Module System Architecture Context

Gerrit Muller
Buskerud University College and Buskerud University College
Hasbergsvei 36 P.O. Box 235, NO-3603 Kongsberg Norway
gaudisite@gmail.com

Abstract

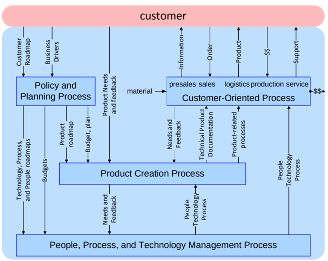
This module zooms in from the company scope to the architecting scope. The business is described by a simplified decomposition in four processes. The *Product Creation Process* is described and decomposed further in three processes. The architecting process is described in relation to the four business processes.

version: 1.1 status: concept March 27, 2021

Contents

Chapter 1

Process Decomposition of a Business



1.1 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.

1.2 Process Decomposition

The business process can be decomposed in 4 main processes as shown in Figure 1.1. 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.

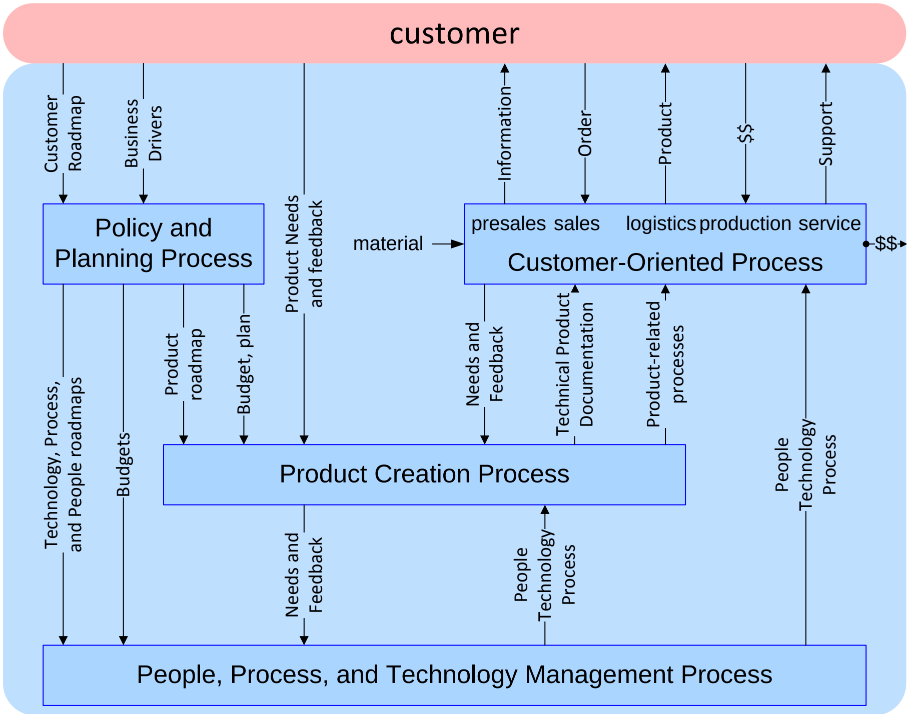


Figure 1.1: Simplified decomposition of the business in 4 main processes

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.

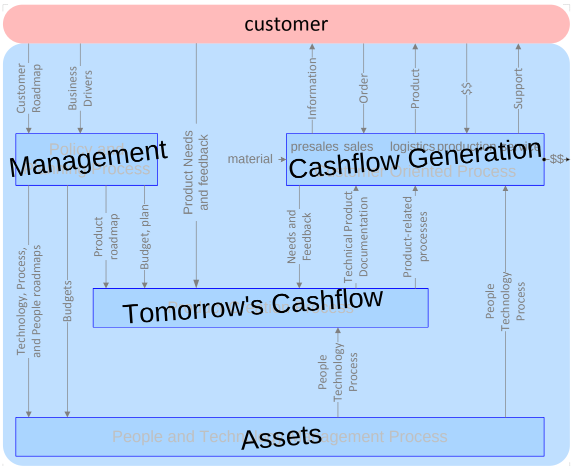


Figure 1.2: Decomposition of the business in 4 main processes, characterized by their financial meaning

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.

1.3 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.

1.4 Value Chain and Feedback

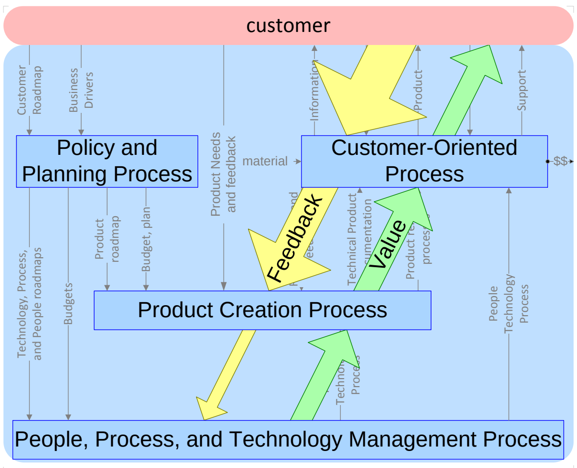


Figure 1.3: The value chain and the feedback flow in opposite direction

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 1.3 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.

1.5 Decomposition of the Customer Oriented Process

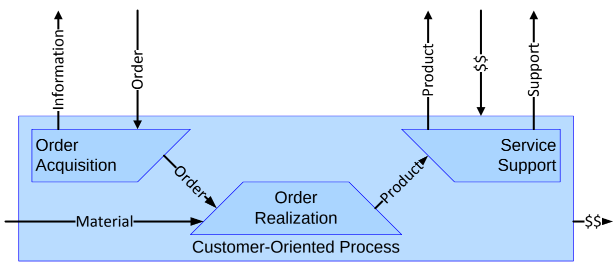


Figure 1.4: Decomposition of the Customer Oriented Process

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 1.4 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.

1.6 Extended Process Decomposition; Generic Developments

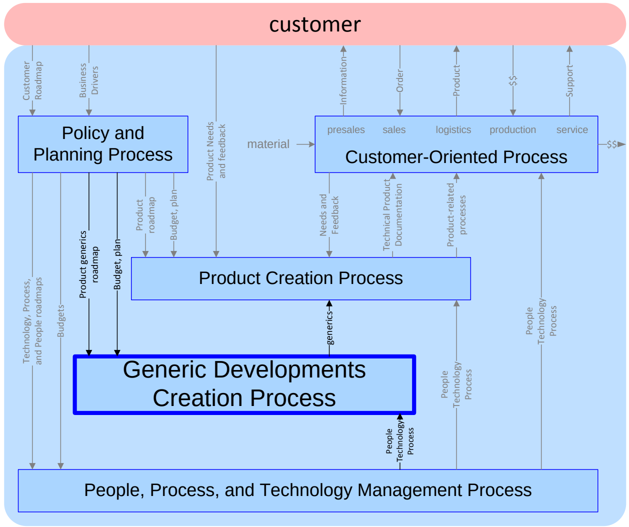


Figure 1.5: The Process Decomposition extended with a generic developments creation process

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 1.5 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.

