scoping.

As a rule of thumb, you should deal with the system level when:

- Thinking about a system helps to better understand the vision
- The system level helps to explore the potential solution space and capabilities of digital technologies as part of the Digital Design brief to make them usable for the vision
- The realization of the vision depends on special system circumstances and the feasibility should be clarified before the start of the concept work

Design freedom at the system level is high during scoping but must be done in concert with the solution level. The system level defines the overall structure of the technical system with its users, existing systems/objects, and own elements. When it comes to vision, or planned change, many people find it easy to describe change as a kind of system. In the context of scoping, such ideas can be recorded by means of appropriate sketches, but not primarily in order to implement them one-to-one as a design, rather as an idea for an implementation that should be analyzed, discussed, and distilled with other ideas until a clear vision for the solution has emerged.

9.4.2 Concept work at the system level

In concept work, the system level is highly relevant. On the one hand, it is a matter of designing the system aspects of the solution and recording them in the system design concept and, on the other hand, designing the technical realization of the solution and documenting it in teamwork in the form of a system realization concept. At the end of the concept work, the question arises whether the solution developed is technically feasible and whether you decide for or against the start of the realization of the solution. To answer this question soundly, a very clear understanding of the attractiveness, economic viability, and feasibility of the system designed is required.

The freedom of design at the system level is high during the concept work, whereby the interaction with the solution level is important, analogous to the scoping of the assignment. We consider details of the interaction in section 11.1. Design work is basically a mixture of analytical activity, conceptual design activity on the system design concept, and evaluation of the system design concept. Analytical means that the technical context of the planned solution must be adequately understood, since any system must embed itself in an existing technical environment. In particular, this environment defines the constraints that a system must fulfill.

The conceptual design activity means an intensive interplay between goals formulated on the basis of the solution design concept for the system and possible designs for achieving the goals in terms of the form, function, and quality of the system (see section 11.1). For the fundamental procedure for designing at the system level in concept work, the rapid elaboration of different variants of designs in the form of system design concepts in combination with evaluations based on prototypes is very useful for adopting a wide variety of perspectives on implementation and corresponding alternatives. We introduced a template in terms of an example structure in section 5.3.2.

With regard to economic viability and technical feasibility, intensive collaboration between design and construction is important (see section 4.3.2) to design a system that is economically and technically feasible and also makes good use of the potential of the technology available. A good practice here is that system design and system realization concepts are developed in close coordination and, if possible, in close collaboration between the activity areas of design and construction. With digital solutions in mind, experience shows that little creative energy is invested at this point in the building process in collaboration between design and construction to realize the full potential of innovative technologies.

When designing at the system level, it is important to get the scope for decision-making and level of detail right. The relevance of the system level for the success of a solution often leads to the reflex to describe the system level as precisely and completely as possible. However, this procedure leads to a very detailed system design concept and to decisions that are not necessary for the actual goal of the concept work (decision for or against realization), but actually belong to the work on the own elements. Analogous to the solution level, the level of detail should be based on the complexity of the system and the handling of potential risks. The more critically the system is to be considered in the evaluation, the more comprehensive and detailed the system design concept should be elaborated and evaluated.

9.4.3 Development and operation at the system level

During development and operation, the focus is on the realization and further development of the solution by the technical system. Accordingly, the system level is highly relevant in this step.

Analogous to the solution level, the design freedom at the system level decreases with increasing development progress, as changes become more and more complex. We consider the four phases of this step below in relation to the system level:

- During the *preparation of the development*, the system, and especially its own elements, must be worked out to such an extent that the development can start. Analogous to the solution level, changes to the system design concept can be made at this stage with comparatively little effort. Major causes for change at the system level usually do not emerge from the system level in this phase but arise from new insights that emerge at the solution or element level.
- With the *start of development*, work continues to flesh out the system level to support the work at element level. Analogous to the solution level, the scope for decision-making changes with the start of development with regard to the costs for far-reaching changes to the system. Furthermore, important decisions for the later operation of the solution are made during the development of the first version. These issues do not belong exclusively to Digital Design, but are generally dealt with in the activity areas of construction and realization (see section 4.2.1). Nevertheless, Digital Design plays a role in these decisions, as they are also part of the holistic design of a solution and have an impact on the business model (in terms of cost) as well as on the customer and user experience (e.g., through the performance of the operating infrastructure or through the quality of the end devices).
- With the *launch of the first version*, the nature of the building process changes analogously to the solution level. Now the phase of optimization and improvement in conjunction with the further development of the system begins. With regard to the system level, further development consists not only of the further development of the elements, but also of the further development of the system as a whole.

Analogous to the solution level, comprehensive changes to the system level and the system design concept should not be handled as part of the continuous development process, but rather through a second parallel build process (see section 11.3).

- When *a solution reaches the end of its life*, the system also goes out of service. Depending on the type of system, final activities may be pending here. Existing hardware must be switched off and any terminal equipment must be disposed of. Likewise, contracts with service providers for connected systems must be terminated. These activities are not necessarily part of Digital Design, but are important, for example, to obtain a holistic perspective. For example, for sustainability, old equipment can be reused sensibly or handed over for recycling.

10 Design work at the element level

At the element level, the system's own elements are considered at a level of detail that should enable the technical implementation of the elements.

In the following, we first present fundamental aspects for designing at the element level (section 10.1) and how these aspects interact (section 10.2). We then address the evaluation work at the element level (section 10.3) and finally, we consider the design work at the element level during the building process together with the work on element design concepts (section 10.4).

10.1 Fundamental aspects of designing at the element level

EO 10.1	Explain the key aspects of conceptual design at the element level (L2)

The element level can be divided into the following aspects for the design:

- Element level goals
- User interfaces
- Technical interfaces
- Data and data structures
- Physical structure
- Use cases
- Technical functions
- Quality requirements for the element
- Element constraints

The goal is to design the solution's own elements with respect to the above aspects and document them in element design concepts. The scope and level of detail of element design concepts depend on several factors, such as whether the element is critical to the success of the solution (see section 10.4.4).

We have already encountered some of these aspects in a similar form at the system level. Related to the model of form, function, and quality (see section 3.1), user interfaces, technical interfaces, data, and the physical structure make up the form of the element. The function is described by use cases and technical functions, and the quality requirements describe the quality.

In the following, we explain these aspects and give examples for documenting them. The examples are not relevant for the foundation level, but serve as references to further literature and to make the contents of the exemplary structure in the element design concept more specific (see section 5.3.2).

10.1.1 Goals (of an element)

Goals of an element are, analogously to goals of the system, an expression of what is to be achieved with the element as part of the system. What is important for the change between element and system level is the clear perspective on the goals of an element, that is, on the question of what is/should be achieved with the element under consideration as part of the system?

Templates such as SMART [Wake2003] or even goal models [GLSB2022] [GLSB2022]can be used to document goals for an element.

10.1.2 User interface (perceivable form)

User interfaces are part of the perceivable form of an element. User interfaces define how a user interacts with the element. We often think of user interfaces as screens, mice, and keyboards. Here, it is important to note that digital technologies now also offer completely different possibilities for interaction; interaction via speech in particular is an exciting area that is often forgotten.

An important factor when working with user interfaces at the element level is the goal of design feasibility. The element level serves as the basis for implementation, so at the element level, user interfaces must be defined such that all details are available for implementation. This often means that user interfaces are described as detailed designs, including shape, colors, data presented, and interaction possibilities. This also includes, for example, the access options and the display of data for different user types, provided that different user types are intended for the element at the system level.

Sketches and specifications for the structure of the user interface are suitable for documenting user interfaces [Laue2005]. For a systematic design of user interfaces, design systems (cf. e.g., [Fros2020]) are a helpful support.

10.1.3 Technical interface (underlying form)

The counterpart to user interfaces are technical interfaces as part of the underlying form of an element. Technical interfaces define how an element can interact with other elements of the system. For a holistic design of an element, the technical interfaces are therefore just as important as the user interfaces and must also be designed at a comparable level of detail. If this detail were missing, important information for later realization would be missing.

The description of a technical interface includes, in addition to the task of the interface, in particular the direction of the interface, the type of call, and the data that passes through the interface. The direction of the interface means whether the interface addresses another element from the element under consideration or whether the element under consideration is addressed by another element. The type of call means whether the interface returns a response immediately (synchronous call) or only after a defined time (asynchronous call). Furthermore, the data required for communication must be described appropriately (e.g., data type or data format).

This perspective may be surprisingly detailed, but it is imperative for holistic design and downstream realization. Similar to the user interface, the communication must be presented precisely in order to enable implementation and evaluation based on this.

Textual specifications [GLSB2022], but also models (e.g., UML sequence diagrams [OMG2017]) are suitable for documenting technical interfaces.

10.1.4 Stored data (underlying form)

The aspect of the data defines what information is stored in an element. Just like technical interfaces, data is part of the underlying form of an element, as the stored data is not immediately apparent, whereas the perceptual data is designed as part of the user interfaces.

Data can be defined well by entities. An entity is understood to be a *something* for which data is stored. This something is further characterized by a name and attributes. An attribute consists of an identifier and a type. The identifier is the name of the attribute and the data type defines what kind of data should be stored in the attribute. There are various types of data. Knowledge about digital material, especially about programming languages, gives you a wide range of data types.

This level of detail for designing an element is also required to achieve the necessary level of detail for the feasibility of an element design concept.

Textual specifications, but also models (e.g., class diagrams [OMG2017]) are suitable for documenting data.

10.1.5 Physical structure (perceivable and underlying form)

The physical structure of an element becomes relevant when an element of the system is not pure software, but a real device. Examples include: smart speakers, fitness watches, or even control devices in a smart home.

Relevant questions in the physical structure concern the form (e.g., the housing with material and color) but also technical components (e.g., keys, screens, processors, memory, communication devices, power supplies, etc.).

As soon as Digital Design involves the physical construction of an element, physical product experts come into play alongside software experts. These must then be involved in the design of the element in the same way as experts for the software part.

For the foundation level in Digital Design, the most important thing with regard to the physical structure is the awareness that in hybrid products, the physical structure is just as important as the digital part.

Building plans, technical specifications, and all other product design techniques are used to document the physical part.

10.1.6 Use cases (perceivable function)

Use cases are a representation that depicts the function of an element in terms of how it interacts with the user. Use cases are an established technique from requirements engineering and are used to describe the interaction of a system with its environment as comprehensively as possible.

Understanding use cases in terms of the interaction between user(s) and element(s) is a very central design aspect. A use case is used to design the concrete behavior of an element in interaction with the user in order to illustrate how the added value of the solution is actually realized.

Use cases describe the interaction between user and element as completely as possible based on main scenarios and possible alternative scenarios. Furthermore, use cases always refer to user types. This reference allows you to specifically define which user types can perform which use cases. This way, if there are several user types per element, you can clearly distinguish which user type is allowed to use which functions.

Here, what is more important than the form of description is the goal of completeness at a meaningful level of detail. Use cases are the basis for the realization of the behavior of an element and thus the basis for the evaluation of the realized element. The goal of completeness can be captured well in interaction with other aspects and is considered in section 10.2.

Textual specifications (use case templates [GLSB2022]) and models (e.g., UML activity diagrams [OMG2017]) are particularly suitable for documenting individual use cases. Use case diagrams [OMG2017] can be used for the interaction of multiple use cases and user groups.

10.1.7 Technical function (underlying Function)

The counterpart to the use case is the technical function for designing the underlying function of an element. Technical functions describe processes that take place within an element and do not involve user interaction. Good examples can be calculations on data and thus modification of the data, but also obtaining data via technical interfaces.

Even if this aspect of an element also seems abstract when first read, it is nevertheless just as important in digital systems as the interaction with the user. Even in the simplest systems, technical functions play an important role in the design of an element that is ready for realization.

