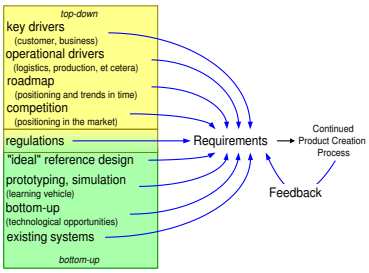
Requirements Capturing by the System Architect

-



Gerrit Muller
Embedded Systems Institute
Den Dolech 2 (Laplace Building 0.10) P.O. Box 513, 5600 MB Eindhoven The Netherlands
gerrit.muller@embeddedsystems.nl

Abstract

The basis of a good system architecture is the availability and understanding of the requirements. This presentation shows how a system architect can capture the requirements and how to use these requirements in the context of the product creation process.
The notion of ”business key drivers” is introduced and a method is described to link these key drivers to the product specification.

version: 1.3 status: concept 5th September 2008

1 Introduction

The basis of a good system architecture is the availability and understanding of the requirements. This article describes how a system architect can capture the requirements and how to use these requirements in the context of the product creation process. This article builds upon the architecture process positioning as described in [4].

This article is part of the deliverables of the Gaudí project [3], which will describe other processes and methods mentioned in this article, like roadmapping.

2 Definition of Requirements

The term requirement is quite heavily overloaded in Product Creation context. One major interpretation is :

The requirements describe the needs of the customer.

In this article this meaning will be captured in the term *Customer Requirements*.

Another major interpretation is:

The requirements describe the characteristics of the final resulting product.

This article will use the phrase *Product Specification* when this interpretation is intended.

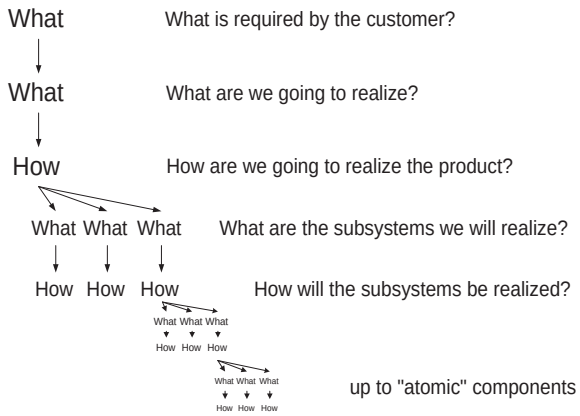


Figure 1: The flow of requirements

In the system engineering world the term *Requirement Management* or *Requirement Engineering* is being used. This term goes much farther than the two previous interpretations. The requirement management process or requirement engineering process deals with the propagation of the requirements in the product specification towards the requirements of the subsystems defined by the first design decomposition finally towards the requirements of the atomic components. In fact the

definition of the Product Specification is recursively applied for every decomposition level. On top of that the management process manages the relationships of the different aggregation levels.

Figure 1 show the flow of the requirements starting at the customer level.

A consensus seems to be present about the fact that requirements deal with the *what* and do not describe the *how*.

Besides the customer an important source of requirements is the producing company itself, the needs of the company itself are described in this article as *Operational Requirements*.

This article will address all the interpretations of requirements given above.

3 System as a black box

One of the main characteristics of requirements is that they describe what has to be achieved and not it how this has to be achieved. In other words system requirements describe the system as *black box*. Figure 2 provides a starting point to describe system requirements.

