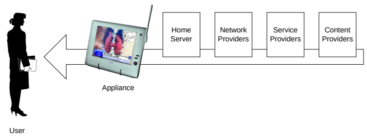
From the soft and fuzzy context to SMART engineering

-



Gerrit Muller
University of South-Eastern Norway-NISE
Hasbergsvei 36 P.O. Box 235, NO-3603 Kongsberg Norway
gaudisite@gmail.com

Abstract

The customer needs are often ill-defined or fuzzy. The need for specific and verifiable requirements is described, but it also shown that less SMART descriptions have complementary value.
A *Mobile Display Appliance* and *Mediascreen* are used to illustrate the translation of user experience (fuzzy) into (SMART) device requirements.

version: 1.4 status: finished September 9, 2018

1 Introduction

Engineering education and process improvement actions always stress the importance of "SMART"ness of requirements. Table 1 shows the meaning of SMART. The original use of this acronym was by George T. Doran, in an article about management goals and objectives [1]. Today the acronym is mostly used to stress the **specificity** and **measurability**, the "ART" part is used in many variations as shown in this list.

- Specific
- Measurable
- Assignable (Achievable, Attainable, Action oriented, Acceptable, Agreed-upon, Accountable)
- Realistic (Relevant, Result-Oriented)
- Time-related (Timely, Time-bound, Tangible, Traceable)

Table 1: *The meaning of SMART*

It is a sound advice to write product specifications with the SMART acronym in mind, every violation is a potential problem. However it is also important to try to capture and communicate the understanding on which these smartened specifications are based. This understanding often deals with much more fuzzy issues.

By following one example it is shown which fuzzy inputs are provided in the beginning, what the different stakeholder needs are and how these inputs can be transformed into more SMART specifications.

2 Case description

Figure 1 shows the type of products used as an example: a *Mobile Display Appliance* and *Mediascreen*. The function of these products can vary from portable television to home control to web browser et cetera.

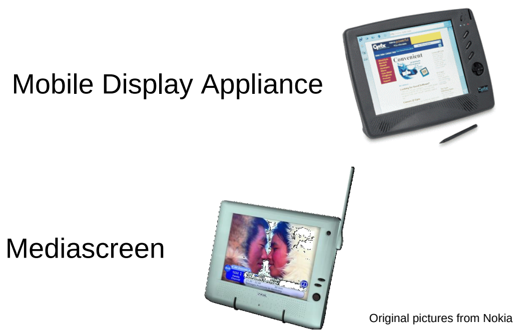


Figure 1: What are the requirements for these products?

Characteristic of these kind of products is that the functionality is typical a function of the appliance itself plus functions and content provided by the context. Figure 2 shows the total chain of systems which can be involved in the functionality.

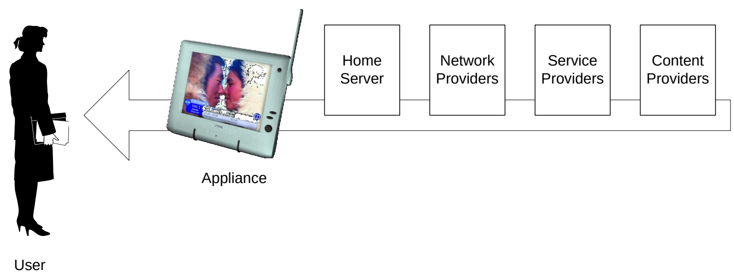


Figure 2: User access point to long food-chain

The architect of this appliance will receive many fuzzy expectations as inputs and for some parts he will get SMART descriptions. Figure 3 shows the fuzzy input as clouds and the SMART input as rectangles. Clearly a significant amount of fuzzy input is provided.