Similar to use cases, technical functions must be designed as completely as possible at a meaningful level of detail to support the realization of the function and the evaluation of the function realized. We consider the goal of completeness in interaction with the other aspects in section 10.2. Textual specifications or models (for example, activity diagrams or state models, cf. [OMG2017]) can be used to document technical functions.

10.1.8 Quality requirements for an element

Quality requirements describe qualitative aspects of the form or function of an element in analogy to the system level. The aspect of quality requirements is important for the design of an element because quality requirements have a great influence on technical decisions (e.g., response time, performance, availability).

Analogous to the system level, working on quality requirements is a team effort, but the responsibility for formulating good quality requirements lies with Digital Design.

The same techniques that are used at the system level can be used for documentation. In addition, there are a number of norms and standards that specify various aspects of quality (cf. e.g., [ISO2011] or [ISO2019]).

10.1.9 Constraints for an element

Analogous to the system level, constraints of the element mean (almost) unchanging specifications that the element must fulfill. Constraints are often of a technical, organizational, or legal nature, and the formulation of meaningful constraints is the responsibility of Digital Design.

The same techniques used at the system level can be used for documentation (see section 9.1.7).

10.2 Using the aspects in interaction when designing an element

EO 10.2 Apply conceptual design to create design concepts at the element level (L3)

The presentation of the various aspects for designing at the element level shows how complex the design work on an element is. For newcomers to Digital Design, keeping track of this scope is usually a major challenge, but one that can be mastered with practice and application of the structures presented.

In addition, the scope and diversity of the aspects show that a variety of disciplines must be involved in the design and thus in the design work at the element level, for example, interaction design for the design of good user interfaces or data science expertise for the design of data-intensive technical functions. Teamwork is of great importance for designing at this point, but is not considered further here, as the topic goes beyond the foundation level. However, we revisit the fundamental importance of teamwork as an attitude in section 13.4.

An important factor for understanding the aspects for designing the element level is the knowledge of the structural dependencies of the aspects around form and function. We have already mentioned the dependencies in parts in the description of the aspects. Nevertheless, knowledge of these dependencies is very important in order to develop a design that is as consistent and implementable as possible at an appropriate level of detail. Therefore, we explain the dependencies again in the following.

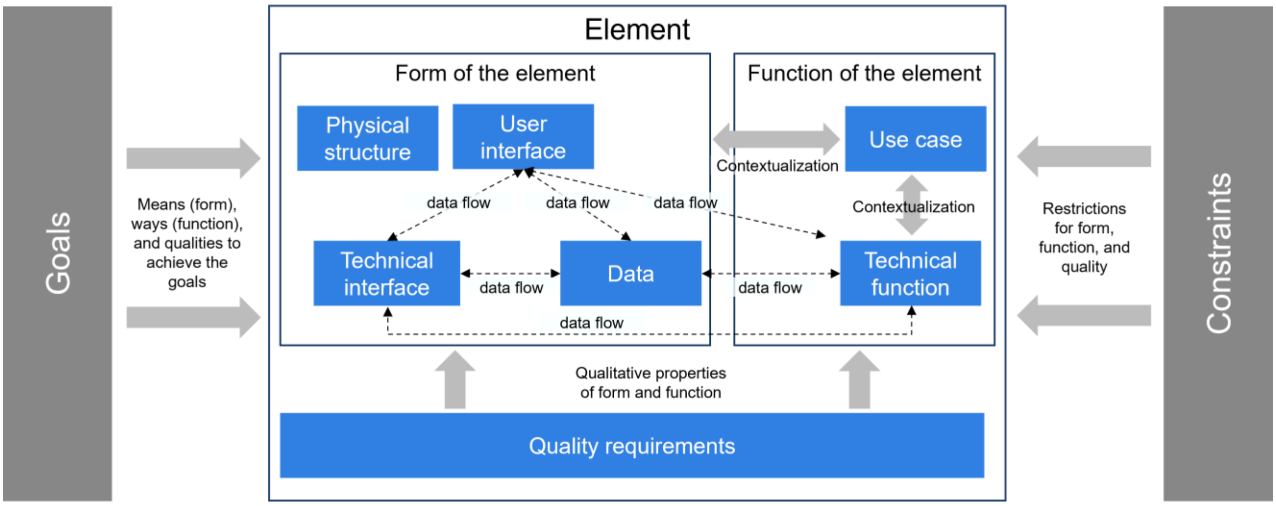

Figure 1.10 - Aspects for designing an element in interaction

Figure 1.10 also summarizes the dependencies visually once again:

- User interfaces as a perceivable form establish the relationship with the user. For this purpose, a user interface displays various information. This information must come from one source. Sources for this can be: stored data (underlying form), delivery of information via technical interfaces (underlying form), or calculation of information by technical functions (underlying function).
- Technical interfaces as an underlying form provide information into or out of the element. The source or target of this information can be the stored data, user interfaces, technical functions, or other technical interfaces.
- Stored data as an underlying form represents the information contained in an element. This information comes from sources and is used in the element. Sources or recipients of the information can be user interfaces, technical interfaces, and technical functions.
- Physical structure as part of the perceivable or underlying form can be technical building blocks of the element and thus carriers for user interfaces or technical interfaces.
- Technical functions as part of the underlying function serve the calculation in the background. They use data from various sources or deliver data to recipients. Sources or recipients can be user interfaces, technical functions, technical interfaces, and stored data.
- Use cases as part of the perceivable function provide the framework for the form and technical functions. By describing use cases, the form and the technical functions are given a context, that is, put into context. Through this contextualization, use cases define how an element achieves the defined goals.

Design work at the element level, analogous to the system level, is an iterative process that works on the form, function, and quality of the element based on the element's goals and constraints.

For beginners in Digital Design, the way to go is through goals and use cases. First, goals are defined for the element and, based on the goals, use cases and quality requirements are developed to define how these goals will be achieved. Based on the use cases, constraints can be identified, and user interfaces, data, technical interfaces, technical functions and, if necessary, the physical structure can be designed. With an initial understanding of an element's form, function, and quality, an iterative design process then begins, moving back and forth between layered aspects. Just as useful in this process can be jumping back to the solution and system level. This interaction is highlighted in chapter 11.

10.3 Evaluation at the element level

EO 10.3 Explain the essential questions for evaluation work at the element level (L2)

Analogous to the description of the evaluation at the solution and system level, we describe essential questions of the evaluation work at the element level below. We do not describe specific techniques as they go beyond the foundation level.

Evaluation with a view to the people perspective

Based on the people perspective, the design work can be evaluated using the following key questions:

- *Is an element desirable and attractive to the client and relevant stakeholders?* Analogous to the solution and system level, the design of an element should be evaluated from the perspective of the client and relevant stakeholders to determine whether the defined form, function, and quality are attractive and desirable.
- *Is an element desirable and attractive to users?* Analogous to the solution and system level, the evaluation of the design of an element questions whether the defined form, function, and quality are desirable and attractive to the user. In this context, analogous to the system level, in literature, the evaluation is also referred to as the evaluation of the user experience (cf. [CPUX2022]).

Evaluation with a view to the business perspective

Based on the business perspective, the design work can be evaluated against the following key questions:

- *Can an element be operated in an economically viable manner?* Analogous to the solution and system level, the evaluation of the viability of an element's design questions whether an element can be operated economically from a technical point of view with the given resources and constraints. This becomes relevant whenever the operation of an element incurs costs beyond the operation of the element as part of the system level (e.g., regular licensing costs for specific technologies of the element).
- *Can an element be realized in an economically viable manner?* Analogous to the solution and system level, the economic feasibility of the element designed should also be evaluated to ensure the economic viability of the solution as a whole. In particular, this involves the question of whether an element is feasible under the given

resources and constraints. Here, both the costs for the initial realization and the costs for potential maintenance and further development must be considered, since the costs for maintenance and further development of a solution over the entire life cycle of a solution often exceed the costs for the initial development.

Evaluation with a view to the technology perspective

Based on the technology perspective, the design work can be evaluated against the following key questions:

- *Is an element as a whole technically feasible?* Analogous to the solution and system level, the evaluation of technical feasibility at the element level questions whether the designed form, function, and quality of an element are technically feasible. This issue is of great importance when using innovative technologies and unprecedented technologies to ensure the feasibility of the solution as a whole.
- *Does an element make good use of the technology's potential?* At the element level, as at the solution and system levels, you should evaluate whether an element designed adequately exploits the potential of existing technologies. This can relate to both the technology chosen and whether alternative technologies offer more potential.

10.4 Design work at the element level during the building process

EO 10.4	Explain the design work at the element level along the three steps of the building process (L2)

10.4.1 Scoping at the element level

Analogous to the system level, the element level has a low relevance in scoping. The design freedom for the elements of the system is high but must take into account the solution and system level. For example, design freedom at the element level may be limited by decisions already made at the solution and system levels, especially by the vision. You can look at the element level to better understand the vision (for example, if a particular element is central to the vision) or to explore the potential solution space in more detail. If the feasibility of the vision depends on an element aspect, then the element level can also already be considered in the context of the scoping.

The comparatively high level of detail of the element level in terms of user interfaces, use cases, etc. is important for the understanding of the element level and the consideration of the element level in the context of scoping. If this level of detail is required as part of the scoping, then typically, an effort is made that is comparable to the concept work. In such a case, the building process should consciously ask whether or not this level of detail is necessary for the next step. Often, in such a situation, it makes more sense to consciously start the step of concept work in order to clarify the necessary details with adequate resources.

10.4.2 Concept work at the element level

The goal of the concept work is a conscious decision for or against the start of the realization. For an informed decision, the details of the element level can make a useful contribution, therefore the element level has a medium relevance for the concept work. For example, element design concepts (see section 5.3.2 for an example template) and prototypes of elements can give a first impression of what the solution realized may look like.

There is a lot of freedom for the design of the elements at the element level, although the design of the solution and the system must also be considered here, see section 11.1. An essential factor for the elaboration of the element level in the concept work is the appropriate level of detail. A complete elaboration of all own elements in the sense of an element design concept is rarely purposeful, since this is associated with a high effort compared to the benefit (see section 10.4.4). Some frameworks prescribe this level of detail, others deliberately refrain from it (for details, see section 12.1).

Basically, as part of the concept work, the level of detail of the element level should be elaborated with a view to contributing to an informed decision for or against the development. This means in particular, that parts of an element with low risks for solution that are already understood (e.g., known processes, such as registration of users or login) should be left out in the concept work. Instead, focus only on those aspects that make a significant contribution to the success of the solution. These aspects can then be captured and evaluated in corresponding element design concepts.

10.4.3 Development and operation at the element level

During development and operation, the focus is on realizing the solution through the technical system. The element level is highly relevant here, as all details for the implementation of an element are defined here. Furthermore, there is a medium freedom of design at the element level, as there are many possibilities for the concrete design of an own element. The design freedom is limited by the previously defined system and the elements already realized as well as the technologies selected (defined by element realization concepts).

An essential factor for the work at the element level is the intensive cooperation with the other activity areas in the building team (see section 4.2) in order to work out the details required for the realization. The focus of the work is the question of when which details of an element are worked out. In the following, we consider the four phases of this step in relation to the element level:

- In *preparation for development*, it is necessary to work out the details of an element, which should be realized as a first step. Depending on the framework, there are different approaches here (see section 12.1). For an efficient and effective development process, from a Digital Design perspective, care should be taken to ensure that a sufficient level of element detail is elaborated and evaluated in appropriate element design concepts (see section 5.3.2) so that development can be sensibly started.