1.7 Acknowledgements

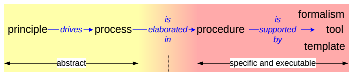
Discussions with and critical comments from Rard de Leeuw, Jürgen Müller, Henk Obbink, Ben Pronk and Jan Statius Muller helped to shape, to improve the structure and to sharpen the contents of the article "Positioning the System Architecture Process". This intermezzo is based on the first sections of 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 2

What is a Process?



2.1 Introduction

We rely in this part heavily on the notion of a process. This intermezzo is defining "process" for the context of this book. We define "process", since this word is heavily overloaded in our daily world. We also discuss the relationship of processes with organizations and the drive for process improvement.

2.2 What is a process

We use process as an abstracted way of working. A process can be characterized the attributes shown in Figure 2.1

In [4] the following definition is given:

A process is an activity which takes place over time and which has a precise aim regarding the result to be achieved. The concept of a process is hierarchical which means that a process may consist of a partially ordered set of subprocesses.

This definition parallels the characterization above. It adds explicitly the potential hierarchical decomposition of the process itself.

The notion of a process can be seen as one step in an abstraction hierarchy, as shown in 2.2. The most abstract notion in this hierarchy is the "principle". A principle is a generic insight that can be used for many different purposes. An example of a principle is *decomposition*: Whenever we have something big, e.g.

Purpose What is to be achieved and why

Structure How will the goal be achieved

Rationale What is the reasoning behind this process

Roles What roles are present, what responsibilities are associated, what incentives are present, what are the criteria for these roles

Ordering What phasing or sequence is applied

Figure 2.1: Process Attributes

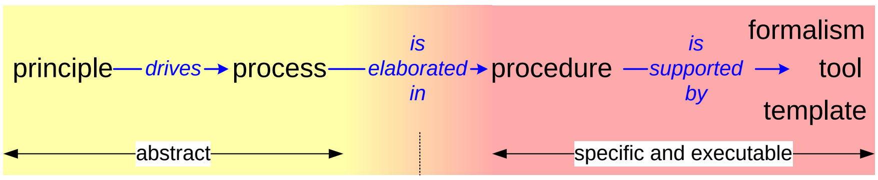


Figure 2.2: A process within an abstraction hierarchy

a problem or project, then we can decompose it in smaller pieces. These smaller pieces are easier to solve or create than the original big one.

A process is rather abstract. It describes the essentials of the purpose, structure, rationale, roles and timing, leaving plenty of implementation freedom. The power of a process is its abstraction, which enables its application in a wide range of applications, by tailoring its implementation to the specific application.

A process can be tailored and elaborated in one or more procedures that describe cookbook-like what needs to be done when and by whom. The why in a procedure has often disappeared, to be replaced by practical information for the execution.

The implementation of a procedure can be supported by tools, notations, templates and other means.

In practice managers and employees ask for tools (means) and procedures (what and how). However, without understanding of the thinking behind the procedure (why), as given in the process, these tools and procedures can be meaningless. The process captures the rationale behind procedures, tools, notations, templates, and other means.

2.3 The relation between Processes and Organizations

Traditional management is focused on “organizations”. Where organization are characterized by the attributes shown in Figure 2.3.

What **functions** are needed?

Who is **responsible** for this function?

What is the **hierarchical relation** between the functions?

What **meeting structure** is required?

Figure 2.3: Organization Attributes

This management views is insufficient in today’s fast moving complex world. The weak spots of the organizational view are shown in Figure 2.4.

Many activities cut arbitrarily through the 1-dimensional hierarchy, causing

- lack of ownership, unclear responsibilities
- high impedance transitions at organizational boundaries

Functions are a combination of tasks, where, in most cases, no human exists with the required skills

Meeting structures are insufficient and inefficient to get things done

Figure 2.4: Weaknesses of the organizational view

Processes are more modern instruments for management. Many processes are required to ensure the effective functioning of an organization. These processes are interrelated and overlapping. Processes are non-orthogonal and don’t fit in a strict hierarchical structure.

Most complex product developments don’t fit in the classical hierarchical organization model, but require a much more dynamic organization model, such as the currently popular more chaotic network organization. Processes are the means which help to ensure the output of dynamic organization models such as a network organization.

Processes can be seen as the blueprint for the behavior of the people within the organization. People will fulfill multiple roles in multiple processes. The process description is intended to give them an hold on what is expected from them.

All important activities will be covered by a process, requiring the definition of ownership, relation with other processes et cetera. The allocation of roles to people is much more dynamic than in conventional hierarchies. More dynamic allocation enables a better match between personal capabilities and required skills. In practice dynamic allocation leads to more distribution of responsibilities, making it more feasible to match capabilities and skills.

The 80/20 rule is also valid for processes: 80% of the behavior is covered by the processes, while 20% requires independent creative behavior. An organization without processes drowns in chaos, while an organization which blindly implements them will be killed by its own inertia, its inability to adapt to the fast changing world.

For reasons of continuity and stability an hierarchical organization will remain. The slowest evolving dimension is mostly used as a basis for this hierarchy. This hierarchy functions as anchor point for people in the continuously changing process world, but should play only a minor role in the entire operation.

The **Centurion** turn around operation within Philips, orchestrated by CEO Jan Timmer in the early nineties, urged the Philips managers and employees to change from an introvert organization point of view to an external result oriented process point of view.

2.4 Process Improvement

Urged by competitive pressure organizations look for ways to improve their efficiency. Many opportunities for improvement have a strong process component.

The 7S model by McKinsey gives a practical way to improve an organization in a balanced way. The message behind this model is that at least 7 views must be balanced when changing an organization. See Figure 2.5 for the 7 views.

The most common pitfall in improvement programs is the over-emphasis on the process component, or worse the isolation of the process improvement. Organizations assessing their maturity level, for instance by Maturity Models [6], quite often get too much process focus. The Process Improvement Officer[1] is focused on process issues only. Hence where the process view is introduced as an extrovert result oriented approach, it suddenly turns into an introvert improvement program, where business goals and drivers are unknown.

This is a quite sad situation: The opportunities for improvement are ample with a strong process component, however due to the wrong focus a negative effect is obtained (such as rigid procedures).

Recommendation: Process improvements should originate from the directly involved people, for instance project leaders, engineers, architects et cetera. Invite participation by this group in such a way that they feel the ownership.

