CTT Course System Architecting SARCH for management teams

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

Information about the course System Architecting for management teams.

January 22, 2023
status: planned
version: 0

logo
TBD

Course System Architecting for Management Teams

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

This article describes the condensed version the course System Architecture by the Center for Technical Training CTT. Trainer is the author of this article Gerrit Muller. At this moment this course is only accessible for Philips Employees.
The course is set up to make the art of system architecting more accessible. The course will address a wide spectrum of issues in relation with system architecture, such as: processes, business, role and task of the system architect (team), generic Developments (re-use, platforms) requirements, roadmapping, skills, and psycho social factors.

January 22, 2023
status: draft
version: 0.1
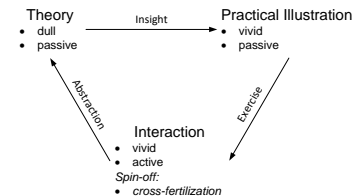
Program

session	subject
day 1 morning	positioning the System Architecture Process Product Creation Process product families, generic developments
day 1 afternoon	role and task of the system architect profile of the system architect documentation, reviewing and other supportive processes
day 2 morning	requirements capturing, roadmapping
day 2 afternoon	HRM aspects; selection, appraisal, career path, etcetera wrap up, expectations, how to continue, evaluation

Exercises

- 1 Make a map of the operational organization, from portfolio down to components, with specific products, names and roles; Discuss the relations in one of the core teams.

- 2 Role play, marketing manager + projectleader + system architect + observer; prepare initial product definition (=business relevance+specification+critical design issues+plan indication)

- 3 Determine Requirements and key drivers and show the relationship

- 4 Show the roadmap as far as known now

Structure

Theory
- dull
- passive

Insight →

Practical Illustration
- vivid
- passive

Abstraction

Exercise

Interaction
- vivid
- active

Spin-off:
- *cross-fertilization*

USN ESI
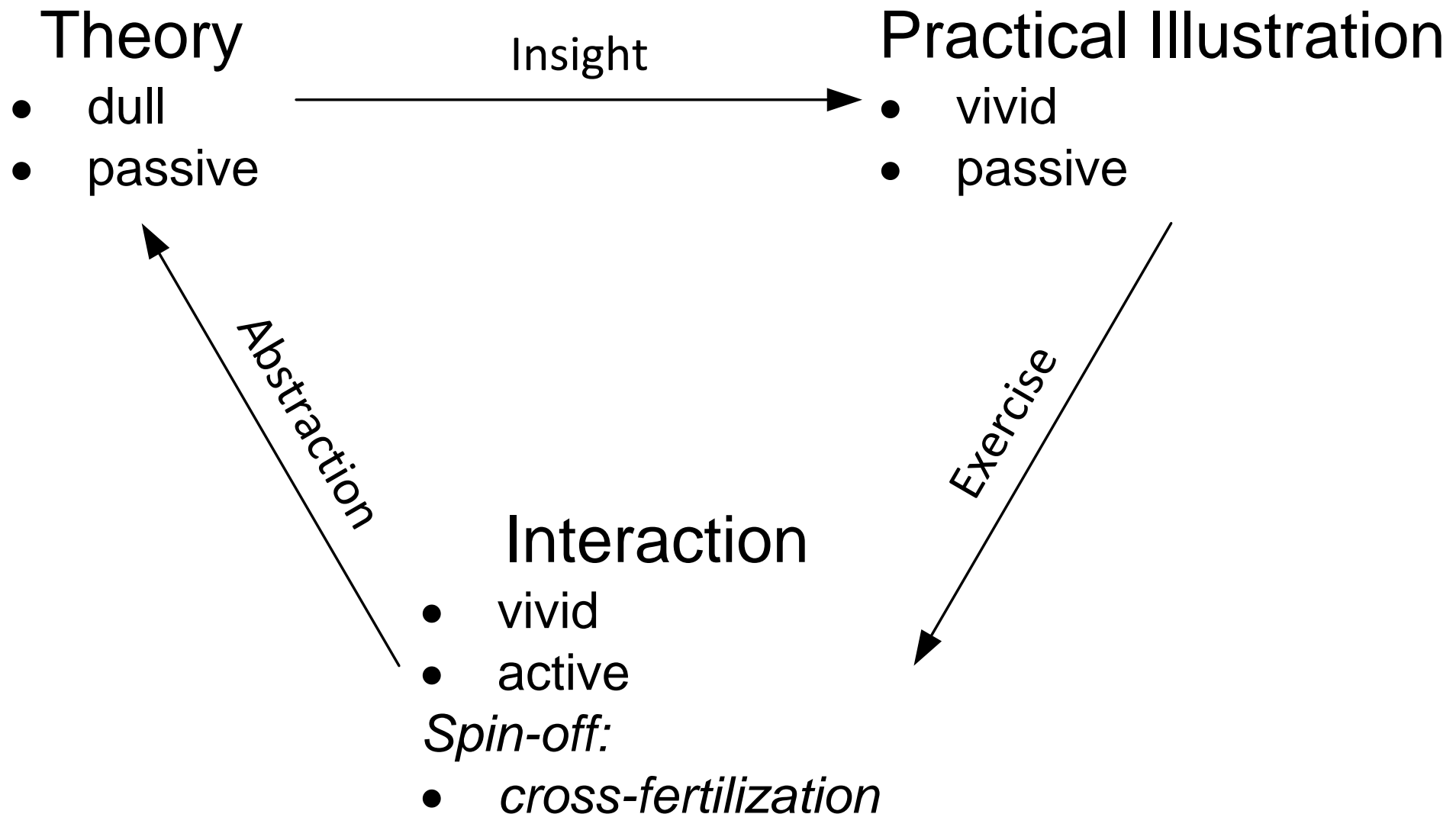
Timing Template of one subject

| Interactive Exploration | Broadcast | break | Broadcast | Interactive discussion | break | groupwork |

9:00   10:00   11:00   12:00   12:30

13:30   14:00   15:00   16:00   17:00

Rules of the Interactive Parts

- Your contribution is essential.
- Don't monopolize the time, everyone also the quiet people should have the opportunity to contribute;
  The facilitator will intervene if the contribution is limited to a small group of participants.
- Respect the contribution of others;
  Opinions can't be wrong, difference of opinion is normal and called pluri-formity.
- The course format is highly experimental and based on improvisation, constructive proposals are welcome;
  it is your course! Regular evaluations will give the opportunity to influence the rest of the course.

Rules of the Broadcast Parts

- Please write your questions/remarks/statements on yellow stickers and attach them at the end on the P-flip.
  These will be used in the interactive section for discussion and to increase insight.
- Short clarification questions are welcome,
  discussion will take place in the interactive part.
- Stupid questions don't exist. Learning is based on **safe** and **open** interaction.
  Very individual oriented questions can be referred to a break or after the session.

The Gaudí Project

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`
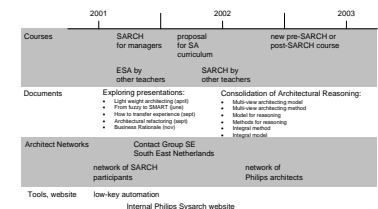
`www.gaudisite.nl`

Abstract

The Gaudí project is described. The goals of the project, the way of working, and an outline for the period 2001 to 2003. The deliverables in terms of documents are positioned by means of a two-dimensional map. Courses based on the Gaudí material are described. The current status of the courses is given.

January 22, 2023
status: draft
version: 3.1

Goals of the Gaudí Project

- **Consolidate existing Systems Architecting Methods**

  evaluate, reflect, generalize

- **Make the Systems Architecting art more accessible**

  case descriptions

- **Enable the education of (future) System Architects**

  curriculum, course material

- **Research new or improved Systems Architecting Methods**

  industry as laboratory

Modular approach
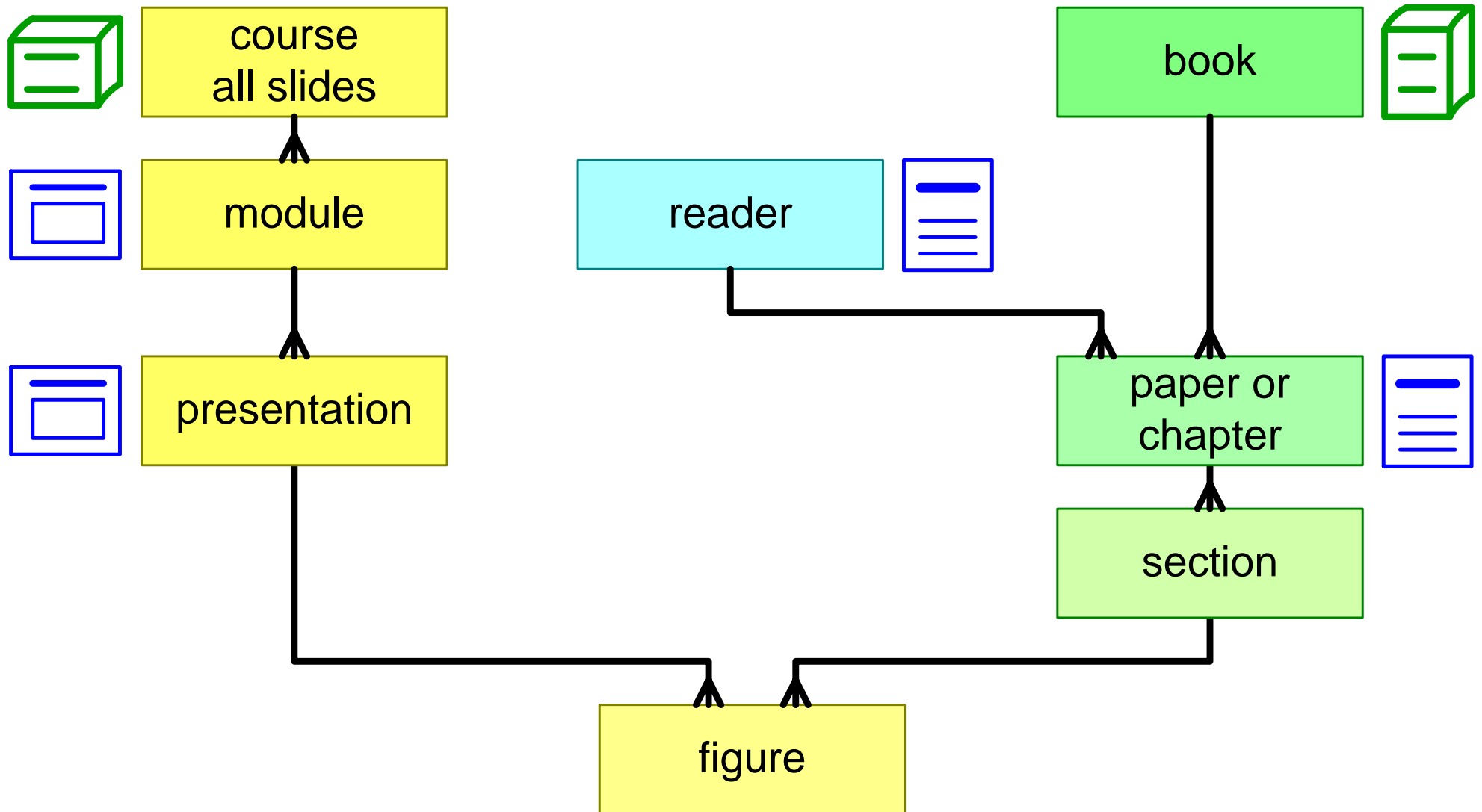
Show Early to Get Feedback

idea

planned

show idea

preliminary
draft

usable

draft

concept

leading principle:

stable

but still
alive

finished

show documents under construction
but clearly show their status

USN ESI
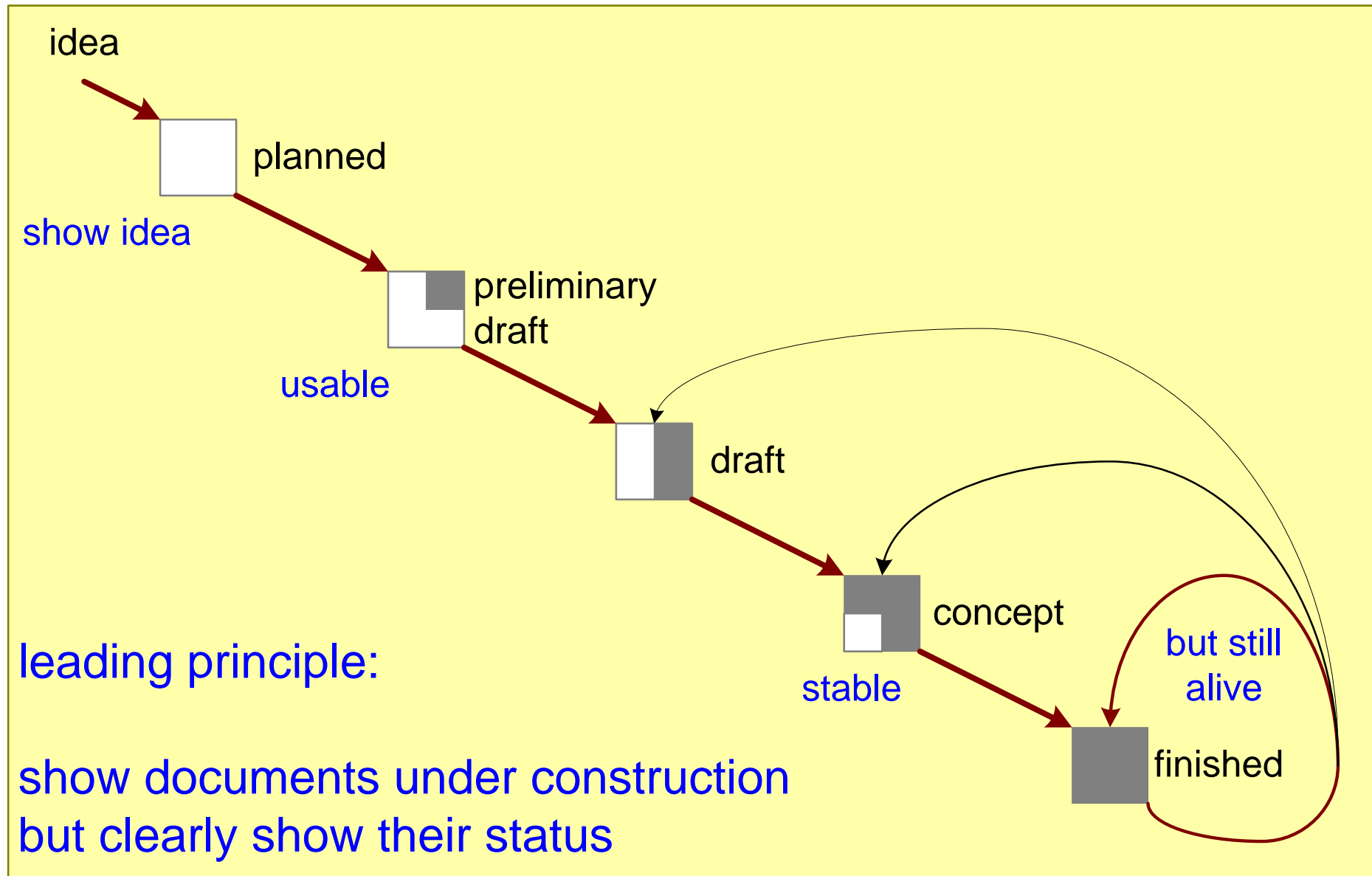
Growth of the System Architect

root technical knowledge → generalist technical knowledge → business, application insight / process insight → psychosocial skills