- As *development begins*, the element level work for Digital Design changes as new tasks are added. This includes, in particular, the clarification of questions, the decision on details for the realization of an element, and the participation in the evaluation of components of the element already realized. These additional tasks mean that the existing work capacity in Digital Design must be divided sensibly between the further elaboration of the element design concepts and the new tasks.
- With the *commissioning of the first version*, the system and, with it, the elements come to life. For the work at the element level and on the element design concepts, this means intensive content work on the details. Errors need to be clarified and fixed, optimizations and improvements need to be incorporated into the element design concepts, and at the same time, the next steps for further development need to be designed and evaluated.
- With the *end of life of a solution*, the system goes out of service and with it, the elements. From a Digital Design perspective, final changes may still need to be made at the element level. For example, user interfaces can be supplemented with notices that the solution is going out of service. Likewise, the need for new features with which users can backup, export, or delete their previous data may arise. If a new solution replaces the old solution, then the development of interfaces and functions for transferring data to the new system may also be required.

10.4.4 Level of detail and scope of element design concepts in the building process

Defining the level of detail and scope of element design concepts is a major challenge for Digital Design and should always be defined in conjunction with the entire building team. Concrete procedures depend on a wide variety of factors and go beyond the foundation level. The following are four typical strategies that can be used to guide the level of detail and scope of documentation:

- *Orientation towards risk:* Aspects of an element that are critical to success are defined with the highest possible level of detail in order to enable a meaningful evaluation on the one hand and, on the other, to provide the most concrete specifications possible for realization. Aspects with lower risk can be worked out in less detail in order to define them in direct cooperation during realization.
- *Orientation to the prior knowledge of the building team:* If the building team has a great deal of prior knowledge, for example, because the team or individual members have already built comparable solutions, less detailed concepts are sufficient and it is more efficient to clarify queries about details during realization. Building teams with little previous knowledge (e.g., teams from external companies), on the other hand, need much more detailed concepts.
- *Orientation to the degrees of freedom for the building team:* The more the building team should bring its own creativity and ideas into the realization, the fewer details an element design concept needs. The more concrete the ideas for realizing the elements are, the more details a concept must provide.

- *Orientation to costs of subsequent changes*: The more costly subsequent changes to a solution already realized are, the more precise the concepts for realization should be. If subsequent changes can be made cheaply and quickly, then less detail is needed.

Regardless of the procedure chosen, it is important that the element design concepts are elaborated to an appropriate level of detail at the latest after realization has taken place. This is the only way to ensure consistency between concept and solution and also to provide an appropriate reference for evaluating the solution.

11 Holistic design work in the building process

In chapters 8, 9, and 10, we considered the three levels of a solution in the context of the building process. In this chapter, we take the higher-level perspective to provide a holistic view of design work.

The model of the three perspectives of people, business, and technology (see section 5.15.1) and the model of the three levels of solution, system, and element (see section 5.25.2) are used as the basis for holistic design. Based on the three perspectives, the goal of holistic design work can be defined such that a solution achieves an appropriate balance of what is technically feasible, what is desirable, and what is economically viable.

The goal of this chapter is to create a comprehensive awareness of the various facets of holistic design work. This awareness is important for the foundation level in Digital Design and forms the basis for competencies in holistic design work:

- Section 11.1 shows how holistic design work is achieved across the three levels of solution, systems, and element (see section 5.2).
- Section 11.2 discusses the holistic design of a solution along the three perspectives of people, business, and technology (see section 5.15.1) at the three levels.
- Section 11.3 uses the example of the three steps of the building process (see section 0) to show how holistic design work benefits from an iterative process.

11.1 The importance of collaborative design work across the three levels

EO 11.1	Explain the importance of co-design across the solution, system, and element level for holistic design work (L2)

In the following, we consider design work along the three levels to show how important the interaction at these levels is for a holistic design. We use the working model for the design of digital solutions from section 3.3 to structure the consideration.

Looking at the foundation level in Digital Design, it is important to understand that this section is not about having a complete understanding of the interdependencies between levels, but rather about having an understanding of the importance of collaborative design work (co-design). Therefore, in this section, we discuss only the main relationships with examples.

The following figure summarizes the main relationships between the levels in an overview.

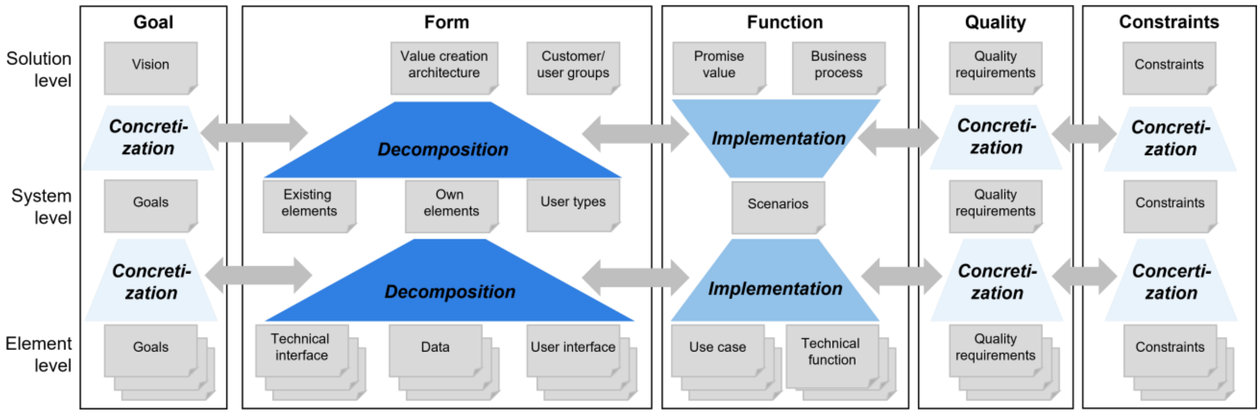

Figure 11.1 - Essential relationships of the aspects across the three levels of a digital solution

Goal setting at the three levels

Goal setting plays an important role at all three levels. At the solution level, the goal is formulated as a vision for the solution, at the system level by goals for the system, and at the element level by goals for the element.

In holistic design work on goal setting, the three levels help to provide a clear focus at the appropriate level and support the transition between two levels. In the process, the goals are specified from level to level.

When working on the vision, for example, goals for the system and the elements can already be considered in order to immediately think about how a system, with its elements, could implement the vision. Conversely, when working on element-level goals, system-level goals and solution-level vision must be considered. For example, goals for an element must not conflict with the system level or the solution level.

Example: The vision is to realize a digital solution for a hotel in which a hotel guest can act as independently as possible (solution level). Accordingly, a system could be implemented in the hotel that allows the guest to check in, check out, and pay the bill (system level). Likewise, the system will also replace room keys. For this purpose, an app could be defined that displays the PIN for the customer's room lock (element level).

Design of the form at the three levels

Design work at the three levels regarding form is, at its core, about breaking down the solution into a technical system and then further breaking down its own elements into their required components.

The decomposition work is illustrated by three important examples: the technical structure of the solution, the user structure perspective, and the information structure perspective.

In terms of the *technical structure*, the value creation architecture defines the structure (partner organizations, organizational structures, etc.) at the solution level to realize the value proposition. Frequently, relevant parts of the value creation architecture at the system level are realized not only by the system's own elements, but also by existing elements that have to be integrated into the system. At the element level, technical interfaces to own elements must be provided accordingly in order to connect these existing elements.

Example: A digital solution for a hotel provides for the customer to pay for their room in advance. The partner organization at solution level is thus a payment service or a bank. At the system level, the payment service provider's system must be taken into account as an existing system and addressed at the element level by means of a suitable technical interface.

In terms of *user structure*, customer/user groups required for the realization of the solution are defined at the solution level. For all customers who interact directly with elements of the system, appropriate user types must be defined at the system level. The same applies to all user groups that are involved in value creation as part of the organization. Based on the user types, corresponding user interfaces must be designed at the element level to enable interaction with the elements.

Example: A digital solution for a hotel allows both guests (customers) and employees of a hotel to book rooms for guests. Accordingly, user types for guests and for employees of the hotel are required, with corresponding elements and user interfaces for booking the rooms.

With regard to the *information structure*, it is only at the element level that concrete data that an element is to store or process and the structure of this data are defined. Nevertheless, information structures also play an important role at the solution and system level. At the solution level, the value proposition, value creation architecture, and business processes formulate requirements for data that the system must provide. At the system level, this raises the question of what elements are required to obtain, manage, and process the data.

Example: A digital solution provides for customer health data to be collected and analyzed in order to provide fitness tips. At the system level, the question arises as to how this data is collected, stored, and evaluated. Assuming the data is collected by a smartwatch, the data can be stored on the smartwatch, an associated smartphone app, or on a central server. Depending on the decision, the appropriate data structures must be defined for the respective element (e.g., the smartphone app).

Design of the function at the three levels

In terms of function, design work at the three levels defines the implementation of the value proposition and business processes. This is illustrated by the following examples.

In terms of *customer/user functionality*, business processes are defined at the solution level to determine how the value proposition is realized for the customer. At the system level, these business processes are illustrated with the help of scenarios and fully defined at the element level in the form of use cases. Business processes, scenarios, and use cases must be defined consistently with each other. When developing use cases, user types as well as existing systems and objects must be taken into account, since they can become actors in the use case through user interfaces and technical interfaces.

Similarly, business processes may require technical operations to realize the value proposition. Here, corresponding *technical functions* must be defined in the elements.

Example: A digital solution for a hotel provides for customers to check in independently at the hotel, receive the room key, and also check out again. At the system level, this process is described by corresponding scenarios, and at the element level, use cases are required that describe check-in, key transfer, and check-out. Let us assume the keys in the hotel are realized by PIN codes on the doors. Technical functions are then required that generate a guest-specific PIN so that this PIN can be communicated to the guest and also set in the door lock.

Design of quality at the three levels

The quality of a digital solution has a significant impact on its acceptance and success. For good quality, it is important to know that the digital solution and the digital system are not identical, and that good quality can be achieved only if it is addressed at all three levels. The digital system is the instantiation of the technical aspects (that is, the hardware and software) of the digital solution and is therefore only part of the digital solution (see section 5.25.2). There are qualities of a digital solution that are independent of the qualities of the digital system. However, the qualities of a digital solution can have an impact on and influence the qualities of the digital system and its elements. Vice versa, the qualities of a digital system become part of and add to the qualities of a digital solution.

For example, an online hotel booking service for tourists is a digital solution. This digital solution could, for example, be realized by a smartphone app that provides the features for booking a hotel online. This app is the corresponding element for the hotel booking service. The online booking service itself has qualities of its own, such as the freedom to book a hotel from all over the world, searching for hotels in various countries. These qualities can be defined independently without having a particular digital system and its qualities in mind. However, these qualities might have an impact on the requirements of a digital system. The corresponding digital system can have its own qualities. For example, the booking app should be easy to use, demonstrate good performance, and provide good aesthetics. These qualities of the digital system add to the experience and quality of the digital solution as a whole.

The role of the constraints at the three levels

Analogous to the goals, constraints are considered at all three levels of a digital solution and must be made more specific consistently.

In terms of working with constraints, it is important to note that constraints exist independently at all three levels and can influence each other.

Typical examples of this situation are country-specific requirements and laws. As long as a solution is only offered in one country, the respective laws apply (e.g., storage periods for data or rates for VAT). As soon as a solution is used in several countries, country-specific differences may have to be taken into account.

11.2 The importance of integrated consideration of the three design perspectives for holistic design

EO 11.2 Explain the importance of the joint consideration of people, business, and technology across the three levels for holistic design work (L2)

An essential factor to understanding holistic design in terms of the three perspectives of people, business, and technology is that all three levels of a solution contribute individually to the success of a solution in each of the three perspectives. Holistic design in the three perspectives of people, business, and technology therefore means the collaborative consideration of all three levels within each perspective and that of each perspective at all levels.

To this end, the following section discusses—for each of the three perspectives—how the three levels of solution, system, and element can affect each perspective. This section concludes with a discussion of the importance of considering all three perspectives in an integrated manner.

