Tutorial Roadmapping for Strategy Support

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

Abstract

Formulating and deploying a strategy requires a combination of vision and analysis. Roadmapping is a tool to explore and articulate future needs and trends for different dimensions, such as the market and customer context, the product portfolio, the technology, competences and supply chain, and processes. Roadmapping helps by relating these different dimensions in time, with a horizon of many years. We will discuss how to create and maintain roadmaps and give practical tips on the format.

version: 0.1 status: draft July 9, 2023

Contents

Chapter 1

Introduction

In the interactive session of the tutorial we start with a brief exploration of the roadmaps that participants ahve seen in practice. The theory of the tutorial starts with some background theory on business processes and key drivers, see Figure 1.1 steps 2 and 3. Then we discuss the core roadmapping approach, step 4.

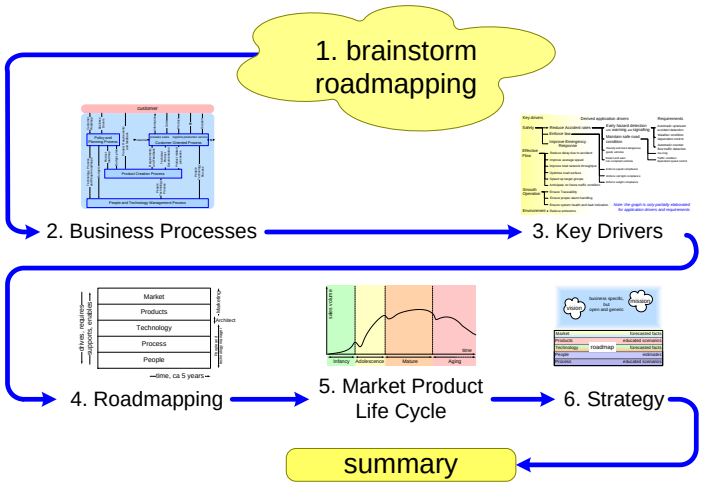
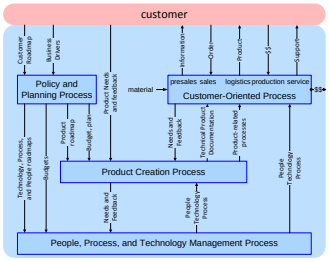


Figure 1.1: Overview of Roadmapping Tutorial

Step 5 provides insight in different kind of products and markets, which is relevant for many aspects of roadmap and strategy. In step 6 we revisit strategy and then we summarize.

Chapter 2

Process Decomposition of a Business



2.1 Introduction

This chapter positions the system architecting process in a wider business scope. The objective of this chapter is to provide system architects insight in the business processes and especially in the processes where system architects actively contribute.

The focus is on companies that create physical products. Other types of businesses, such as solution providers, services, courseware, also need systems architecting. The process structure will deviate somewhat from the structure presented here. See Intermezzo “Products, Projects, and Services” for a discussion on the processes in these other businesses.

2.2 Process Decomposition

The business process can be decomposed in 4 main processes as shown in Figure 2.1. We have on purpose ignored the supporting and connecting processes. This simplification will allow us to get a number of more fundamental insights in the main processes.

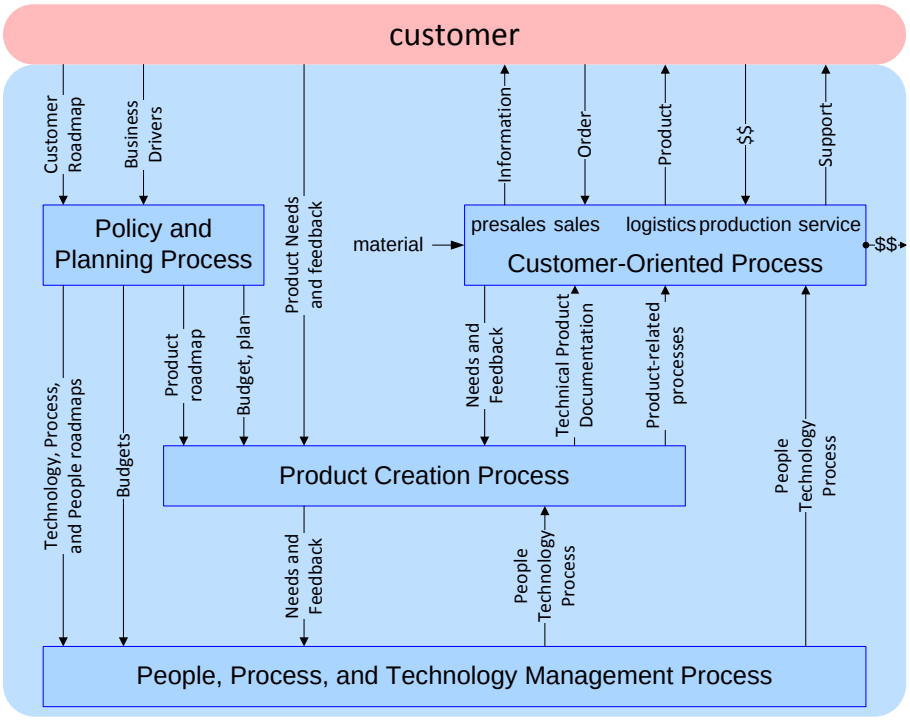


Figure 2.1: Simplified decomposition of the business in 4 main processes

The function of the 4 main processes is:

Customer Oriented Process performs in repetitive mode all direct interaction with the customer. This process is the cash flow generating part of the enterprise. All other processes only spend money.

Product Creation Process feeds the Customer Oriented Process with new products. This process ensures the continuity of the enterprise by creating products that keep the company competitive. In this way the Product Creation Process enables the Customer Oriented Process to generate cash flow in the near future as well.

People, Process, and Technology Management Process manages the competencies of the employees and the company as a whole. The competencies of the employees and the company are the main assets of a company.

Policy and Planning Process is the management process. The Policy and Planning Process defines the strategy, the long term direction of the company, and it balances the shorter term tensions between the three other main processes. The Policy and Planning Process uses roadmaps and budgets to define the

direction for the other processes. Roadmaps give direction to the Product Creation Process and the People, Process and Technology Management Process. For the medium term these roadmaps are transformed in budgets and plans, which are committal for all stakeholders.

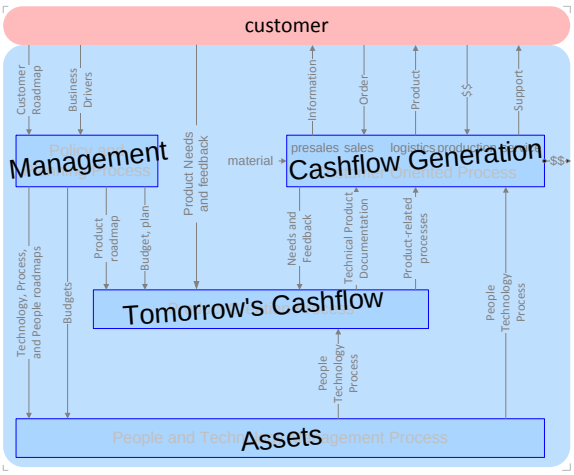


Figure 2.2: Decomposition of the business in 4 main processes, characterized by their financial meaning

The 4 processes as described here are different in nature. The Customer oriented process executes over and over a well defined set of activities. The system architect does not participate in active role in this process. However since the Customer Oriented Process is the main customer of the Product Creation Process, it is crucial that the system architect understands, or better has experienced, the Customer Oriented Process.

The system architect is in continuous interaction with many stakeholders, mostly about technical aspects. From this perspective the architect will generate inputs for the People and Technology Management Process. This might even result in participation in this process for instance by coaching, participation in the appraisal process, or participation in technology studies.

The number of instances of each process is related to different entities:

Customer Oriented Process: Depends on geography, customer base, and supply chain.

Product Creation Process: One per entity to be developed, where such an entity can be a product family, a product, or a subsystem.

People and Technology Management Process: One per "competence", where a competence is a cohesive set of technologies and methods.

Policy and Planning Process: One per business. This is the pro-active integrating process.

The evolutionary developments of product variants and new releases are seen as individual instances of the Product Creation Process. For example the development of a single new feature for an existing product is performed by following the entire Product Creation Process. Of course some steps in the process will be (nearly) empty, which does not cause any harm.

2.3 Process versus Organization

This process decomposition is not an organization, see Intermezzo "What is a Process". A single person can (and often will) fulfill several roles in different processes.

System architects specifically spend most of their time in Product Creation Process (circa. 75%), a considerable amount of time in the Policy and Planning Process (circa 20%), and a small fraction of their time in the People, Process and Technology Management Process.

Most engineers will spend a small amount of time in the People, Process, and Technology Management Process, working on technologies and capabilities, while the majority of their time is spend in the Product Creation Process.

2.4 Value Chain and Feedback

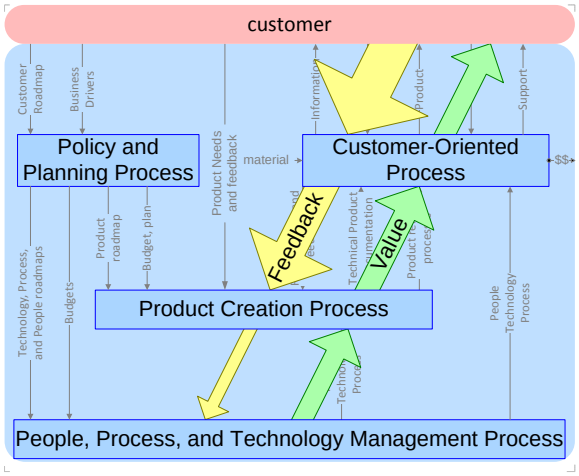


Figure 2.3: The value chain and the feedback flow in opposite direction

The value chain in these processes starts at the assets in the People, Process, and Technology Management Process. The assets are transformed into potential money by the Product Creation Process. The Customer Oriented Process finally turns it into real money. Figure 2.3 shows the value chain.