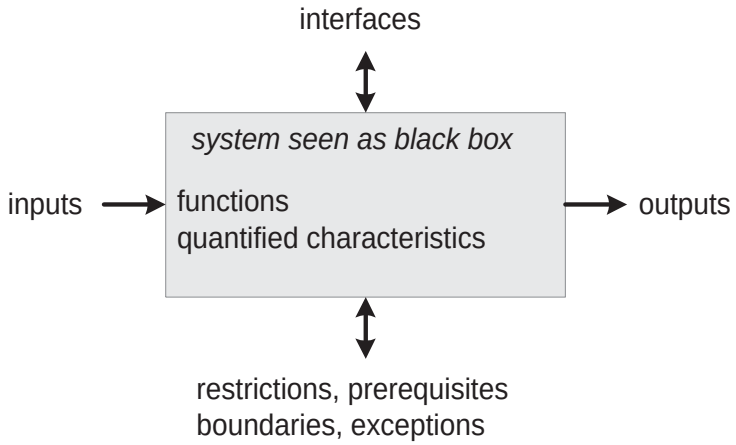


Figure 2: System as a Black Box

The system is seen as black box. What goes into the box, what comes out and what function has to be performed on the inputs to get the outputs. Note that the function tells something about the black box, but without prescribing how to realize. To get the requirements more specific all interfaces are identified; Human interfaces as well as interfaces to other systems. Specifying only the functions is insufficient. The specification must also describe the desired quantified characteristics, such as how fast, how much, how large, how costly, et cetera.

Another class of information are prerequisites and constraints enforced on the system. Further scoping can be done by stating boundaries and desired behavior in case of exceptions.

4 Stakeholders

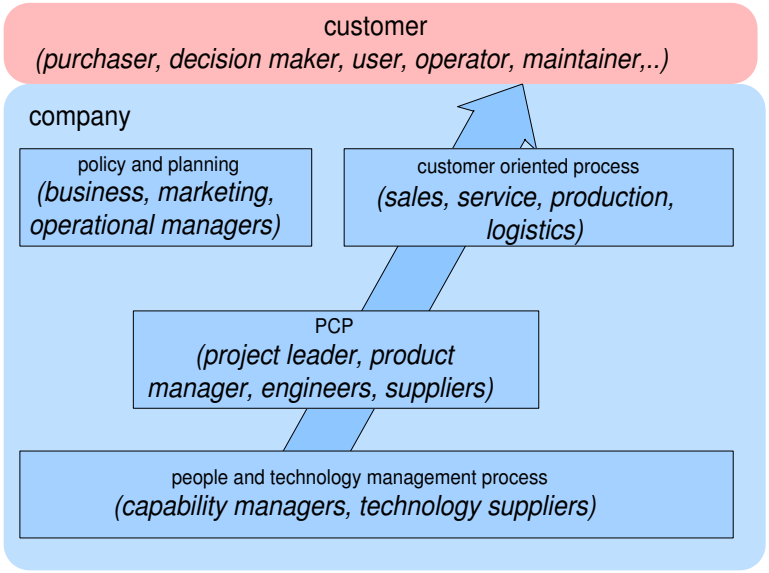


Figure 3: A simplified process decomposition of the business. The stakeholders of the requirements are beside the customer self, mainly active in the customer oriented process and the product creation process.

A simplified process model is shown in figure 3. The stakeholders of the requirements are of course the customers, but also a number of representatives in the customer oriented process and most people active in the Product Creation Process.

For convenience the word customer is used, although the customer can be a business or even a group of businesses. A good understanding of the customer business is required in order to identify the customer-stakeholders.

5 Requirements for Requirements

Standards like ISO 9000 or methods like CMM prescribe the requirements for the requirement management process. These requirements are:

- Specific (1)

- Unambiguous (2)
- Verifiable (3)
- Quantifiable (4)
- Measurable (5)
- Complete (6)
- Traceable (7)

Unfortunately these requirements are always biased towards the formal side. A process which fulfils these requirements is from technical point of view sound and robust. However an important aspect which is forgotten quite often is that product creation is a human activity, with all their human capabilities and constraints. The Human point of view adds a number of requirements, which are required for **every** stakeholder:

- Accessible (8)
- Understandable (9)
- Low threshold (10)

These requirements, which are imposed by the human element, can be conflicting with the requirements which are prescribed by the management process. Many problems can be traced back to violation of the human imposed requirements. For instance a customer requirement which is described so abstract that no real customer can understand it anymore is a severe risk, because early validation is impossible.

6 Viewpoints on Requirements

Many complementary viewpoints are required to collect the requirements. Figure 4 shows a useful number of viewpoints when collecting requirements.