USN ESI

Positioning Courses

root technical designer

technical generalist

business, application insight

process insight

psycho-social interests

Gaudi courses

Performance
Architecting System Performance

Modeling
System Modeling and Analysis

CAFCR
Multi-Objective System Architecting and Design

SARCH
System Architecting

Platforms and Evolvability
Architecting Evolvable Product Families

USN ESI

Positioning Books

root technical designer → technical generalist → business, application insight / process insight → psycho-social interests

Gaudí books

- Architecting System Performance
- System Modeling and Analysis: a Practical Approach
- CAFCR PhD thesis
- CAFCR Architectural Reasoning
- Systems Architecting
- Supporting Processes
- Composable Architectures
- Human Measure

Productivity: number of new entries
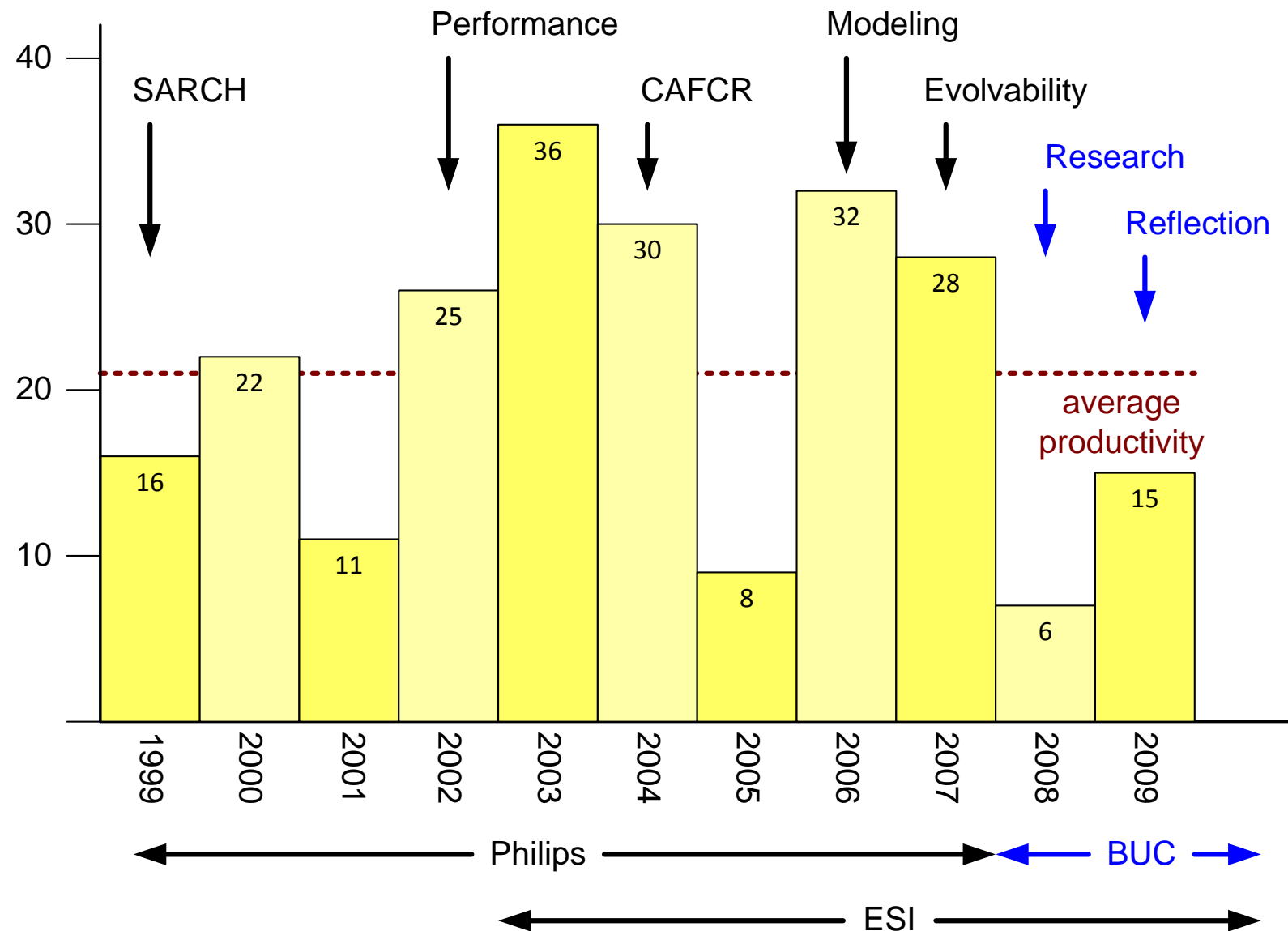
Module System Architecture Context

by *Gerrit Muller*   Buskerud University College and Buskerud University College

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

The system architecture process is positioned in a wider context: First in the business context, then in the Product Creation Process context.

January 22, 2023
status: concept
version: 1.1

Process Decomposition of a Business

by *Gerrit Muller*   USN-SE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

This article positions the system architecture process in a wider business scope. This positioning is intended to help understanding the processes in which the system architect (or team of system architects) is involved.

It focuses on an organization that creates and builds 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.

January 22, 2023
status: concept
version: 1.1
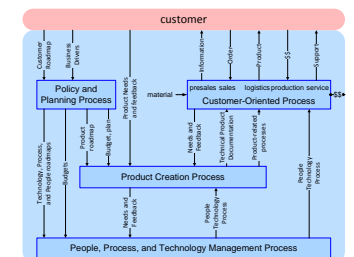
Simplified Decomposition of the Business

customer

Customer Roadmap → Policy and Planning Process

Business Drivers → Policy and Planning Process

Product Needs and feedback

Information

Order

Product

$$

Support

material → Customer-Oriented Process

presales sales logistics production service

Customer-Oriented Process → $$

Policy and Planning Process:
- Technology, Process, and People roadmaps
- Budgets
- Product roadmap
- Budget, plan

Product Creation Process

Needs and Feedback

Technical Product Documentation

Product-related processes

People Technology Process

Needs and Feedback

People Technology Process

People, Process, and Technology Management Process

USN ESI

Financial Characterization of Decomposition

Diagram showing a decomposition of business processes. The top pink box is labeled "customer". Below is a large blue region containing boxes labeled "Management" (Policy and Planning Process), "Cashflow Generation" (Customer Oriented Process), "Tomorrow's Cashflow", and "Assets" (People and Technology Management Process).

Arrows from customer into the region: Customer Roadmap, Business Drivers, Information, Order, Product, $$, Support.

Management connects with: Product roadmap, Budget, plan, Technology, Process, and People roadmaps, Budgets.

Cashflow Generation labels: presales, sales, logistics, production, service, material, $$.

Other flows: Product Needs and feedback, Needs and Feedback, Technical Product Documentation, Product-related processes, People Technology Process.

USN   ESI

Multiple Instances per Process

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 Value Chain and the Opposite Feedback Flow

customer

Customer Roadmap

Business Drivers

Information

Product

Support

Policy and Planning Process

Customer-Oriented Process

-$$

material

Product Needs and feedback

Product roadmap

Budget, plan

Technical Product Documentation

Product re... proces...

Feedback

Value

Product Creation Process

Technology, Process, and People roadmaps

Budgets

People Technology Process

Techno... Proce...

People, Process, and Technology Management Process

USN   ESI

Decomposition of the Customer Oriented Process

Extended with Generic Developments

The Product Creation Process

by *Gerrit Muller*   USN-SE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

The Product Creation Process is described in its context. A phased model for Product Creation is shown. Many organizations use a phased model as blueprint for the way of working. The operational organization of the product creation process is discussed, especially the role of the operational leader.

January 22, 2023
status: concept
version: 2.2

The Product Creation Process in Business Context

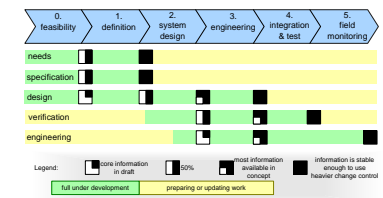**Customer**

Customer Roadmap

Business Drivers

Product Requirements and feedback

Information

Order

Product

$$

Support

Policy and Planning Process

presales sales logistics production service

material

Customer Oriented Process

$$

Technology, Process and People roadmaps

Budgets

Product roadmap

Budget, plan

Requirements and Feedback

Technical Product Documentation

Product related processes

People Technology Process

Product Creation Process

People Technology Process

People and Technology Management Process

USN ESI

Phasing of the PCP at Business Level

| 0. feasibility | 1. definition | 2. system design | 3. engineering | 4. integration & test | 5. field monitoring |

sales

logistics

production

service

development & engineering: marketing, project management, design

USN   ESI

Phasing the Design Control Process

Advantages and Disadvantages of a Phased Process

benefits	disadvantages
blueprint: how to work	following blueprint blindly
reuse of experience	too bureaucratic
employees know *what* and *when*	transitions treated black and white
reference for management	

USN ESI

Characteristics of a Phase Model

large impact
decisions

phase transitions
check points

order
long-lead
items

order
high-cost
items

product
announcement

| 0. feasibility | 1. definition | 2. system design | 3. engineering | 4. integration & test | 5. field monitoring |

concurrency

needs

specification

design

verification

engineering

iteration

USN ESI

Decisions and Phase Transitions

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.

Evolutionary PCP model

test and
evaluate

requirements
specification

**2% of budget (EVO)
2 weeks (XP)
up to 2 months
per cyclus**

build

design

Decomposition of the Product Creation Process

Product Creation Process

Operational Management

specification
budget
time

— planning

— progress control

— resource
 management

— risk management

— project log
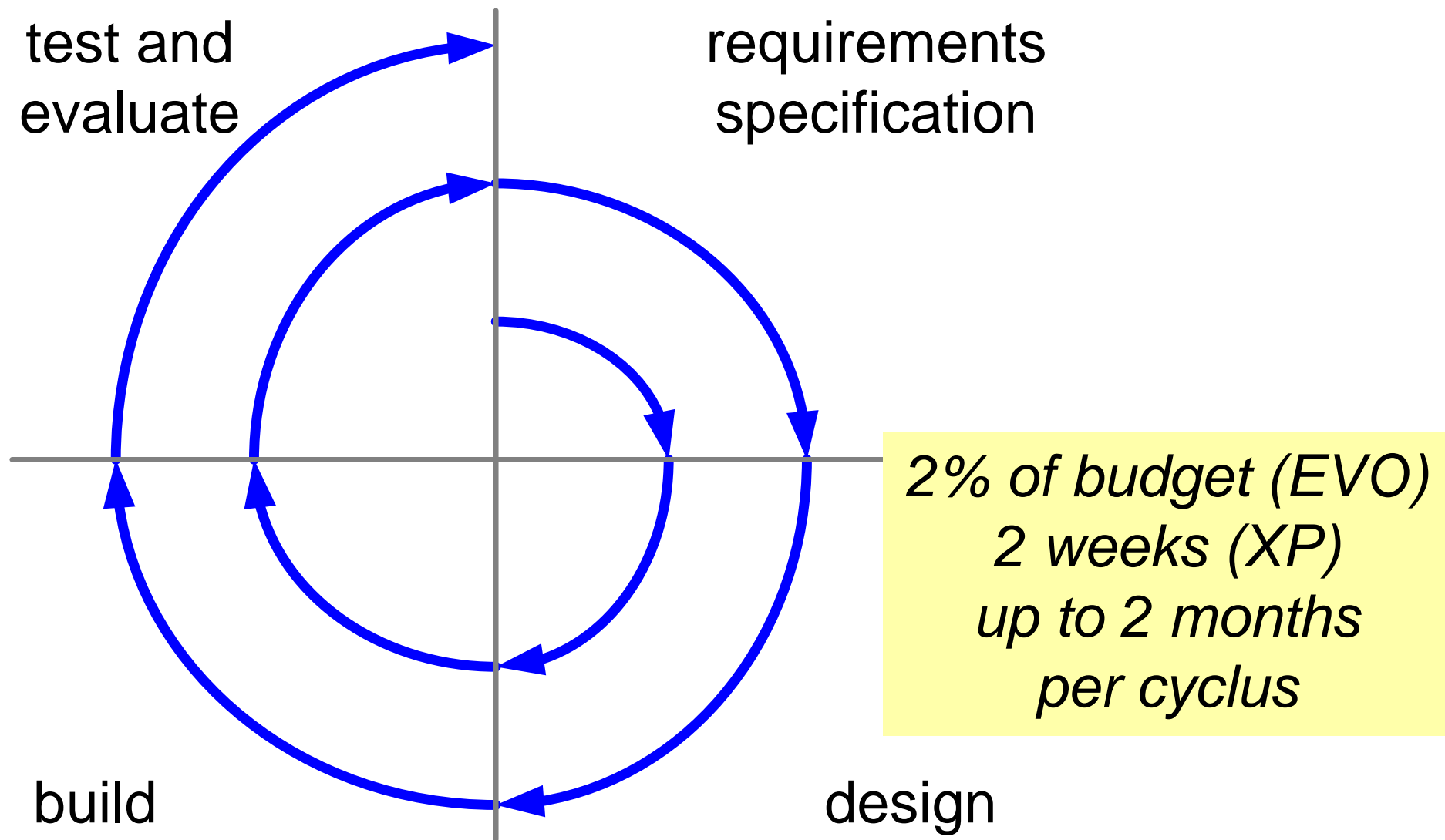
Design Control

technical

— needs
    what is needed

— specification
    what will be realized

— design
    how to realize

— verification
    meeting specs
    following design

— engineering
    how to produce
    and to maintain

Marketing

profitability
saleability

— customer input

— customer expectations

— commercial structure

— product pricing

— market introduction

— introduction at customer

— feedback

USN   ESI

Operational Organization of the PCP

	operational	technical	commercial
entire portfolio	portfolio **operational manager**	portfolio **architect**	portfolio **marketing manager**
product family	family **operational manager**	family **architect**	family **marketing manager**
single product	(single product) **project leader**	product **architect**	product **manager**
subsystem	subsystem **project leader**	subsystem **architect**	
module		developers	

USN ESI

Specification

Quality

Resources   Time

The Rules of the Operational Game

business management

project leader

define project

update project

specification, resources, time →

assess risks

determine feasibility

← accept or reject

accept ↓

execute project
within normal
quality rules

USN   ESI

Operational Teams

Sales Manager

Logistics

Quality Assurance

Application Manager

Operational Support (project manager)

Requirements Analyst

Marketing or Product Manager

Operational Leader (project leader)

Subsystem Operational Leaders

Architect

Test Engineer

Service

Technology-Specific Architects

Subsystem Architects

Manufacturing

Development support

The System Architecture Process

by *Gerrit Muller*   USN-SE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

The System Architecture Process is positioned in the business context. This process bridges the gap between the Policy and Planning Process and the Product Creation Process.
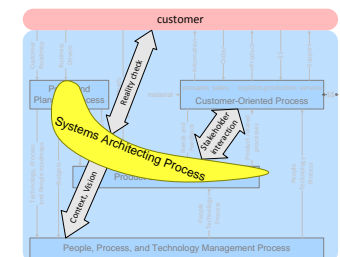
The purpose of the System Architecture Process is to provide the Integral Technical overview and consistency, and to maintain the integrity over time. Subjective characteristics as elegance and simplicity are key elements of a good architecture.