The feedback flows in the opposite direction, from customer via the Customer Oriented Process and the Product Creation Process to the People Technology and Process Management Process. Customer will communicate mostly with sales and service people. Needs and complaints are filtered by the reporting system before the information reaches Product Creation Teams. Only a small part of the customer feedback reaches the People, Process, and Technology management.

This simple model explains why the knowledge about the customer gets less deeper in the organization. The consequence is that internal technology and process provides show to little concern for urgent customer or business challenges; the sense of urgency seems to be lacking. We can take preventive measures, such as sending process and technology managers to customer sites, once we are aware of the gap caused by this natural information flow.

2.5 Decomposition of the Customer Oriented Process

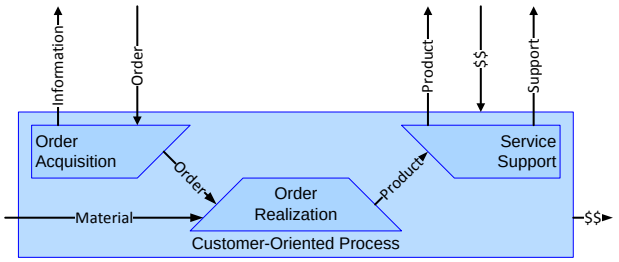


Figure 2.4: Decomposition of the Customer Oriented Process

The Customer Oriented Process is often the largest process in terms of money. From business point of view it is an oversimplification to model this as one monolithic process. Figure 2.4 shows a further decomposition of this process.

The Order Acquisition Process and the Service Support Process are operating quite close to the customer. The Order Realization Process is already somewhat distant from the customer.

The owners of all these three processes are stakeholders of the Product Creation Process. Note that these owners have different interests and different characteristics.

2.6 Extended Process Decomposition; Generic Developments

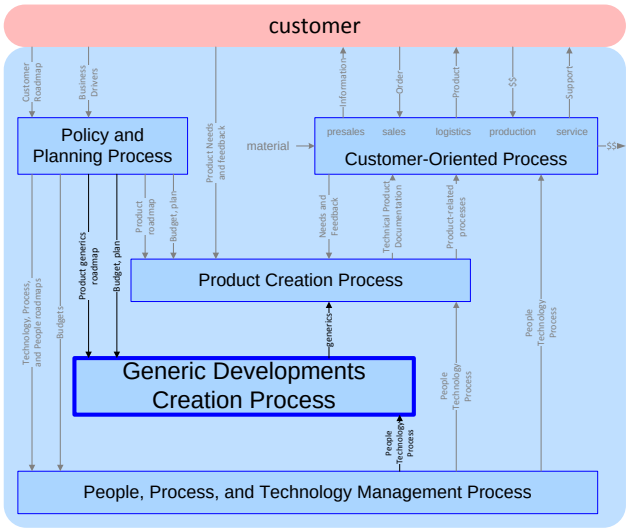


Figure 2.5: The Process Decomposition extended with a generic developments creation process

Companies which develop product families try to capitalize on the commonality between the members of the product family. This is often implemented by the development of common subsystems or functions. In the diagram 2.5 this is called **Generic Developments Creation Process**. A wide variety of names is used for this phenomena, such as re-use, standard design, platform et cetera.

2.7 Acknowledgements

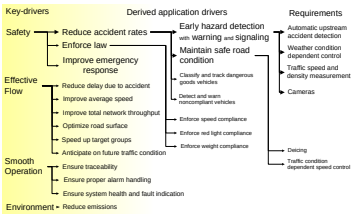
Discussions with and critical comments from Rard de Leeuw, Jürgen Müller, Henk Obbink, Ben Pronk and Jan Statius Muller helped to shape, to improve the structure and to sharpen the contents of the article "Positioning the System Architecture Process". This intermezzo is based on the first sections of this article. I am grateful for their contribution.

Discussion with Ab Pasman helped to remove some architect bias from the process decomposition, by providing a further decomposition of the Customer Oriented Process.

Jaap van der Heijden helped to improve the layout of the diagrams and with the document structure.

Chapter 3

Key Drivers How To



Note: the graph is only partially elaborated for application drivers and requirements

3.1 Introduction

A key driver graph is a graph that relates the key drivers (the essential needs) of the customer with the requirements in the product specification. This graph helps to understand the customer better, and the graph helps to assess the importance of requirements. The combination of customer understanding and value assessment makes the graph into an instrument to lead the project.

We will discuss one example, a Motor way management system, and we will discuss a method to create a customer key driver graph.

3.2 Example Motor Way Management

In this section we discuss an example from practice. The graph discussed here was created in 2000 by a group of marketing managers and systems architects. Creating this version took a few days. Note that we only show and discuss a small part of the entire graph to prevent overload.

Figure 3.1 shows an example of a key driver graph of a motor way management system. A motor way management system is a system that provides information to traffic controllers, and it allows traffic controllers to take measures on the road

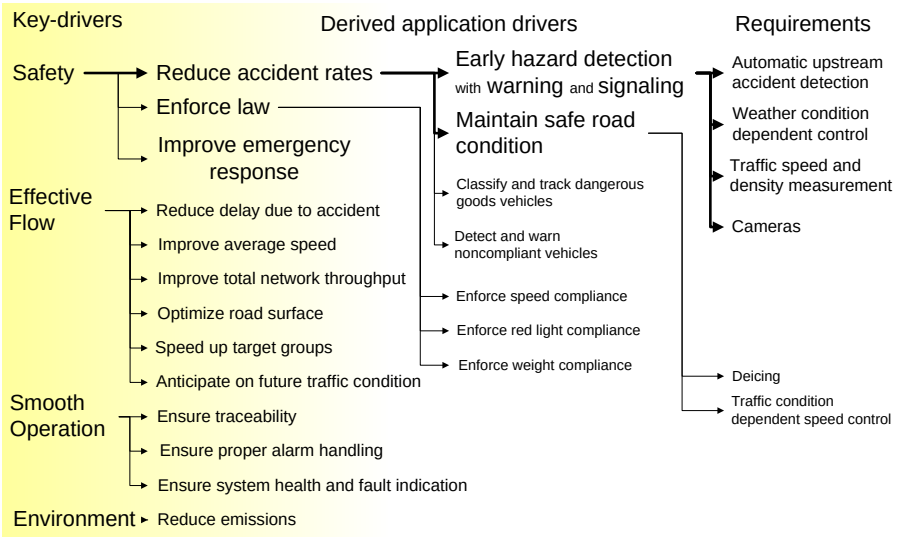


Figure 3.1: The key driver graph of a Motor way Management System

or to inform drivers on the road. As driver we typically see electronic information and traffic signs that are part of these systems. Also the cameras along the road are part of such system.

The key drivers of a motor way management owner are:

Safety for all people on the road: drivers and road maintainers.

Effective Flow of the traffic.

Smooth Operation of the motor way management.

Environment such as low emissions.

To realize these key drivers the owner applies a number of application processes. For example the traffic controllers can improve safety by reducing the accident rate. The accident rate can be reduced by detecting hazards and warning drivers about the hazards. Examples of hazards are accidents that already have happened and in turn may trigger new accidents. Another example of a hazard are bad weather conditions. Hence the automatic detection of accidents and controls that are weather dependent will help to cope with hazards, and hence will reduce accident rates and improve the safety.

Note that the 4 key drivers shown here are the key drivers of the motor way management system. Other systems will also share these concerns, but might not have these as key drivers. For example, smart phones will have a completely different set of key drivers. Do not use this example as template for your own key driver graph, because it biases the effort.

3.3 CAF-views and Key Drivers

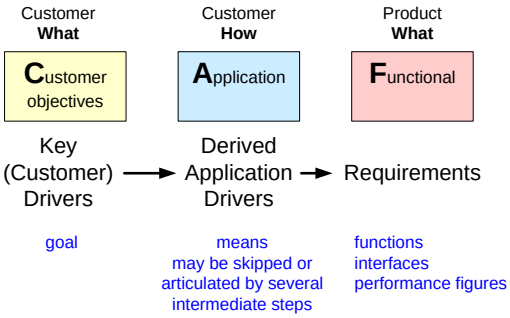


Figure 3.2: The flow from Key Drivers via derived application drivers to requirements

We can capture the essence of the customer world in the *Customer Objectives* view of the CAFCR model by means of customer key drivers. The customer will organize the way of working such that these key drivers are achieved. Figure 3.2 shows how the key drivers as part of the *Customer Objectives* view are supported by application drivers. The application drivers are means to satisfy the customer key drivers. These application drivers in turn will partially be fulfilled by the system-of-interest. Appropriate requirements, e.g. specific functions, interfaces or performance figures, of the system-of-interest will help the customer to use the system to satisfy their customer key drivers. The key drivers are one of the submethods in the Customer Objectives view.

Figure 3.3 shows a method to define key drivers.