[1] The existence of this function in itself is quite dangerous, it invites the unbalanced isolated "improvement" behavior

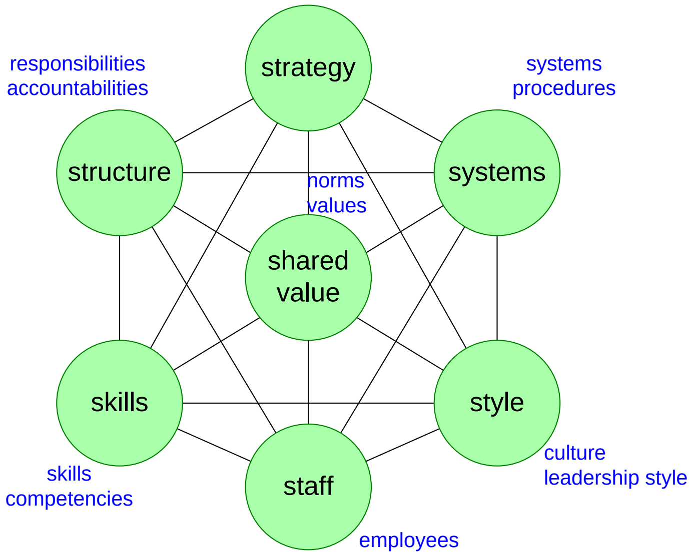


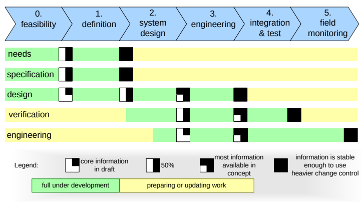
Figure 2.5: McKinsey 7S model

2.5 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.

Chapter 3

The Product Creation Process



3.1 Introduction

The Product Creation Process described how an organization gets from a product idea to a tested system and all product documentation that is required for the Customer Oriented Process. System Architects spend most of their time in the Product Creation Process. This chapter describes the Product Creation Process, including organizational aspects and the roles of people within the process.

3.2 The Context of the Product Creation Process

Figure 1.1 shows the context of the Product Creation Process in the decomposition of the business in 4 main processes. From Product Creation Process point of view the Policy and Planning Process determines the charter for the Product Creation Process. The Technology and People Management Process supplies people, process and technology enabling the Product Creation. The Customer Oriented Process is the customer: it receives and uses the results of Product Creation.

The Product Creation Process has a much wider context than the conventional "Research and Development" or "Development and Engineering" departments. The Product Creation Process includes everything that is needed to create a new product, for instance it includes:

- Development of the production process

- Design of the logistics flow and structure
- Development of required services
- Market announcement
- Market introduction

In other words the Product Creation Process is a synchronized effort of nearly all business disciplines within a company.

The term Product Creation is not only used for the development of entirely new products, but applies also to the development of variations of existing products or the development of upgrades or add-on products. The implementation of the Product Creation Process can vary, depending on the product being developed; a small add-on product will use a different organization than the development of a large new complex product.

3.3 Phases of the Product Creation Process

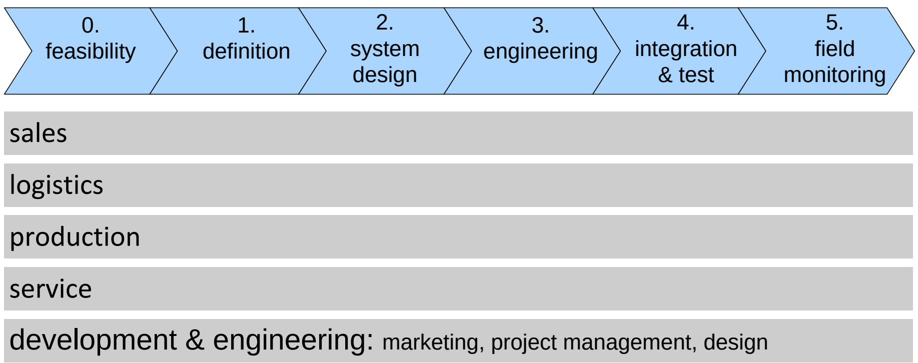


Figure 3.1: A phased approach of the Product Creation Process, showing the participation of all disciplines during the entire process

The Product Creation Process can be structures by using a phased approach. Figure 3.1 shows the phases as used in this book. The figure shows the participation of all business disciplines during this process.

These phases are used across all business functions which have to participate in the Product Creation Process. It is a means to manage the relations between these functions and to synchronize them. Note that sales, production, logistics and service people are involved in the Product Creation Process. Their participation is required to understand the needs of the Customer Oriented Process. A good understanding of these needs is required to develop the new procedures and processes for the customer oriented process, such as ordering, manufacturing, and installation.

Figure 3.2 zooms in on the expected progress for the design deliverables. We use the term work flows for the horizontal classes of activities: *needs analysis*, *product specification*, *design*, *verification and validation*, and *engineering*. Note that needs analysis, product specification, and design progress concurrently. Also note that the first review typically takes place long before any of the work flows is complete. The main question for the first review is: does it make sense to invest in the later phases?

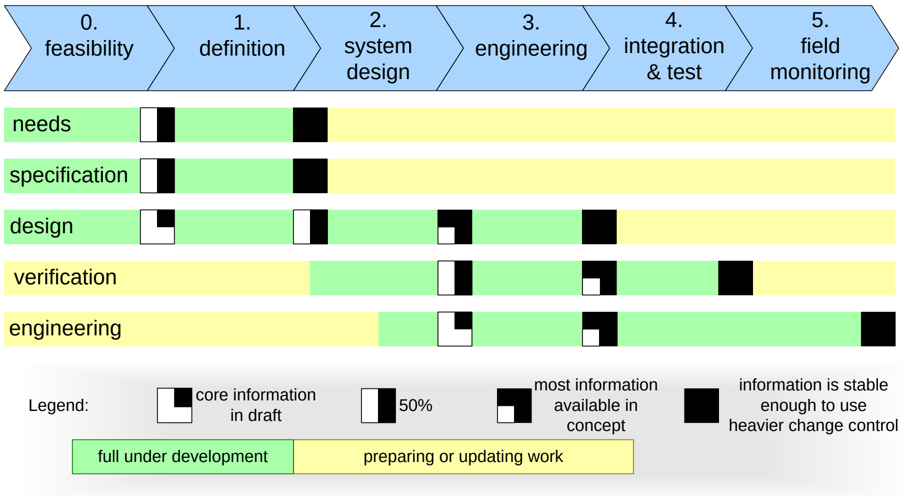


Figure 3.2: A phased approach of the Product Creation Process, showing the progress of the different design deliverables

The advantages of a phased approach are shown in Figure 3.3. The project members get guidelines from the phase model,: *who* does *what* and *when*. At the same time the check lists per phase provide a means to check the progress for the management team. The main risk is the loss of common sense, where project members or management team apply the phase model too dogmatic.

Customization of the phase model to the specific circumstances is always needed. Keep in mind that a phased process is only a means.