11.2.1 Designing an attractive solution (people perspective)

The basis for an attractive solution is the solution level with the value proposition for the customer groups and with them, the business processes that enable an attractive customer journey. Holistic design at the solution level in terms of an attractive solution means thinking about the customer groups, the value proposition, and the customer journey together to define a good value proposition for the customer groups. At the same time, the customer journey must be considered in a meaningful way so that customers learn about the solution and actually use it.

As the basis for the implementation of the solution, the system level makes a significant contribution to a good customer journey. Meaningfully defined user types and a good implementation of the business process through sensibly realized elements make an important contribution to a positive customer experience. The element level is also directly involved here.

The appropriate realization of the processes through the elements of the solution not only leads to a good user experience, but also improves the customer experience. Here, the difference between user and customer is an essential aspect. If the customer and user are one and the same person, then the customer and user experience together affect the same person.

Even if the customer and user are different people, the customer and user experience can still influence each other. Here is an example: a service employee in a restaurant has to serve a customer's order with a poorly designed software. The order can only be taken slowly and awkwardly. First, from the customer's perspective, this is a poor customer experience of the process (slow order intake). Furthermore, the service employee could be annoyed by the user experience of the software and thus additionally contribute to a poor customer experience.

Holistic design in terms of a good customer and user experience means designing the business processes at the solution level and the implementation of the business processes through the system and element level collaboratively and paying attention to the mutual effects.

11.2.2 Designing an economically viable solution (business perspective)

The solution level forms the basis for an economically viable solution. In terms of a holistic design, the value creation architecture and the business processes must be geared to ensuring that the solution can be operated such that costs are covered and such that it generates profit or meets cost expectations in a business context. Above all, this requires efficient and effective business processes and a sustainable value creation architecture.

The efficiency of business processes is largely determined by the implementation at the system and element level. The meaningfully structured system with user groups in combination with efficient use cases (system and element level) contributes to a high degree of efficiency.

The viability of the value creation architecture is largely determined by generating adequate revenues to meet the costs of development and operation and, if applicable, the revenue expectations of the client and other stakeholders. A relevant factor here is, on the one hand, pricing or the tapping of other sources of financing if direct revenue from customers is not envisaged. Other important factors may include the cost of building and operating the solution. In the case of new developments in particular, the building process can generate high costs, which must be recouped later through the solution.

In terms of an economically viable solution, holistic design at the three levels means recognizing and taking into account the influence of all three levels on the efficiency and effectiveness of business processes and the viability of the value creation architecture.

11.2.3 Designing a feasible solution (technology perspective)

The question of the feasibility of a solution begins at the solution level with the question of the technical feasibility or technical support of the business processes and the value creation architecture. The technical feasibility of the business processes is based on the fact that a feasible technical system can be designed that implements the business processes.

At the system level, the question of feasibility can go in a wide variety of directions. The following examples are intended to illustrate the broad spectrum:

- *Is the necessary equipment available to implement the business process?* Online shopping was not possible until a sufficient mass of people was equipped with Internet connections and corresponding PCs.
- *Are the required technologies powerful enough to realize the business process?* Video streaming could only be realized with the corresponding availability of powerful Internet connections.

- *Do the available technologies provide the expected functions?* Translation services based on machine learning could not be enabled until the technology could reliably translate languages.

For the system level, holistic design in the sense of a feasible solution means critically examining the availability and expected capabilities with regard to the planned business processes and considering the availability and potential of the technologies when designing the business processes.

At the element level, there is also the question of the feasibility of the elements and their functions in detail. Analogous to the system level, the spectrum of possible questions is very broad. Therefore, we give some examples here for clarification:

- *Is the planned user interface feasible?* An interaction with touch control could only be realized meaningfully with the corresponding end devices.
- *Are the available end devices powerful enough to enable the planned functions of the elements?* Voice control of smartphone apps only became possible when smartphones had correspondingly powerful processors.
- *Are the functions designed basically realizable by software?* Sending photos over the Internet was made possible by suitably powerful image compression algorithms.

For the element level, in the sense of a feasible solution, holistic design means that the feasibility of the designed form and function is critically examined and the possibilities at the element level are taken into account when designing the business processes and the system.

11.2.4 Holistic design at the intersection of people, business, and technology

For a comprehensive holistic design, this section considers the interplay of all three perspectives. In terms of holistic design, an optimal solution addresses all three perspectives in an optimal way. Based on the best available technology, it creates an economically maximally successful solution that is attractive to the maximum number of people. This ideal image of an optimal solution is easy to explain, but probably unattainable in practice. The following two examples illustrate how the three perspectives can influence each other.

At first glance, the highest possible automation rate for processes can have a positive impact on the value creation architecture (technology to business). However, poor implementation (e.g., many errors with high manual effort to rework) will reduce customer and user satisfaction (technology to people). This in turn influences the number of customers and thus the value creation (people to business).

Use cases that are as simple and usable as possible improve customer and user satisfaction and thus have a positive impact on the value creation architecture (people to business). Designing and implementing these simple use cases can generate a lot of effort on the part of the development team in the building process, thus reducing the supposed gain on the value creation architecture side or even turning it into a loss (technology to business).

Holistic design at the intersection of people, business, and technology means, above all, an awareness that advantages from one perspective are almost always bought at the expense of disadvantages in the other perspectives, and that holistic design work requires an ongoing balance between the perspectives.

Thus, a much more realistic goal of holistic design work is not to search for the optimum in each perspective, but rather to find a good compromise between the three perspectives. Understanding the three levels of a solution goes a long way toward working out this tradeoff.

11.3 The Importance of an iterative approach to the building process for holistic design

EO 11.3 Explain the iterative nature of scoping, concept work, and development and operation for holistic design work (L2)

In section 7.1.1, we stated that the design process is by its nature an iterative process. With the goal of achieving a meaningful compromise between the perspectives of people, business, and technology through holistic design work in mind, this section explains how iterative approaches to the building process can help achieve this goal.

For a general understanding of the building process, in section 0 we presented the three-step model (scoping, concept work, development and operation). Iteration, in the sense of frequent repetition of an activity, can be applied to the building process in two ways: iteration over all steps and iteration within a single step. The following figure shows schematically how iterations can proceed within the three steps.

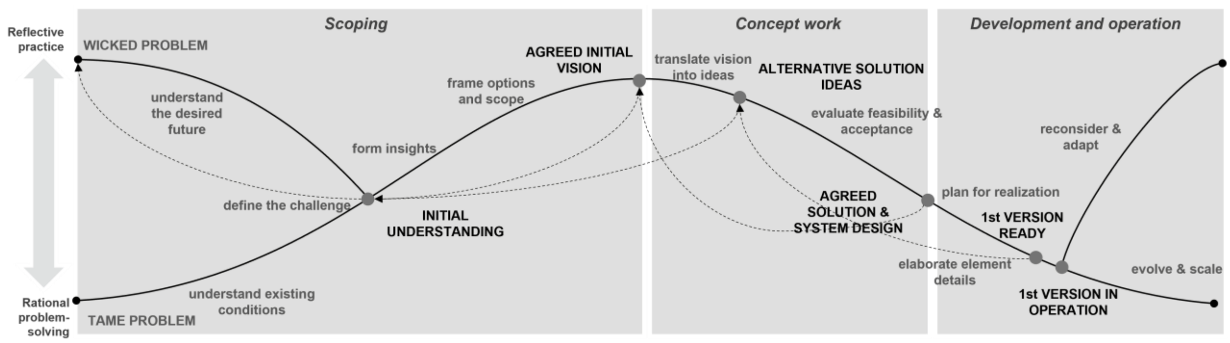

Figure 2.11 - An idealized model of the iterations in the building process

We describe the details of this figure briefly below, starting on the left with the scoping.

Scoping

At the beginning of the building process, we are in a situation where we face an abstract and open space of possibilities. To view the initial situation in a structured way, the distinction between tame problems and wicked problems (see section 7.1.37.1.3) is useful to note the two extreme perspectives in the building process.

When facing a wicked problem, the reflective practice mode (see section 7.1.2) is the correct approach. We have to understand the desired change together with all relevant stakeholders in order to reach an initial understanding of what the digital solution is about. Means for this understanding are early concepts (e.g., sketches) and prototypes (e.g., paper prototypes or storyboards).

When facing a tame problem, the rational problem-solving mode is the correct approach. We have to understand the existing conditions in which the problem exists for two reasons: first, to understand what has to be achieved, and second, to make sure that the problem is not a hidden wicked problem.

No matter what type of problem we face, we refine our understanding of the overall problem by analyzing tame and wicked parts and try to reach an initial mutual understanding of what we want to achieve. If it is not possible to reach this mutual understanding, the scoping must start all over again under the assumption that we are facing a wicked problem.

By mutual agreement, we again work in the reflective practice mode and start to make our understanding of the scope even more specific. This is done by discussing the problem from various perspectives (e.g., by looking at competitors or analyzing digital technologies that might help solve the problem). At the end of this process, a concrete and agreed vision for the digital solution is defined in the form of a design brief. If it is not possible to reach this agreement among all relevant stakeholders, the building process must start another iteration to define a new vision. In Figure 2.11, the iterations are represented by dashed lines. The visualization of these iterations describes the latest point in time where the need for an iteration can be recognized.

Concept work

With an agreed initial vision (and the other details from the Digital Design brief), the concept work can start. At the beginning of the concept work, we are once again in an abstract and open situation, as there are several alternative ways of achieving the vision. The reflective practice mode is the correct working mode in this situation since we have to translate the vision into various solution ideas (see section 7.1.2). For this purpose, we can develop initial concepts and create fundamentally different prototypes to explore these various solution directions. Details were presented in chapters 8, 9, and 10.

The understanding of the different solution ideas becomes more and more concrete during this work. An important result of this work is alternative solution ideas. Development and evaluation of alternative solution ideas is a key approach when performing design work. Alternative ideas allow us to explore the possible solution space in a systematic way and increase the possibility of finding a good solution idea. If the solution ideas developed are not promising enough, the initial vision should be questioned, and the building process should go back to the scoping in order to develop a new vision for the digital solution.

During this process of concretization, the working mode shifts from the reflective practice to the rational problem-solving mode. The various solution ideas should be translated into system ideas. The feasibility and acceptance of these ideas must then be evaluated with the relevant stakeholders. At the end of this process, agreed initial solution and system designs for the digital solution are created that are sufficiently detailed to accept the risk of starting the development. If it is not possible to reach such a design, the development of alternative ideas must start again.

Development and operation

Although the design and realization concepts created are concrete from the perspective of concept work, the start of development again means an abstract and open situation. There are multiple ways to start development. Therefore, an initial plan for the realization of the digital solution must be defined at the start of development.

In this situation, the rational problem-solving mode is the proper way of working, since many detailed design decisions have to be made in order to elaborate and evaluate the various details of the digital solution. For digital solutions, this part of the process is a real challenge and requires experts from various domains.

From an idealized perspective, the end of this work must be a first version of the digital solution that is ready for operation. In the course of this process, a critical decision must therefore be made: is the solution realized so far ready for operation or can an operational solution be achieved under the given constraints and on the basis of the previous ideas?

If the answer to this question is no, then the building process can either just be aborted or the process goes back to concept work. These options seem radical, but they are the only sensible paths. If no solution can be realized on the basis of the previous concepts and under the given constraints, then further development does not make sense and the existing concepts must be questioned through intensive concept work or the process must be aborted altogether.