Define the scope specific . Identify a specific customer and within the customer a specific stakeholder to make the graph. Choosing a customer implies choosing a market segment. A narrow well defined scope results in a more clear understanding of the customer. The method can be repeated a few times to understand other customers/stakeholders. Products normally have to serve a class of customers. A common pitfall is that the project team too early “averages” the needs and by averaging compromises the value for

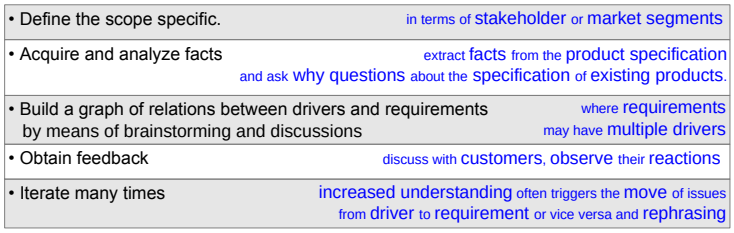


Figure 3.3: Method to define key drivers

specific customers. We recommend to first create some understanding of the target customers before any compromising takes place.

Acquire and analyze facts We recommend to start building the graph by looking for known facts. For example, in most organizations there is already an extensive draft product specification, with many proposed requirements. For every requirement in the draft specification the *why* question can be asked: "Why does the customer need this feature, what will the customer do with this feature?". Repeating the *why* question relates the requirement in a few steps to a (potential) key driver.

Note that starting with facts often means working bottom-up[1]. When marketing and application managers have a good understanding of the customer, then the facts can also be found in the CA-views, allowing a more top-down approach. Iteration, repeated top-down and bottom-up discussions, is necessary in either case.

Build a graph of relations between drivers and requirements by means of brainstorms and discussions. A great deal of the value of this method is in this discussion, where team members create a shared understanding of the customer and the product specification. Note that the graph is often many-to-many: one requirement can serve multiple key drivers, and one key driver results in many different requirements.

Obtain feedback from customers by showing them the graph and by discussing the graph. Note that it is a good sign when customers dispute the graph, since the graph in that case apparently is understandable. When customers say that the graph is OK, then that is often a bad sign, mostly showing that the customer is polite.

Iterate many times top-down and bottom-up. During these iteration it is quite normal that issues move left to right or opposite due to increased under-

[1]Every time that course participants ignore this recommendation, and start top-down while lacking customer insight, they come up with a set of too abstract not usable key drivers.

standing. It is also quite normal that issues are rephrased to sharpen and clarify.

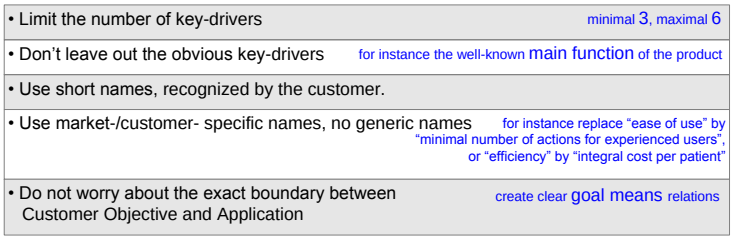


Figure 3.4: Recommendations when defining key drivers

Figure 3.4 shows some recommendations with respect to the definition of key drivers.

Limit the number of key drivers to maximal 6 and minimal 3. A maximum of 6 Key Drivers is recommended to maintain focus on the essence, the name is on purpose **Key** driver. The minimum (three) avoids oversimplification, and it helps to identify the inherent tensions in the customer world. In real life we always have to balance objectives. For example, we have a strong need to maximize safety and performance, while at the same time we will have cost pressure. A good set of key drivers captures also the main tensions from customer perspective.

Do not leave out the obvious key drivers such as the main function of the product. For example, the communication must be recognizable when discussing smart phones; the focus might be on all kinds of innovative features and services, while the main function is forgotten.

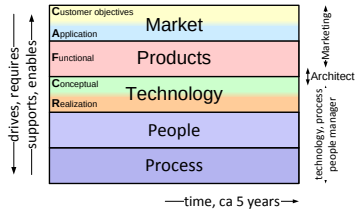
Use short names, recognized by the customer. Key drivers must be expressed in the language of the customer so that customers recognize and understand them. The risk in teams of engineers is that the terminology drifts away and becomes too abstract or too analytical. Another risk is that descriptions or sentences are used in the graph to explain what is meant. These clarifying texts should not be in the graph itself, because the overview function of the graph gets lost. The challenge is to find short labels that resonate with customers.

Use market/customer specific names, no generic names . The more specific a name or label is, the more it helps in understanding. Generic names facilitate the "escape" of diving into the customer world. For example, the term *ease of use* is way too much of a motherhood statement. Instead *minimal number of actions (for experienced users)* might be the real issue.

Allocation to Customer Objectives or Application View Do not worry about the exact boundary between Customer Objective and Application The purpose of the graph is to get a clear separation of goals and means, where goals and means are recursive: an application driver is a means to achieve the customer key driver, and at the same time it is a goal for the functions of the system of interest. Sometimes we need five steps to relate customer key drivers to requirements, sometimes the relation is obvious and is directly linked. The CAFCR model is a means to think about the architecture, it is not a goal to fit everything right in the different views!

Chapter 4

Roadmapping



4.1 Introduction

The definition of new products is a difficult activity, which frequently ends in a stalemate: “It must be don” versus “It is impossible to realize in such a short time frame”. The root cause of this frustrating stalemate is most often the fact that we try to solve a problem in a much too limited scope. Roadmapping is a method to prevent these discussions by lifting the discussion to a wider scope: from single product to product portfolio and from a single generation of products to several generations in many years.

The roadmap is the integrating vision shared by the main stakeholders. A shared vision generates focus for the entire organization and enables a higher degree of cooperating concurrent activities.

We discuss what a roadmap is, how to create and maintain a roadmap, the involvement of the stakeholders and gives criteria for the structure of a roadmap.

4.2 What is in a roadmap?

A roadmap is a visualization of the future (for example 5 years) integrating all relevant business aspects. Figure 6.7 shows the typical contents of a roadmap. At the right hand side the owner of the view is shown, while the left hand side shows

the asymmetry of the views: the market is driving, while technology people and process are enabling.

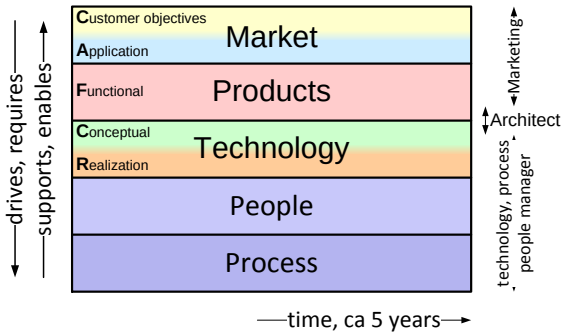


Figure 4.1: The contents of a typical roadmaps

Key to a good roadmap is the skill of showing the important, relevant issues. The roadmap should provide an immediate insight in the most relevant developments from the 5 mentioned points of view. These issues are primarily related by the time dimension.

The convention used in this article is to show products, technologies, people or process when they are or should be available. In other words the convention is to be extrovert, be oriented to the outside world. The introvert aspect, when and how to achieve these items, are not directly shown. This information is often implicitly present, since people and process often have to be available before the availability of the technology, and technology often precedes the product.

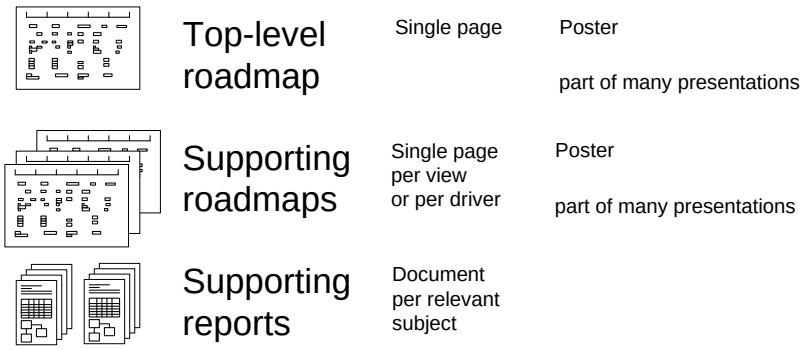


Figure 4.2: The roadmap is documented at several levels of detail

A good roadmap is documented and presented at top level and at a secondary level with more details. Figure 4.2 shows the desired granularity of the roadmap documentation, the secondary level is called supporting roadmaps. The top level

is important to create and maintain the overview, while the more detailed levels explain the supporting data. The choice of the decomposition into supporting roadmaps depends on the domain. Typically, the supporting roadmaps should maintain an integrated view. Examples of decomposition are:

- One supporting roadmap per key driver.
- One supporting roadmap per application area.

4.3 Why Roadmapping?

The Policy and Planning process as discussed in Chapter 2 relies heavily on roadmapping as tool. The main function of roadmapping is to provide a shared insight and overview of the business in time. This insight and overview enables the management of the 3 other processes:

- the Customer Oriented Process
- the Product Creation Process
- the People, Process, and Technology management Process

Where managing these processes means defining the charter and the constraints for these processes in terms of budgets and results: Where do we spend our money and what do we get back for it?

When no roadmapping is applied then the following problems can occur:

Frequent changes in product policy due to lack of anticipation.

Late start up of long lead activities , such as people recruitment and process change.

Diverging activities of teams due to a lack of shared vision.

Missed market opportunities , due to a too late start.