The phase process is used as a means for the management team to judge the progress of the Product Creation Process. That can be done by comparing the actual progress with the checklists of the phase model, at the moment of a phase transition. The actual progress is measured at the moment of transition. Normally the development will continue after the phase review, even if some deliverables are behind schedule. In that case the problem is identified, enabling the project team to take corrective action. Some management teams misinterpret the phase transition

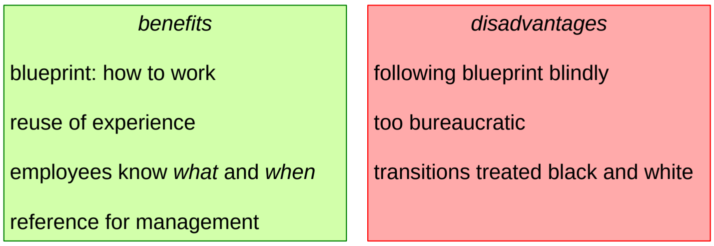


Figure 3.3: Advantages and Disadvantages of a phased approach

as a milestone with mandatory deliverables. Based on this misinterpretation the management team might demand full compliance with the checklist, disrupting the project. This kind of interference can be very counterproductive. See section 3.5 for a better management method with respect to milestones.

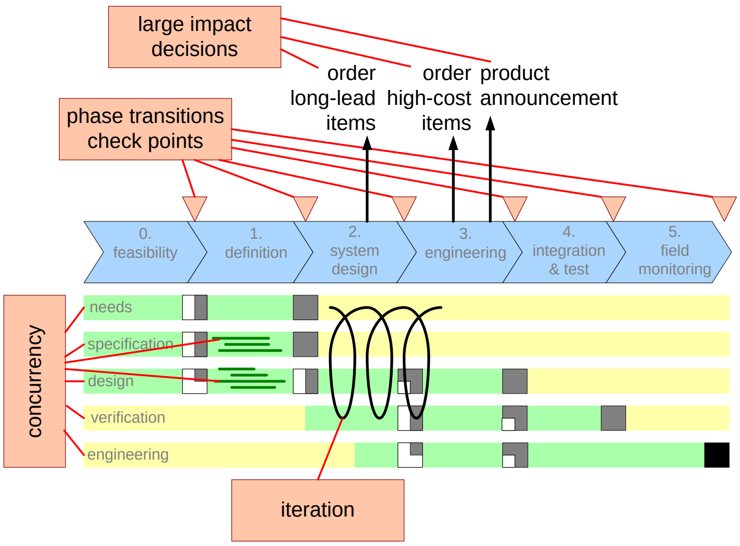


Figure 3.4: Characteristics of a phase model

Important characteristics of a phase model are shown in Figure 3.4:

Concurrency of need analysis, specification, design, and engineering, and concurrency between activities within each of these work flows.

Checkpoints at phase transition. Often more checkpoints are defined, for instance halfway a phase.

Iteration over the work flows and over activities within the work flows.

Large impact decisions that have to be taken, often long before the full consequence of the decisions can be foreseen.

3.4 Evolutionary models for Product Creation

The phase model stresses and supports concurrent activities, see also [3]. A common pitfall is a waterfall interpretation of a phased approach. Following a strict top-down approach can be a very costly mistake, because feedback from implementation and customers is in that case too late in the process. Early and continuous feedback both from implementation as from customer point of view is essential, see Intermezzo 4.

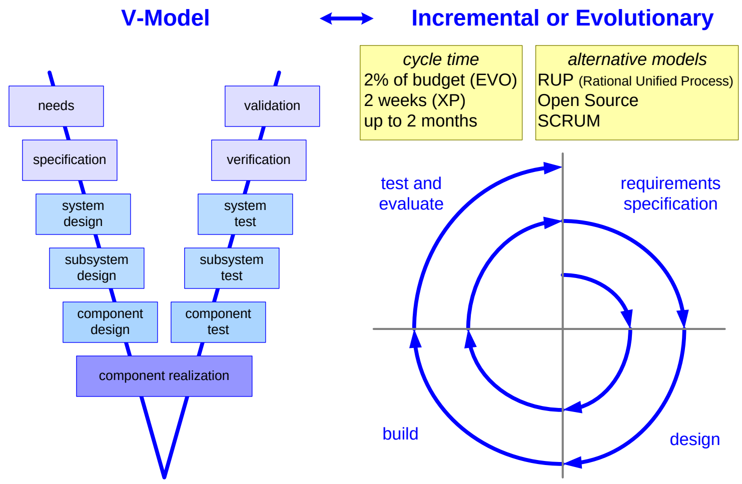


Figure 3.5: V-model versus Incremental or Evolutionary development models

High market dynamics exposes one weakness of the phased approach: market and user feedback becomes available at the end of the creation process. This is a significant problem, because most product creations suffer from large uncertainties in the specifications. Discovering at the end that the specifications are based on wrong assumptions is very costly.

Figure 3.5 show the V-model and evolutionary model side by side. Evolutionary methods focus on early feedback creation. EVO [2] by Gilb recommends to use evolutionary development steps of 2% of the total development budget. In every step some product feedback must be generated. Extreme Programming (XP) [1] by Beck is based on fixed duration cycles of two weeks. XP requires additional customer value in every increment.

The class of agile product creation approaches is struggling with the architecting process. The leaders of these communities dislike the “big design up-front”.

However, running in a treadmill of small increments may cause the loss of the "big picture". Architecting and short cycles, however, are not in conflict. The architecture also has to grow in incremental steps.

3.5 Milestones and Decisions

The project team is faced with a limited number of large impact decisions during the creation process. The decision in general engage the organization with a commitment somewhere in the future. For example:

Ordering of long lead items where changes in specification or design might obsolete ordered items. Re-ordering will cause project delay. Using the initially ordered items might decrease system performance.

Ordering of expensive materials where changes in plan, specification or design might obsolete the ordered materials.

Product announcement can not be reversed once the outside world has seen the announcement. Note that announcing a new product often impacts the order intake of existing products. Announcing a new product late might cause competitive risks.

Define a minimal set of *large-impact* decisions.

Define the mandatory and supporting information required for the decision.

Schedule a decision after the appropriate phase transition.

Decide explicitly.

Communicate the decision clearly and widely.

Figure 3.6: How to deal with large impact decisions

An explicit decision can be planned as a milestone in the project master plan. Information should be available to facilitate the decision: some of the information is mandatory to safeguard the company, some of the information is only supportive. In general the mandatory information should be minimized to prevent a rigid and bureaucratic process, causing the company to be unresponsive to the outside world. These decisions can be planned after the phase transition when most of the required supportive information will be available in an accessible way. Figure 3.6 shows the recommendations how to deal with large impact decisions.