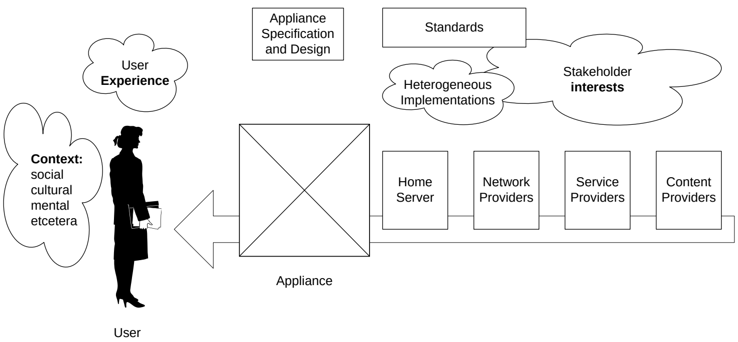


Figure 3: ”Fuzzy expectations” and ”SMART descriptions”

3 Why SMART?

The supply chain stakeholders of the design process are shown in figure 4. The product Creation Process takes care of the decomposition of the system in subsystems and components as well as the integration and test of the system. Later during manufacturing the components and subsystems are ordered from suppliers and assembled into a system. Finally the sales channels sell the systems and deliver the systems to the customers.

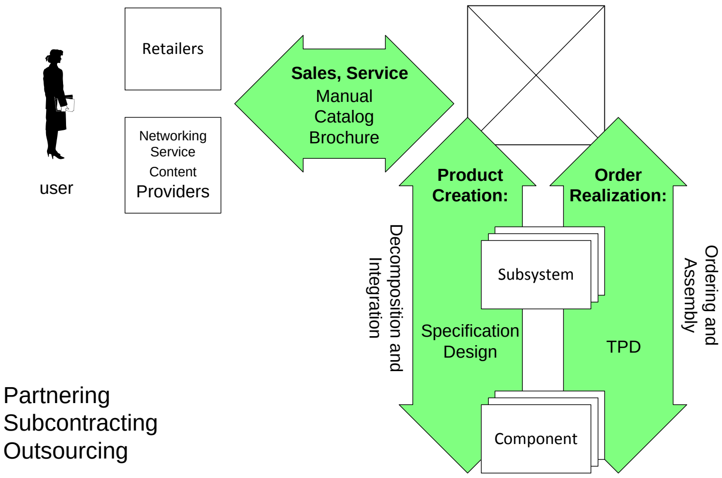


Figure 4: Supply Chain Stakeholders

All the stakeholders involved in this supply chain need specific and verifiable data to order, assemble, test and sell the product. (Did you ever try to tell a sales manager: ”don’t worry the product will be fast, will have a nice image quality and it will be very fashionable”, without any further hard facts?)

Figure 5 shows the problem statement by visualizing all the fuzzy needs at the one hand and the SMART facts at the other hand.

Figure 6 sharpens the problem statement by showing the fuzzy elements playing mostly in a creative world (imagination) and the formalizations often used to make things work in a supply chain environment.

One very specific stakeholder is the supplier. Often outsourcing or purchasing processes are highly formalized, to prevent problems. Figure 7 shows this relationship, which often causes redundant specifications at the interface (one from the integrator point of view and one from the supplier point of view). The final formalization is laid down in a contract.

To make this relationship work it is important that the integrator has know-how and understanding of the supplier and vice versa, as shown in figure 8.

The decomposition and integration of the system requires SMART data at

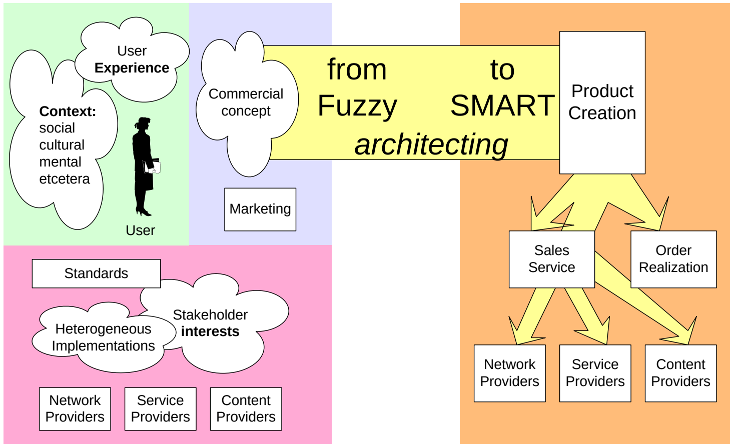


Figure 5: Problem Statement

all aggregation levels, both for product creation as well as for the supply chain. Figure 9 shows the functions in both processes, where SMART data are important.