The frequent changes in the product policy are caused by the lack of time perspective. In extreme cases the planning is done with a limited time horizon of, for instance, 1 year. External events which are uncertain in time can shift into view within the limited horizon when popular and disappear again when some other hype is passing by. This effect is shown in Figure 4.3

The availability of a roadmap will help the operational management to apply a low pass filter on their decisions. The control becomes more analog rather than discrete, where the amount of people can be increased or decreased dependent on the expected delivery date, as shown in figure 4.4.

An inherent benefit of roadmapping is the anticipation, which is especially important for all long lead time aspects. Examples are technology, people and

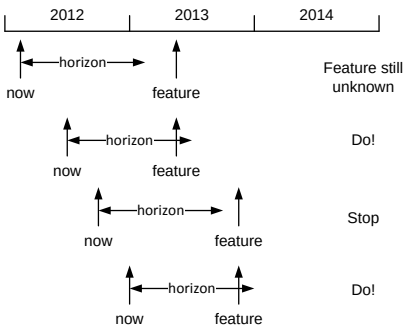


Figure 4.3: Management based on a limited horizon can result in a binary control of product policy decisions

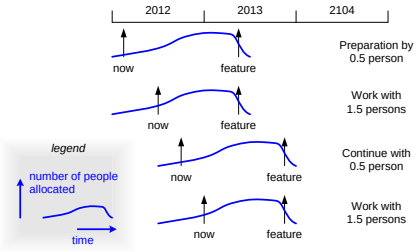


Figure 4.4: Management with a broader time and business perspective results in more moderate control: work with some more or some less people on the feature

process. This is not limited to development activities only; market preparation, manufacturing and customer support also require anticipation. For example, reliable mass production has a significant lead time.

4.4 How to create and update a roadmap

A roadmap is a joint effort of all relevant stakeholders. Typical stakeholders for roadmapping at a typical high-tech company are

business manager , overall responsible for the enterprise

marketing manager(s)

people, process, and technology manager(s) , often called line or discipline managers

operational manager(s) , e.g. program managers or project leaders

architect(s)

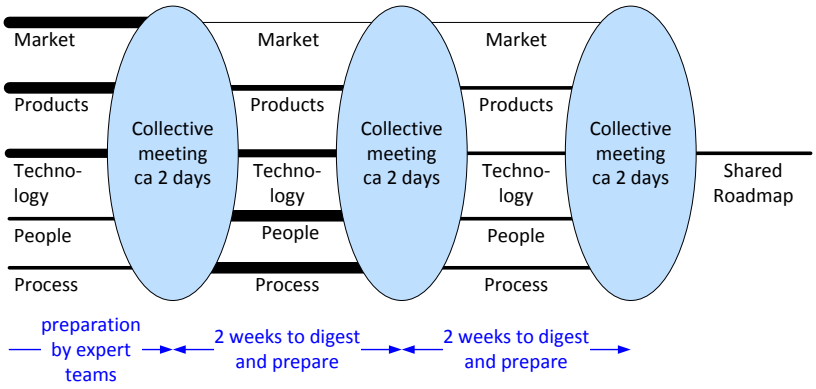


Figure 4.5: Creation or Update of a roadmap in "Burst-mode"

An efficient way to create or update a roadmap is to work in “burst-mode”: concentrate for a few days entirely on this subject. To make these days productive a good preparation is essential. Figure 4.5 shows the roadmap creation or update as three successive bursts of 2 days.

The input for the first days is prepared by expert teams. The expert teams focus on the *market*, the *products*, and the *technology* layers of the roadmap. The current status of *people* and *process* should be available in presentable format. The target of the first burst is:

- to get a shared vision on the market
- to make an inventory of possible products as an answer to the needs and developments in the market
- to share the technology status, trends and ongoing work, as starting point for technology roadmap
- to explore the current status of people and process and to identify main issues

Between the first and second burst and between the second and third burst some time should be available, at the one hand to digest the presented material and the discussions, at the other hand to prepare the next session. The target of the second burst is:

- to obtaining a shared vision on the desired technology roadmap
- to sharing the people and process needs for the products and technology defined in the first iteration
- to analyze a few scenarios for the layers *products*, *technologies*, *people*, and *process*

The thickness of the lines in figure 4.5 indicates the amount of preparation work for that specific part of the roadmap. It clearly shows the the shift in attention from the market side in the beginning to the people and process side later. This shift in attention corresponds with the asymmetry in figure 6.7: the market is driving the business, the people and processes are enabling the business.

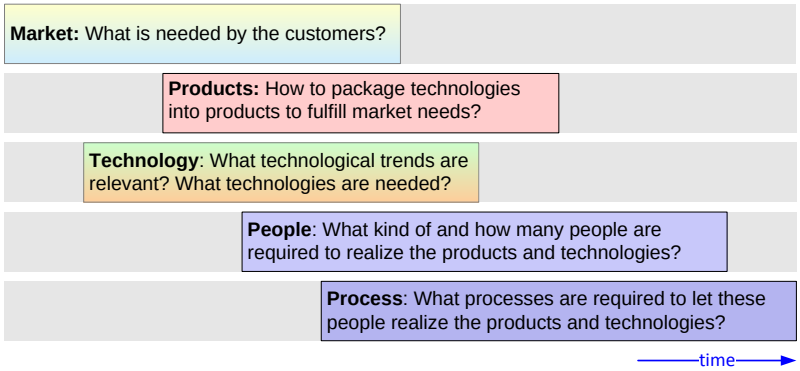


Figure 4.6: The roadmap activities visualized in time.

The function of the collective meetings is to iterate over all these aspects and to make explicit business decisions. The *products* layer of the roadmap should be consistent with the *technology*, *people* and *process* layers of the roadmap. Note that the marketing roadmap may not be fulfilled by the products roadmap, an explicit business decision can be made to leave market segments to the competition.

Figure 4.6 shows the roadmap activities in time. Vertical the same convention is used as in figure 6.7: the higher layers drive the lower layers in the roadmap. This figure immediately shows that although "products" are driving the technology, the sequence in making and updating the roadmap is different: the technological opportunities are discussed before detailing the *products* layer of the roadmap.

4.5 Roadmap deployment

The roadmap is a shared vision of the organization. This vision is implemented in smaller steps, for instance by defining outputs per program and the related resource allocations per program. In Figure 4.7 it is shown that roadmap updates are performed regularly, in this figure every year. After determining the vision a "budget" is derived that sets the charter for the programs. The budget is revised with an higher update frequency, typically every 3 months. The budget itself sets goals and constraints for the operation. The programs and projects in the operation have to realize the outputs defined in the budget. The operational activity itself uses detailed schedules as means for control. The schedules are updated more

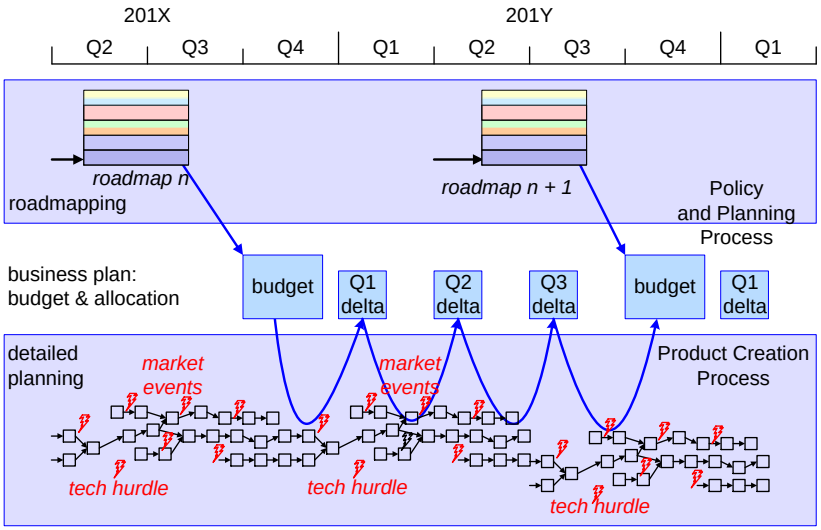


Figure 4.7: The roadmap is used to create a budget and resource allocation. The operational programs and projects use more detailed plans for control.

frequently than the budget update. Within the operational activity the updates are mostly event driven: changes in the market, technology or resources that render the existing plan obsolete.

	horizon	*update*	*scope*	*type*
roadmap	5 years	1 year	portfolio	vision
budget	1 year	3 months	program	commitment
detailed plan	1 mnth-1yr	1 day-1 mnth	program or activity	control means

Figure 4.8: Three planning tiers and their characteristics

From long term vision to short term realization is a 3-tier approach as shown in Figure 4.8. The roadmap provides the context for the budget, the budget defines the context for the detailed plans. The highest tier, the roadmap, has the longest horizon, the slowest update rate, and the broadest scope. When going down in tiers, the horizon tends to decrease, the update rate increases, and the scope decreases. The roadmap provides a vision, and as such is not committal. A budget is a commitment to all involved parties. Plans are means to realize the programs and projects, and tend to adapt frequently to changed circumstances.

4.6 Roadmap Essentials

We recommend to create a roadmap that fulfills the following requirements:

- Issues are recognizable for all stakeholders.
- All items are clearly positioned in time; uncertainty can be visualized explicitly.
- The main events (enabling or constraining) must be present.
- The amount of information has to be limited to maintain the overview.

4.6.1 Selection of most important or relevant issues

The art of making a roadmap is the selection of the most relevant issues. It is quite easy to generate an extensive roadmap, visualizing all marketing and technological information. However, such superset roadmap is only the first step in making the roadmap. The superset of information will create an overload of information that inhibits the overview we strive for.

