

2 System and Context Boundaries

The requirements for a system to be developed do not simply exist, they have to be elicited. The purpose of defining the system and context boundaries in requirements engineering is to identify the part of the environment that influences the requirements for the system to be developed.

2.1 System Context

In the development process, requirements engineering fulfils the task of identifying all those material and immaterial aspects that have a relationship to the system. In order to do that, it is anticipated what the system will be like once it becomes real. By doing so, those parts of the real world which will potentially influence the requirements of the system can be identified. To be able to specify the requirements for a system correctly and completely, it is necessary to identify the relationships between individual material and immaterial aspects as precisely as possible. The part of reality that is relevant for the requirements of a system is called the system context.

*Anticipate the system
in operation*

Definition 2-1: System Context

The system context is the part of the system environment that is relevant for the definition as well as the understanding of the requirements of a system to be developed.

Among others, the following possible aspects of reality influence the context of a system:

*Context aspects in the
system context*

- People (stakeholders or groups of stakeholders)
- Systems in operation (other technical systems or hardware)
- Processes (technical or physical processes, business processes)
- Events (technical or physical)
- Documents (e.g., laws, standards, system documentation)

Consequence of erroneous or incomplete context consideration

If the system context is incorrectly or incompletely considered during requirements engineering, it may result in incomplete or erroneous requirements. This leads to the system operating on the basis of incomplete or erroneous requirements, which is often the reason for system failure during operation. Such errors often remain undetected during the validation procedures, which determine if the system meets the specified requirements, and occur only during operation, sometimes entailing catastrophic consequences.

System context and requirement context

The origin of the system's requirements lies within the context of the system to be developed. For example, stakeholders, pertinent standards, and legal guidelines demand particular functional properties that the system to be developed must possess at its interfaces. A requirement is therefore defined for a specific context and can only be interpreted correctly in regard to this specific context. The better the context of a requirement is understood (e.g., why is the technical system "X" in the system context the origin of some requirement), the lower the likelihood of incorrect interpretation of the requirement. Therefore, a purpose-driven documentation of the system context or information about the system context is of particular importance.

2.2 Defining System and Context Boundaries

It is within the responsibility of the requirements engineer to define the system context properly. In order to do so, it is necessary to separate the system context from the system to be developed as well as from the parts of reality that are irrelevant for the system (see figure 2-1):

- *Defining the system boundary:* When defining the system boundary, a decision has to be made: Which aspects pertain to the system to be developed and which aspects belong in the system context?
- *Defining the context boundary:* When defining the context boundary, the question to be answered is: Which aspects pertain to the system context (i.e., have a relation to the system to be developed) and which aspects are part of the irrelevant environment?

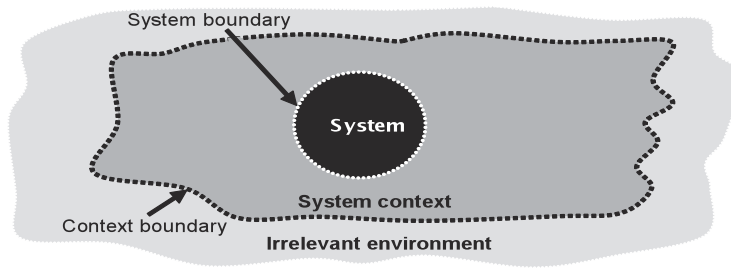


Figure 2-1 System and context boundary of a system

Thus, system and context boundaries define the system context. The system context comprises all aspects that are relevant with regard to the requirements for the system to be developed. These aspects cannot be altered or modified by the system development process.

System and context boundaries define the system context.

2.2.1 Defining the System Boundary

The system boundary separates the object of concern (i.e., the system) from its environment. When the system boundary is defined, the scope of the development (i.e., the aspects that are covered by the system to be developed) as well as the aspects that are not part of the system are determined. We therefore define the system boundary as follows:

Definition 2-2: *System Boundary*

The system boundary separates the system to be developed from its environment; i.e., it separates the part of the reality that can be modified or altered by the development process from aspects of the environment that cannot be changed or modified by the development process.