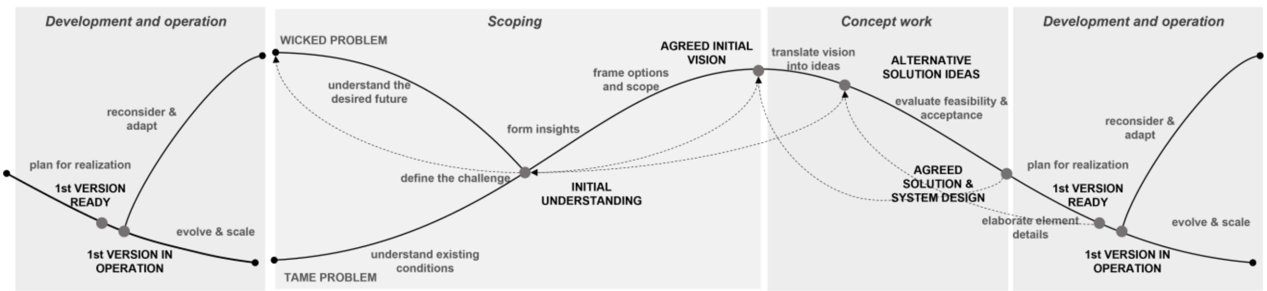

Figure 3.11 - The end of the building process is a new beginning

Once the solution is in operation, feedback can be gathered from real customers and users. In general, there are now two options: one option is that the solution is accepted by customers and users and creates the desired change so that the solution can be further developed and hopefully scales to become successful. The alternative outcome is that the solution is not or is only partially accepted and the goals defined are not or only partially fulfilled, although all previous evaluation measures indicated that customer acceptance was very likely. We consider such a situation as a kind of wicked problem since we have to reconsider our whole understanding of the solution.

Basically, the building process can now be thought of further as follows:

- The required changes are small and manageable: the building process remains in the *development and operation* step and the solution is continuously developed.
- The required changes are so extensive that they require a radical rebuild of the entire solution: in this case, a new scoping must take place and the idealized model of the building process (see Figure 2.11) should be run through again.
- Individual parts of the solution require comprehensive adaptation or the solution must be expanded to include new components: in this case, a new building process with a new scoping is started for the part to be changed (see Figure 3.11, right part) and the further development of the previous solution remains in the *development and operation* step.

What is important to understand for the foundation level at this point is that the building process can continue in a number of ways. However, the concrete procedure for parallel processes goes beyond the foundation level and is not considered further here.

12 Frameworks for the building process from a Digital Design perspective

In this chapter, we present examples of different frameworks that can be used to build digital solutions from a Digital Design perspective. Due to the different levels of detail, in this chapter we use the term *framework* as a generic term for procedure models, process models, and other working models.

The main goal of this introduction is to get an overview of possible frameworks and their possible applications in the building process and, in particular, to locate the perspective of design in the sense of Digital Design in these frameworks. We then give examples to show how several frameworks can be combined in a meaningful way within the building process.

12.1 Selected frameworks from a Digital Design perspective

EO 12.1	Compare and contrast future search, design thinking, design sprint, plan-driven development, scrum, and lean startup approaches as frameworks for the development process from a Digital Design perspective (L2)

There are a variety of frameworks in literature that present procedures and processes for building digital solutions in whole or in part. Each framework has its own strengths, advantages, and disadvantages. From a Digital Design perspective, it is important to know that there are different frameworks and that frameworks can be used differently in each step of the building process.

This section presents well-known frameworks that can be useful even at the foundation level.

12.1.1 Future search

Future search [WeJa2010] is a framework for engaging large groups in a planning process. A major application area for future search is citizen participation and urban planning.

Future search defines a process along a time frame of three days in which typically, a large group of stakeholders (50 people and more) works on a question in a structured way and derives concrete measures for next steps from this question. Key principles of future search are:

- The whole system (in terms of the environment and relevant stakeholders) should be brought into one room.
- The group should think globally but act locally.
- The focus should be on the future, not on current problems.
- All participants work in self-guided groups.

The three days of a future search proceed along the following steps:

- Understanding the past
- Understanding the present
- Focus on the future
- Identification of commonalities
- Measures planning

The literature provides specific processes, templates, and guidance for facilitation, space, and process planning for all steps to guide the large group of people through the process efficiently and effectively. The effort required for a future search is very high, depending on the number of people involved, due to the space, resources, and personnel required.

From a Digital Design perspective, future search can be used especially for content work with large groups during the building process. For example, a future search can be used to engage a large group in developing a shared vision at the beginning of the building process during scoping. Similarly, smaller groups can be surveyed through a future search; this requires correspondingly fewer resources, but also provides a less broadly based result.

12.1.2 Design thinking

Design thinking [Brow2009] is a framework for solving problems or developing innovative ideas. Key aspects of this approach are an interdisciplinary team, a space to work in, and a defined process.

With a view to the problem or the goal, the design thinking team should be specifically set up in an interdisciplinary manner in order to include the broadest possible spectrum of people, ideas, and perspectives in the process.

Space means a shared workspace where the team can work together as part of the process. The workspace is considered to be of great importance for creativity in design thinking, so that the team feels comfortable and has all the necessary resources (technology, material for prototypes, etc.) to work on the problem.

The process of design thinking is defined slightly differently in literature depending on the source, but essentially consists of the following steps:

- *Develop emp*athy: In this step, the team should develop empathy for the people affected, for example, by analyzing their needs, desires, and motivations.
- *Define the problem or point of view*: In this step, the team should translate the insights gained into a common problem definition or point of view to decide what the problem is or what issues should be addressed through the process.
- *Find ideas*: In this step, the team develops as many ideas for possible solutions as possible in relation to the problem and prioritizes them.
- *Develop prototypes*: In this step, the team develops the simplest possible prototypes for the most promising ideas. The goal here is deliberately not perfection, but testing and quick feedback.

- *Test prototypes*: In this step, the team tests the prototypes with relevant stakeholders to get feedback on the ideas and to validate them.

The process in design thinking should be followed but applied flexibly and iteratively. For example, it is possible to jump back in the process at any time if this serves to gain understanding. For example, when building a prototype, if the team realizes that a modified problem statement fits better, then the team should go back and revise the original problem statement.

From a Digital Design perspective, design thinking is a useful approach that can be brought to bear in a variety of situations. What is important for the use of design thinking in the building process is the understanding that design thinking is meant for developing ideas and as a result, delivers ideas that have been evaluated with stakeholders and especially end users based on simple prototypes. Design thinking can thus be applied usefully in all steps of the building process.

Key aspects in relation to design thinking are the resources and time involved [LiOg2011]. If you take the approach seriously, design thinking engages a larger team over several weeks to achieve the desired result. Therefore, the specific questions that are addressed in design thinking should be critically selected and the process itself should be sensibly planned into the overall project.

It follows that design thinking should preferably not be used in a critical implementation phase and for clarifying detailed questions. For such situations, for example, the design sprint presented below is more suitable.

12.1.3 Design sprint

The *design sprint* [KnZK2016 can be understood as a compact variant of design thinking. The essential feature of the design sprint is a clearly defined execution of the process steps based on a plan for exactly five days:

- Day 1: Understand the problem
- Day 2: Outline solution ideas
- Day 3: Choose best solution ideas
- Day 4: Build prototype
- Day 5: Test prototype with five customer representatives

Successful implementation of this strict schedule depends on a number of factors. The people necessary for execution and the relevant stakeholders must be available. Furthermore, stakeholders are needed who are authorized to make decisions with regard to the problem, the solution ideas, and the prototype. In addition, literature provides a number of tools and checklists for successful implementation.

From a Digital Design perspective, the design sprint can be used in the same way as design thinking. Due to limited time and resources, the range of ideas and the degree of evaluation and feedback are limited.

Therefore, the design sprint may be used more for smaller and more focused issues that can be addressed in a week as part of the approach. In principle, the design sprint can also be applied in all steps of the building process. Due to the firmly defined time and resource requirements, however, a design sprint can be used in particular in the *development and operation* step to develop solutions for smaller but important problems.

12.1.4 Plan-driven development

The term *plan-driven development* refers to process models that implement a solution according to a highly structured plan and in defined phases [Royce1970].

Plan-driven development is based on the assumption that the requirements for a solution can be formulated as completely as possible in advance. The requirements formulated are then transferred, in a step-by-step process, into a design ready for implementation, which is then finally implemented and rolled out. An essential feature of plan-driven development is that the results of the respective phase (requirements, design, etc.) must be fully developed and approved before the next phase can be started. If errors or incompletions are discovered in a downstream phase, the process can go back to previous phases and correct the errors or work out the incompletions.

Plan-driven development is used when external constraints require a strong structuring of the building process (e.g., building process in a public context) or when very high demands are made on safety and functional quality (e.g., in aerospace).

For Digital Design, a strict interpretation of plan-driven development means that the essential design work takes place at the beginning of the process. The goal of the design work is then to develop a conceptual design of the digital solution that is as complete as possible in terms of a requirements specification and to coordinate it with the client. In terms of the building process, this means that the scoping and concept work are performed in the design phase. Furthermore, all solution elements that are relevant for the realization must be fully elaborated.

In the downstream phases, this requirements specification (see above) is transformed into a technical design, which is then technically implemented. The Digital Design work essentially consists only of clarifying open questions and details that are relevant in the context of further implementation. In a very simplified way, plan-driven development can be visualized with the three core activities of the building process (see section 4.2) as follows:

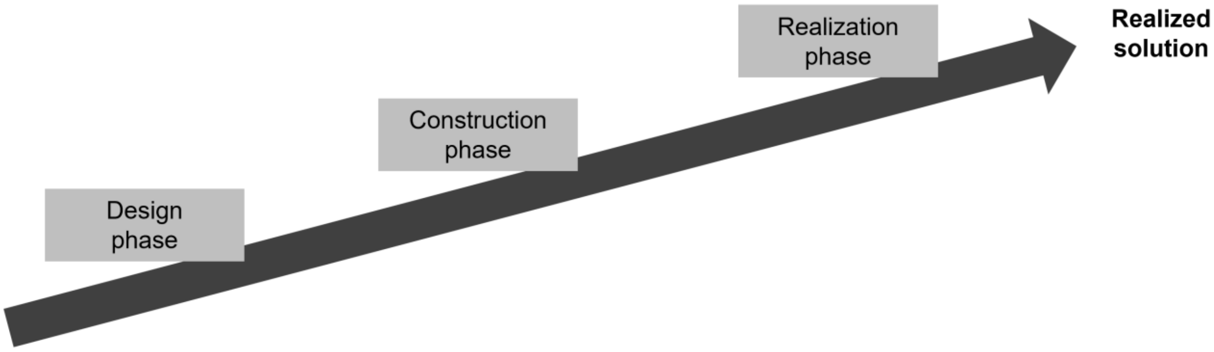

Figure 12.1 - Simplified model for plan-driven development

From a Digital Design perspective, plan-driven development is challenging in that all relevant details of a solution must be worked out and fully defined in advance. Even though the models basically provide that errors or incompletions from previous phases can be corrected, such phase reversals typically mean a very high effort, because the plan previously made has to be changed.

12.1.5 Scrum

Scrum is a very popular example of an agile development framework for developing complex products [ScSu2020]. The term *product* is defined very broadly. A product does not necessarily have to be software; a product can also be a concept or a physical object. Importantly, it must be possible for a product to evolve incrementally and iteratively, with each increment delivering value in terms of the product.

The work process is organized based on a prioritized product backlog and structured in fixed time periods (known as sprints). Scrum provides that a product owner defines the work for the next sprint together with the scrum team. Backlog items are taken from the product backlog and prioritized in the sprint backlog. The sprint backlog is processed according to priority in the sprint. At the end of a sprint, the result is the realized increment, which is reviewed by the stakeholders and the scrum team. The findings from this test are prioritized for further processing in the product backlog.