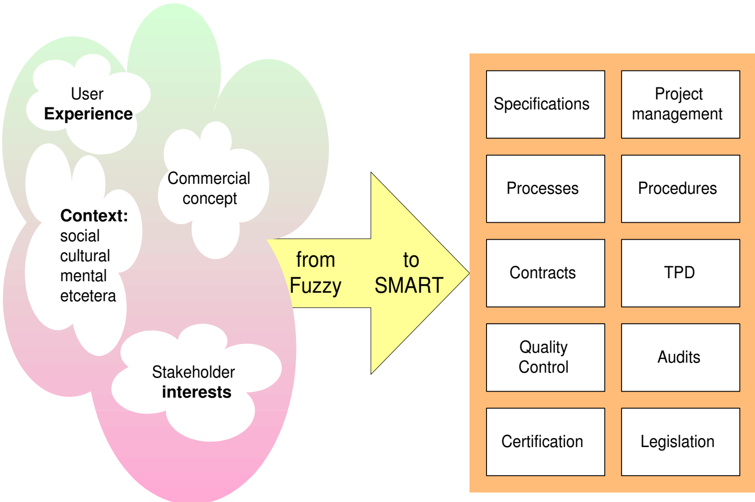


Figure 6: Problem (2): From Imagination to Formalization

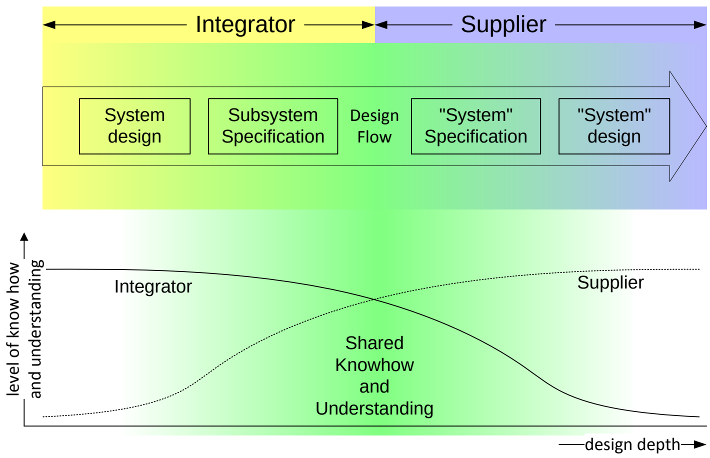


Figure 7: Theory: Subcontractors require SMART relation

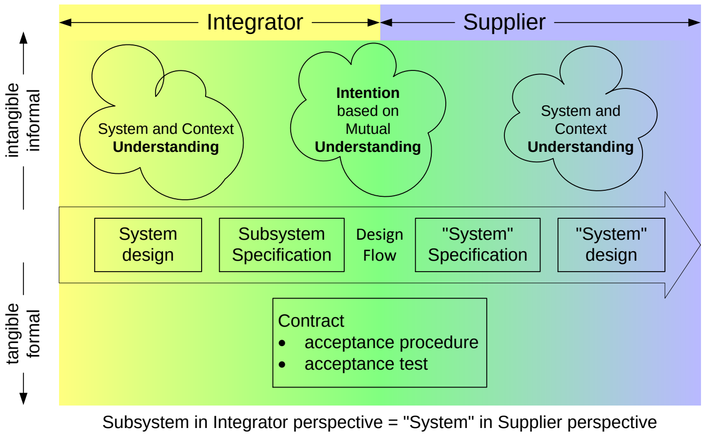


Figure 8: Critical Success Factor: Mutual understanding

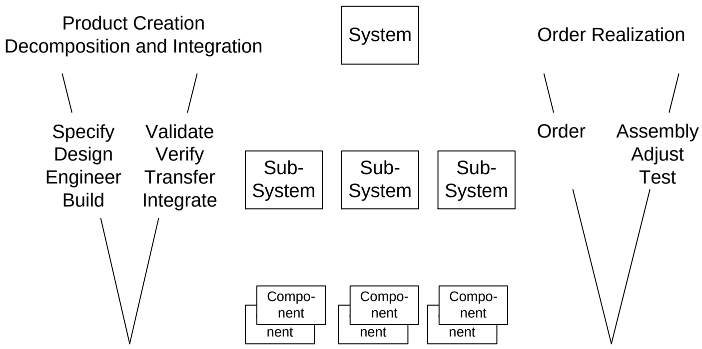


Figure 9: Views on Aggregation; Why SMART is needed

4 Examples of smartening fuzzy requirements

The user or consumer needs are shown in figure 10. As shown in this figure none of these requirements is specific nor measurable et cetera. However the qualified