3.6 Organization of the Product Creation Process

The Product Creation Process requires an organizational framework. The organizational framework of the Product Creation Process is independent of the Organizational frameworks of the other processes[1]

3.6.1 Hierarchical decomposition

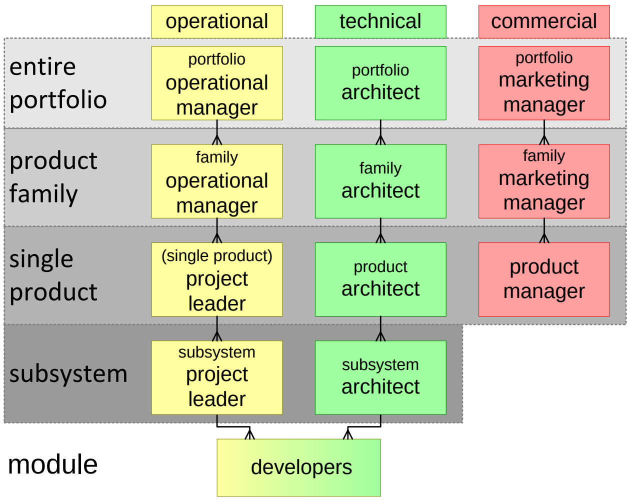


Figure 3.7: The simplified hierarchy of operational entities in the Product Creation Process form the core of this process.

The operational organization is a dominant organizational view on the Product Creation Process. In most organizations the operations of the Product Creation are decomposed in multiple hierarchical levels, at the highest level the entire product portfolio at the lowest level the smallest operational entity for instance a subsystem. Note that in figure 3.7 the hierarchy stops at subsystem level, although for large developments it can continue into even smaller entities like components or modules. The hierarchy is simply the recursive application of the decomposition principle.

Figure 3.7 is simplified by assuming that a straight forward decomposition can be applied. This assumption is not valid when lower level entities, e.g. subsystems, are used by multiple higher level entities, e.g. systems. For instance, if one subsystem is used in different products. In Chapter **??** we elaborate this aspect further.

[1] Quite often a strong link is present between People and Technology Management Process and the Product Creation Process; Using similar frameworks can be quite counterproductive, because these processes have quite different aims and characteristics. Of course, nearly all people are part of both organizational frameworks.

3.6.2 Further decomposition of the Product Creation Process

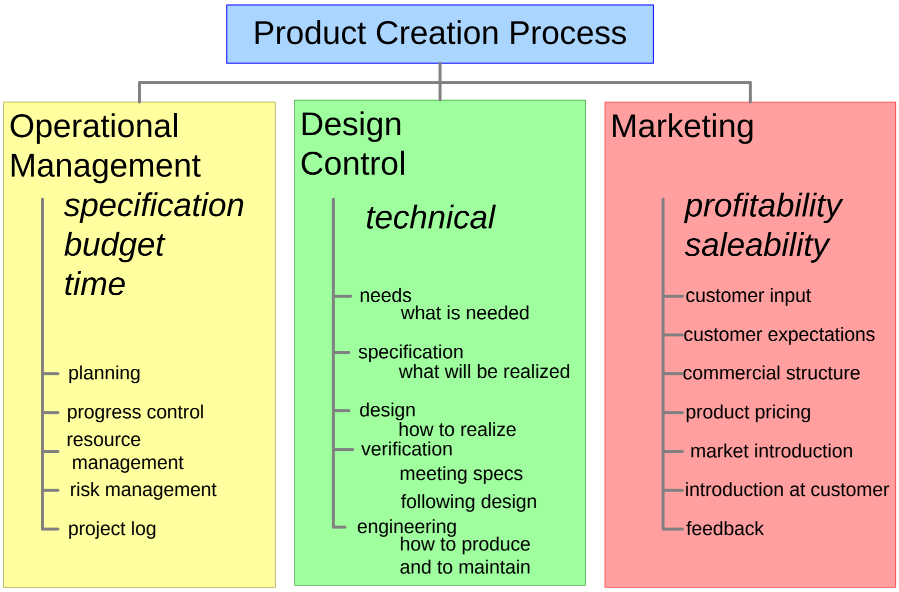


Figure 3.8: Decomposition of the Product Creation Process

The Product Creation Process can be decomposed in 3 processes as shown in 3.8:

Marketing: Defining how to obtain a sellable profitable product, starting with listening to customers, followed by managing the customer expectations, introducing the product at the customer and obtaining customer feedback.

Project Management: Realizing the product in the agreed triangle of

- specification
- resources
- amount of time

Design Control: Specifying and designing the system. The Design Control Process is that part of the Product Creation Process that is close to the conventional R&D activities. It is the content part of the Product Creation Process.

The functions mentioned in figure 3.7 map directly on the processes in figure 3.8:

- The *operational* or *project leader* is responsible for the *operational management*
- The *architect* is responsible for the *design control*
- The *marketing* or *product manager* is responsible for the *commercial* aspects

3.6.3 Design Control

The ISO 9000 standard has a number of requirements with respect to the *design control* process. The design control process is a core content oriented process, it is the home base of the system architect. The system architect will support the project management and the commercial process.

The design control process itself is further decomposed, also shown in figure 3.8:

- Needs
- Specification
- Design
- Engineering
- Verification

The needs express what the stakeholders of the system need, not yet constrained by business or technical considerations. Most development engineers tend to forget the original needs after several iterations of commercial and technical trade-offs.

The specification describes what will be realized, in terms of functionality and performance. This specification is the agreement with all stakeholders. The difference between the needs and the specification is that in the specification all trade-offs have been made. See also Chapter **??** where we elaborate the process of needs analysis and requirements management.

The design is the description how the specification will be realized. For instance, the physical and functional decomposition and the budgets for critical technical resources belong to the design.

Needs, specification and design are documented in development documents. The main function of these documents is to streamline the Product Creation Process. During this process these are living documents fulfilling an important communication function, while at the same time they play an important role in the control aspect of the design process.

The verification process verifies that the implementation meets the specification in the way it is specified in the design.

The engineering process provides the foundation upon which the Customer Oriented Process works for the entire life-cycle of the product. The documentation generated in the engineering process is the output of the Product Creation Process.

3.6.4 Operational Management

The operational management is governed by a simple set of rules, see Figure 3.9. These rules combine a number of very tightly coupled responsibilities in one function,

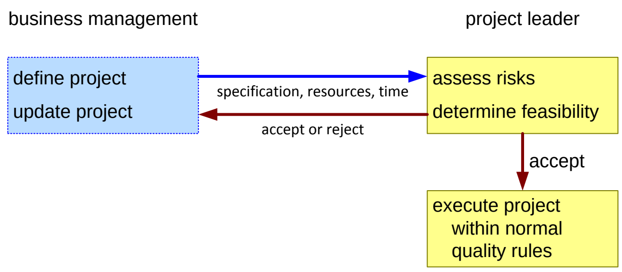


Figure 3.9: Commitment of the operational leader to the project charter

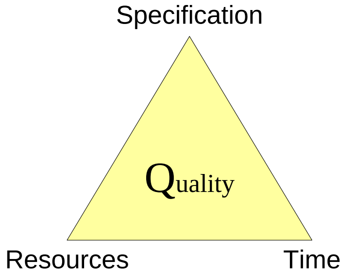


Figure 3.10: The Operational Triangle of responsibilities; The operational leader commits to the timely delivery of the specification within the agreed budget, with the "standard" quality level

to enable a dynamic balancing act by the operational leader. These responsibilities form the operational triangle as shown in figure 3.10.