From a Digital Design perspective, scrum allows design work to be done by both the product owner and the developers. The task of the product backlog must be understood as the central means of work organization in scrum. According to the Scrum Guide [ScSu2020], the product backlog is *a living, ordered list of things needed to improve the product*. From a Digital Design perspective, the *things* in the product backlog can be primarily as follows:

- *Description of an aspect to be implemented for the digital solution:* In this case, the product owner has defined an implementation-ready design in terms of Digital Design (see section 7.27.2) and gives the scrum team the task of implementing this aspect of the digital solution. The value of this increment then consists of a concrete further development of the solution.
- Description of an aspect of the digital solution to be designed for downstream implementation in a later sprint: In this case, the scrum team is tasked with design

work in the sense of Digital Design and develops a design for one aspect of the planned solution. The value of this increment is new insight for all stakeholders in terms of further development of the digital solution. The findings can then be incorporated into the further development of the product backlog.

- *Description of a prototype to be implemented for a defined aspect of the solution:* In this case, the scrum team is tasked with prototyping work in terms of Digital Design. The value of this increment, as with concept work, lies in new insights for all involved through realization and testing of the prototype. The findings can then be incorporated into the further development of the product backlog.

An essential idea of scrum is to approach a permanently executable, value-adding, and potentially deliverable part of a product in defined and small iterations (the sprints). In this process, design concepts can represent a value-creating intermediate step for complex solutions. In the sense of Digital Design, design concepts in the context of scrum should therefore be understood as value-creating products that are developed collaboratively by the scrum team during the sprints.

All those involved are also required to be transparent and to continuously reflect on and adapt their own working methods. This also refers in particular to the work with design concepts. The scope and level of detail of concept work are continuously reviewed and adjusted through this scrum feedback process.

With regard to the building process, scrum is particularly well-suited for the *development and operation* step of the building process. The main limitation is the size and complexity of the solution to be implemented. When a digital solution becomes so large and complex that it requires more than one scrum team, it is called *scaled agile development*. However, this topic is far beyond the focus of this handbook and is not considered further here. Further details can be found in [CPRE2022].

12.1.6 Lean startup

Lean startup [Ries2011] is an approach to developing companies and their products using the shortest possible development cycles. At the core of lean startup is the idea that every aspect of a product is a hypothesis that needs to be validated with real customers as quickly as possible. These short cycles enable a company to learn quickly and develop its products further in the interests of its customers.

The process of lean startup is described with the three steps *build-measure-learn*. In concrete terms, the aim here is to realize, based on a defined hypothesis (What do customers want? What problem do they have? What added value do they need?) a *minimum viable product* (MVP), on the basis of which, data is then collected as objectively as possible to validate the hypothesis. The company can then learn from this data and formulate new hypotheses so that the process can start all over again. Figure 2.12 shows the process in simplified form.

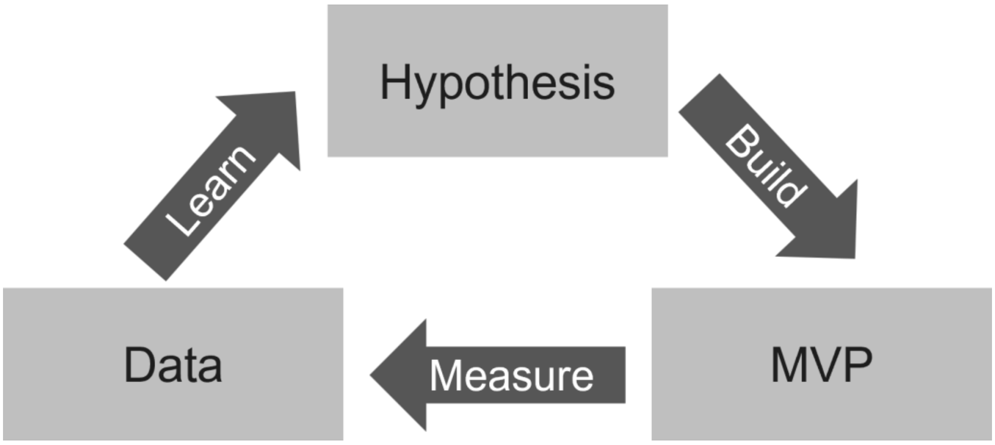

Figure 2.12 - The build-measure-learn process in lean startup

This process in lean startup is also described by the term *validated learning*. This is to express that companies should develop specifically on the basis of data and experiments. The MVP, along with the process, is an essential aspect of understanding lean startup.

In lean startup, the MVP is defined as a version of a product that allows the team to gain the greatest amount of insight with the least amount of effort. The important thing here is the fact that the MVP is not a prototype, but actually goes live to get real feedback from customers. At this point, to understand lean startup, it is important to know that the MVP is defined and used differently in other contexts (cf. [CPRE2022]).

From a Digital Design perspective, lean startup is a framework that places a special emphasis on evaluating the solution with real customers in real operations. Using the terms of the three steps of the building process, the process in lean startup continuously goes through the steps: scoping, concept work, and development and operation. The existing digital solution (the current MVP) remains in operation and is further developed with all the advantages and disadvantages of operation.

The quick feedback on the current MVP and the resulting insights for the further development of the MVP are the big advantages for the design work in Digital Design. At the same time, however, the pace and the focus on a rapid realization of the MVP is also challenging. Concept work and prototyping take a back seat in lean startup in favor of rapid MVP development. Of course, this does not mean that there is no concept work in lean startup. However, concept work should focus on implementing a meaningful MVP that maximizes learning.

Lean startup is particularly well-suited as an approach if a startup is actually to be founded, as it structures all aspects of such a startup. However, the approach can also be used in contexts where the fast cycles and delivery of an MVP to real customers and further development in real operations are possible or accepted by the clients and customers.

12.2 Combining frameworks in a building process

EO 12.2 Explain the possibility of combining frameworks in the building process (L2)

The previous section presented various frameworks that can be applied in a building process. Each framework has its advantages and disadvantages and natural environments in which it works particularly well. Likewise, the frameworks have specifics when it comes to design work and the Digital Design perspective.

The selection of frameworks for the building process is an issue to be decided with all relevant stakeholders.

Knowledge of the various frameworks is important from a Digital Design perspective in order to participate in the decision about how to proceed with a project or organization, and to properly consider the Digital Design perspective in the decision. What is important to understand in this context is that frameworks can be used as a kind of construction kit from which an organization or product development can draw.

Of course, the ability to plan building processes goes far beyond the skills of an entry-level digital designer, and far beyond Digital Design itself. Nevertheless, it is important to be aware of the ability to combine frameworks in order to see the potential and opportunities that arise from this combination.

The following two examples illustrate this.

Example 1: Design thinking, lean startup, and scrum

Let us imagine that a group of people in a company has a good idea for a digital solution. The group uses design thinking as an approach over an extended period of time to look at different solution ideas. At the end of design thinking, the group has three good ideas for a solution.

The group decides to bring the most promising solution to market as an MVP, following the lean startup model. After several iterations and good feedback from customers, a good and very clear concept of the solution emerges. The group elects a product owner from among its members and switches to scrum with the approach to further develop the MVP into a fully-fledged product in rapid cycles.

This could be an example of how three models seamlessly merge and play to their respective strengths with the design of the solution in mind. Design thinking was used to design an initial good direction for the solution. This direction was then brought into operation with lean startup and critically examined. When the idea stabilized, further development based on scrum could start.

Example 2: Plan-driven development and scrum in parallel

Now let us imagine an organization that wants to digitialize their entire business processes with new software. Since this project represents a high risk for the organization's business operations, the decision is made to go for a plan-driven development and start working out the requirements. It turns out that three out of five business units have very clear ideas about what the software should be able to do. The elaboration of the requirements is very fast for these three areas. Unfortunately, the situation is different for the other two areas. Here, the requirements are only vaguely tangible and there is some uncertainty as to what a software solution for the area might look like.

From the company's perspective, all five divisions have fairly clear interfaces for collaboration, so the company decides to split the implementation of the solution. The realization for the three understood areas continues to follow plan-driven development, whereas the other two areas each work according to the scrum approach. As an overarching goal, all five areas define that a first version of the solution should be ready for the areas in 12 months at the latest, so that a trial run of the solution can start.

13 The social dimension in the building process

For the building process and the success of a digital solution, the previous chapters presented different practices and approaches to the content dimension of a solution. The successful application of practices depends not only on methodological skills, but also on the respective people working together in different situations (e.g., departments, projects, or even project steps). This human perspective is also referred to as the *social dimension* of the building process (cf. e.g., [VPGV2008]).

At foundation level, in Digital Design, you should have an understanding of the importance of mindset and develop an awareness of the relevance of the social dimension. For this purpose, we explain important basics about mindset in section 13.1. Then, section 13.2 shows which factors are relevant to the three steps of the building process from a social perspective. This awareness is necessary in order to recognize effects and influences of the social dimension in the building process and to further develop your own handling of the social dimension based on this awareness. Finally, we discuss essential aspects for a design mindset in section 13.3.

13.1 Mindset basics

EO 13.1	Know the importance of the mindset as the foundation for the social dimension (L1)

The essential prerequisite that people bring to the table is their personality or mindset, that is, in particular, their mentality, their value system, and their attitude in dealing with problems and with other people. Everyone thinks and communicates differently depending on their personality, activity area, and situation. People are influenced differently by education, culture, and their individual life experience. People can acquire specialized knowledge and skills. Based on their individual mindset, people find it easier or harder to learn and apply something.

A typical example of the difference between mindset and skills can be observed when filling roles in projects. In simple situations, roles are requested and suitable people are sought who have the required skills for the role (e.g., interaction designer). In difficult situations, we often observe that no roles are requested, but instead, specific people with their names. For example: *The situation is so complex that we need someone like Betty for this.* These formulations show that the competence requirements often contain a combination of experience, skills, and also personality or mindset.

It is also helpful to understand a person's individual mindset. Mindset describes a person's way of thinking, that is, the way they take in information, what things they give importance to, and how they make decisions. On the one hand, the mindset is represented by the temperament that people *bring with them from home* and by what people are *born with*. On the other hand, however, it is also due to the attitude that a person develops in the course of their life. Different mindsets can be recognized, for example, in thinking, in values, in language or terminology, in behavior, or in the way of dealing with stress.

To give some examples: some people prefer to pay more attention to details. Others find it easier to see the big picture. Some rely on knowledge and experience from the past. In return, these people have difficulty developing their own imagination for the future. Others, on the other hand, find it easy to look to the future. Personality models such as the Keirsey temperament sorter [Keir1998], MBTI, or the big five personality dimensions are used to identify mindset traits and thus understand people's behaviors and personal motivations. Knowledge of these personality indicators can be useful in Digital Design to guide communication and understand team dynamics and leadership in the building process.

13.2 The three steps of the building process from the perspective of the social dimension

EO 13.2	Justify the importance of the social dimension in the building process for a digital solution (L2)

So far, we have considered the three steps of the building process from a process perspective (What frameworks exist?) and from a content perspective (What aspects need to be clarified and to what depth/maturity?).

The aim of the social dimension is to gain a better understanding of the people involved in the building process in order to address the challenges that people bring with their different backgrounds and individual contributions.

For this purpose, we consider the three steps of the building process from the perspective of the people in order to show important aspects of the social dimension. Generally, we refer to *people* in this section, as the aspects can be relevant for the building team as well as for clients, customers, and users. If a stakeholder role is particularly affected, we highlight it accordingly.

13.2.1 Social dimension in scoping

In the scoping, the goal is to work with the client and relevant stakeholders to develop a shared understanding of the need for change (Why are we starting a new building process? What is the reason for action?), the target picture for the digital solution (What do we want to achieve?), and the available staff and resources.

Dealing with the initial situation—tame vs. wicked problem

Following the design squiggle, scoping is characterized by an intensive orientation process for all stakeholders. At the beginning of this orientation process is the question of what situation you find yourself in. The distinction between tame and wicked problems that we have already introduced is helpful here. Depending on this situation analysis, different personalities are needed.