needs do provide insight in the motivation of users and in that way are useful to help others to obtain a better understanding.

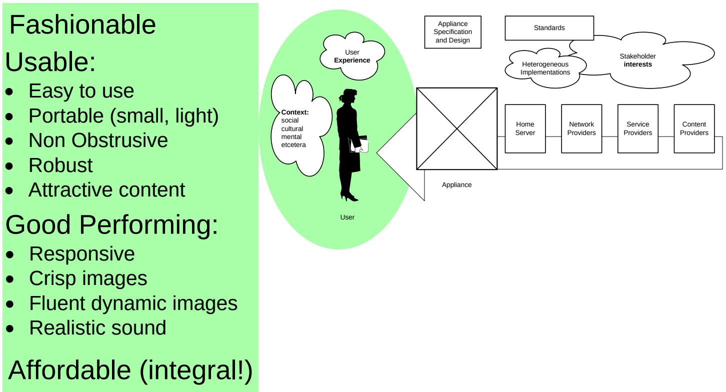


Figure 10: The "Fuzzy" needs of the User

Figure 11 shows in the same way the fuzzy needs of the providers and the retailers, which are also stakeholders of these products.

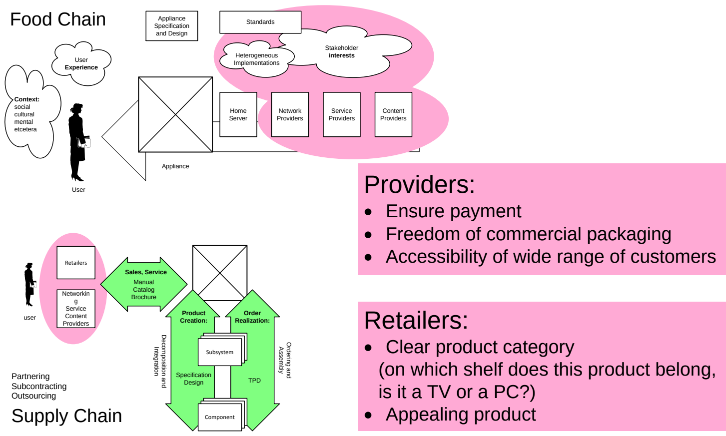


Figure 11: The "Fuzzy" needs of the Provider

The world of the designers is much less fuzzy. Systems are decomposed and interfaces are defined in SMART terms. Figure 12 shows the decomposition for this kind of product.