The scope of the system architecture process is illustrated by showing 5 views used in a reference architecture, ranging from Customer Business to Realization.
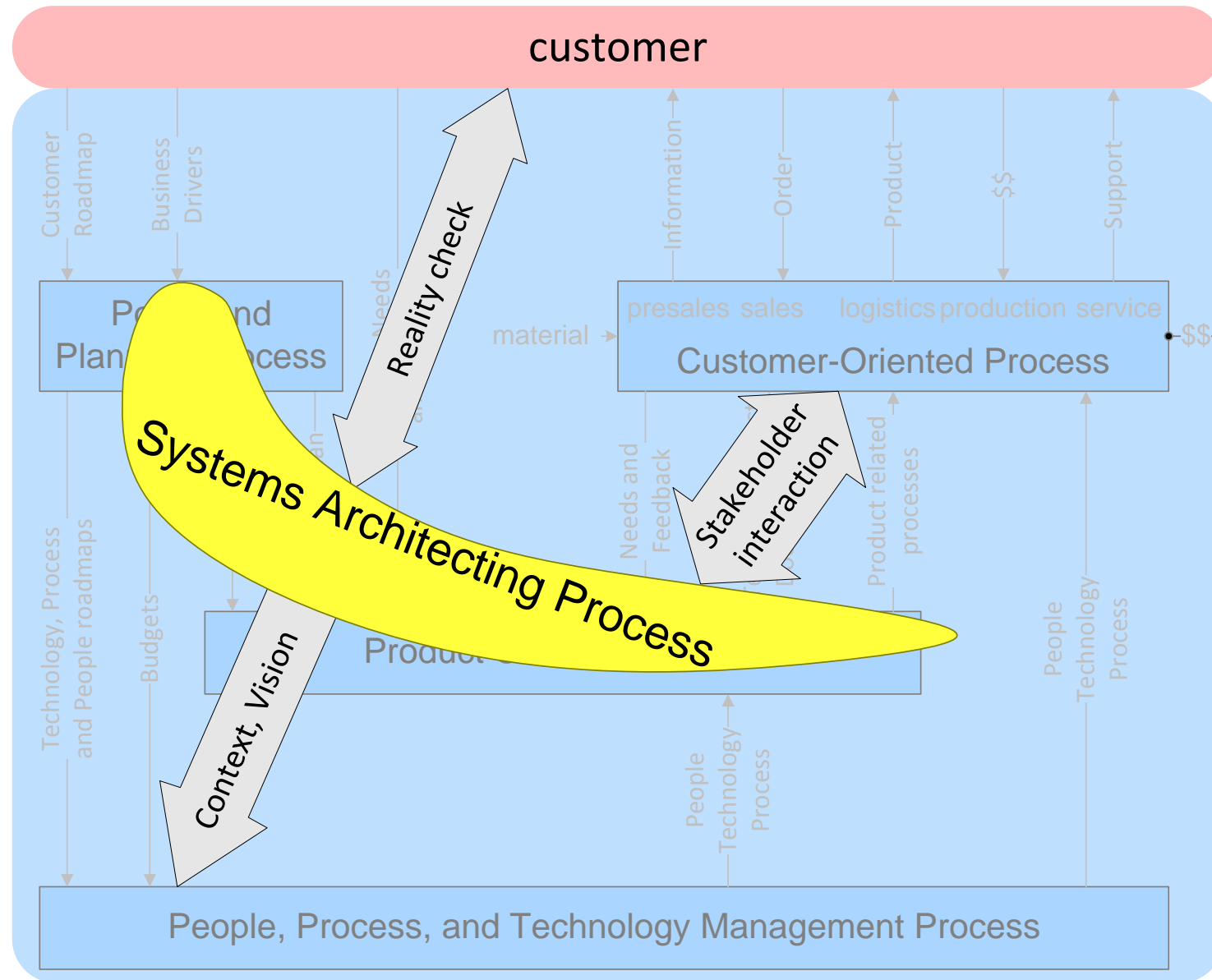
Distribution

January 22, 2023
status: concept
version: 2.3

System Architecting Process in Business Context

customer

Customer Roadmap
Business Drivers
Information
Order
Product
$$
Support

Po... nd Plan... ocess

presales sales   logistics production service

material →

Customer-Oriented Process

→ $$ →

Reality check

Needs and Feedback

Stakeholder interaction

Product related processes

Systems Architecting Process

Technology, Process and People roadmaps
Budgets

Product ...

Context, Vision

People Technology Process

People Technology Process

People, Process, and Technology Management Process

USN   ESI

Map of System Architecting Process and Neighborhood

Business

Marketing

Systems Architecting

Budget

Road-mapping

People and Technology

requirements

specification

design

engineering

verification

Design Control

Project Management

Policy and Planning

Product Creation

USN ESI
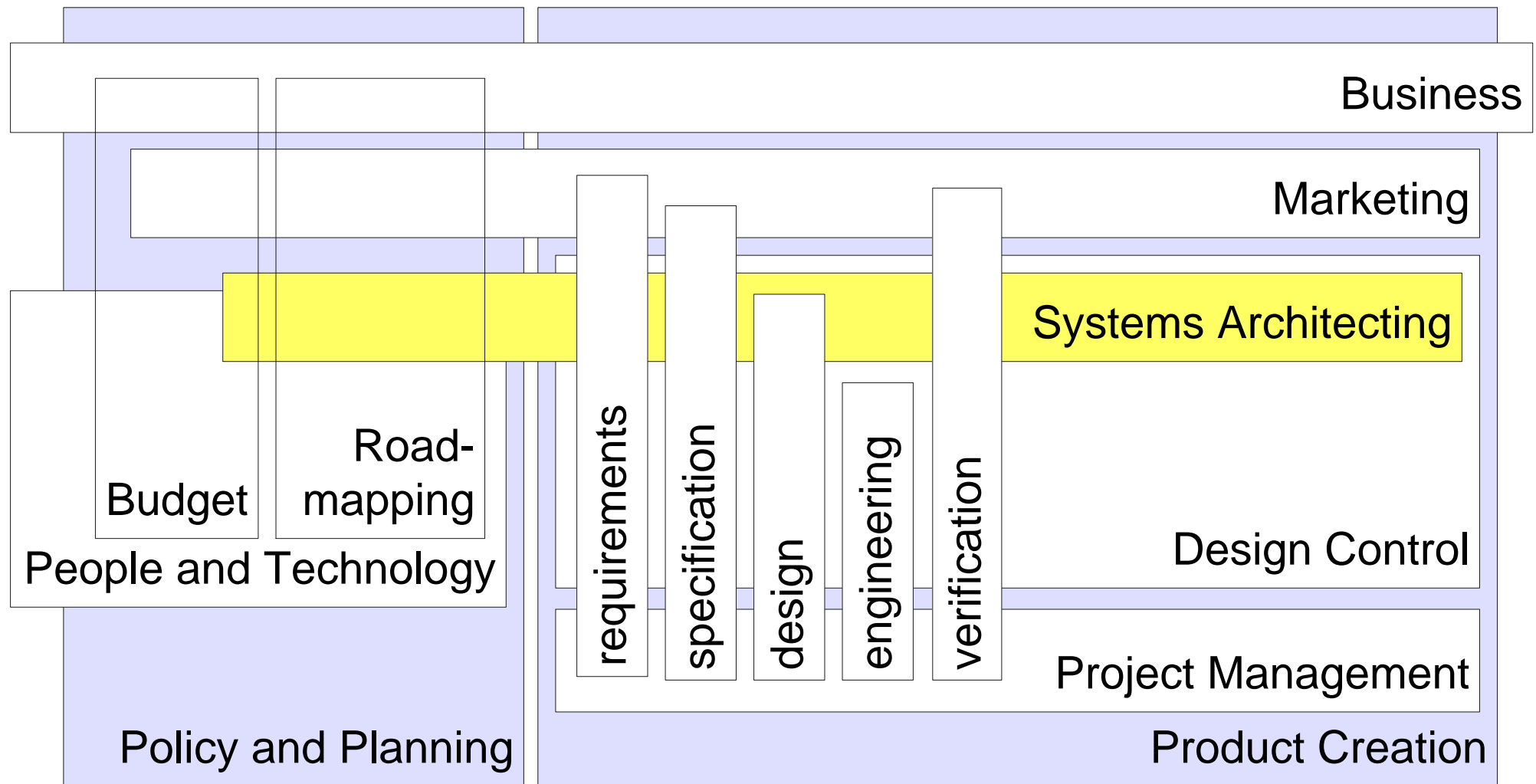
System Architecting Relation between PPP and PCP

Policy and Planning Process → **Context: Product Portfolio, Time** → Product Creation Process

Vision, Policy, Intention

Practical Knowledge ←

Feedback from Reality
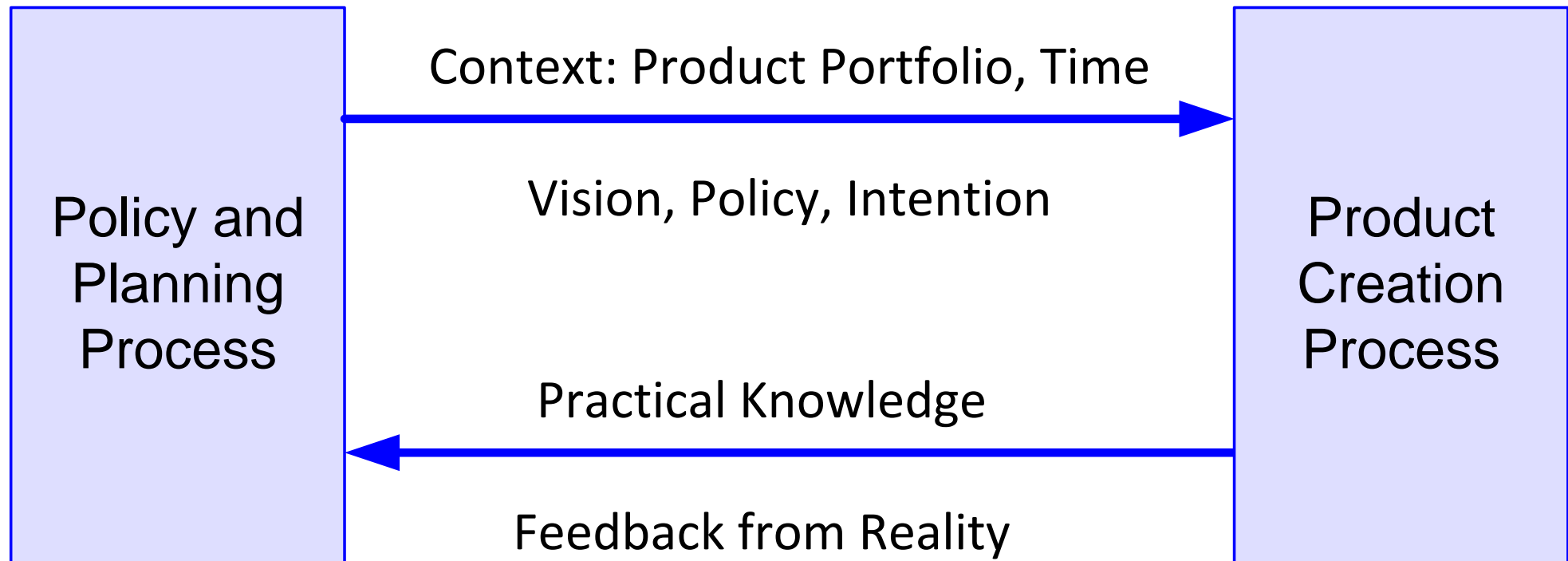
System Architecting Key Issues

key words

balance

consistency

integrity

simplicity

elegance

stakeholder
satisfaction

balancing acts

External ⟷ internal requirements
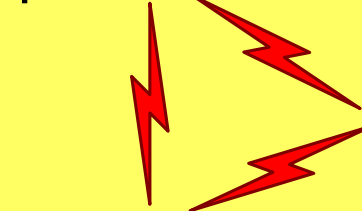
Short term needs ⟷ long term interests

Efforts ⟷ risks from requirements to verification

Mutual influence of detailed designs

Value ⟷ costs

example trade-offs
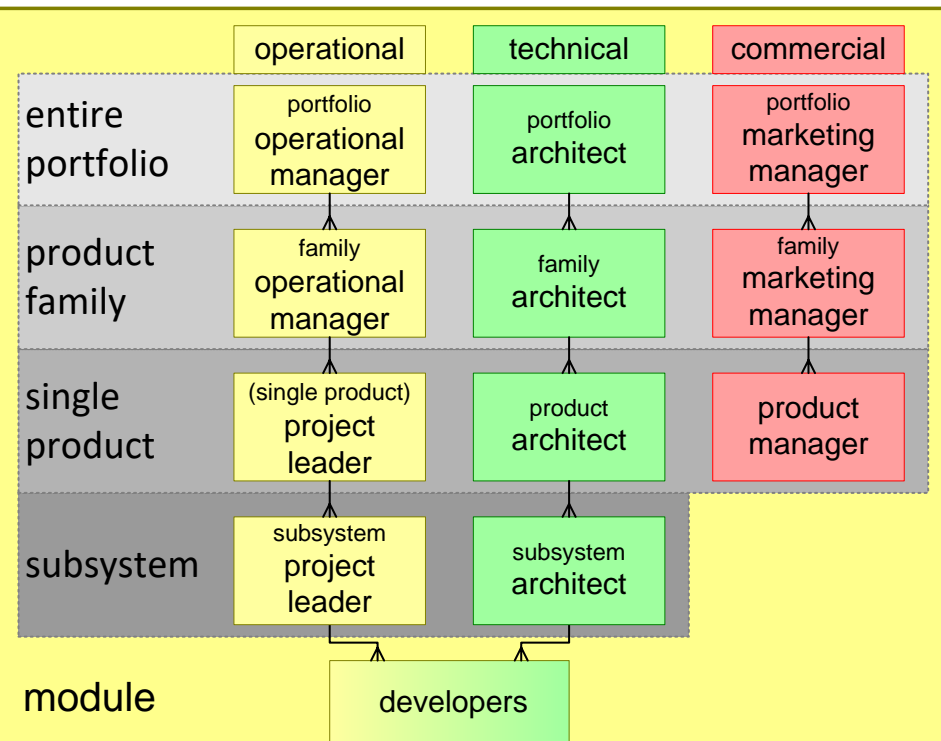
performance

functionality

qualities

synergy

specific solution

Exercise Product Creation Process

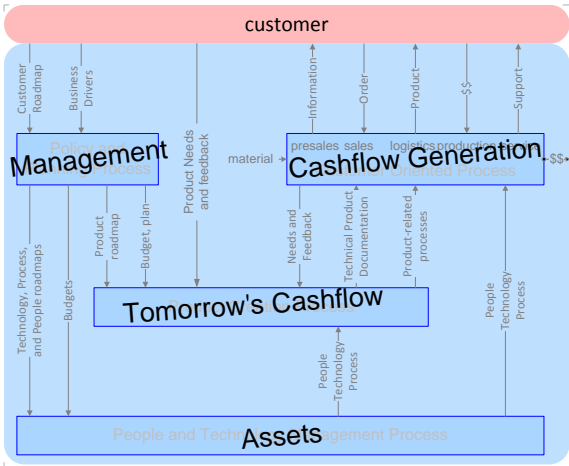Make a map with names of individuals in the **operational organization** of one project and its context

Identify the **relationships** of the **project core team**:

- **geographical**

- **organizational**

- **psychological**

	operational	technical	commercial
entire portfolio	portfolio operational manager	portfolio architect	portfolio marketing manager
product family	family operational manager	family architect	family marketing manager
single product	(single product) project leader	product architect	product manager
subsystem	subsystem project leader	subsystem architect	
module	developers		