The rules ensure that the operational leader takes ownership of the timely delivery of the specification within the agreed budget, with the "standard" quality level. Transfer of one of these responsibilities to another person change the system in an open loop system[2].

3.6.5 Marketing

The marketing manager knows the market: who are potential customers, what are their needs, what is of value in the market, what are commercial partners, what is the competition. This knowledge is "future" oriented and is used to make choices for future products. What are feasible products, what are the features and perfor-

[2] Many conventional development organizations have severe problems with this aspect. The most common mistake is that either the quality responsibility or the resource(budget) responsibility is transferred to the People and Technology Management Process. The effect is that excuses are present for every deviation of the commitment, for instance *I missed the timing because the people were working on non project activities.*

mance figures for these products, based on choices where value and cost are in a healthy balance. Hence the marketing manager is involved in packaging and pricing of products and options. A good marketing manager looks broader than the current products. Most innovations are not "more of the same", but are derived from new opportunities, technical or in the application.

Note that most sales managers are much more backward oriented: we sell what we have to customers who know existing systems. Good sales persons are often not good marketing persons!

3.6.6 Product Creation Teams

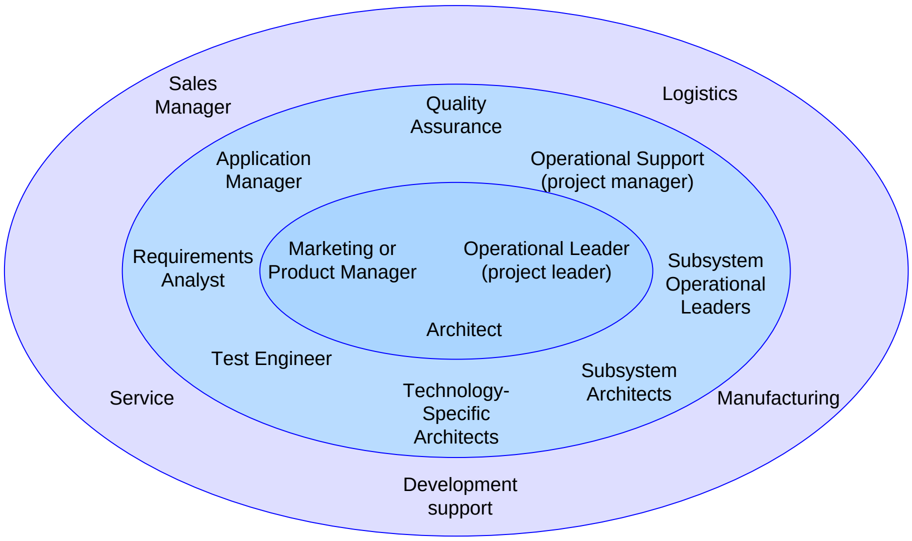


Figure 3.11: The operational teams managing the Product Creation Process

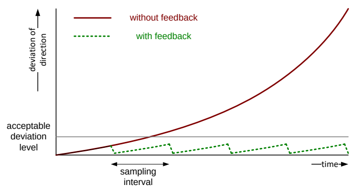
So far we have discussed *Operational management*, *Design Control* and *Marketing*. However, in the Product Creation Process more specialized functions can be present. Figure 3.11 shows a number of more specialized functions as part of a number of concentric operational teams. The amount of specialization depends on the size of the operation. In very small developments none of the specializations exist and is even the role of project leader and architect combined in a single person.

3.7 Acknowledgements

Rahim Munna suggested to add a short description of Marketing.

Chapter 4

The Importance of Feedback for Architecture



4.1 Introduction

Feedback is a universal principle that is applied in highly technical domains such as control engineering, but also in social sciences. This Intermezzo discusses feedback as part of the Systems Architecting Process and explains its importance.

4.2 Why Feedback?

4.2.1 Control

Feedback is used in control systems to ensure that the actual direction corresponds to the desired direction. In general the deviation from the desired direction grows exponentially in time, see Figure 4.1.

Many control systems implement a feedback loop to force the system back in the desired direction. Figure 4.1 also shows the effect of a discrete feedback system over time. It will be clear that the sampling interval is determined by the time constant of the deviation and the acceptable deviation level.

Product development can be seen as an ordinary system that can be controlled analog to technical control systems. Product developments without feedback result

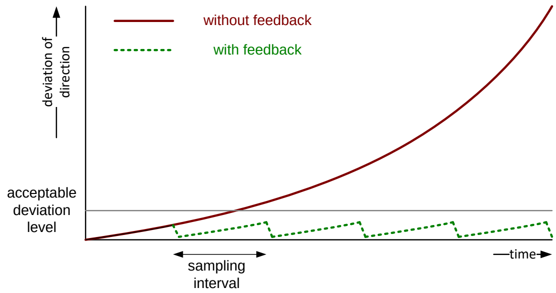


Figure 4.1: The deviation of the actual direction of product development with respect to the desired direction as function of the time

in products that are out of specification (too late, too slow, too expensive, too heavy et cetera). Sound development processes contain (often multiple) feedback loops.

4.2.2 Learning

Human beings learn from their mistakes, *provided that they are aware of them*. Feedback is the starting point of the learning process, because it provides the detection of mistakes. Efficiency of individuals and organizations can be increased by learning. Without learning similar mistakes are repeated: a waste of resources.

4.2.3 Applicability

The principle of feedback can be applied on **any** activity. The higher the uncertainty or the larger the duration of an activity is, the more important feedback becomes.

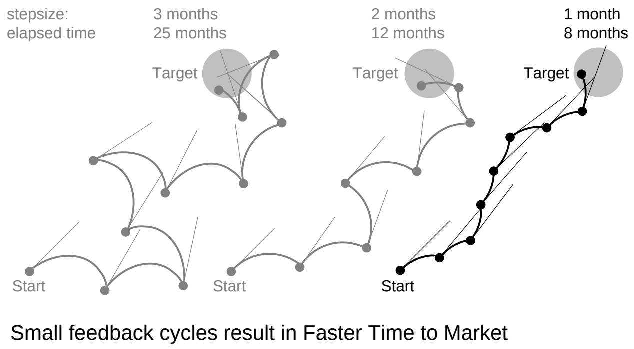


Figure 4.2: Example with different feedback cycles (1, 2, and 3 months) showing the time to market decrease with shorter feedback cycles

Figure 4.2 shows an example of a development with three different feedback cycle times, respectively three, two, and one months. The three month feedback cycle results in an project duration of 25 months. Decreasing the feedback cycle to 2 months brings the total project duration down to 12 months. One month feedback cycles give a total time of only 8 months. This simple model ignores the cost of obtaining feedback, but it clearly illustrates the essence of short feedback cycles.