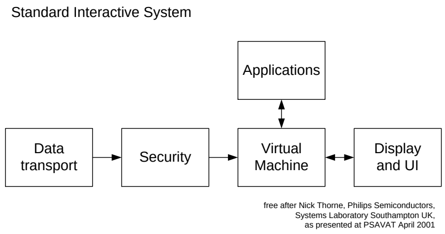


Figure 12: The "SMART" world of the Design

In such a decomposition many specification items can be defined. Figure 13 shows the decomposition annotated with specification items. Also a typical flow is shown for some user interaction: some request enters via the user interface, flows through al the blocks to the relevant service, and the answer travels in the opposite way. As an example we will have a look at the performance in terms of the response time.

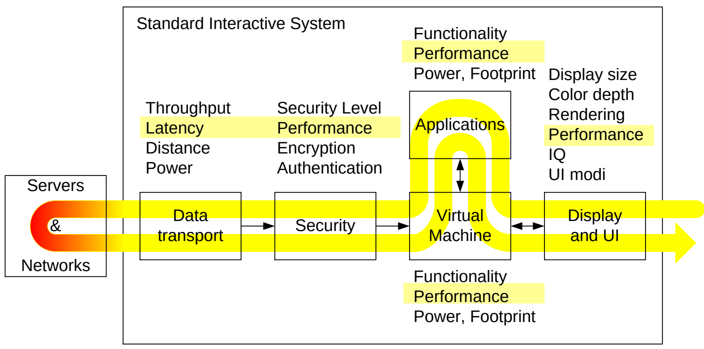


Figure 13: Specifiable characteristics

All functions in the chain contribute to the response time, figure 14 shows an (academic) budget for this response time. Note that the top part of the budget is well defined and in control of the appliance designer, however the lower levels of the budget are assumptions made by the appliance designer about the context. The wider the context becomes the more uncertainty will be present in the numbers.

The designer of the appliance need to make assumptions about the context in order to make a good design. These assumptions will be based on a model of the context. Such a model can be calibrated by measuring the context for as far as it exists already. However the designer should stay aware (as with all of his models) that this model is a tremendous simplification of reality. Reality is infinitely complex due to the possible variations in the food chain and the dynamics (changes over time).

times in milliseconds	Message Latency	Response Time
Appliance	40	100
Data transport	10	20
Security	10	20
Virtual Machine	10	20
Application	10	30
Graphics and UI	0	10
Home Network	20	50
Home Server	10	30
Network contention	10	20
Provider Infrastructure	50	160
Last-Mile network	10	20
Backbone network	20	40
Service server	10	50
Content server	10	50
Total	110	310
User need		200

All numbers are imaginary and for illustration purposes only

Figure 14: Response Time: Latency Budget

One of the important characteristics for the user of the appliance is the response time. As shown in figure 15 the model indicates good response times for functionality which stays within the appliance, but poor response times for functions which need interaction with far away servers. This insight will influence a lot of specification and design decisions. For all functions which require true interactive responses (i.e. less than 200 ms), some local solution is required, maybe supported by all kind of clever tricks such as look ahead, caching et cetera.

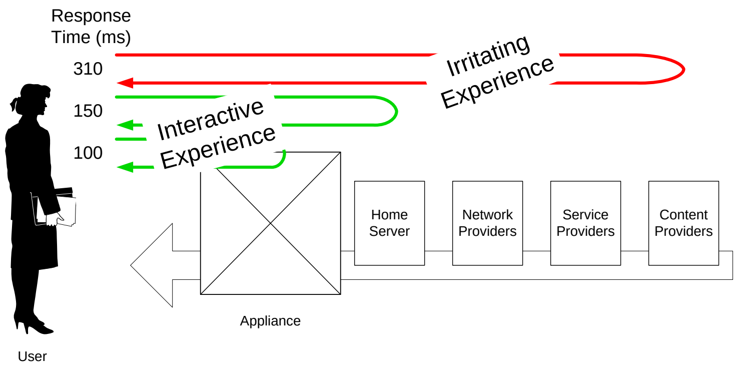


Figure 15: Interaction or Irritation?