4.6.2 Key drivers as a means to structure the roadmap

In [2] key drivers are explained as an effective method to elicit and understand requirements. Key drivers can also be very helpful in the creation and update of the roadmap. At the marketing side the trend in these key drivers must be visible in the roadmap. Showing key driver trends also helps to structure the roadmap.

The supporting roadmaps can clarify how the key driver trends will be supported For instance, a technology roadmap per key driver is a very explicit way to visualize the relationship between the market in terms of key drivers, the products with the expected performance levels, and enabling technologies.

4.6.3 Nothing is certain, ambiguity is normal

A roadmap is a means to share insight and understanding in a broader time and business perspective. Both dimensions are full of uncertainties and mostly outside the control of the stakeholders. It can not be repeated often enough that a roadmap is **only** a vision (or dream?).

The only certainty about a roadmap is that reality will differ from the vision presented in the roadmap.

As a consequence the investment in making the roadmap more accurate and more complete should be limited. Nobody can predict the future, we will have to live with rather ambiguous visions and expectations of the future.

4.6.4 Use facts whenever possible

The disclaimer that *ambiguity is normal* can be used as an excuse to deliver sloppy work Unfortunately, a sloppy roadmap will backfire to the creators. It is recommended to base a roadmap on facts whenever possible Examples of sources of facts are:

- Market analysis reports (number of customers, market size, competition, trends)
- Installed base (change requests, problem reports, historical data)
- Manufacturing (statistical process control)
- Suppliers (roadmaps, historical data)
- Internal reports (technology studies, simulations)

Use of multiple data sources enable cross-verification of the sanity of assumptions For instance, predictions of the market size in units or in money should fit with the amount of potential customers and the amount of money these customers are capable (and willing) to spend.

4.6.5 Do not panic in case of impossibilities

It is quite normal that the roadmap layers appear to be totally inconsistent For instance, a frequent occurring effect is that the budget estimate in response to the market requirements is 3 times the available budget[1]. Retrospective analysis of past roadmaps shows that the realized amount of work for the given budget is often twice the estimate made for the roadmap. In other words, due to a number of effects the roadmap estimates tend to have a pessimistic bias. The overestimation can be caused by:

- Quantization effects of small activities (the amount of time is rounded to person weeks/months/years).
- Uncertainty is translated into margins at every level (module, subsystem, system).
- Counting activities twice (e.g., in technology development and in product development).
- Quantization effects of persons/roles (full time project leader, architect, product manager, et cetera per product).

[1] This factor 3 is an empirical number which of course depends on the company and its culture

- Lack of pragmatism, a more extensive technical realization than required for the market needs.
- Too many bells and whistles without business or customer value.

Initial technical proposals might be more extensive than required for market needs, as mentioned in the lack of pragmatism. Technical ambition is good during the roadmap process, as long as it does not pre-empt a healthy decisions. The roadmapping discussions should help to balance the amount of technology anticipation with needs and practical constraints.

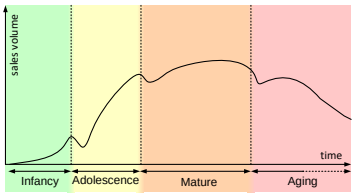
4.7 Acknowledgements

The insight that a roadmap should cover all 5 views form market to process came to me via Hans Brouwhuis. Roadmapping as a business tool gained momentum within Philips during the quality actions inspired by Jan Timmer.

The critical and constructive remarks by Jürgen Müller helped to shape this article.

Chapter 5

Market Product Life Cycle Consequences for Architecting



5.1 Introduction

A class of products serving a specific market evolves over time. This evolution is reflected in the sales volume of these products. The systems architecting approach depends where products are in this evolution.

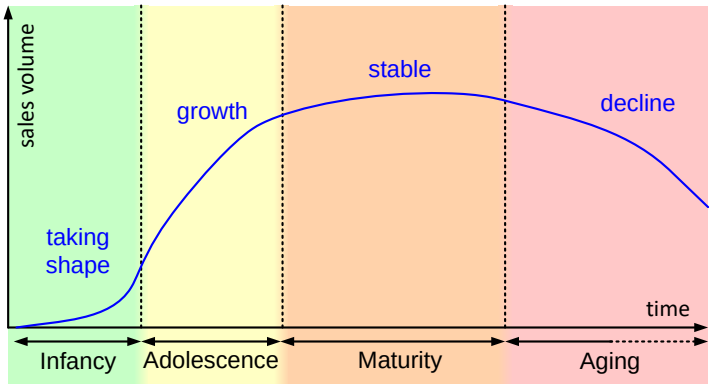


Figure 5.1: Compared with ideal bathtub curve

The life cycle of a product market combination can be visualized by showing the sales volume as a function of the time. In literature the form of the curve of the sales volume as function of the time is described as bathtub, see figure 5.1. It is customary to recognize four phases in this curve:

- The life cycle starts with very small sales in the **infancy** phase, where the product finds its shape.
- A fast increasing sales volume in the **adolescent phase**.
- A more or less stable sales volume in the **mature** phase.
- A decreasing sales volume in the **aging** phase.

The curve and its phases represent the theoretical evolution. In the next paragraphs we will discuss observations in practice and an explanation, and we will show that the class of products and the market themselves also evolve on a macro scale.

5.2 Observed Life Cycle Curve in Practice

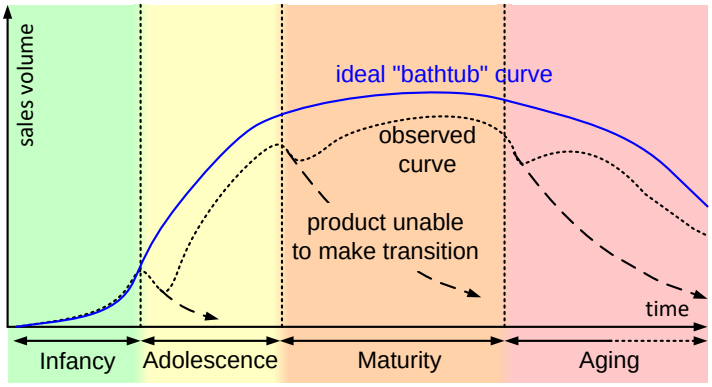


Figure 5.2: Market product life cycle phases

Henk Obbink (Philips Research) observed dips in the sales volume, as shown in figure 5.2. The transition from one phase to the next does not seem to happen smoothly. In some cases the sales drops further and the product does not make the transition at all.

The hypothesis for the dips in the curve is that characteristics of all stakeholders are different for the different life cycle phases. If the way of working of an organization is not adopted to these changes, then a mismatch with the changed circumstances results in decreasing sales. Figure 5.2 also indicates that, if no adaptation to

the change takes place, that the sales might even drop to zero. Zero sales effectively is killing the business, while still plenty of market opportunity is present.

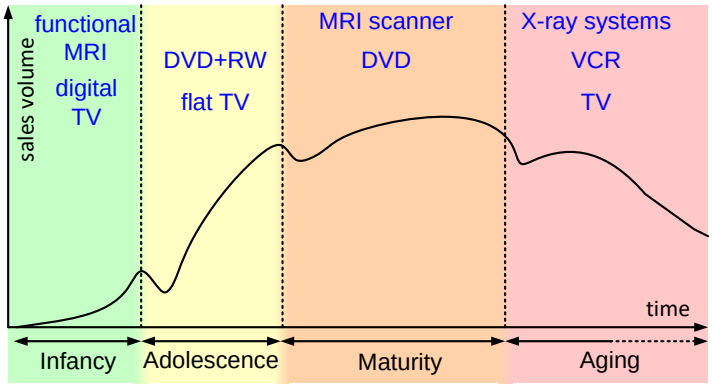


Figure 5.3: Examples of product classes on the curve

Figure 5.3 annotates the life cycle graph with a number of products and their positioning in the life cycle. As can be seen products can move backwards in the phases (i.e. become "younger") by the addition of innovative features. For instance MRS scanners moved backwards when *functional imaging* was added, an innovative way to visualize the activity of specific tissues. Similarly, conventional televisions rejuvenated multiple times by adding digital processing, flat screens, and digital interfaces.

5.3 Life Cycle Model

Figure 5.4 shows typical attributes of the life cycle phases.

The *infancy* phase is characterized by uncertainty about the customer needs, and therefore the product requirements. Essential is that the creator/producer is responsive to the customer needs, which will provide insight in needs and requirements. The way of working in this phase reflects the inherent uncertainty, the chaotic development, and the innovative and pioneering mind set. Product cost is still less of an issue, the risk related to the uncertainty is the dominant concern. The design copes with the uncertainty by over-dimensioning those aspects which are perceived to be the most uncertain.

The *adolescent* phase is characterized by strong (exponential) growth of the sales volume, concurrent with an increase in performance, features and product variants. The challenge is to cope with this strong growth in many dimensions. With respect to the requirements a strategic selection is needed, to serve the growing customer base, without drowning in an exploding complexity. The technical and process challenge is to scale up in all dimensions at the same time. Up-scaling

	Infancy	Adolescence	Mature	Ageing
Driving factor	Business vision		Stable business model	Harvesting of assets
Value from	Responsiveness	Features	Refinements / service	Refining existing assets
Requirements	Discovery	Select strategic	Prioritize	Low effort high value only
Dominant technical concerns	Feasibility	Scaling	Legacy Obsolescence	Lack of product knowledge Low effort for obsolete technologies
Type of people	Inventors & pioneers	Few inventors & pioneers "designers"	"Engineers"	"Maintainers"
Process	Chaotic		Bureaucratic	Budget driven
Dominant pattern	Overdimensioning	Conservative expansion	Midlife refactoring	UI gadgets

Figure 5.4: Attributes per phase

the Customer Oriented Processes and the Product Creation Process requires more shared structure between the participants. This involves a mind set change: less inventors, more designers. The design pattern used frequently in this phase is conservative extension of a base design.