USN ESI

Process Decomposition of a Business

Importance in Financial terms

Management

Cashflow Generation

Tomorrow's Cashflow

Assets

Value Chain and Feedback Flow

Policy and Planning Process

Customer-Oriented Process

Product Creation Process

Feedback

Value

People, Process, and Technology Management Process

intentionally left blank
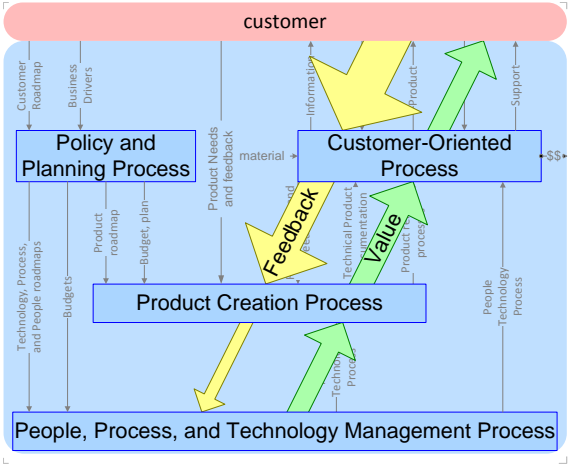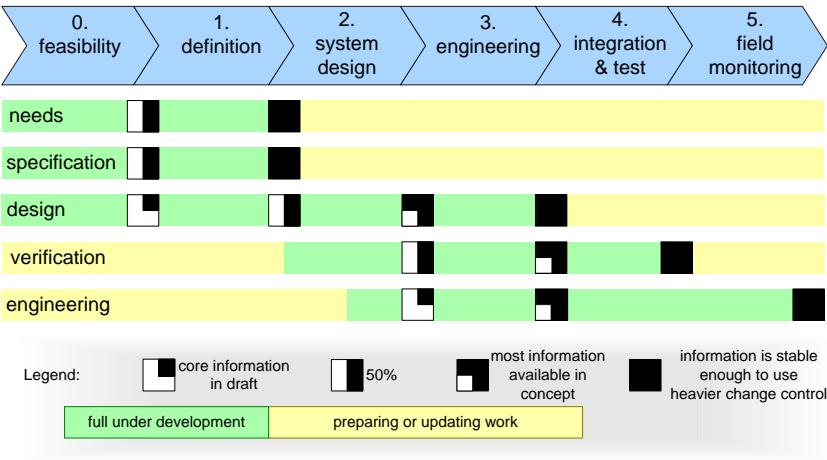
intentionally left blank

Product Creation Process

PCP involves **all** disciplines, much more than D&E

| 0. feasibility | 1. definition | 2. system design | 3. engineering | 4. integration & test | 5. field monitoring |

sales

logistics

production

service

development & engineering: marketing, project management, design

Phased Process

| 0. feasibility | 1. definition | 2. system design | 3. engineering | 4. integration & test | 5. field monitoring |

needs

specification

design

verification

engineering

Legend:

core information in draft

50%

most information available in concept

information is stable enough to use heavier change control

full under development

preparing or updating work

Incremental Development

test and evaluate

requirements specification

build

design

2% of budget (EVO)
2 weeks (XP)
up to 2 months
per cyclus

intentionally left blank
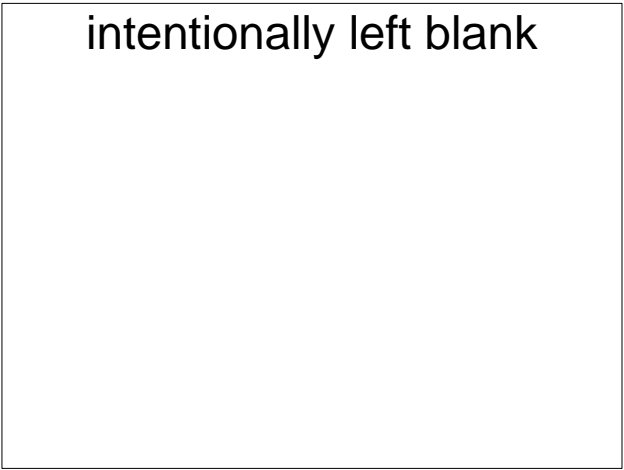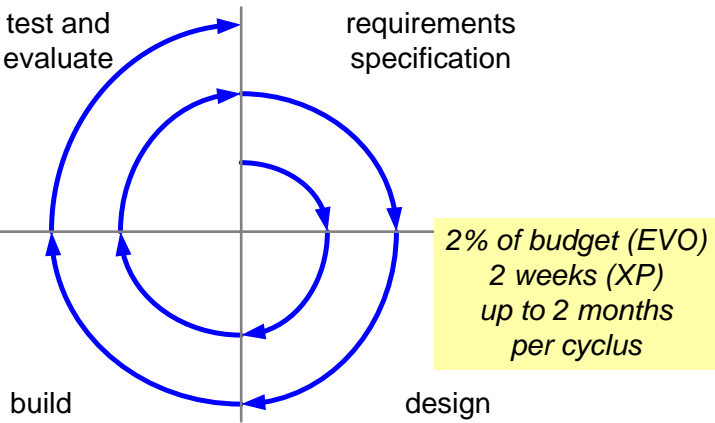
PCP Decomposition and Operational Management

PCP decomposition

```
                 Product Creation Process
         ┌───────────────────┼───────────────────┐
  Operational          Design              Marketing
  Management           Control

  specification        technical           profitability
  budget                                   sellability
  time
```
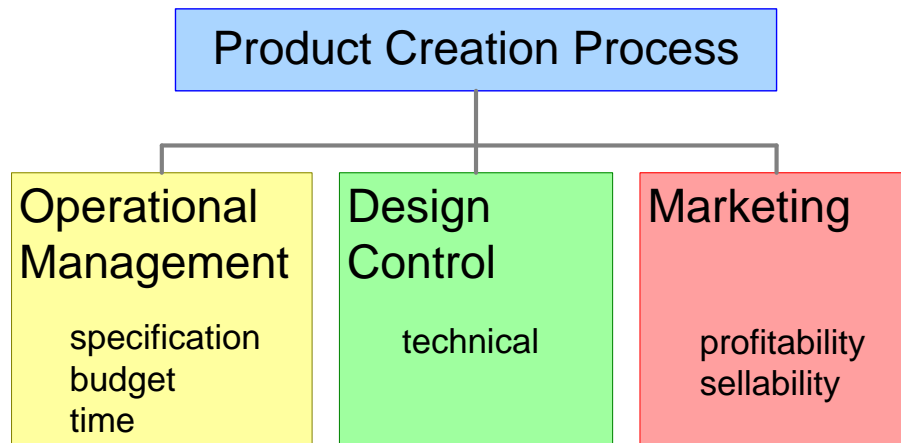
Architecture at all levels; From portfolio to subsystem

	operational	technical	commercial
entire portfolio	portfolio operational manager	portfolio architect	portfolio marketing manager
product family	family operational manager	family architect	family marketing manager
single product	(single product) project leader	product architect	product manager
subsystem	subsystem project leader	subsystem architect	
module	developers		

Operational Commitment

```
          Specification
               /\
              /  \
             /    \
            / Quality \
           /          \
          /_____\
   Resources          Time
```

Core: Operational + Technical + Commercial

Sales Manager, Quality Assurance, Logistics, Application Manager, Operational Support (project manager), Requirements Analyst, marketing manager, project leader, Subsystem Operational Leaders, Test Engineer, architect, Subsystem Architects, Service, Technology-Specific Architects, Manufacturing, Development Support

USN   ESI

System Architecture Process

In Business Context

Key Issues

key words

balance

consistency

integrity

simplicity

elegance

stakeholder satisfaction

balancing acts

External ⟷ internal requirements
Short term needs ⟷ long term interests
Efforts ⟷ risks from requirements to verification
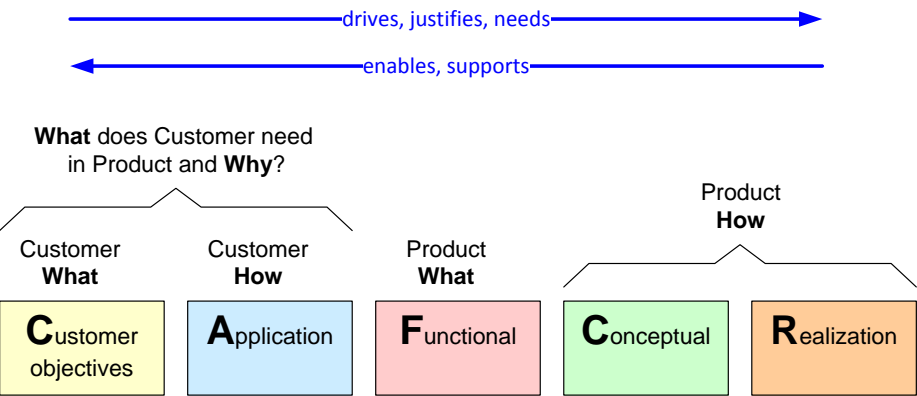Mutual influence of detailed designs
Value ⟷ costs

example trade-offs

performance   synergy

                functionality
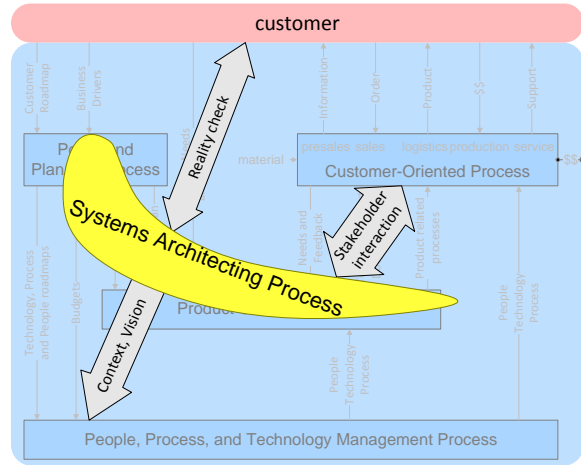
qualities   specific solution

5 Views

drives, justifies, needs →

← enables, supports

What does Customer need
in Product and **Why**?

Customer **What** | Customer **How** | Product **What** | Product **How**

Customer objectives | **A**pplication | **F**unctional | **C**onceptual | **R**ealization

intentionally left blank

Module Product Families and Generic Developments

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

This module addresses product families and generic developments.

January 22, 2023
status: preliminary draft
version: 1.3

USN ESI

Product Families and Generic Aspects

by *Gerrit Muller*   USN-SE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

Most products fit in a larger family of products. The members of such a product family share a lot of functionality and features. It is attractive to share implementations, designs et cetera between those members to increase the efficiency of the entire company.

In practice many difficulties pop up when product developments become coupled, due to the partial developments which are shared. This article discusses the advantages and disadvantages of a family approach based on shared developments and provides some methods to increase the chance on success.

January 22, 2023
status: concept
version: 2.3
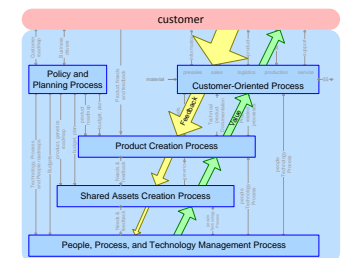
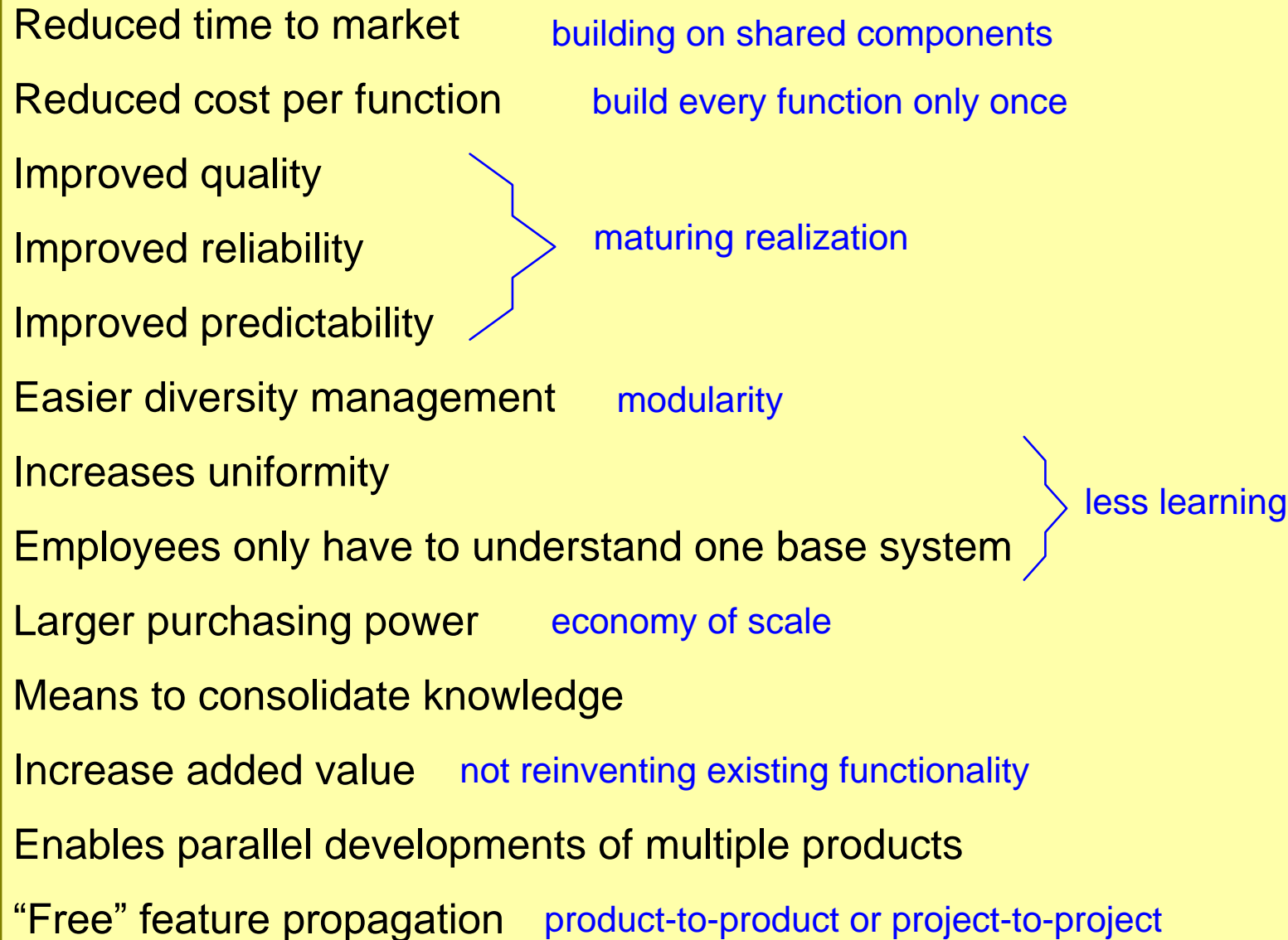Platform

Common components

Standard design

Framework

Family architecture

Generic aspects, functions, or features

Reuse

Products (in project environment)

Claimed Advantages of Generic Developments

Reduced time to market — building on shared components

Reduced cost per function — build every function only once

Improved quality
Improved reliability — maturing realization
Improved predictability

Easier diversity management — modularity

Increases uniformity
Employees only have to understand one base system — less learning

Larger purchasing power — economy of scale

Means to consolidate knowledge

Increase added value — not reinventing existing functionality

Enables parallel developments of multiple products

"Free" feature propagation — product-to-product or project-to-project

USN ESI

Experiences with reuse, from counterproductive to effective

<table>
<thead>
<tr><th>bad</th><th>good</th></tr>
</thead>
<tbody>
<tr><td>longer time to market</td><td>reduced time to market</td></tr>
<tr><td>high investments</td><td>reduced investment</td></tr>
<tr><td>lots of maintenance</td><td>reduced (shared) maintenance cost</td></tr>
<tr><td>poor quality</td><td>improved quality</td></tr>
<tr><td>poor reliability</td><td>improved reliability</td></tr>
<tr><td>diversity is opposed</td><td>easier diversity management</td></tr>
<tr><td>lot of know how required</td><td>understanding of one base system</td></tr>
<tr><td>predictable too late</td><td>improved predictability</td></tr>
<tr><td>dependability</td><td>larger purchasing power</td></tr>
<tr><td>knowledge dilution</td><td>means to consolidate knowledge</td></tr>
<tr><td>lack of market focus</td><td>increase added value</td></tr>
<tr><td>interference</td><td>enables parallel developments</td></tr>
<tr><td>but integration required</td><td>free feature propagation</td></tr>
</tbody>
</table>