For a tame problem, people who can perceive and name concrete challenges are more helpful. In a wicked problem situation, people who can approach the situation in the abstract, as well as summarize the multiple aspects and issues of the situation, are more likely to be helpful.

Dealing with risks during scoping

People deal with risk differently. Risk-averse people may tend to demand a lot of information in order to make the most information-based decision possible about a risk. Risk-takers may tend to trust their intuition rather than obtain and evaluate extensive information.

For people in the client role in particular, managing risk is critical to the way we work in scoping. Risk-averse clients tend to demand early decisions and clarity regarding the assignment to be able to control the situation. Risk-taking clients may be more likely to waive this control, allowing greater latitude. The same idea can be applied to people who are in the role of the customer, the user, or the building team in the process.

Dealing with the "blank slate"

Another factor in scoping is dealing with what tends to be a wide variety of options. Figuratively speaking, this situation is also referred to as a blank slate that needs to be filled.

The task of filling this free space has different effects on people. One extreme is a strong aversion to such situations. Such people tend to feel negatively about the space that is opening up and seek the advice of third parties to fill the space. On the other hand, there are people who are open to free space and see it as an opportunity to create something new, explore possibilities, and fill the blank sheet of paper with their own ideas.

On the part of the building team, people who can handle this free space well are therefore helpful.

Conclusion for the social dimension in scoping

Dealing with the initial situation, dealing with risks, and dealing with freedom that presents itself are three examples of human factors related to scoping. For the foundation level, the following findings emerge from these examples.

Design problems carry a high percentage of wicked problems. Therefore, people who are willing to take risks, who can question the status quo, and who find it easy to design a big picture are important for scoping. This allows them to give detail-focused people an overview of the context.

13.2.2 The social dimension in concept work

The goal of the concept work is to develop a sufficient understanding of the solution and the underlying technical system among all relevant stakeholders. Based on this understanding, the client must decide whether or not to take the risk of implementing the solution.

Combination of divergent and convergent thinking

In the spirit of the design squiggle, concept work is also characterized by an initially rather chaotic orientation process. Different ideas for the solution are designed, investigated, and discarded. At the end of the concept work, however, there must be a meaningful design for a good solution.

In this process, two different ways of working in the building team are highly relevant. One is the ability to open up the realm of possibilities, that is, to try out and explore different possibilities (divergent thinking). On the other hand, there is the ability to think an idea through to its conclusion and bring it to a level that allows an informed decision (convergent thinking).

Time orientation for finding solutions

Another important perspective is the direction of thinking with regard to the past and the future. There are people who orient their thinking primarily on existing things that are tried and tested and socially accepted. These people like to transfer this experience to the future.

On the other hand, there are people who base their thinking primarily on trends and potential future developments. This thinking tends to generate new ideas, most of which have yet to prove their potential.

The third category of people is good at switching between the two perspectives, combining experiential knowledge and knowledge of future trends.

Theory focus and practice focus as a working style

Concept work is the playground of people with a theory focus, as it involves designing a solution in terms of ideas and thoughts. These people are often seen in the role of architect, inventor, critic, or visionary. In contrast, the practice-focused person directs their energy toward realizing and engaging with the real and tangible world.

Theory-focused people find it easier to change perspectives. As a result, they are more likely to succeed in questioning the status quo when finding solutions and thus find innovative solutions. Practice-focused people, on the other hand, are better able to focus on an existing solution variant and are thus less likely to get bogged down in finding a solution.

The difference in the two working styles often leads to misunderstandings and communication problems, as theory-focused and practice-focused people perceive and give meaning to things differently.

Conclusion for the social dimension in concept work

Convergent or divergent thinking, the preference for temporal orientation, and the distinction between theory and practice are three examples of human capabilities that are significant for concept work. In Digital Design at foundation level, the following key lessons emerge from these examples.

In concept work, people are needed who, as a group, have both skills in a balanced measure. Too much divergent thinking would lead to a never-ending process, as no idea is thought through to the end and you get bogged down. Too much convergent thinking risks not exploring the potential realm of possibilities in a meaningful way and committing to a suboptimal potential solution too early.

The relationship between people with a theoretical and practical background should be understood differently in concept work. First, a greater proportion of theory-focused people are needed to meaningfully open, explore, and prioritize the realm of ideas. We know from experience that the prioritization step in concept work is initiated by practice-focused people. In order to bring the concept work to a meaningful conclusion, practice-focused workers are needed at a certain point to bring the process to a close (What are we really doing now?).

Depending on the desired degree of innovation of the solution, more or fewer people with a future orientation are needed in the concept work. If the degree of innovation is high, the proportion of people with a future orientation should be high, and if the degree of innovation is low, it should be correspondingly smaller.

13.2.3 The social dimension in development and operation

The goal of development and operation is the actual implementation of the digital solution and with it, the achievement of the planned change as the goal of the design work. Therefore, this step is the playground of practice-focused people (see section 13.2.2).

In the spirit of the design squiggle, the goal now is to take the building process to an important milestone with the operation of the first version of the solution.

Your own demand for perfection

The demand for perfection in the sense of finding the best solution plays an important role in development. On one side of the spectrum here are people who claim to deliver the best solution in the broadest sense. These people invest a lot of time in finding the best solution. On the other side are the pragmatists. For them, the motto *good enough is good enough* applies.

Willingness to make mistakes and receive feedback

People behave differently when it comes to mistakes, feedback, and criticism. There are people who shy away from conflict that can be associated with mistakes and negative feedback or criticism because they take these mistakes personally. On the other hand, there are people who consciously demand feedback and see mistakes as an opportunity to learn in order to design future-proof solutions.

Conclusion for the social dimension in development and operation

The demand for perfection and the willingness to *make mistakes* are two examples of human skills that are important for the development and operation of a solution. For Digital Design at foundation level, these examples yield the following important lessons.

A *high percentage* of people striving for perfection during development enables the best possible solution to be created. A combination of people striving for perfection and low willingness to make mistakes in the building team carries the risk that a solution will be finished too late.

A *small percentage* of people striving for perfection bears the risk that the next best solution will be realized and that the quality of the solution will suffer. The goal for development and operation should therefore be to strive for a good balance between perfectionism and willingness to make mistakes.

13.3 Essential aspects for a creative mindset

EO 13.3	Describe key aspects of the mindset that are important when designing a digital solution (L1)

In the following, we consider seven aspects that are important for design work in terms of the mindset. In section 0, we defined design competence as the capability to design. With regard to the mindset, the idea of design and the associated holistic claim is important. Therefore, in the following, we use the term *design* and we speak of a *design mindset*.

13.3.1 Design work means looking through other people's eyes

Design work should always be understood as work that you do for other people. Above all, this means learning to look through the eyes of the people for whom a solution is being designed. In general systems theory, this is called a second-order perspective. Let us clarify the core of this idea and of the distinction into first and second order with an example.

A first-order activity is an activity that a person performs in relation to themselves. When someone buys a jacket or other piece of clothing for themselves, that person takes knowledge about themselves (clothing size, color taste, intended use, etc.) as the basis for the purchase decision and makes the decision based on their own experience.

A second-order activity is an activity that one person A performs for another person B. In this case, person A must either have knowledge about person B or make assumptions about person B. For example, using the example of buying clothes, person A needs to know or at least guess the clothing size, color taste, and intended use of the clothing. The better person A's knowledge about person B is, the better person A can buy a garment.

Design work is always done with the stakeholders in mind (clients, customers, users). This brings us to the area of a second-order activity. At first glance, this statement seems philosophical. However, this finding is of great importance for the work in the building process, since essential aspects for the work in the building process can be derived from this finding:

- *Thinking from the stakeholder perspective:* Design requires a deep understanding of the stakeholders for whom the solution is being designed. In terms of Digital Design, you have to be aware of this fact and actively work on being able to empathize with the different stakeholder perspectives.
- *Designs must always be understood as hypotheses:* Any decision in the design to shape a solution is always made based on the potentially limited and error-prone

knowledge about the stakeholders. In the sense of Digital Design, designs must therefore be understood fundamentally as hypotheses that need to be confirmed or refuted by appropriate procedures.

- *No clarity without evaluation:* If designs are understood as hypotheses, then it follows that the evaluation of a design is just as important as the work on the design itself. This makes evaluation as essential a part of the design work as the design work itself. Only appropriate evaluation with the right stakeholders can verify whether a design is good or bad.

13.3.2 Cultivate good communication

The first aspect was aimed at the attitude towards your own design work. The second aspect is about meaningful communication with stakeholders.

In terms of the social dimension, the building process can be understood as a system of individuals communicating with each other. Good communication is important for interdisciplinary work in Digital Design, so you should be aware of how important good communication is for togetherness and how much you can also irritate or support people with your own communication and thus disrupt or promote togetherness.

In addition to this interpersonal level, the aspect of mutual understanding is of great importance for design work. In terms of Digital Design, you have to understand that people will always communicate and, above all, understand communication based on their individual backgrounds and experiences. Summarized in one sentence, this means: *The receiver and the recipient decide on the message*.

In this context, good communication means paying attention, in the sense of Digital Design, to whether the other person understands my statements in my sense, can follow me, and wants to follow me. This stance is relevant for two reasons.

First, digital solutions are usually complex for people outside the field. Therefore, stakeholders often need time to understand a solution or design in its depth. This complexity arises from the demand to understand essential details in interaction and to grasp the big picture at the same time.

Second, stakeholders often have to make important decisions regarding a digital solution. In order to make meaningful and good decisions, stakeholders must have an adequate understanding of the issues to be decided.

13.3.3 Experimenting with alternatives for good solutions

Designing good solutions is strongly related to considering diverse alternatives for a solution in order to choose the best possible solution from a range of possibilities [NeSt2014]. Behind this insight is the attitude of being able to work with different alternatives and not committing to one solution option unnecessarily early on.

Experimenting with alternatives refers to all levels of a solution. The question of added values or variants for business models is just as relevant for the success of a solution as the question of the system structure or the question of the design details of a user interface.

Experimenting with alternatives requires courage to step outside your comfort zone and to question what seems to be undisputed in expert circles. Likewise, thinking in terms of alternatives is closely linked to the awareness that discarding conceived alternatives is not a failure, but must be understood as an essential part of design work. A rejected alternative (e.g., for a system structure) must therefore by no means be understood as a waste of time, but rather as part of a cognitive process.

However, the design of alternatives must also not be understood and forced as an end in itself. For example, it is not necessary to develop alternative designs for proven and understood functions of a solution. Working on alternatives is especially important when the problem under consideration is particularly critical to the success of a solution, or when a solution to a problem seems self-evident.

13.3.4 Promote creative tension (creative traction)

Every building process means a change and this change is realized by the people who are involved in the building process. The motivation of these people can be characterized by the concept of creative tension.

Creative tension is defined, according to Peter Senge [Seng2006], as the discrepancy between the actual state and the goal of a group, that is, the difference between vision and reality. In this sense, an attractive and good vision generates tension that motivates people to participate and contribute to the design and change, to design a good solution, and thus realize meaningful change. At the same time, there are many factors that can counteract this tension and thus hinder or even completely prevent change.

For the foundation level and to understand the building process of a solution, we can deduce from the concept of creative tension that in terms of good Digital Design, we must adequately observe and continuously encourage the energy of the people involved. In this way, the motivation and commitment of all participants can be maintained in a targeted manner. An important tool is the formulation and communication of a good vision and a meaningful reason to act as part of the scoping. In this way, a clear picture can be defined that serves as the goal of the building process for all parties involved and opens up the right scope for the design work.

13.3.5 Pay attention to the right leadership and team composition

In chapter 2, we defined design as bringing about intentional change. Thus, design work is not only mental work in the sense of drafts and ideas, but rather pursues the goal of bringing about real, manifested change.