The *mature* phase is characterized by more stability of the business model and the market, while the market has become much more cost sensitive. Instead of running along in the feature race more attention is required to optimize the specification and development choices. The value can be shifting from the core product itself to services and complements of the product, while the features of the product are mostly refined. The age of the product starts to interfere with the business, obsolescence problems occur, as well as legacy problems. Innovative contributions become counterproductive, more rigid engineers are preferred above creative designers. The cost optimization is obtained by process optimization, where the processes also become much more rigid, but also more predictable, controllable and executable by a large community of less educated engineers. The design copes with the aging technology by performing limited refactoring activities in areas where return on investment is still likely.

The *aging* phase is often the phase where the product is entirely seen as cash cow, maximize the return on (low) investments. This is done by searching all the low effort high value requirements, resulting mostly in small refinements to the existing product. Often the integral product know how and even specialist know how has been lost. Only very important obsolescence problems are tackled. Again the mind set of the people working on the product is changing to become more maintenance oriented. Cost is a very dominating concern, budgets are used to

control the cost. Many changes are cosmetic or superficial, taking place in the most visible parts of the product: the user interface and the outer packaging.

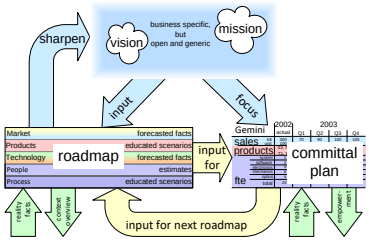
5.4 Acknowledgements

Henk Obbink observed the discontinuity of market success at the phase transitions. The analysis of this phenomenon was carried out by Jürgen Müller, Henk Obbink and Gerrit Muller.

Pierre America improved the layout of the diagrams.

Chapter 6

The role of roadmapping in the strategy process



6.1 Process decomposition of a business

The business process for an organization which creates and builds systems consisting of hardware and software is decomposed in 4 main processes as shown in figure 6.1.

The decomposition in 4 main processes leaves out all connecting supporting and other processes. The function of the 4 main processes is:

Customer Oriented Process This process performs in repetitive mode all direct interaction with the customer. This primary process is the cashflow generating part of the enterprise. All other processes only spend money.

Product Creation Process This Process feeds the Customer Oriented Process with new products. This process ensures the continuity of the enterprise by creating products which enables the primary process to generate cashflow tomorrow as well.

People and Technology Management Process Here the main assets of the company are managed: the know how and skills residing in people.

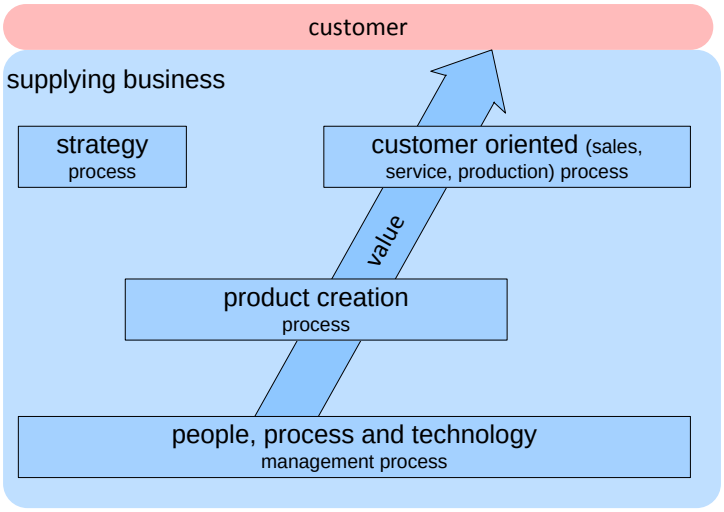


Figure 6.1: Simplified decomposition of the business in 4 main processes

Policy and Planning Process This process is future oriented, not constrained by short term goals, it is defining the future direction of the company by means of roadmaps. These roadmaps give direction to the Product Creation Process and the People and Technology Management Process. For the medium term these roadmaps are transformed in budgets and plans, which are committal for all stakeholders.

Figure 6.2 characterizes the processes from the financial point of view. From bottom to top soft or latent value (the assets) are transformed in harder value, to become true money when the customers are paying for the products and services (the cashflow).

At the same time figure 6.2 shows that the feedback flow from the customer into the organization moves in the opposite direction. A nasty phenomenon is the deformation and loss of feedback information while it flows through these processes. The further away from the customer, the less sense of urgency and the less know how of the customer needs. In many organizations this is a significant problem: competence organizations which have lost the sight of the customer and become introvert.

In many companies the value chain is optimized further, by using the synergy between products and product families. Figure 6.3 shows that the simplified process decomposition model can be extended by one process *component or platform creation* to visualize this strategy. This optimization is far from trivial. At the one hand synergy must be used, most companies cannot afford to create everything from scratch all the time. At the other hand is the consequence of the set up shown here that the value chain becomes longer (and takes somewhat longer), while

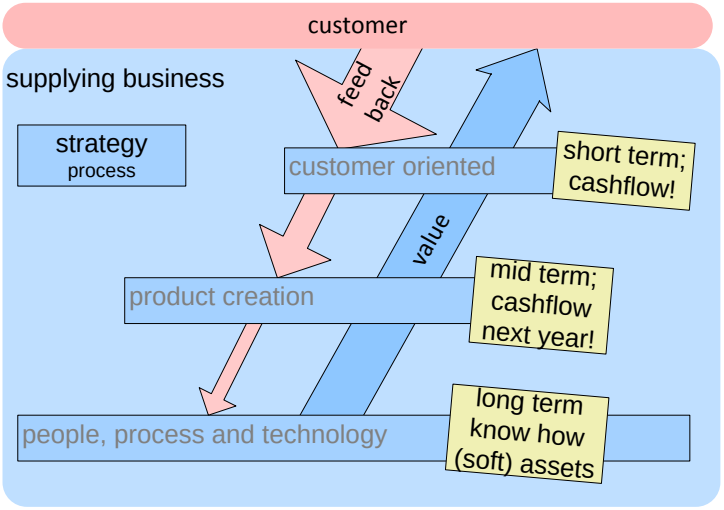


Figure 6.2: Tension between processes

the feedback deformation and loss increases even further! A more elaborated discussion on these aspects can be found in [1].

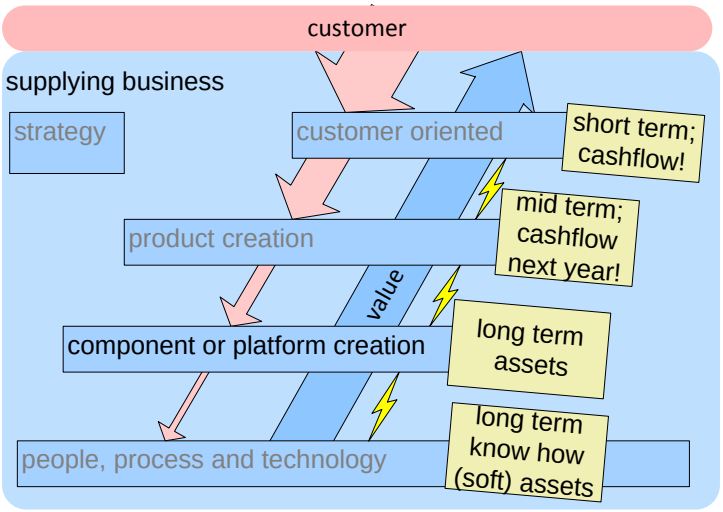


Figure 6.3: Platform strategy adds one layer

6.2 Framework for architecting and roadmapping

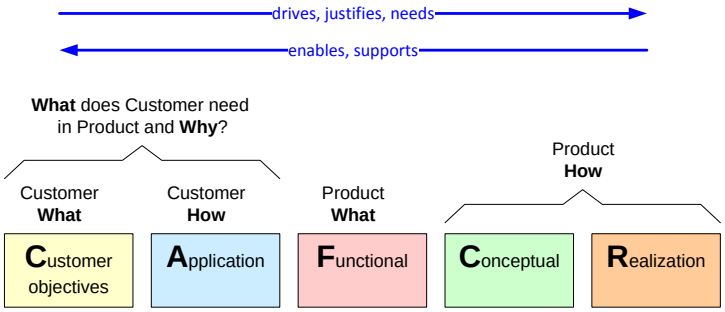


Figure 6.4: CAFCR framework for architecting

Figure 6.4 shows the ”CAFCR” framework for system architecting, see [5]. The *customer objectives* view and the *application* view provide the **why** from the customer. The *functional* view describes the **what** of the product, which includes (despite the name) also the *non functional* requirements. The **how** of the product is described in the *conceptual* and *realization* view, where the conceptual view is changing less in time than the fast changing realization (Moore's law!).

The job of the architect is to integrate these views in a consistent and balanced way. Architects do this job by *frequent viewpoint hopping*, looking at the problem from many different viewpoints, sampling the problem and solution space in order to build up an understanding of the business. Top down (objective driven, based on intention and context understanding) in combination with bottom up (constraint aware, identifying opportunities, know how based), see figure 6.5.