USN   ESI

Successful examples of reuse

homogeneous domain
cath lab
MRI
television
waferstepper

hardware dominated
car
airplane
shaver
television

limited scope
audio codec
compression library
streaming library

USN ESI

Limits of successful reuse

struggle with integration/convergence with other domains

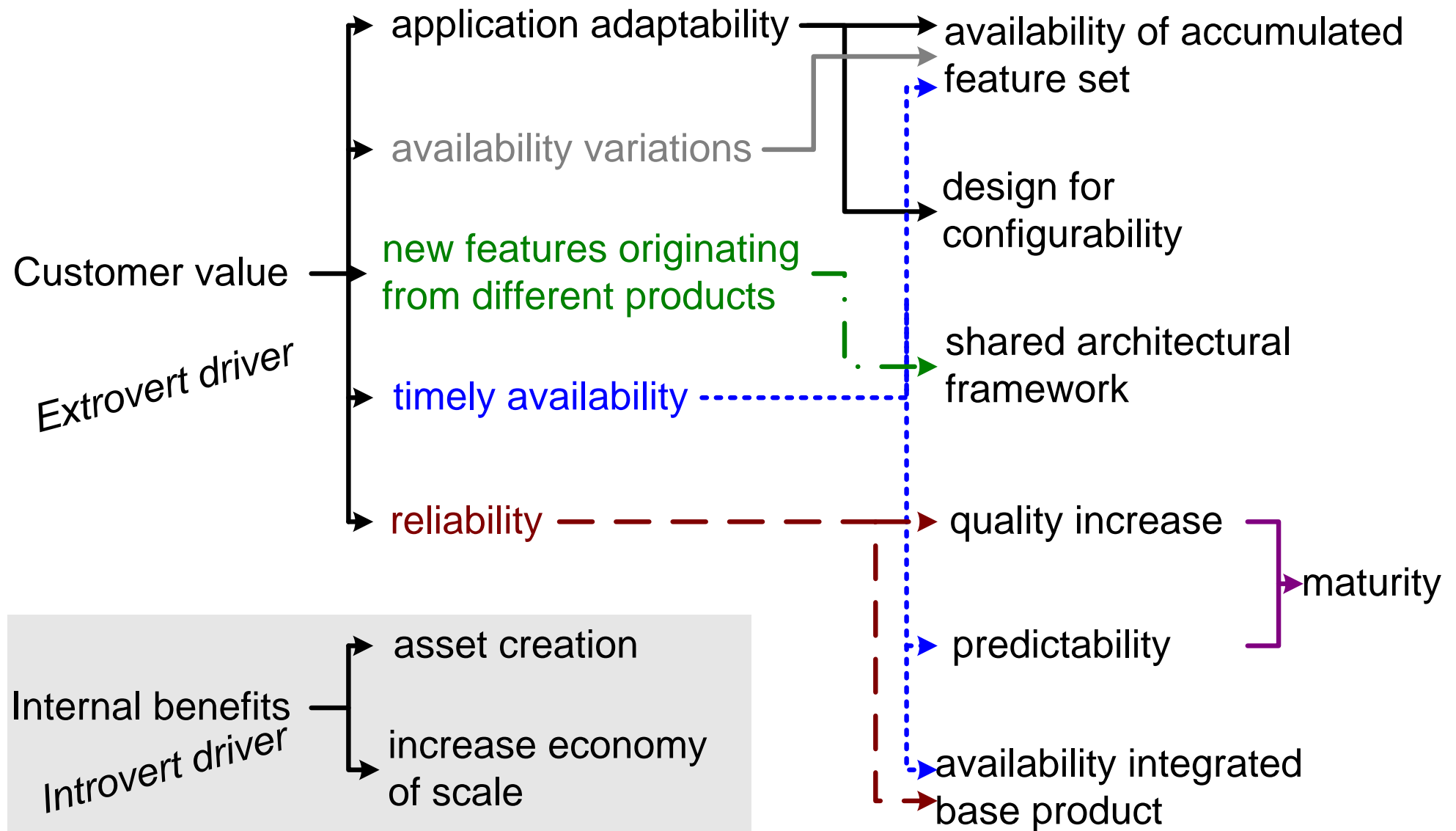   TV: digital networks and media
    cath lab: US imaging, MRI

poor/slow response on paradigm shifts

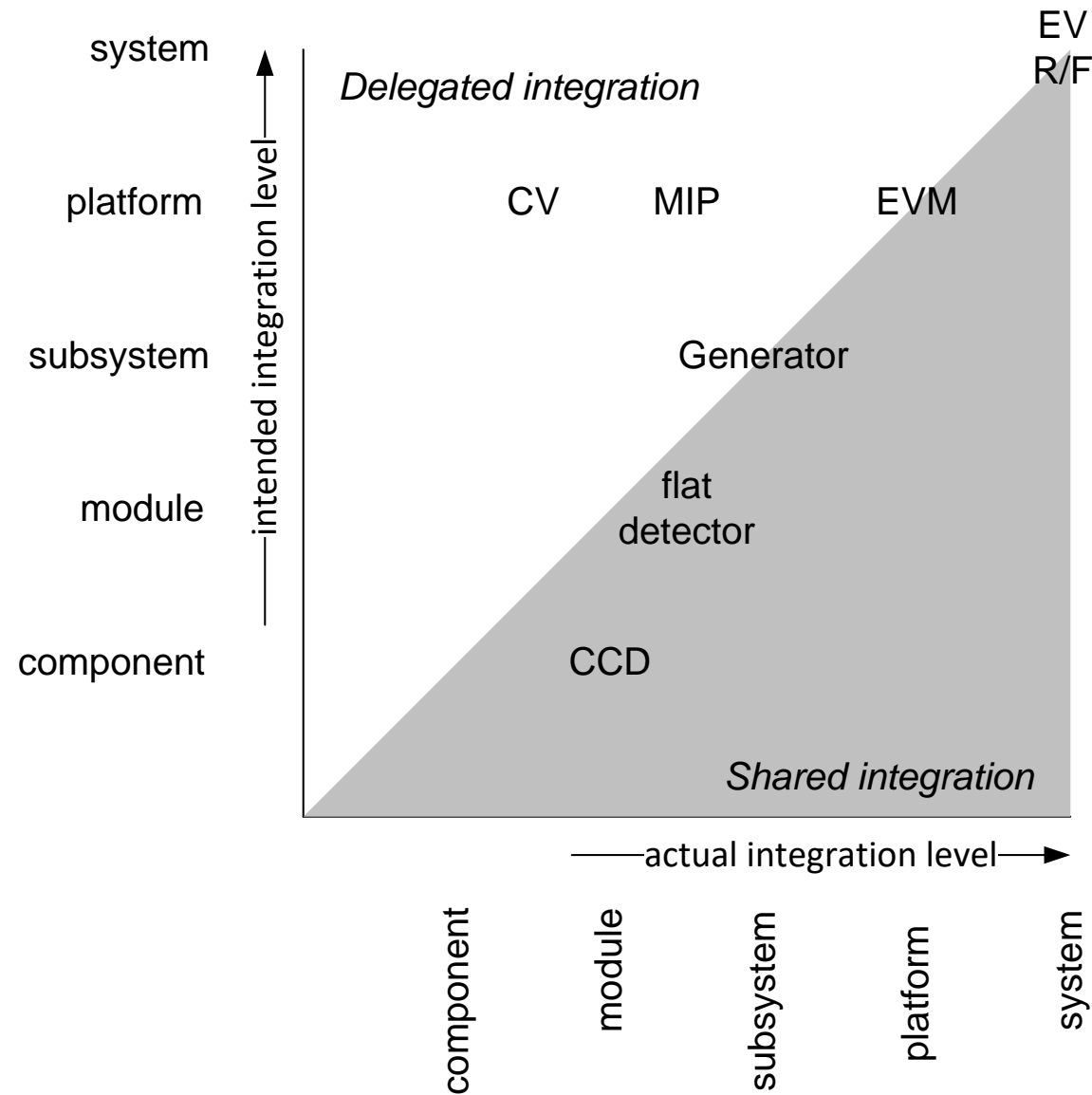   TV: LCD screens
    cath lab: image based acquisition control

software maintenance, configurations, integration, release

    MRI: integration and test
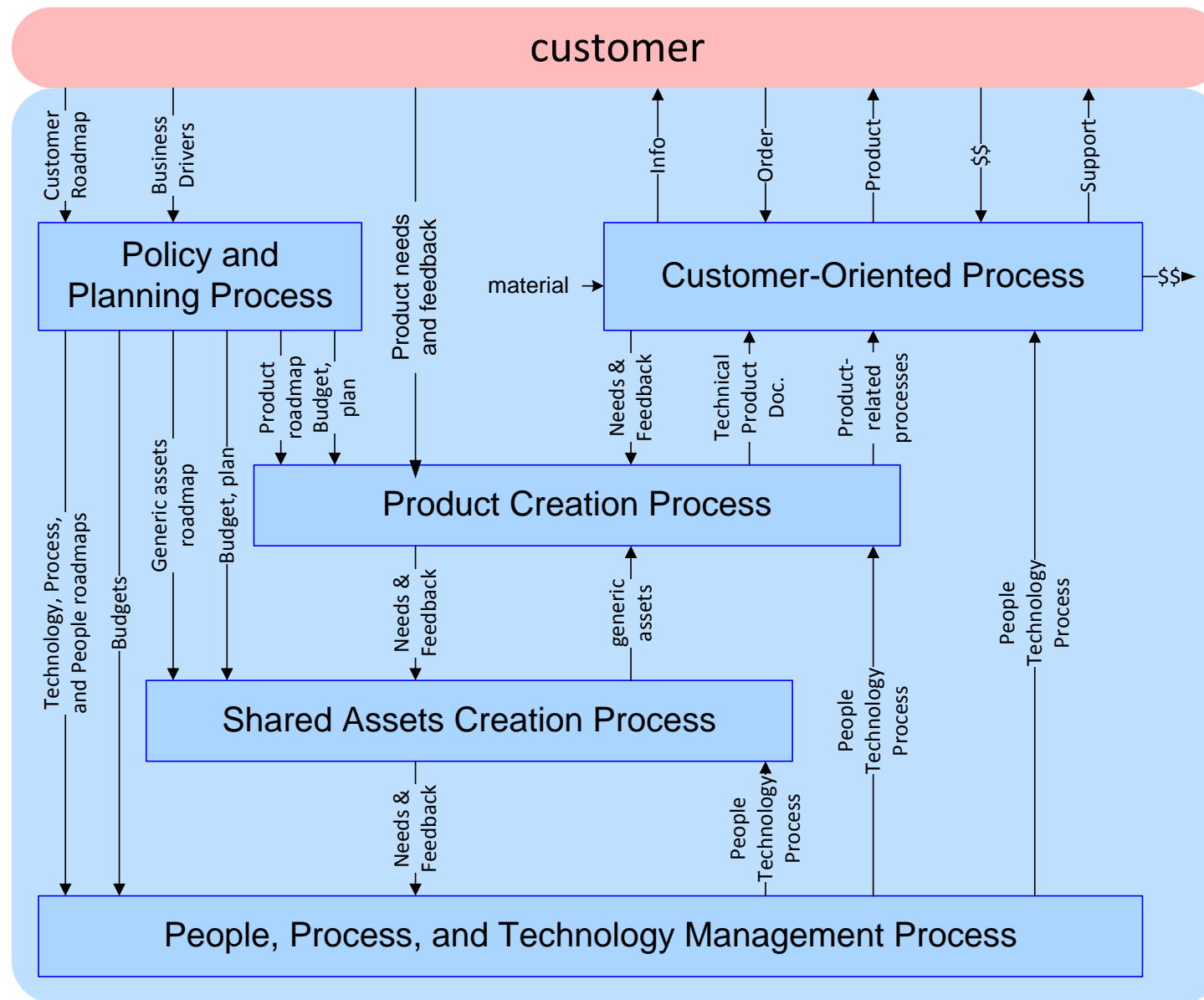   wafersteppers: number of configurations

how to innovate?