The **key-driver** viewpoint and the **operational** viewpoint are the viewpoints of the stakeholders which are "consuming" or "using" the output of the product creation process. These viewpoints represent the "demanding side".

The **roadmap** and the **competition** viewpoint are viewpoints to position the requirements in time and in the market. Those viewpoints are important because they emphasize the fact that a product is never made in isolation, but in a rather dynamic and evolving world.

Regulations result in requirements both top-down, as well as bottom-up. A top down example are labor regulations that can have impact on product functionality

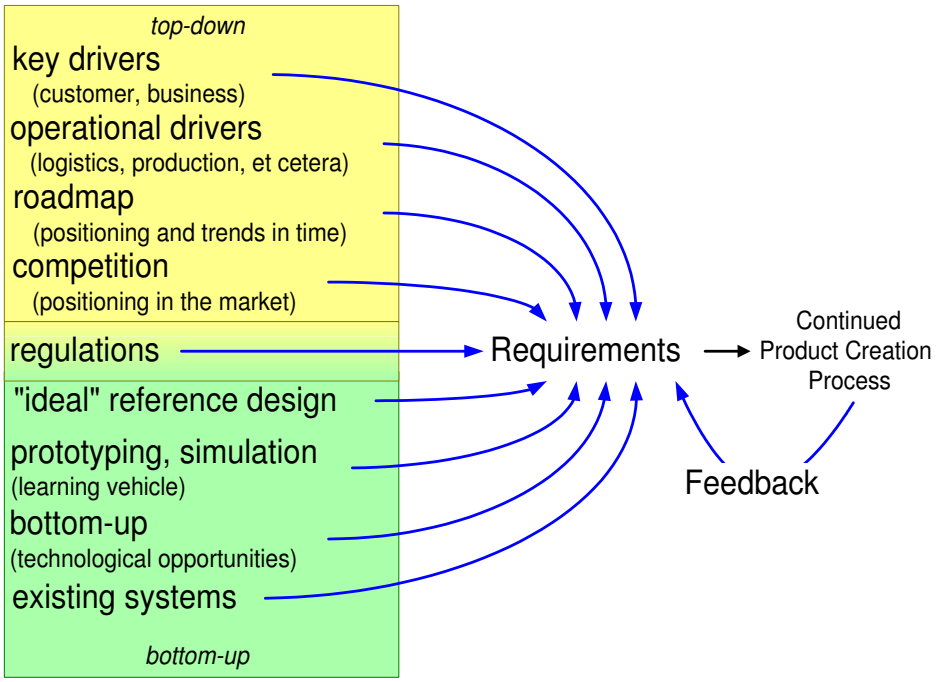


Figure 4: Complementary Viewpoints to collect requirements

and performance. A bottom up example are materials regulations, for instance don't use lead, that may strongly influence design options.

The **"ideal" reference design** is the challenge for the architect. What is in his vision the perfect solution? From this perfect solution the implicit requirements can be reconstructed and added to the rest of the requirements.

Prototyping or simulations are an important means in communication with customers. This "pro-active feedback" is a very effective filter for nice but impractical features at the one hand and it often uncovers many new requirements, which do not appear with a pure paper approach.

The **bottom up** viewpoint is the viewpoint which takes the technology as the starting point. This viewpoint sometimes triggers new opportunities which are overlooked by the other viewpoints due to an implicit bias by today's technology.

The **existing system** is one of the most important sources of requirements. In fact it contains the accumulated wisdom of years of practical application. Especially the large amount of small but practical requirements can be extracted from existing systems.

The requirement specification is a dynamic entity, because the world is dynamic: the users change, the competition changes, the technology changes, the company itself changes. For that reason the **Continuation of the Product Creation Process** will generate input for the requirements as well. In fact nearly all viewpoints are

present and relevant during the entire Product Creation Process.

7 Reference Architecture and Key Drivers

A system architect must look at the product from multiple complementary viewpoints. Figure 5 shows 5 useful views for a reference architecture.