In other words the views must be used concurrently, not top down like the waterfall model. However at the end a consistent story must be available, where the justification and the needs are expressed in the customer side, while the technical solution side enables and support the customer side.

The term *customer* is easily used, but it is far from trivial to determine the customer. The position in the value chain shows that multiple customers are involved. In figure 6.6 the multiple customers are addressed by applying the CAFCR model recursively.

The customer is a gross generalization. Marketing managers make a classification of customers by means of a market segmentation. Nevertheless stay aware of the level of abstraction used when discussing **the** customer/market/market segment.

The viewpoints of the ”CAFCR” framework are useful for setting up a roadmap as well. However on top of these views also *business*, *people* and *process* views are needed in a roadmap, see figure 6.7 and [3].

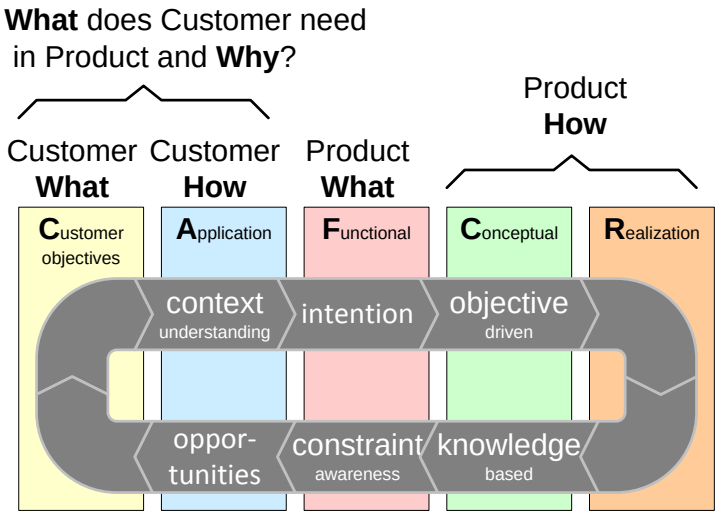


Figure 6.5: Five viewpoints for an architecture. The task of the architect is to integrate all these viewpoints, in order to get a *valuable*, *usable* and *feasible* product.

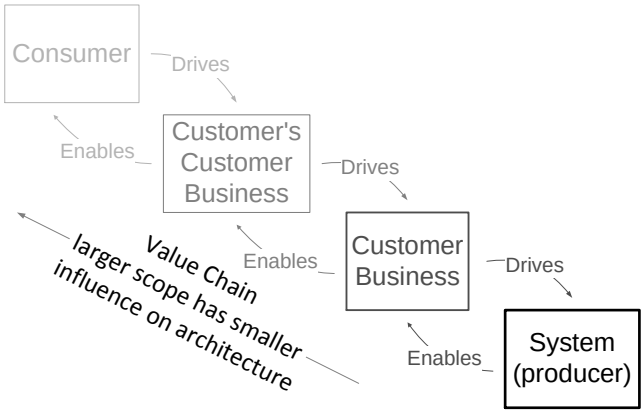


Figure 6.6: CAFCR can be applied recursively

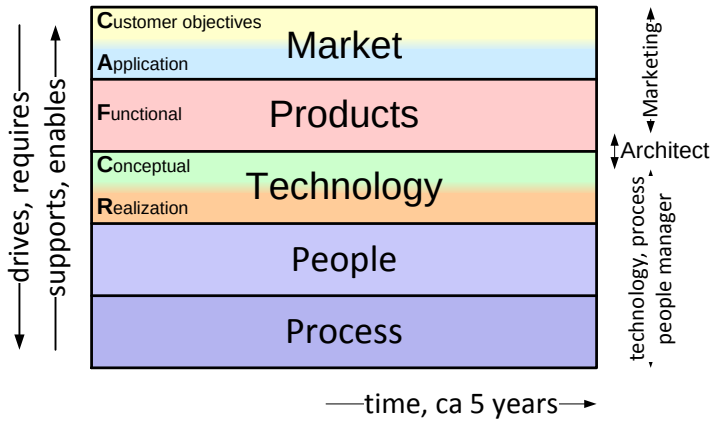


Figure 6.7: Structure of a roadmap

6.3 From vision to roadmap to plan and further

The identity or the main focus of a company is often expressed in a mission statement, supported by a vision on the market, the domain and its own position in market and domain. The nature of both mission and vision is highly generic, although business specific. Mission and vision is a compact articulation of the company and its strategy.

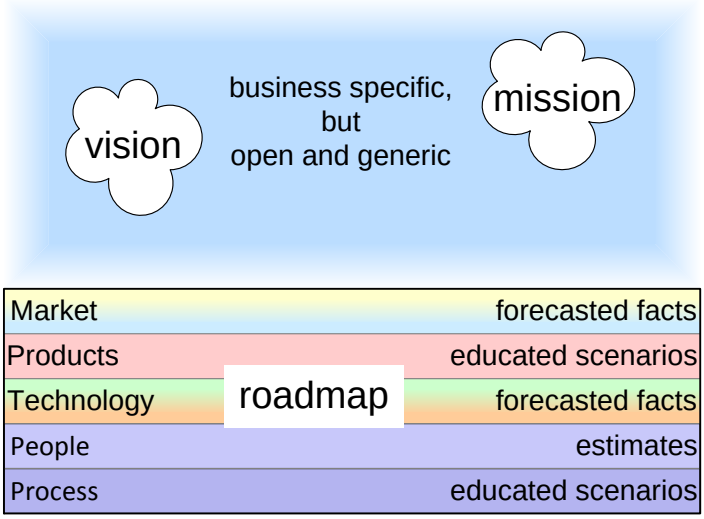


Figure 6.8: From generic mission to factual roadmap

The roadmap builds on vision and mission and makes the strategy much more specific in time as well as in contents. Figure 6.8 shows the generic mission and vision statement as overarching entities for the roadmap. As indicated within the roadmap segments its content is much more specific, containing (forecasted) facts, (educated) scenarios and estimates.

An integrated roadmap is made in steps:

1. Explore *market*, *product* and *technology* segments; what is happening in the outside world, what is needed, where are opportunities in market and/or technology.

2. Estimate *people* and *process* needs for the identified *product* and *technology* needs. These estimates should be made without constraints. The question is what is **needed**, rather than what is **possible**.

3. Determine a balanced, economic attractive and skills wise feasible content for *product*, *technology*, *people* and *process*. Here trade-offs have to be made

and creative marketing as well as technological skills are required to define an effective product roadmap, which is at the same time realistic with respect to the people and processes.

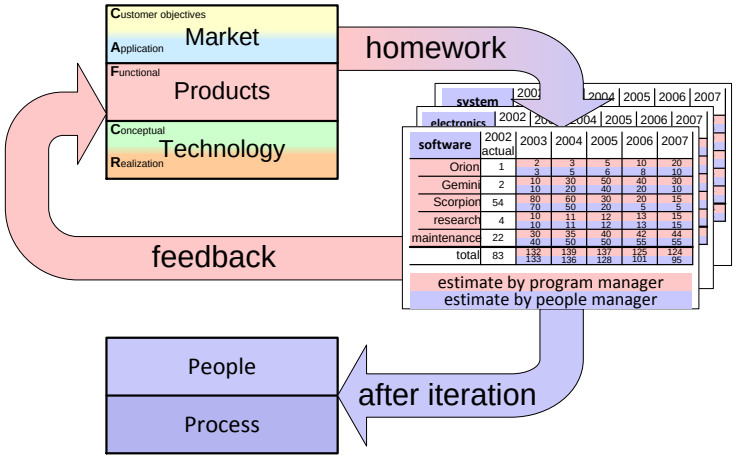


Figure 6.9: From Market, Product, Technology to People, Process

Figure 6.9 shows how to make the last few steps. The estimations for the amount of people are made from 2 viewpoints: the people and technology manager (the supplier of resources) and the operational manager (responsible for the timely and reliable result of the product creation process and hence the ”consumer” of these resources).

The people and technology manager will make estimates which are discipline specific, decomposed towards the programs, see figure 6.10

The operational manager (or program manager) will make an estimate which is program specific. A program is a cohesive set of products, where the program manager is responsible for the timely development and quality of all products within the program. This estimate will be decomposed into disciplines, see figure 6.11.

Every activity is estimated twice via this approach. In both figure 6.10 and figure 6.11 the corresponding second estimate is shown as well, in other words the results are merged. This merge immediately shows differences in interpretation of the input or differences in opinion. These differences should be discussed, so either the inputs are reiterated, resulting in a shared estimate, or the difference in opinion is analyzed and a shared estimate must be the result (although the compromise may be marked as highly uncertain)

After this ”harmonization” of the estimates the real difficult work starts, of tweaking the product program, the required features and being more creative in the solutions in order to come to a feasible roadmap. This step will change the *product* and *technology* segments, with corresponding changes in *people* and *process*.

Orion	1	2 3	3 5		10 8	20 10
Gemini	2	10 10	30 20		40 20	30 10
Scorpion	54	80 70	60 50		20 5	15 5
research	4	10 10	11 11		13 13	15 15
maintenance	22	30 40	35 50		42 55	44 55
total	83	132 133	139 136		125 101	124 95

software	2002 actual	2003	2004	2005	2006	2007
				5 6		
				50 40		
				30 20		
				12 12		
				40 50		
				137 128		

estimate by program manager

estimate by people manager

Figure 6.10: People estimate, discipline view

Figure 6.12 shows the people roadmap from another domain in a more visual format. In this example a clear growth of the staffing is visible, where for instance system and software are growing much faster than electronics. Besides these typical product creation disciplines also the *customer oriented* people and skills are shown. The decomposition choosen here is to the needed or expected education level (high, medium and low). The clear trend here is a significant growth of customer support people, while at the same time it is expected that the education level will decrease significantly[1].