4.3 Theory versus Practice

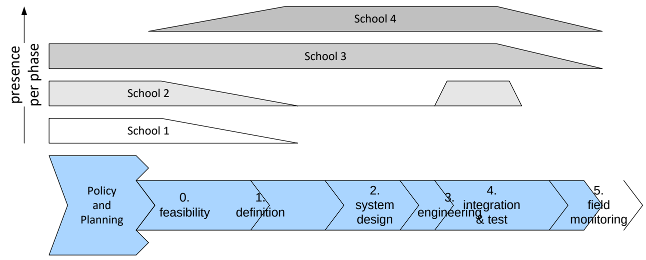


Figure 4.3: Four different schools of architecting, showing the presence of the architect in relation to the policy and planning process and the product creation process

Systems architecting is partially a very conceptual activity. The concepts are theoretical as long as they are part of presentations or specifications. Some architecting schools promote the system architecting function as strategic, providing direction, without being drowned in operational problems. A second school promotes an architect who is active in the definition phase of a product as well as in the verification phase. We argue a third direction: architecting has to be done during the entire development life cycle. In practice many architects function still in a fourth way: entirely in the technical domain. Figure 4.3 visualizes the 4 different schools as function of the process phase.

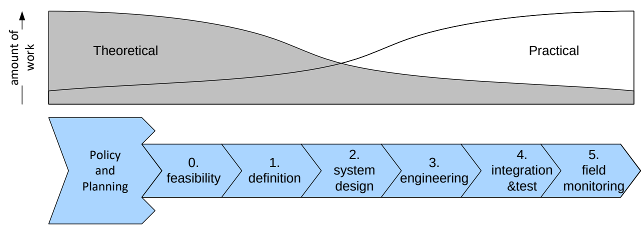


Figure 4.4: Theoretical versus Practical system architecture work in relation to the development life cycle

Figure 4.4 shows the amount of "theoretical" work and the amount of "practical" work also as function of the process phase. Where we use the term "theoretical" for concepts in presentations or specifications that have not been exposed to the physical world. Similarly, "practical" is used for work where the design is realized and tested.

A number of feedback loops can be closed during the Product Creation Process. Normally the next phase in the process provides feedback to the previous phase in the process. This phase transition feedback is often applied. However, feedback from the next phase is a rather indirect measure for the desired direction. The next step provides feedback on the usefulness of the input to continue the work, but the user satisfaction and market success can not be measured by the next step.

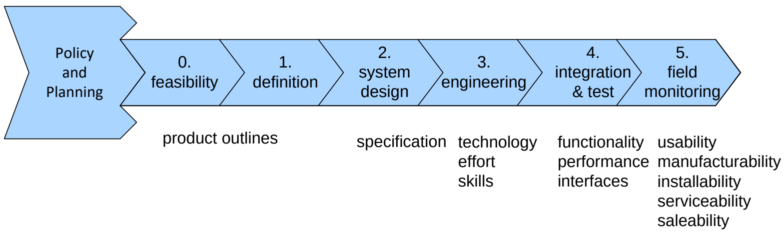


Figure 4.5: Feedback per development phase

The feedback for theoretical work comes from the practical work. Figure 4.5 shows the feedback per development phase. This figure makes it immediately clear that the amount of feedback is proportional to the amount of practical work going on.

4.4 Conclusions

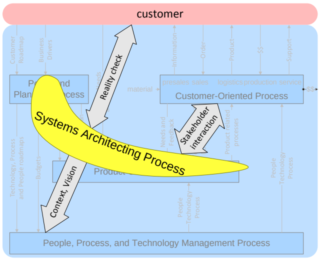
The conclusions of this paper are given here as a set of position statements:

1. For the education of system architects it is essential that they participate in the entire feedback loop.
2. The education of system architects is never finished.
3. System architects must participate in the entire product creation lifecycle for most of their carrier.
4. The value of system architects in the policy and planning process stems from the practical feedback during the product creation process.
5. Feedback can never come too early.

6. System architects can have fantastic dreams, feedback is required to prevent that dreams turn into a nightmares.

Chapter 5

The System Architecture Process



5.1 Introduction

This chapter positions the systems architecting process in a wider business scope. This positioning is intended to help understanding the process itself and the role of the system architect (or team of system architects).

We focus on systems architecting within organizations that create and build systems consisting of hardware and software. Although other product areas such as solution providers, services, courseware et cetera also need system architects, the process structure will deviate from the structure as presented here. See Intermezzo **??** for an elaboration of these other architecting models.

5.2 System Architecture in the Business Context

Figure 5.1 shows the main activities of the System Architecting Process as an overlay over the business decomposition.

Processes are goal oriented, as discussed in Intermezzo 2. The process decomposition is not orthogonal, several processes are overlapping. The System Architecting Process is a clear example of such non-orthogonality. Figure 5.2 shows a map of the System Architecture Process and neighboring processes. Many processes,

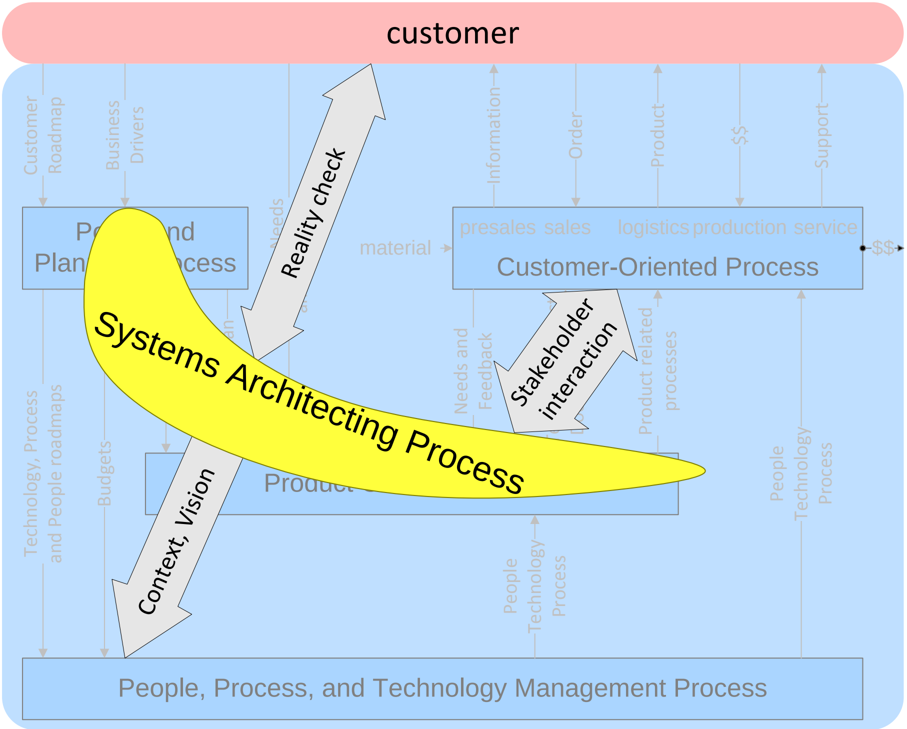


Figure 5.1: The main System Architecture activities in the Business Context

such as manufacturing engineering, service engineering, have been left out of the map, although these processes also have a high architecture relevance.