Drivers for Generic Developments

application adaptability → availability of accumulated feature set

availability variations

Customer value

Extrovert driver

new features originating from different products

timely availability

reliability

design for configurability

shared architectural framework

quality increase

predictability

maturity

Internal benefits

Introvert driver

asset creation

increase economy of scale

availability integrated base product

USN ESI

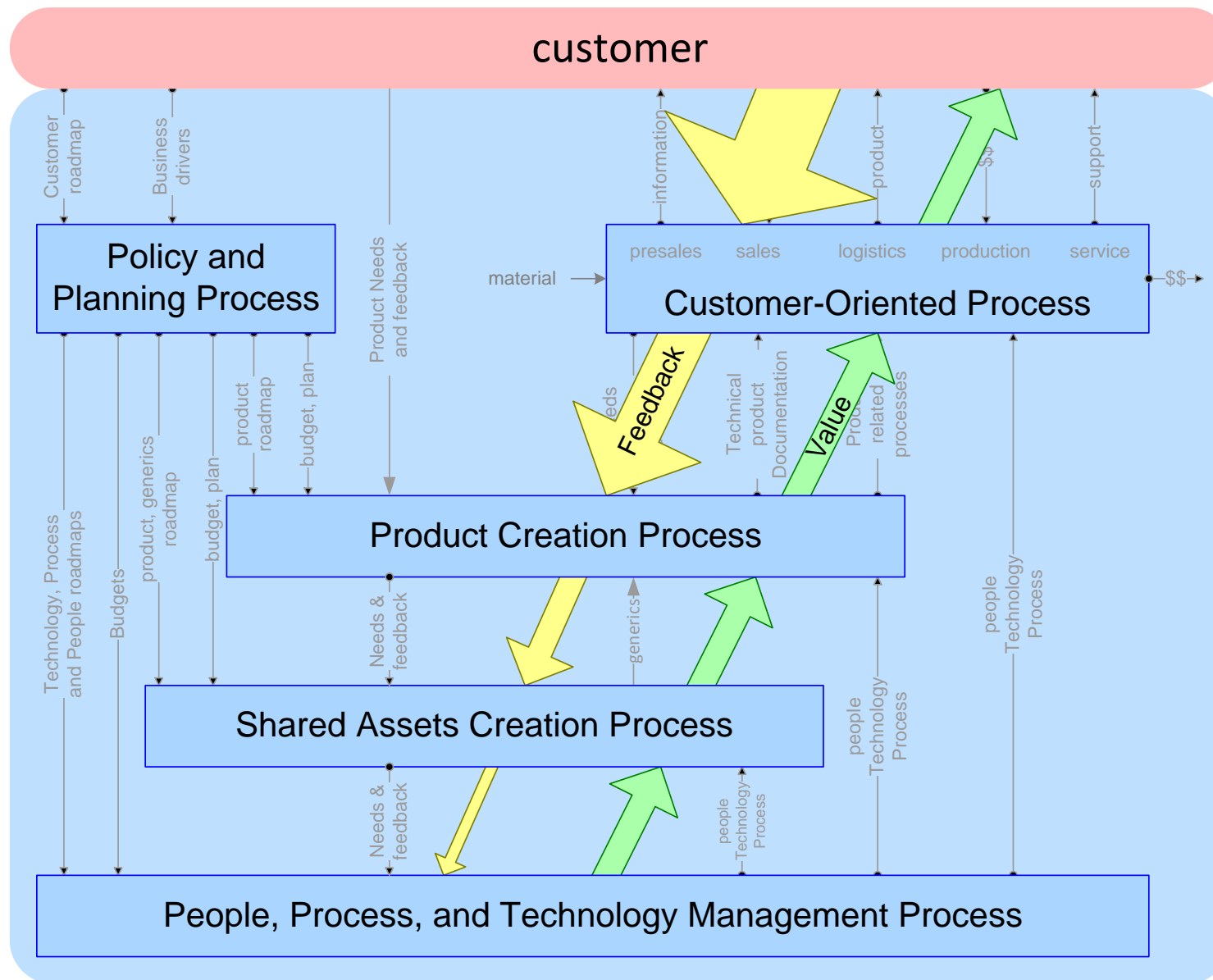# Granularity of generic developments shown in 2 dimensions
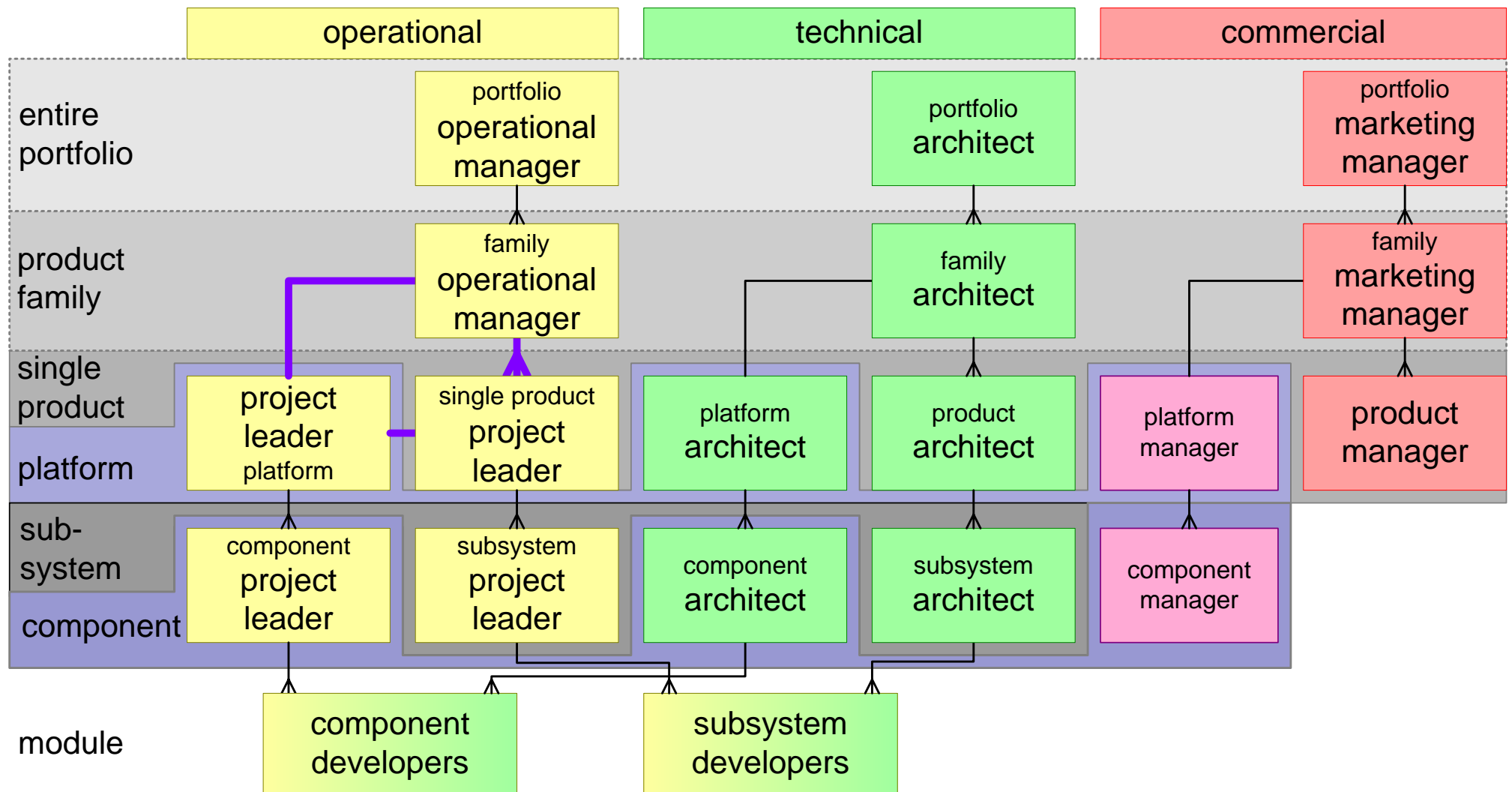
Modified Process Decomposition
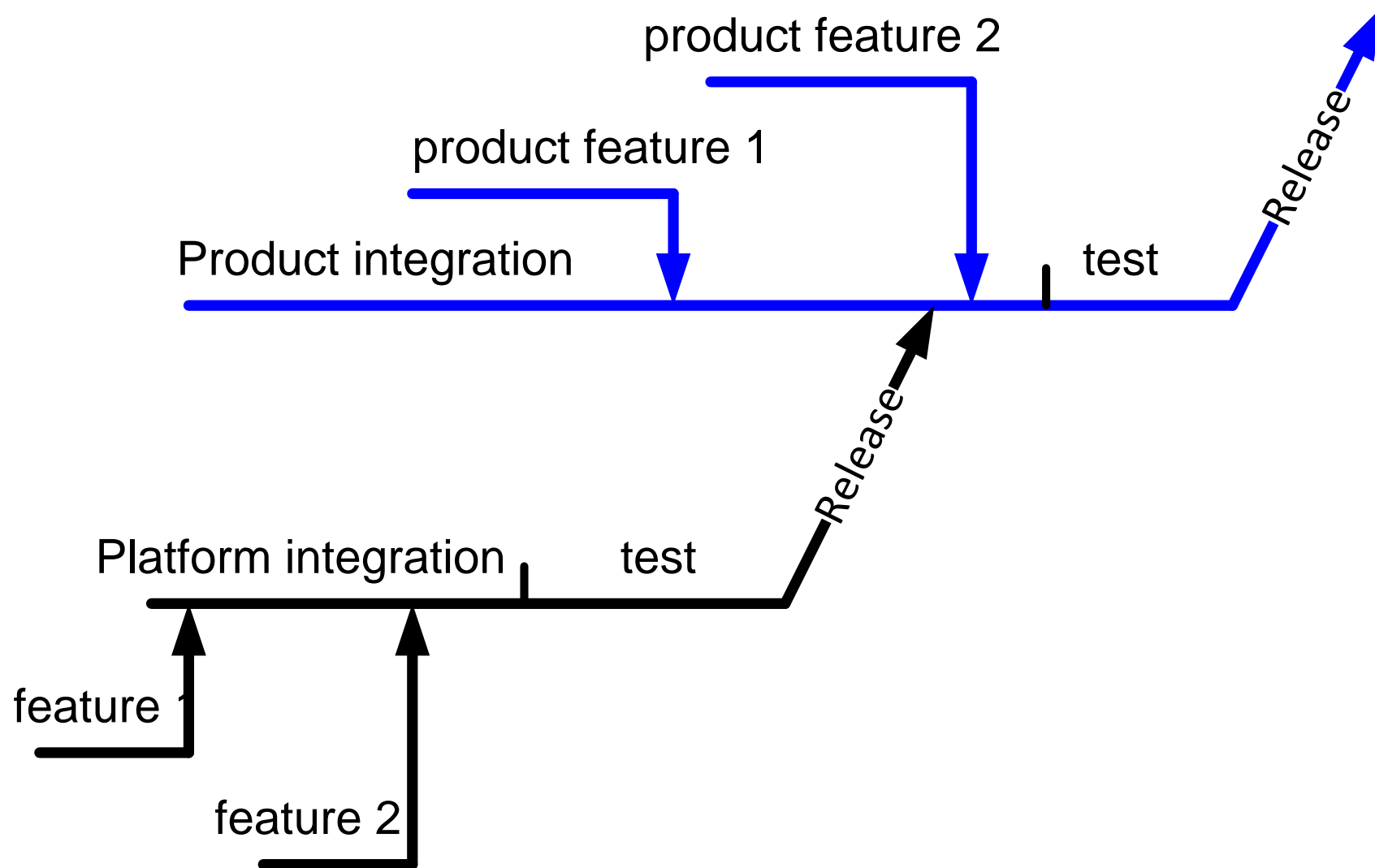
Financial Viewpoint on Process Decomposition
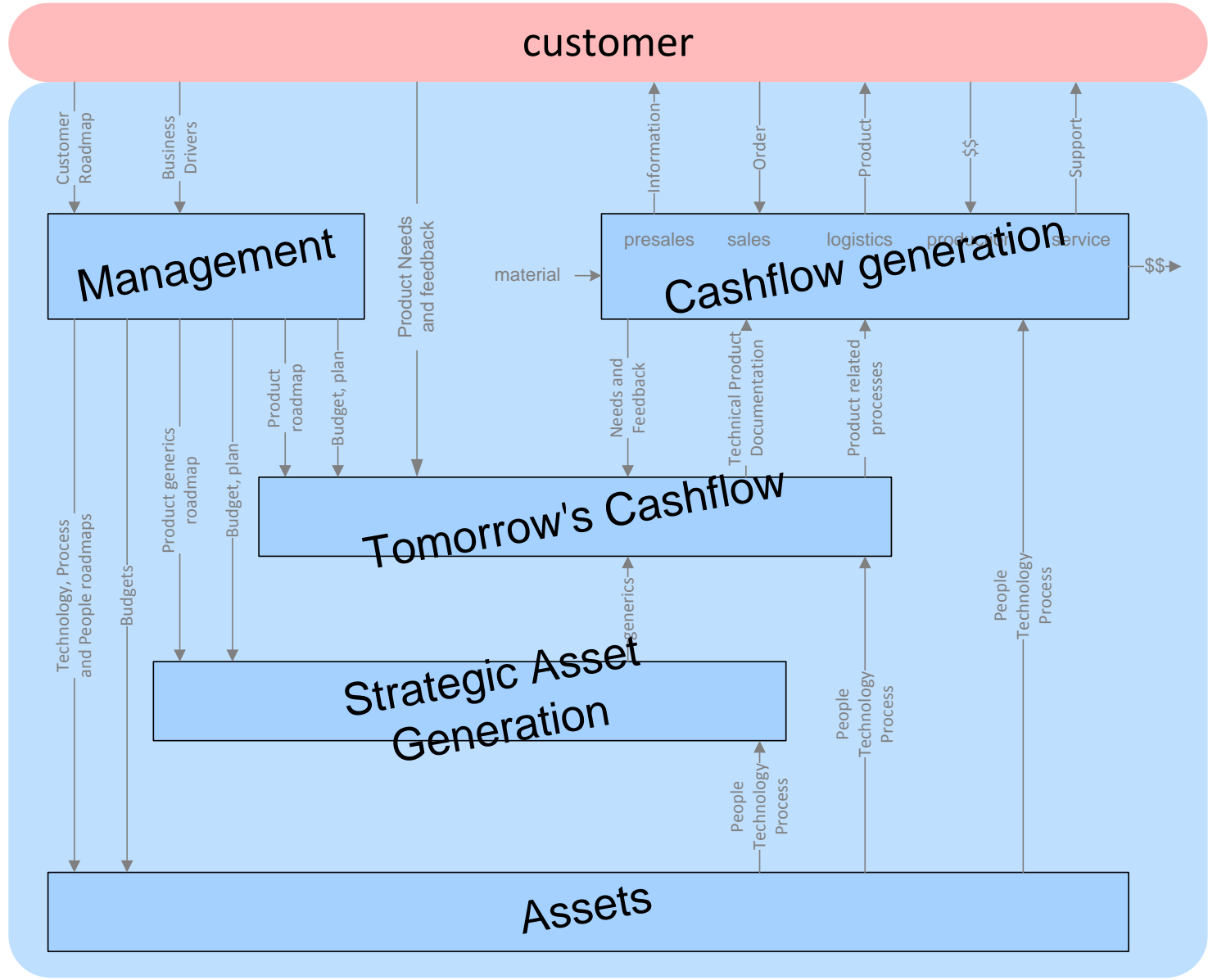
Value and Feedback Flow

Modified Operational Organization PCP

Propagation Delay Platform Feature to Market

product feature 2

product feature 1

Product integration

test

Release

Platform integration

test

Release

feature 1

feature 2

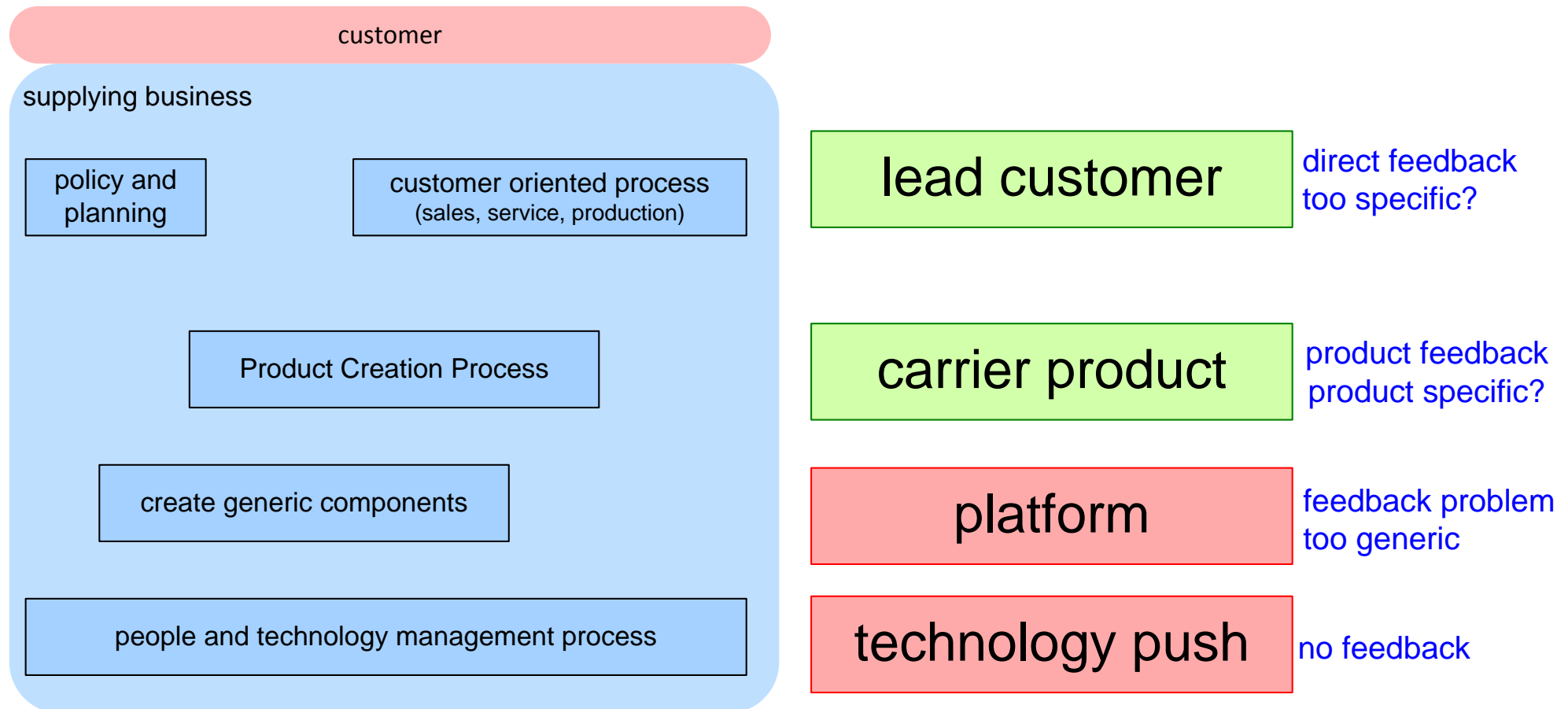# Sources of Failure in Generic Developments

Technical

- Too generic
- Innovation stops
  (stable interfaces)
- Vulnerability

Process/People/Organization

- Forced cooperation
- Time platform feature to market
- Unrealistic expectations
- Distance platform developer to customer
- No marketing ownership
- Bureaucratic process (no flexibility)
- New employees, knowledge dilution
- Underestimation of platform support
- Overstretching of product scope
- Nonmanagement, organizational scope increase
- Underestimation of integration
- Component/platform determines business policy
- Subcritical investment

USN   ESI

Models for Generic Development

customer

supplying business

policy and planning

customer oriented process
(sales, service, production)

Product Creation Process

create generic components

people and technology management process

lead customer — direct feedback too specific?

carrier product — product feedback product specific?

platform — feedback problem too generic

technology push — no feedback

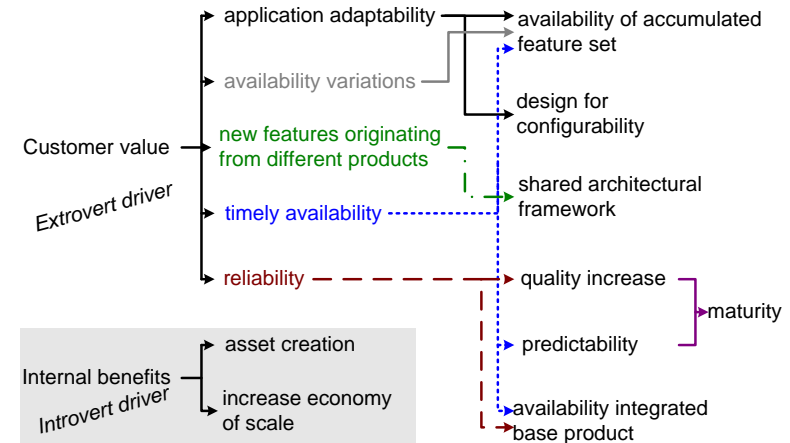USN ESI

Exercise Generic Developments

What are the top 3 benefits for your product family or generic development?