All aspects that are within the system boundary can thus be altered during system development. For instance, an existing system that consists of hardware and software components and is supposed to be replaced by the new system can be within the system boundary. Aspects within the system context can be business processes, technical processes, people and roles, organizational structures, and components of the IT infrastructure. Figure 2-2 schematically shows the system context of a system. The system context consists of other systems, groups of stakeholders that in some way use the interfaces of the system to be developed, and additional requirements sources and their interrelations.

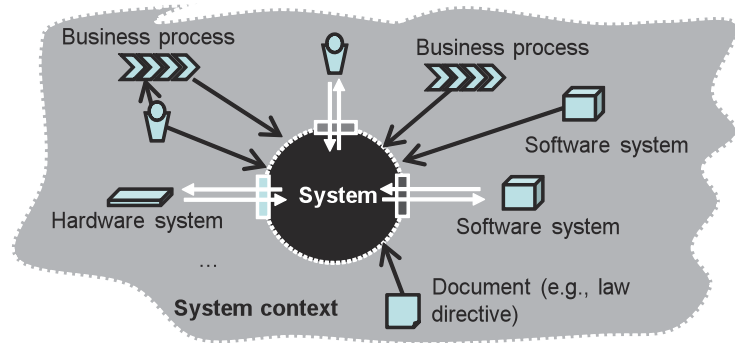


Figure 2-2 Types of aspects within the system context

Sources and sinks as the starting point

Among other things, sources and sinks (see, e.g., [DeMarco 1978]) can be used to identify the interfaces the system has with its environment. Sources provide inputs for the system. Sinks receive outputs from the system. Possible sources and sinks of a system are as follows:

- (Groups of) stakeholders
- Existing systems (both technical and nontechnical systems)

Interfaces: interaction between system and environment

Sources and sinks interact with the system to be developed via system interfaces. Using these interfaces, the system provides its functionality to the environment, monitors the environment, influences parameters of the environment, and controls operations of the environment. Depending on the type of the respective source or sink, the system needs different interface types (e.g., human–machine interface, hardware interface, or software interface). The interface type in turn may also impose specific constraints or additional sources of requirements on the system to be developed.

Gray zone between system and system context

Frequently, the system boundary is not precisely defined until the end of the requirements engineering process. Before that, some or several interfaces as well as desired functions and qualities of the system to be developed are only partially known or not known at all. We refer to this initially vague separation of the system and its context as the gray zone between the system and the context (see figure 2-3). At the beginning of the requirements engineering process, it may, for example, not be clear whether the system should implement a certain function (e.g., “pay by credit card”) or whether there is another system in the system context providing such a function that should be used (e.g., “payment processing”).

The system boundary may not only shift within the gray zone (① in figure 2-3) but also the gray zone itself may shift during the requirements engineering process (② in figure 2-3). This kind of shifting is caused by the fact that aspects, pertaining at first to the system context, now will be modified during system development. Such a situation occurs during requirements engineering, for example, if it is not clear in the system context whether certain activities of a business process should be implemented or supported by the system to be developed or not. In this situation, it is not clear which aspects belong to the system and can thus be changed or modified and which aspects belong to the system context. This causes a corresponding shift of the gray zone between system and system context (see figure 2-3).

Adjusting the gray zone

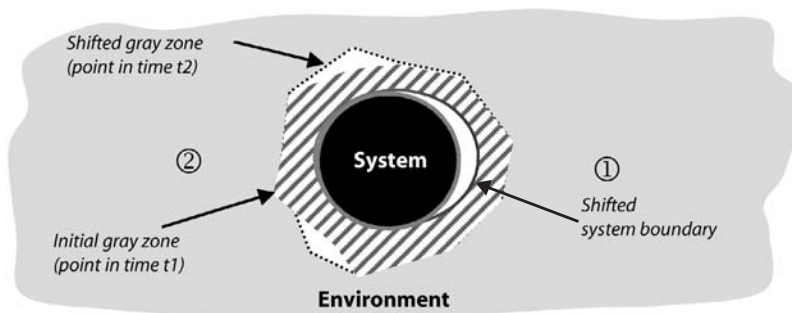


Figure 2-3 Gray zone of the system boundary

The gray zone shifts, for instance, when interfaces are attributed to the system boundary and the gray zone is extended to comprise aspects of the environment that concern these interfaces.