Both figures make it clear that the System Architecting Process contributes heavily to the Product Creation Process, while it plays also an essential role in the Policy and Planning Process. Both contributions are strongly coupled, see figure 5.3

The System Architecture Process bridges the gap between Product Creation Process and the Policy and Planning Process. In many organizations this link is missing. The absence of this link results in:

- re-inventing a (different) product positioning during the Product Creation Process, with a limited context view
- policies which are severely handicapped by a lack of practicality or realism

The overview created by the System Architecting Process also helps in establishing a technology policy.

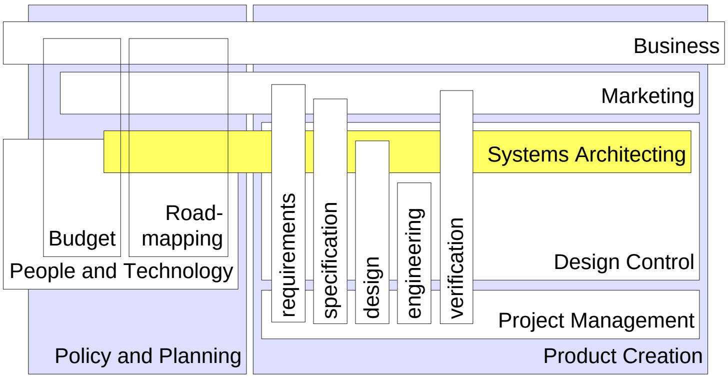


Figure 5.2: Map of the System Architecture Process and neighboring processes

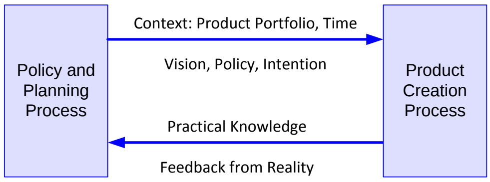


Figure 5.3: Contribution of System Architecting to the the Coupling of Policy and Planning Process and the Product Creation Process

5.3 Purpose of the System Architecting Process

Every business exceeding a few people enables the efficient concurrent work of these people by dividing the tasks in smaller more specialized jobs, the *decomposition principle* in action. This decomposition of responsibilities requires an opposing force integrating the activities in a useful overall business result. Several integrating processes are active in parallel, such as project management, commercial management et cetera.

The System Architecting Process is responsible for:

- the Integral Technical aspects of the Product Creation Process, from requirement to deployment.
- the Integral Technical Vision and Synergy in the Policy and Planning Process.

The System Architecting Process is striving for an optimal overall business result, by creating and maintaining the key issues, such as a balanced and consistent design, selection of the least complex solution, and satisfaction of the stakeholders.

The System Architecture Process is balancing amongst others:

- External and internal requirements

- Short term needs and long term interests
- Efforts and risks from requirements to verification
- Mutual influence of detailed designs
- Value and costs

Such a balance is obtained by making trade-offs, for example *performance* versus *qualities* versus *functionality*, or *synergy* versus *specific solution*

It is the purpose of the System Architecting Process to maintain the consistency throughout the entire system, from roadmap and requirement to implementation and verification. On top of this consistency the integrity in time must be ensured.

An enabling factor for an optimal result is *simplicity* of all technical aspects. Any unnecessary complexity is a risk for the final result and lowers the overall efficiency.

5.4 The System Architect as Process Owner

The owner of the System Architecting Process is the System Architect or the System Architecting Team. Many other people are involved in the System Architecting Process.

The system architect or the team members spent the majority of their time, about 80%, in the Product Creation Process. From the remaining time the majority is spent in the Policy and Planning Process. In 5.2 it is explained that these processes are strongly coupled. This coupling is for a large part implemented by employing the same people in both processes. A small amount of time is spent in People, Process, and Technology Management.

5.5 System Architecting in Product Creation Context

The Systems Architecting Process is striving for consistency and balance from requirement to actual product.

The amount of people working in product creation can vary from a few to tens of thousands of people. All people working on the creation of a new product have only knowledge of a (small) subset of the information. Inconsistencies and local optimal solutions pop up all the time, caused by lack of knowledge of the broader context.

The Systems Architecting Process has to prevent this natural degradation of the system quality. Systems Architecting acts pro-active by clear and sharp requirements, specification and system design as well as reactive by following up the feedback from detailed design, implementation and test.

During the Product Creation Process many specification and design decisions are taken. Quite often these decisions are taken within the scope of that moment. Consecutive decisions can be in contradiction with previous decisions. For instance, a decision is taken to add memory to the product to increase performance, while one month later the amount of memory is decreased to lower the cost. The Systems Architecting Process maintains the integrity over time, by looking at decisions from a broader perspective.

5.6 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 article is based on the last sections of this article. I am grateful for their contribution.

Jürgen Müller spotted hiccups in the flow of the new article, enabling a streamlining and extension of this article. Robert Deckers analyzed the text and pointed out many inconsistencies and poor formulations.

An inspiring presentation by Bud Lawson helped me to make a more complete and balanced list of System Architecture key issues.

Bibliography

[1] Kent Beck. *Extreme Programming Explained: Embrace Change*. Addison-Wesley, Reading, MA, 2000.

[2] Thomas Gilb. *Competitive Engineering: A Handbook For Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage*. Elsevier Butterworth-Heinemann, London, 2005.

[3] Ivar Jacobson, Grady Booch, and James Rumbaugh. *The Unified Software Development Process*. Addison-Wesley, Reading, MA, 1999.

[4] Klaus Kronlöf, editor. *Method Integration; Concepts and Case Studies*. John Wiley, Chichester, England, 1993. A useful introduction is given in Chapter 1, The Concept of Method Integration.

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

[6] Carnegie Mellon Software Engineering Institute SEI. Software engineering management practices. `http://www.sei.cmu.edu/managing/managing.html`, 2000.

History

Version: 1.1, date: December 15, 2018 changed by: Gerrit Muller

- replaced the assignment by a graphical version SEFASorganization
- changed status to concept

Version: 1, date: March 22, 2004 changed by: Gerrit Muller

- Created reader, no changelog yet

Version: 0, date: 24 April 2002 changed by: Gerrit Muller

- Created, no changelog yet