A different area of fuzzy needs is image quality. The user needs good (sharp, bright, smooth moving) image quality. The verifiable image quality is often based on synthetic images, which enable verification for all technical parameters. However a perfect technical image quality does not mean a satisfied user...

Some of the image quality aspects are even more fuzzy, for instance when perception is taken into account (color blind people!), or worse when taste comes into play (this image is too sharp, while someone else finds it too smooth).

Figure 16 visualizes these different types of needs.

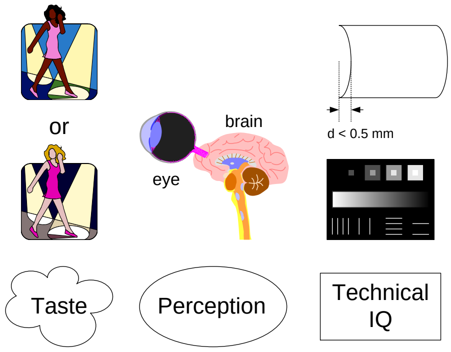


Figure 16: Image Quality

Fashion is also an intangible need of the user. However some product functions can be created to make it possible to follow the fashion, for instance by enabling personalization. A well known example is the exchangeable front of GSM phones. Figure 17 shows downloadable themes as example, which requires all kinds of functions such as format, download, import, scale et cetera.

Figure 17: Fashionable

5 How to verify?

Verification of fuzzy requirements is difficult, while SMART requirements are verifiable by definition. However the fact that a smartened requirements is fulfilled, does not mean that the originating fuzzy requirement is also met.

Confrontation with market and consumers:

Good	**Bad**
Enthusiasm	Critical
Instant playing	Stumbling
Relaxed usage	Tension
Buying	Wait and see

Figure 18: From SMART to Fuzzy

Figure 18 shows that careful observation can be used to obtain insight in the level of fulfillment of the originating fuzzy requirement.

6 Conclusion

”Fuzzy” understanding of requirements and smartened descriptions of requirements are complementary. The smartening process of requirements often significantly increases the understanding of the requirements, mostly due to the need to articulate everything explicit. Unfortunately understanding itself is a non transferable concept, any description always flattens the rich understanding into a limited set of words and definition. Only readers with sufficient a priori knowhow are able to reconstruct the richer understanding again. The essential insights obtained in the requirements analysis are often captured in a few non-SMART statements, where the author hopes to enable the readers to obtain the original understanding.

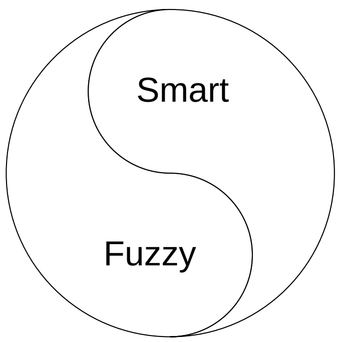


Figure 19: Complementing views

7 Acknowledgements

A long time ago Dieter Hammer send me a presentation explaining the SMART acronym. A vivid discussion with Thomas Gilb triggered this presentation, because I kept struggling with the need for being specific and measurable at the one hand, and the need to understand the less tangible aspects at the other hand. I thank them for their inspiration.

Tom Gilb and Lindsey Brodie helped me to trace back the origin of SMART.

References

[1] George T. Doran. There's a S.M.A.R.T. way to write management's goals and objectives. *Management Review (AMA Forum)*, pages 35–36, November 1981.

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

History

Version: 1.4, date: September 4 2007 changed by: Gerrit Muller

- corrected SMART reference

Version: 1.3, date: September 4 2007 changed by: Gerrit Muller

- added source of SMART

Version: 1.2, date: January 25 2007 changed by: Gerrit Muller

- replaced SMART list with Figure

Version: 1.1, date: January 17 2002 changed by: Gerrit Muller

- Created the article version.
- Added definition of SMART

Version: 0, date: June 2001 changed by: Gerrit Muller

- Created, no changelog yet