If we decompose the people estimates from figure 6.12 in the operational direction then a much more dynamic picture emerges. Operational activities have a faster rhythm than disciplines. Understanding of this dynamics helps in the total balancing act required from the strategy process. Special attention should be given to the often implicit programs, such as:

- installed base management

[1]This is a quite normal trend. Young products are supported by highly skilled people, which is possible because the installed base is still small. When the installed base is growing it is difficult to find sufficient well trained people, who are motivated to work as support personnel. At the same time the cost pressure increases, which makes it economically unattractive to hire expensive support people. All together the consequence is that investments in the product and the processes are required to operate in the more mature phase with less educated customer support people.

Gemini	2002 actual	2003	2004	2005	2006	2007
system	1	2 3	4 5	5 6	4 5	3 4
software	2	10 10	30 20	50 40	40 20	30 10
electronics	5	16 12	20 18	12 16	4 12	2 6
mechanics	8	8 12	5 14	2 8	1 6	1 3
optics	4	6 6	6 6	5 5	4 4	3 3
total	20	42 43	64 63	74 75	52 47	39 26

estimates by program manager

estimates by discipline manager

Figure 6.11: People estimate, program view

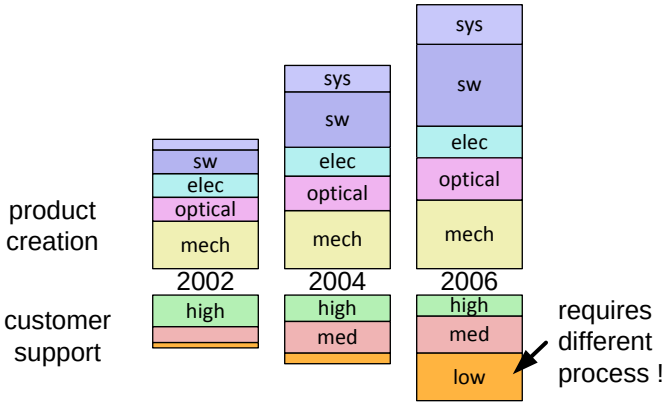


Figure 6.12: Roadmap of people skills

- component and platform creation
- research
- development infrastructure

At the end a sanity check should be made of the balance between the explicit programs and the less explicit programs mentioned here. The explicit, product oriented programs in general should use a significant amount of the total man count, otherwise it is a symptom of an introvert organization (focus on **how** do we do it, instead of **what** is needed).

The roadmap created as described above is a means to share insight in the market and the future and to provide overview and focus to the entire organization,

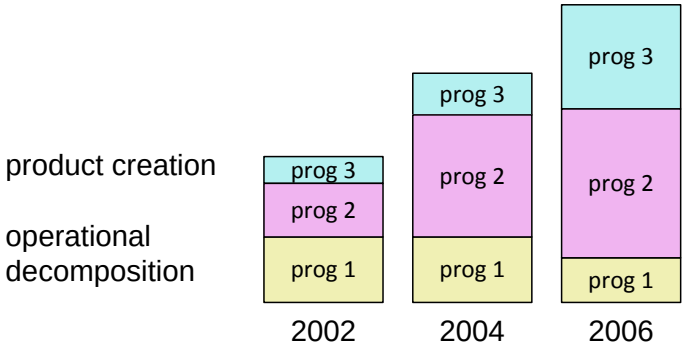


Figure 6.13: Operational axis is more dynamic

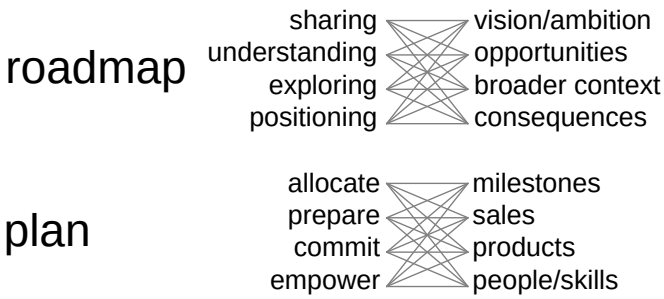


Figure 6.14: From roadmap to planning

in a broad time perspective. This process should take place in an open, explorative atmosphere. This can be achieved by keeping the roadmap as a shared snapshot of the future and not make it a committal plan. In other words nobody gains any right because of the roadmap. The roadmap does not contain hard decisions, it contains shared understanding and expectations.

The roadmap is used as input to create a committal plan, with a shorter time horizon. It does not make any sense to make long term commitments, the future is way too uncertain for hard decisions. The committal plan will typically have a scope of 1 year. Within this year a consistent set of decisions are needed, ranging from sales and turnover commitments to product creation commitments (main product characteristics and timing) to technology, people and process commitments. This commitment serves also as a means to delegate and empower, which also requires allocation of resources. Figure 6.14 shows the essentials of the roadmap and the committal plan.

Figure 6.15 shows an example of a committal plan, containing the business

Gemini		2002 actual	2003 Q1	Q2	Q3	Q4
sales	k$	300	70	90	100	105
	unit	100	20	25	25+3	22+7
products		S2, S3 T1, T4	S4		V6	S6
fte	system	1	2	3	3	4
	software	2	10	18	24	28
	electronics	5	16	17	19	20
	mechanics	8	8	8	6	6
	optics	4	6	6	6	6
	total	20	42	50	58	64

Figure 6.15: Example of committal plan

commitments (sales), the PCP commitments (products to be created) and the people and technology commitments (allocated fte's[2]). Such a plan must be available per program, in this example it is the *Gemini* program.

6.4 Summary

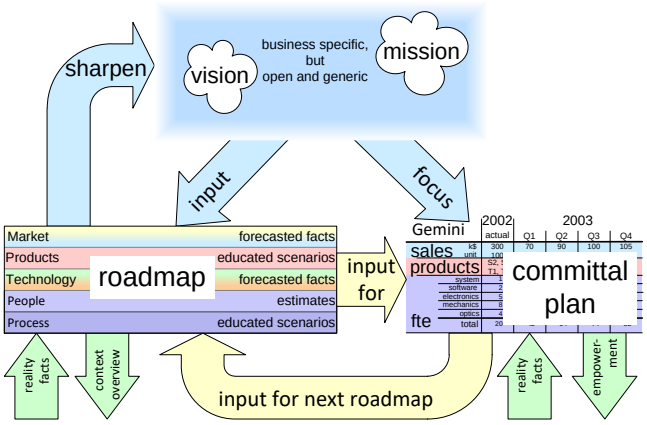


Figure 6.16: Overview of strategic entities

The mission, vision, roadmap and plan will normally be used as part of the business plan, which is used towards the financial stakeholders of the company. These entities together define the strategy and the deployment of the strategy. Figure 6.16 shows an overview of the entities which play a role in the startegy process.

[2] fte = full time equivalents

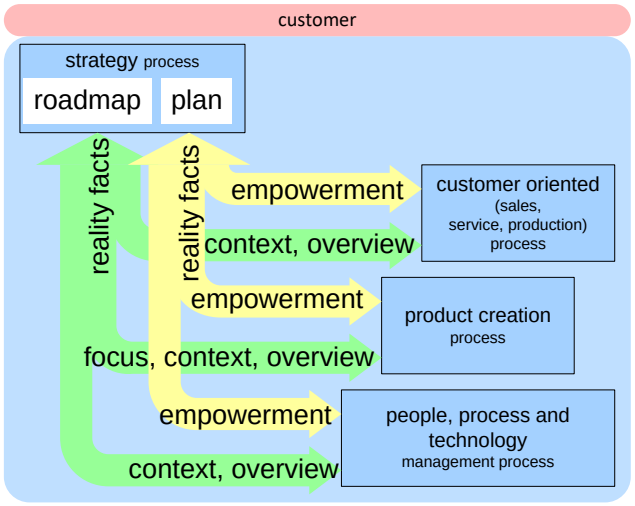


Figure 6.17: Summary of role in business

The value of roadmap for the other processes is to provide context and overview for the specific goal of that process. Especially for the product creation process it also provides focus, the development team can concentrate on the product, which is currently being developed, without discussions of all other alternatives.

The value of the plan for the other processes is that it provides the delegation boundaries, which allows for empowerment. Figure 6.17 shows the value of roadmap and plan for the other processes. In the opposite direction the other processes should provide the reality facts to be used in next roadmap and plan.

6.5 Acknowledgements

Philip Bucher asked me for this presentation for the GATIC workshop, which provided me with the right trigger to write this already long ago planned article. Philip also helped by discussing the purpose and content.

Bibliography

[1] Gerrit Muller. Product families and generic aspects. http://www.gaudisite.nl/GenericDevelopmentsPaper.pdf, 1999.

[2] Gerrit Muller. Requirements capturing by the system architect. http://www.gaudisite.nl/RequirementsPaper.pdf, 1999.

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

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

[5] Gerrit Muller. Architectural reasoning explained. http://www.gaudisite.nl/ArchitecturalReasoningBook.pdf, 2002.

History

Version: 0.1, date: June 8, 2010 changed by: Gerrit Muller

- corrected the title
- included changes applied to original Gaudí material, especially transformation of lists and tables in figures

Version: 0, date: March 22, 2010 changed by: Gerrit Muller

- composed the tutorial from existing Gaudí material
- Created, no changelog yet