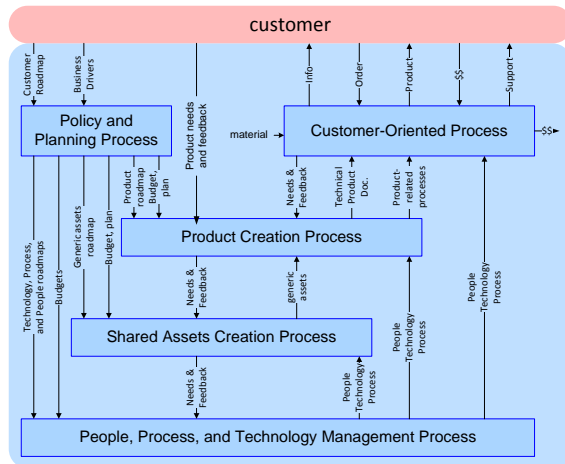What are the top 3 disadvantages?

Harvesting Synergy

Contradicting Experiences

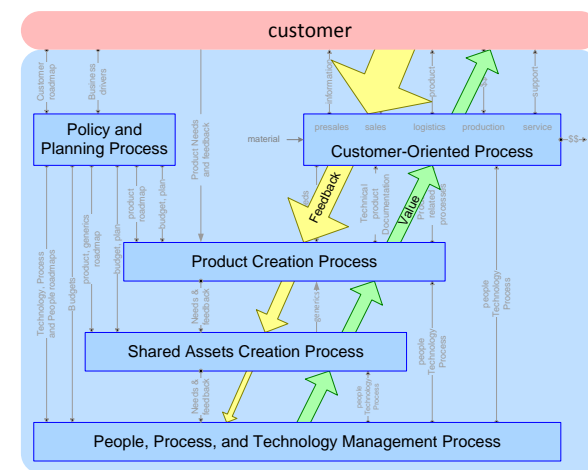| bad	good
longer time to market	reduced time to market
high investments	reduced investment
lots of maintenance	reduced (shared) maintenance cost
poor quality	improved quality
poor reliability	improved reliability
diversity is opposed	easier diversity management
lot of know how required	understanding of one base system
predictable too late	improved predictability
dependability	larger purchasing power
knowledge dilution	means to consolidate knowledge
lack of market focus	increase added value
interference	enables parallel developments
but integration required	free feature propagation

Drivers

Customer value — Extrovert driver:
- application adaptability → availability of accumulated feature set
- availability variations → design for configurability
- new features originating from different products → shared architectural framework
- timely availability
- reliability

Internal benefits — Introvert driver:
- asset creation
- increase economy of scale

quality increase, predictability → maturity
availability integrated base product

Shared Asset Creation Process

Longer Chains

USN   ESI

Some Architecting Means

Organizational Complexity

operational	technical	commercial

entire portfolio	portfolio operational manager	portfolio architect	portfolio marketing manager
product family	family operational manager	family architect	family marketing manager
single product / platform	project leader platform / single product project leader	platform architect / product architect	platform manager / product manager
sub-system / component	component project leader / subsystem project leader	component architect / subsystem architect	component manager

module: component developers / subsystem developers

Delay to Market

product feature 2
product feature 1
Product integration — test — Release

Platform integration — test — Release
feature 1
feature 2

Pitfalls

Technical

• Too generic
• Innovation stops
  (stable interfaces)
• Vulnerability

Process/People/Organization

• Forced cooperation
• Time platform feature to market
• Unrealistic expectations
• Distance platform developer to customer
• No marketing ownership
• Bureaucratic process (no flexibility)
• New employees, knowledge dilution
• Underestimation of platform support
• Overstretching of product scope
• Nonmanagement, organizational scope increase
• Underestimation of integration
• Component/platform determines business policy
• Subcritical investment

Successful and Failing Models
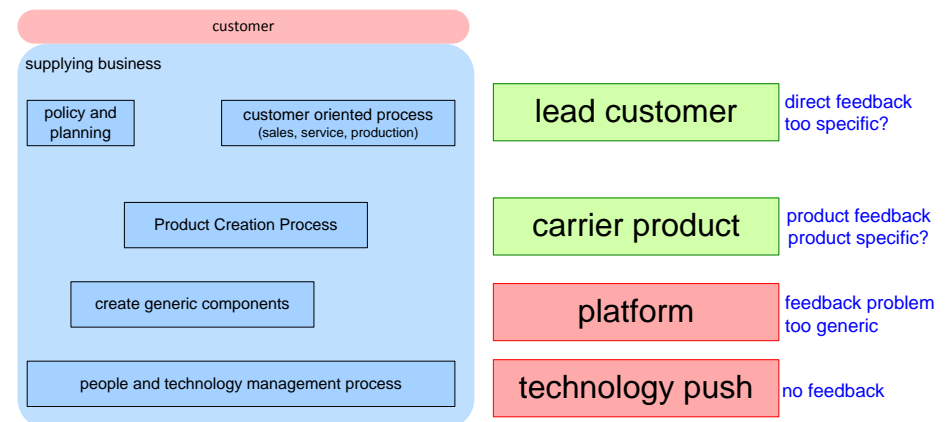
customer

supplying business

policy and planning

customer oriented process
(sales, service, production)

Product Creation Process

create generic components

people and technology management process

lead customer — direct feedback too specific?

carrier product — product feedback product specific?

platform — feedback problem too generic

technology push — no feedback

Role and Task of the System Architect

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

The role and the task of the system architect are described in this module.

January 22, 2023
status: preliminary draft
version: 1.0

The Role and Task of the System Architect

by *Gerrit Muller*   USN-SE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

The role of the system architect is described from three viewpoints: deliverables, responsibilities and activities. This description shows the inherent tension in this role: a small set of hard deliverables, covering a fuzzy set of responsibilities, hiding an enormous amount of barely visible day-to-day work.
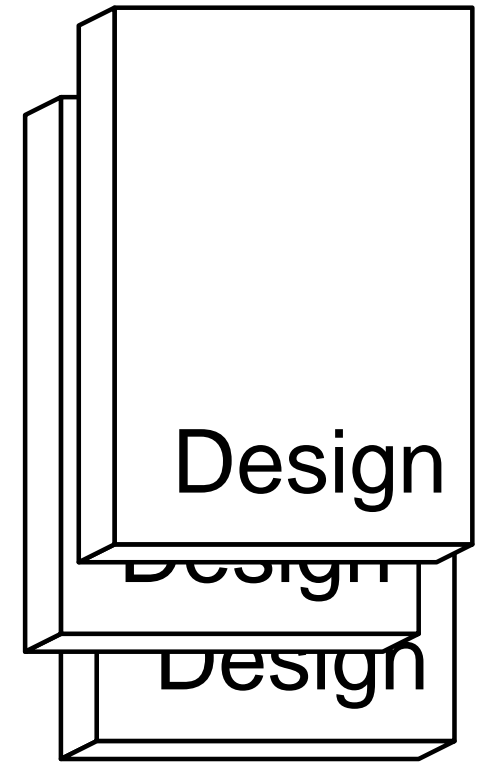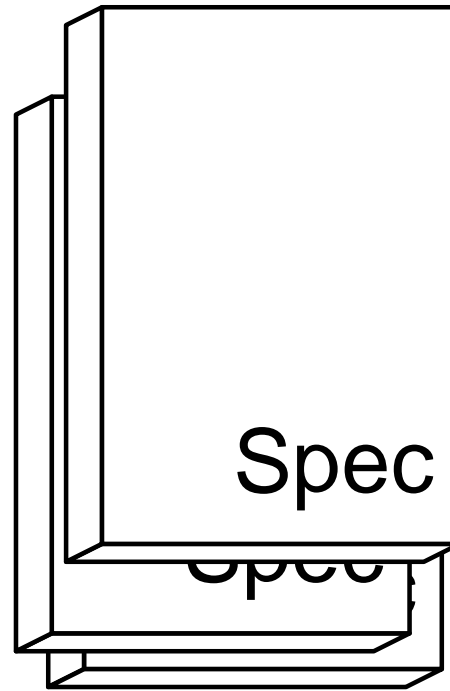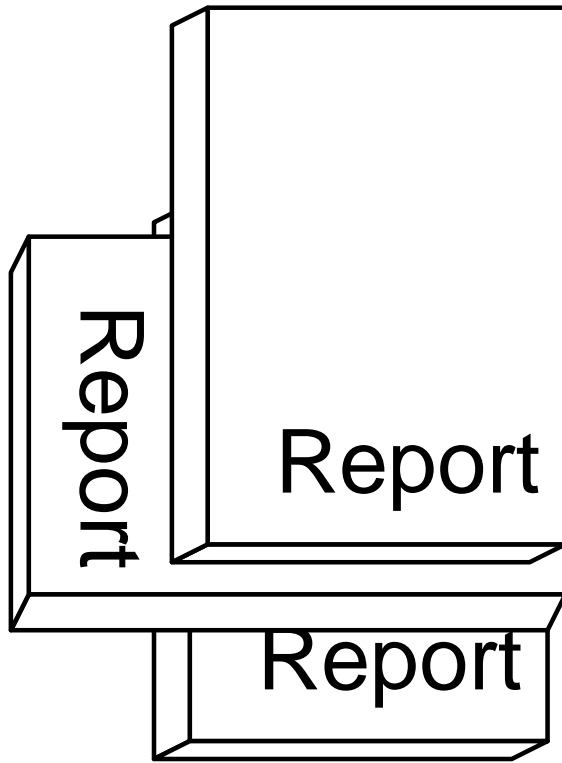
January 22, 2023
status: concept
version: 2.0
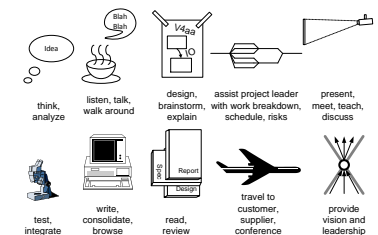
Deliverables of the System Architect

Report

Report

Report

Spec

Design

USN ESI

List of Deliverables

Customer and Life-Cycle Needs *(what is needed)*

System Specification *(what will be realized)*

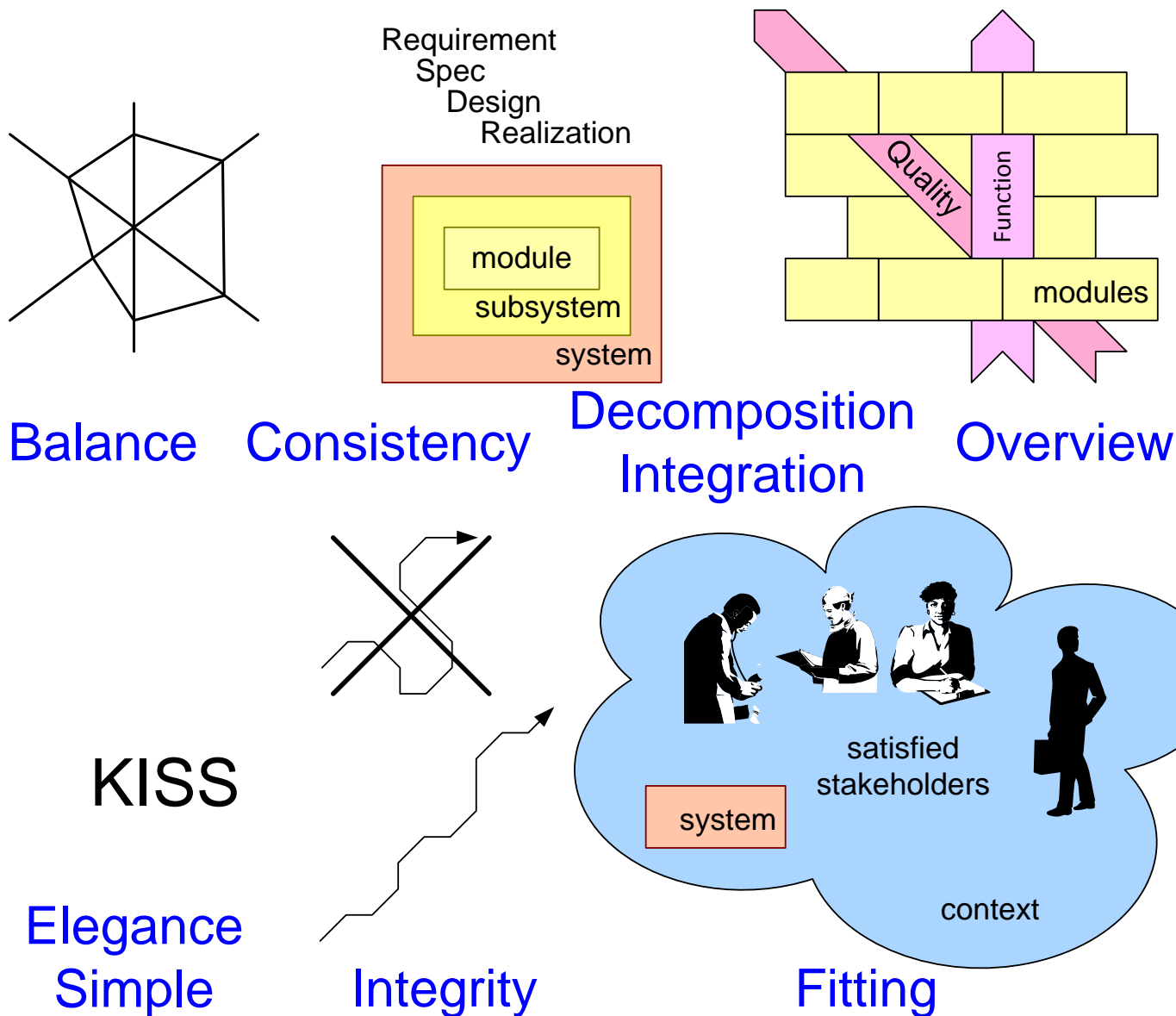Design Specification *(how the system will be realized)*

Verification Specification *(how the system will be verified)*

Verification Report *(the result of the verification)*

Feasibility Report *(the results of a feasibility study)*

Roadmap

Responsibilities of the System Architect

Requirement
Spec
Design
Realization

module
subsystem
system

Quality

Function

modules

Balance

Consistency

Decomposition
Integration

Overview

KISS

Elegance
Simple

Integrity

satisfied
stakeholders

system

context

Fitting

USN   ESI

Examples of Secondary Responsibilities

responsibility	primary owner
business plan, profit	business manager
schedule, resources	project leader
market, saleability	marketing manager
technology	technology manager
process, people	line manager
detailed designs	engineers

What does the System Architect do?

think, analyze

listen, talk, walk around

design, brainstorm, explain

assist project leader with work breakdown, schedule, risks

present, meet, teach, discuss

test, integrate

write, consolidate, browse

read, review

travel to customer, supplier, conference

provide vision and leadership

From Detail to Overview

		Quantity per year (order-of-magnitude)	architect time per item
consolidation in deliverables	driving views	10	100 h
	shared issues	10^2	1 h
meetings	touched details	10^4	0.5 – 10 min
informal contacts	seen details	$10^5 - 10^6$	0.1 – 1 sec
sampling scanning	product details	$10^7 - 10^{10}$	
	real-world facts	infinite	

USN   ESI

Reality or Virtuality?