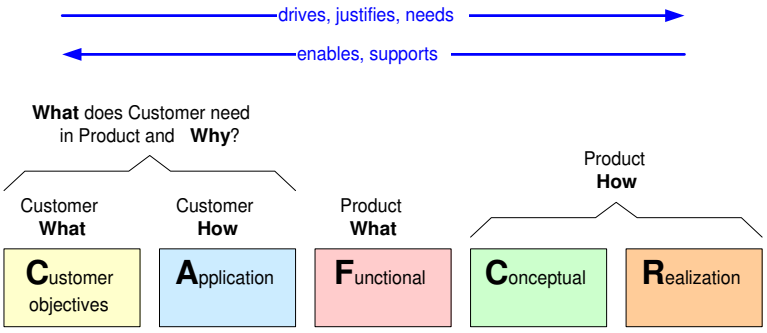


Figure 5: A Reference Architecture views the architecture from 5 viewpoints

The business architecture is the architecture of the business of the customer, in relation with the product. Typically it will describe the flow of information or goods, the business processes and the related roles.

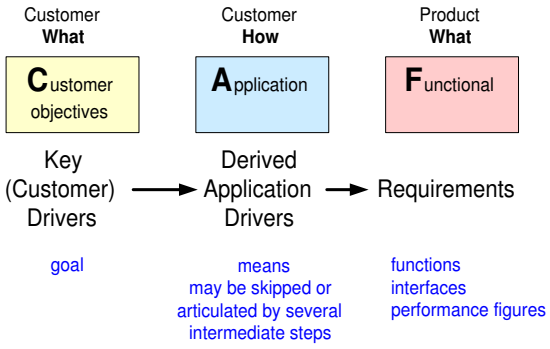


Figure 6: The mapping of Key Drivers via derived application drivers on requirements

A very powerful means to capture requirements is to describe the essence of the business in terms of *Key Drivers*. These drivers must be recognized and understood by the customer, which means that these drivers should be expressed in the language of the customer. A maximum of 5 Key Drivers is recommended to maintain focus on the essence, the name is on purpose **Key** driver. The key drivers are one aspect of the business architecture. Figure 7 shows a method to define key

drivers. Figure 8 shows some recommendations with respect to the definition of key drivers.

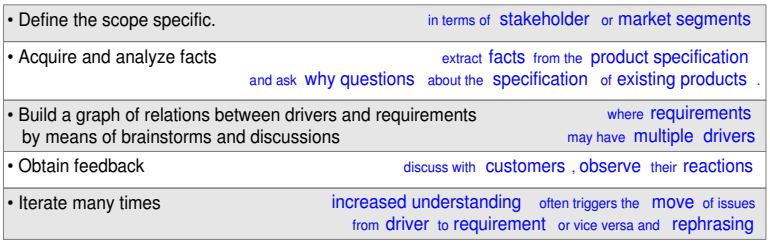


Figure 7: Method to define key drivers

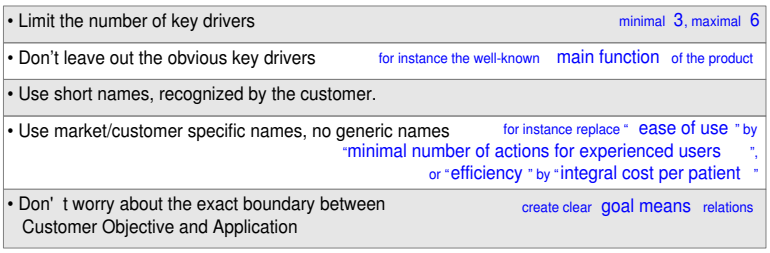


Figure 8: Recommendations when defining key drivers

Key drivers can be mapped on derived application drivers. Which application activities are done to enable the key driver? The derived application drivers must also be expressed in customer language. The explicit description of application drivers will also ease the job of modelling the application domain.

The derived application drivers are implemented or supported by features or functions of the product. This means that the derived application drivers can be translated into customer requirements of the product.