2.2.2 Defining the Context Boundary

The context boundary distinguishes between context aspects, i.e., those aspects of the environment that need to be taken into account during requirements engineering (e.g., as requirements sources) and those aspects that are irrelevant for the system. The context boundary can be defined as follows:

Definition 2-3: Context Boundary

The context boundary separates the relevant part of the environment of a system to be developed from the irrelevant part, i.e., the part that does not influence the system to be developed and, thus, does not have to be considered during requirements engineering.

Concretion and shift of the context boundary

At the beginning of the requirements engineering process, frequently only part of the environment as well as single specific relationships between the environment and the system to be developed are known. In the course of requirements engineering, it is necessary to concretize the boundary between system context and irrelevant environment by analyzing relevant aspects within the environment with regard to their relationships to the system. Besides the system boundary, the context boundary typically also shifts during requirements engineering. For instance, it may be possible that a law directive that was considered to be relevant for the system to be developed no longer impacts the system or is no longer considered relevant. The system context is therefore reduced (① in figure 2-4). If a new law directive is identified that influences the system, the system context is extended accordingly (② in figure 2-4).

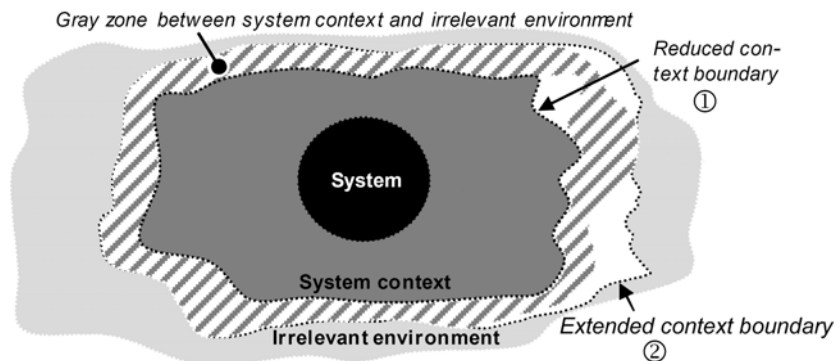


Figure 2-4 Gray zone between system context and irrelevant environment

Gray zone between system context and irrelevant environment

Since the context boundary separates the system context from those parts of reality that are irrelevant to the system, a complete and precise definition of the context boundary for complex systems is virtually impossible. In addition, it may not be possible to clarify for single aspects of the environment whether they influence the system to be developed or are influenced

by it or not. These two observations are the reason for the existence of a gray zone with regard to the context boundary (see figure 2-4).

This gray zone therefore comprises identified aspects of the environment for which it is unclear whether they have a relation to the system or not. In contrast to the gray zone between the system and the system context that must be resolved in the course of requirements engineering, it is not necessary to resolve the gray zone between the system context and the irrelevant environment entirely.

Resolving and shifting of the gray zone

2.3 Documenting the System Context

In order to document the system context (especially the system and context boundaries), “use case” diagrams [Jacobson et al. 1992] (see sections 4.2.3 and 6.3.1) or “data flow” diagrams [DeMarco 1978] (see section 6.6.1) are often used. When the context is modeled with data flow diagrams, sources and sinks in the environment of the system that represent the source or destination of data flows (or flows of material, energy, money, etc.) are modeled. In use case diagrams, actors (such as people or other systems) in the system environment and their usage relationships to the system are modeled. To model the system context, UML class diagrams [OMG 2007] (see section 6.5.2) may also be used. In order to document the system context of a system as thoroughly as possible, typically several documentation forms are used.

2.4 Summary

The system context is the part of the reality that influences the system to be developed and thus also influences the requirements for the system. In order to be able to elicit the requirements for the system to be developed, it is necessary to define the boundary of the system to the system context and the boundary of the system context to the irrelevant environment first. When the system boundaries are defined, the scope of the system is determined. The scope comprises those aspects that can be changed and designed during system development. At the same time, it is also defined which aspects belong to the environment and thus cannot be altered during development and may provide constraints for the system to be developed.

The context boundary separates the part of the environment that influences the requirements for the system to be developed from that part that does not influence the requirements. Typical aspects within the system context are stakeholders (e.g., the users of the system) and documents (e.g., standards that have to be considered) as well as other systems that, for instance, interact with the system to be developed. Defining the system and context boundaries successfully is the foundation for a systematic elicitation of requirements for the system to be developed.