For change to become a reality, people are needed to drive the change regardless of the specific process and circumstances. Who these people are depends, of course, on the specific environment and the individual leadership aspirations of the people involved. In some situations (e.g., during scoping or concept work), you can take over this task in the interest of Digital Design and lead the process. In other situations, care must be taken to hand off leadership to those who can be more effective in a situation.

Good design is achieved through interdisciplinary collaboration. At the same time, each phase in the building process needs a different type of team constellation to be effective. In terms of team composition, attention should be paid not only to disciplines and the associated skills in terms of interdisciplinarity, but also to good composition in terms of mindset.

For the attitude in terms of Digital Design, this aspect means making sure that someone takes on the leadership task, pays attention to the team composition, and drives the change process appropriately. It is important to note that this management task is not necessarily performed by people trained in Digital Design.

13.3.6 Design requires a continuous process of cognition

In addition to the designs and prototypes created, a great deal of implicit knowledge about the solution idea and the context arises during the building process for all participants. This knowledge is not always equally distributed and equally weighted. Much of this knowledge can be lost, especially during transitions in the building process and team changes.

For the foundation level and understanding, it is important to know and minimize this risk of knowledge loss and translation gaps. A useful approach to this is to use dedicated individuals as bridge builders to guide the process. Bridge builders can perceive and smooth these transitions in the building process as a link between the worlds of the *theory focus* and the *practice focus*. They can provide an overview of different expectations, help orient to the big picture of the solution, protect existing ideas, and provide proper translation during transitions.

Similarly, teams can be formed that involve people relevant to subsequent steps in the building process at an early stage. This early involvement allows people to absorb such important tacit knowledge about the solution and context directly as it emerges.

13.4 Design is teamwork

It is important for the Digital Design mindset to recognize that designing a good solution is a task that goes beyond Digital Design. It is a task that requires close cooperation between management, design, construction, and realization. Likewise, it takes diverse professionals to design a good solution, including:

- Data protection experts to achieve data protection and privacy
- Visual design experts to achieve elegant and aesthetic visual design

- Ergonomics and usability experts to achieve accessible, enjoyable, useful, and usable digital solutions
- Requirements engineering experts to identify stakeholders and to understand stakeholder needs
- Social science and anthropology experts to anticipate the impact of a digital solution
- Sustainability experts to assess and improve the sustainability of a digital solution
- Experts in design and construction to achieve an elegant and aesthetic digital solution on the perceivable and underlying levels

Finally, there remains one final important observation for the attitude in Digital Design. This handbook shows that Digital Design is a very diverse profession that requires various skills. We believe it is possible to understand the importance of and the relationship between all these skills. We further believe that it is possible to become a master in some of these skills.

However, becoming a master of the whole spectrum of skills is possible, but only for exceptional talents. For average people, such as we authors are, the following thought remains as a conclusion for the attitude of Digital Design that we wanted to promote in this handbook:

> Good Digital Design can be achieved only through transdisciplinary teamwork that can cover the diversity of Digital Design skills.

References

[Angr2020] Angrave, J.: The Journey Mapping Playbook: A Practical Guide to Preparing, Facilitating and Unlocking the Value of Customer Journey Mapping. De Gruyter, 2020.

[Alex2005] Alexander, I. F.: A Taxonomy of Stakeholders: Human Roles in System Development. International Journal of Technology and Human Interaction, Vol 1, 1, 2005, 23-59.

[BCDE2015] Becker, C., Chitchyan, R., Duboc, L., Easterbrook, S., Prenzenstadler, B., Seyff, N., Venters, C.: Sustainability Design and Software: The Karlskrona Manifesto. International Conference on Software Engineering (ICSE), 467-476, 2015.

[Bitk2017] Bitkom e. V.: Rollenideal Digital Design – Erfolgreiche Digitalisierung und Digitale Transformation erfordern ein Umdenken in der Softwareentwicklung, 2017, https://www.bitkom.org/Bitkom/Publikationen/Rollenideal-Digital-Design.html,
last visited September 2023. Available only in German.

[Bloo2018] Bloomberg, J.: Digitization, Digitalization, And Digital Transformation: Confuse Them At Your Peril, https://www.forbes.com/sites/jasonbloomberg/2018/04/29/digitization-digitalization-and-digital-transformation-confuse-them-at-your-peril/?sh=1fc351092f2c, last visited September 2023.

[Brow2009] Brown, T.: Change by Design. Harper Business, 2009.

[CPRE2022] IREB Certified Professional for Requirements Engineering - Advanced Level RE@Agile, Version 2.0.0, July 2022.

[CPUX2022] UXQB: CPUX-F Curriculum, Certified Professional for Usability and User Experience Foundation Level, Version 4.01, November 2022.

[Cros2006] Cross, N.: Designerly Ways of Knowing. Springer, 2006.

[Demi2000] Deming, W. E.: Out of the Crisis. Reprint. MIT Press, 2018.

[Dick2019] Dickel, S.: Prototyping Society. Available only in German. Bielefeld: transcript, 2019. Available only in German.

[Dors1997] Dorst, K.: Describing Design – A Comparison of Paradigms. Delft University Press, Delft, The Netherlands, 1997.

[Dors2003] Dorst, K.: Understanding Design: 150 Reflections on Being a Designer. BIS, 2003.

[ErMa2008] Erlhoff, M., Marshall, T. (Eds.): Design Dictionary: Perspectives on Design Terminology. Birkhäuser, 2008.

[Fros2020] Frost, B.: Atomic Design. https://atomicdesign.bradfrost.com. Last visited September 2023.

[Glas2006] Glass, R. L.: Software Creativity 2.0. developer.* books, 2006.

[Glin2020] Glinz, M.: A Glossary of Requirements Engineering Terminology, Version 2.0, October 2020. International Requirements Engineering Board (IREB).

[GLSB2022] Glinz, M., Van Loenhoud, H., Staal, S, and Bühne,S.: Handbook for the CPRE Foundation Level according to the IREB Standard Version 1.1.0, September 2022. CPRE by IREB, 2022. https://www.ireb.org/en/downloads/#cpre-foundation-level-handbook. Last visited January 2024.

[GSBF2023] Glinz, M., Seyff, N., Bühne, S., Franch, X., Lauenroth, K.: Towards a Modern Quality Framework. 2nd International Workshop on Requirements Engineering Frameworks, September, 2023.

[IIBA2022] IIBA: The Business Analysis Standard. Version 1.0. IIBA Publications, November 2022.

[ISO2011] ISO/IEC 25010: Software Product Quality. https://iso25000.com/index.php/en/iso-25000-standards/iso-25010, 2011. Last visited September 2023.

[ISO2018] ISO 9241-11: Ergonomics of Human-System Interaction – Part 11: Usability: Definitions and Concepts. 2018.

[ISO2019] ISO 9241-210:2019: Ergonomics of Human-System Interaction — Part 210: Human-Centred Design for Interactive Systems. ISO, 2019.

[ISTQB2023] ISTQB: Certified Tester Foundation Level Syllabus v4.0, International Software Testing Qualifications Board, 2023.

[Keir1998] Keirsey, D.: Please Understand Me II: Temperament, Character, Intelligence (1st ed.). Prometheus Nemesis Book Co. ISBN 1-885705-02-6.

[Kell2016] Kelly, K.: The Inevitable: Understanding the 12 Technological Forces that will shape our Future. Viking, 2016.

[KnZK2016] Knapp, J., Zeratsky, J., Kowith, B.: Sprint – How to solve Big Problems and Test new Ideas in five Days. Bantam Press, 2016.

[Laue2005] Lauesen, S.: User Interface Design. A Software Engineering Perspective. Addison Wesley, 2005.

[Laue2019] Lauenroth, K.: Understanding Digital as a Material with Form, Function and Quality. In Digital Design@Bauhaus, Bitkom, 2019, https://www.bitkom.org/sites/main/files/2019-09/20190910_sammelband-digital-designbauhaus.pdf. Available only in German., last visited January 2024.

[LBGH2018] Lauenroth, K., Bramsiepe, H., Gilbert, D., Hartwig, R., Lehn, K., Schubert, U., Trapp, M.: The Digital Design Manifesto, https://www.digitaldesign.org/en/downloads. Last visited January 2024.

[LiOg2011] Liedtka, J., Ogilivie, T.: Designing for Growth – a Design Thinking Toolkit for Managers. Columbia Business Scholl Publishing, 2011.

[MaLa2015] Margolis, E., Laurence, S. (Eds): The Conceptual Mind: New Directions in the Study of Concepts. MIT Press, 2015.

[McCo2014] McConnell, S.: Software Estimation: Demystifying the Black Art. Microsoft Press, 2014.

[McEl2017] McElroy, K.: Prototyping for Designers. O'Reilly, 2017.

[Meye2014] Meyer, B.: Agile: The Good, the Hype and the Ugly. Springer, 2014.

[MiKi1994] Milgram, P., Kishino, F.: A Taxonomy of Mixed Reality Visual Displays. IEICE Transactions on Information and Systems. Vol. 77, 1994.

[Newm2020] Newman, D.: The Process of Design Squiggle. thedesignsquiggle.com. Last visited September 2023.

[NeSt2014] Nelson, H. G., Stolterman, E.: The Design Way – Intentional Change in an Unpredictable World. 2nd edition, MIT Press, 2014.

[Nobl1996] Noblet, J. de: Industrial Design: Reflection of a Century. Flammarion 1996.

[OMG2017] OMG: Unified Modeling Language (UML). Version 2.5.1, 2017.

[OMG2014] OMG: Business Process Model and Notation (BPMN). Version 2.0.2, 2014.

[OPBS2014] Osterwalder, A., Pigneur, Y., Bernarda, G., Smith, A.: Value Proposition Design: How to Create Products and Services Customers Want. Wiley, 2014.

[OsPi2010] Osterwalder, A., Pigneur, Y.: Business Model Generation. Wiley, 2010.

[PoLR2013] Polaine, A., Løvlie, L., Reason, B.: Service Design: From Insight to Implementation. Rosenfeld Media, 2013.

[PrAd2006] Pruitt, J., Adlin, T.: The persona lifecycle. Keeping People in Mind Throughout the Product Design. Morgan Kaufmann Publisher, 2006.

[Ries2011] Ries, E.: The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses. Crown Books, 2011.

[RiWe1973] Rittel, H. W. J., Webber, M. M.: Dilemmas in a General Theory of Planning. In: Policy Sciences 4(2):155–169, 1973.

[Ross2019] Rossmann, J.: Think Like Amazon: 50 ½ Ideas to Become a Digital Leader. McGraw-Hill, 2019.

[Royce1970] Royce, W.: Managing the Development of Large Software Systems: Concepts and Techniques. In: Proceedings of IEEE WESCOM. IEEE Computer Society Press, Los Alamitos, 1970.

[Seng2006] Senge, P. M.: The Fifth Discipline: The art and practice of the learning organization: Second edition. Random House, 2006.

[ScSu2020] Schwaber, K., Sutherland, J.: The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game. November 2020.

[TuPA2021] Tueanrat, Y., Papagiannidis, S., Alamanos, E.: Going on a journey: A review of the customer journey literature. *Journal of Business Research, 125*, 336–353, 2021.

[VPGV2008] Van de Ven, A., Polley, D., Garud, R., Venkataraman, S.: The Innovation Journey. Oxford University Press, 2008.

[Wake2003] Wake, B.: INVEST in Good Stories and SMART Tasks: 2003, https://xp123.com/articles/invest-in-good-stories-and-smart-tasks/. Last visited September 2023.

[Wein1971] Weinberg, G. M.: The Psychology of Computer Programming. Van Nostrand Reinhold, 1971.

[WeJa2010] Weisbord, M., Janoff, S.: Future Search – Getting the whole System in the Room for Vision, Commitment, and Action. Berrett-Koehler Publishing, 2010.

List of Figures

List of Tables

The Big Picture