From point of view of requirements engineering the customer requirements are used as input to produce a product specification, which controls the entire product creation process. The design of the system will result in a technical architecture, with amongst others a decomposition in subsystems and function allocation. The technical architecture is finally mapped onto an implementation. The relation between requirements at the functional architecture level, the technical architecture level and the implementation is managed by the requirements management process.

Approaching the requirements definition in this way enables the architect to understand a technical feature in relation with the key driver from the customer business. Any feature that cannot be related back to a key driver is suspect: either it should not be there or some requirement or driver is missing.

8 Example Motorway Management

Figure 9 shows an example of the requirements analysis of a motorway management system. The keydrivers of a motorway management owner are:

- Safety
- Effective Flow
- Smooth Operation
- Environment

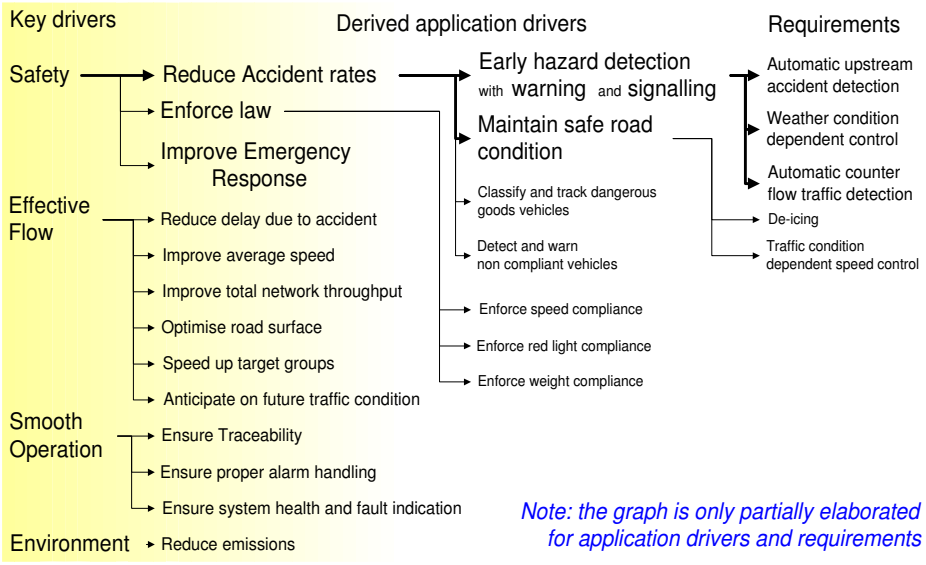


Figure 9: The key drivers, derived application drivers and requirements of a Motorway Management System

To realize these key drivers the owner applies a number of application processes. This leads to the derived application drivers. For instance to realize safety it is important to prevent accidents and to have immediate response by emergency departments in case of accidents.

9 Requirements Value and Selection

The set of customer requirements and operational requirements is often larger than can be realized in the first release of a product. A selection step is required to generate a product specification with the customer and operational requirements as input. Figure 10

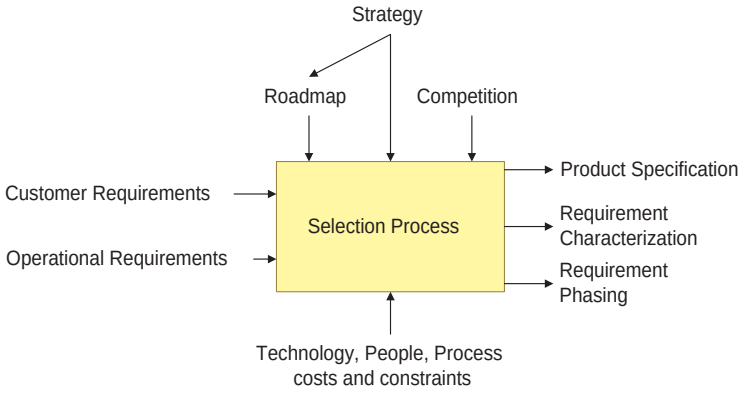


Figure 10: The selection process produces a product specification and to prevent repetition of discussion a phasing and characterization of requirements

The selection process is primarily controlled by the strategy of the company, which determines market, geography, timing and investments. The roadmap, which is in itself based on the strategy, is giving context to the selection process for an individual product. The reality of the competitive market is the last influencing factor on the selection.

The selection will be based on facts and estimates from the technology, people and process world, which will often constrain the possibilities.

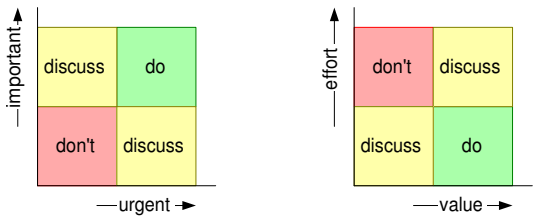


Figure 11: Simple methods for a first selection

The amount of requirements sometimes asks for a first selection step, which determine the "obvious". For some requirements it is immediately obvious that they have to be done anyway, while other requirements can be delayed without any problem. Figure 11 shows a number of qualitative characterizations of requirements, visualized in a two-dimensional matrix. For every quadrant in the matrix a conclusion is given, a requirement must be done, not be done or must be discussed further.

This simple qualitative game can for instance be done with the following criteria:

- importance versus urgency

- customer value versus effort
- must have

In the final selection step a more detailed analysis step is preferable, because this improves the understanding of the requirements and results in a less changes during the development.

A possible way to do this more detailed analysis is to "quantify" the characteristics for every requirement for the most business relevant aspects, for instance:

- Value for the customer
- Selling value (How much is the customer willing to pay?)
- Level of differentiation w.r.t. the competition
- Impact on the market share
- Impact on the profit margin

These quantifications can be given for the immediate future, but also for the somewhat remote future. In that way insight is obtained in the trend, while this information is also very useful for a discussion on the timing of the different requirements. In [1] a much more elaborated method for requirement evaluation and selection is described.

The output of the requirement characterization and the proposed phasing is input for the next update cycle of the roadmap. Note that some companies use the word roadmap for the phasing of the requirements, while this article uses a roadmap in a much broader sense, see the article [2]

10 Acknowledgements

The platform project within Philips Projects provided a clear analysis of amongst others a motorway management process. Wil Hoogenstraaten contributed significantly in the creation of this requirements analysis and sharpened the model from key driver towards features and functions.

Stimulating discussions with Henk Obbink and Jürgen Müller helped to shape this article.

Shakil Ahmed added the regulations viewpoint.

References

[1] Jean-Marc DeBaud and Klaus Schmid. A systematic approach to derive the scope of software product lines. In *21st international Conference on Software Engineering; Preparing for the Software Century*, pages 34–47. ICSE, 1999.

[2] Gerrit Muller. Roadmapping. http://www.gaudisite.nl/RoadmappingPaper.pdf, 1999.

[3] Gerrit Muller. The system architecture homepage. http://www.gaudisite.nl/index.html, 1999.

[4] Gerrit Muller. The system architecture process. http://www.gaudisite.nl/SystemArchitectureProcessPaper.pdf, 2000.

History

Version: 1.3, date: August 28, 2008 changed by: Gerrit Muller

- added system as a black box

Version: 1.2, date: April 20, 2005 changed by: Gerrit Muller

- added regulations viewpoint

Version: 1.1, date: February 17, 2005 changed by: Gerrit Muller

- updated key driver figures
- improved readability of some figures

Version: 1.0, date: May 31 2002 changed by: Gerrit Muller

- added abstract
- increased readability of some figures
- modified stakeholder figure
- updated keydriver figure
- modified Motorway management example

Version: 0.2, date: June 22 2001 changed by: Gerrit Muller

- annotated the figure with the reference architecture
- added a list of recommendations with respect to the definition of keydrivers

Version: 0.1, date: February 16 2001 changed by: Gerrit Muller

- new layout

Version: 0, date: October 25 1999 changed by: Gerrit Muller

- Created, no changelog yet