Abstractions only exist for concrete facts.

Visible Output versus Invisible Work

From Manager perspective

Decreasing Visibility

Deliverables

Report
Spec
Design

Requirement
Spec
Design
Realization

module
subsystem
system

Quality
Function
modules

KISS

Responsibilities

Idea

Bla Bla

V4aa

IO

Spec
Report
Design

Activities

USN   ESI
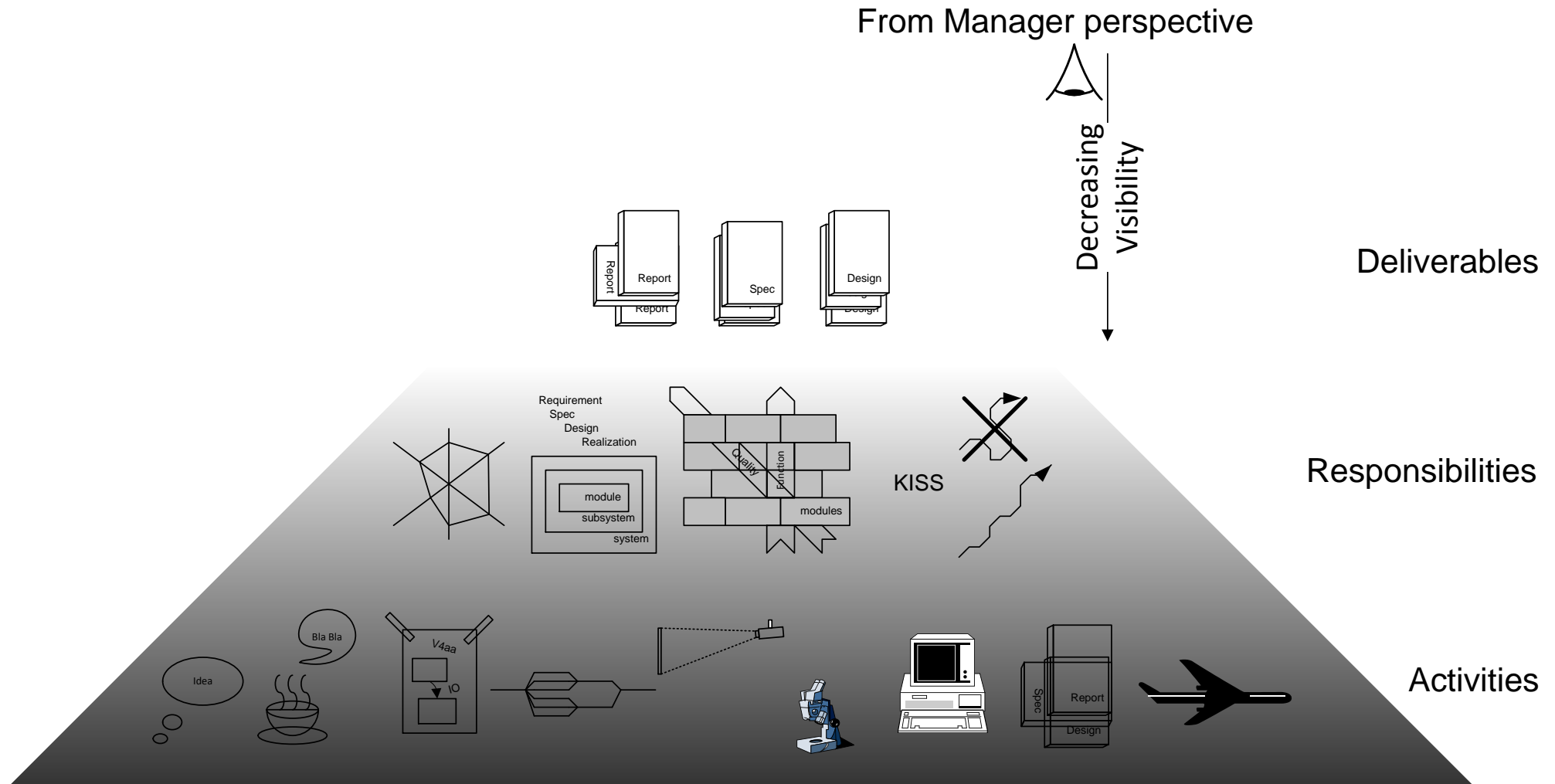
The Awakening of a System Architect

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: gaudisite@gmail.com

www.gaudisite.nl

Abstract

The typical phases of a system architect development are described, beginning at the fundamental technology knowledge, with a later broadening in technology and in business aspects. Finally the subtlety of individual human beings is taken into account.
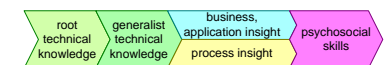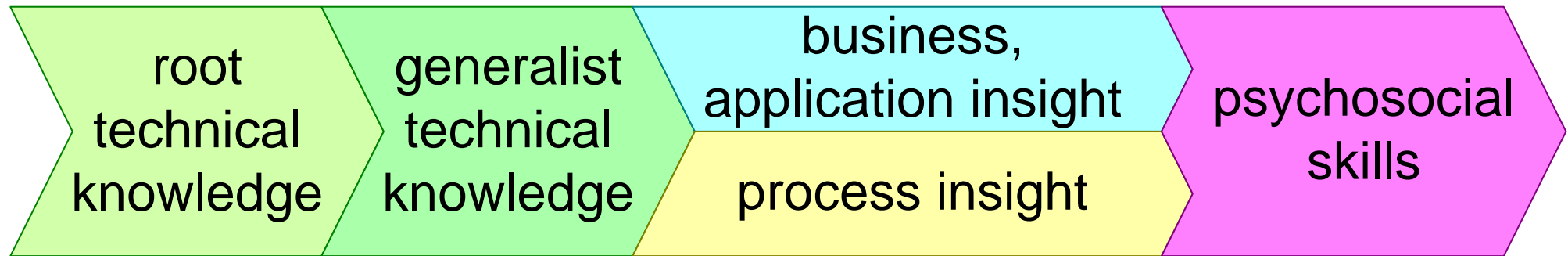
January 22, 2023
status: concept
version: 1.1

Typical Growth of a System Architect

root technical knowledge → generalist technical knowledge → business, application insight / process insight → psychosocial skills

Generalist versus Specialist

breadth of
knowledge

depth of
knowledge

specialist

generalist

root
knowledge
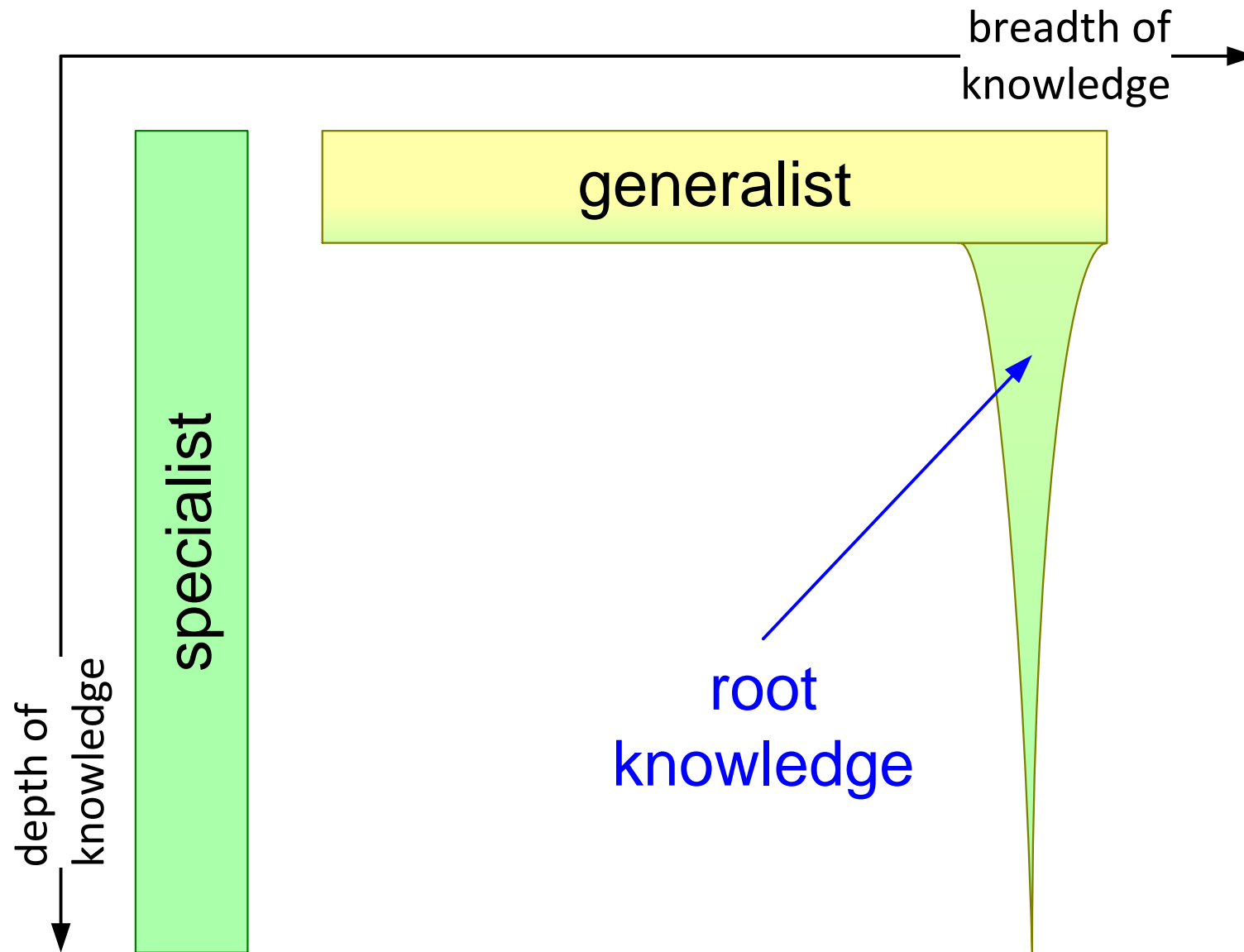
Generalists and Specialists are Complementary

breadth of knowledge

depth of knowledge

generalist

generalist

specialist specialist specialist specialist specialist specialist specialist specialist
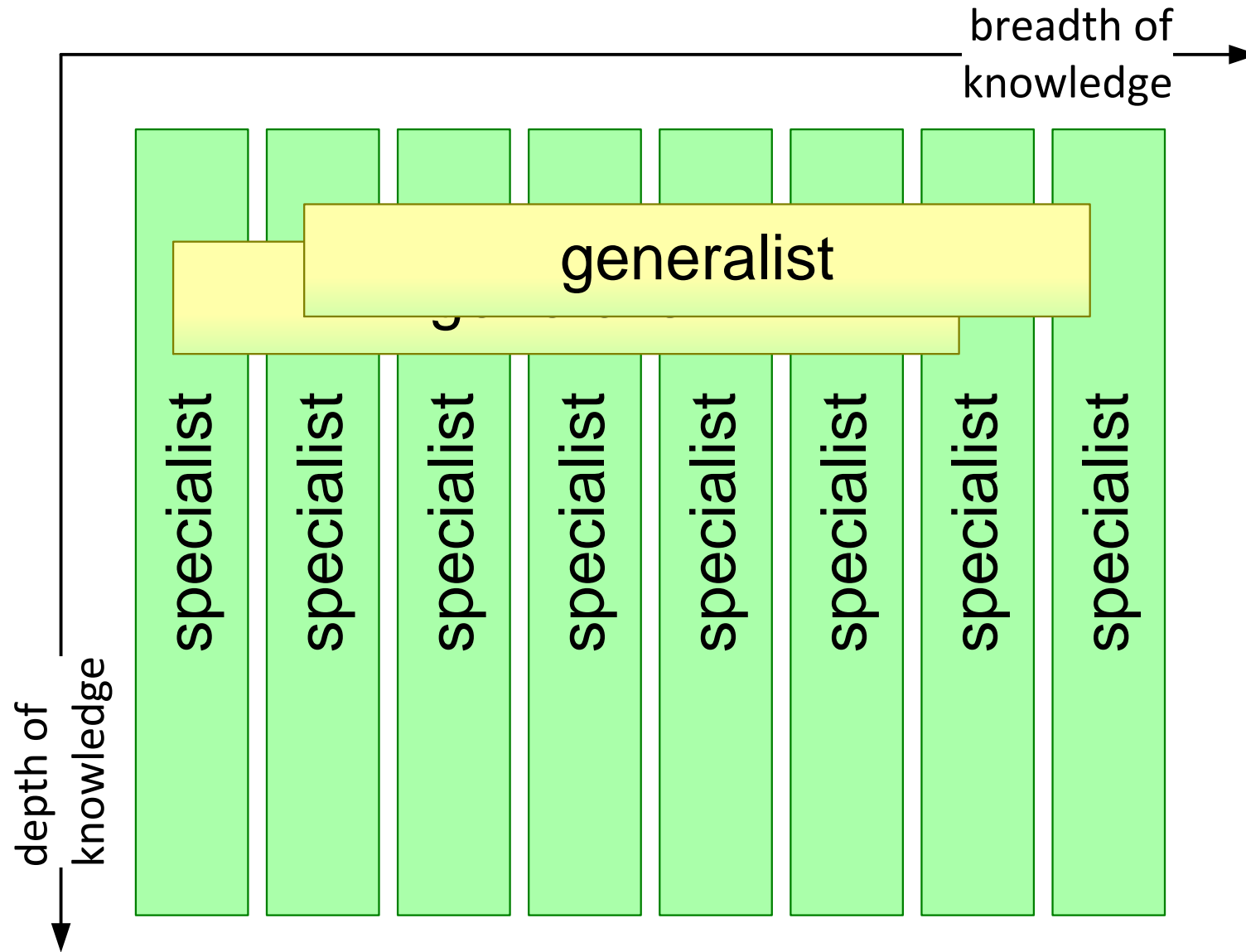
Spectrum from Specialist to System Architect

breadth of knowledge →

depth of knowledge

specialist

all-round specialist

aspect architect

systems architect

root knowledge

USN ESI
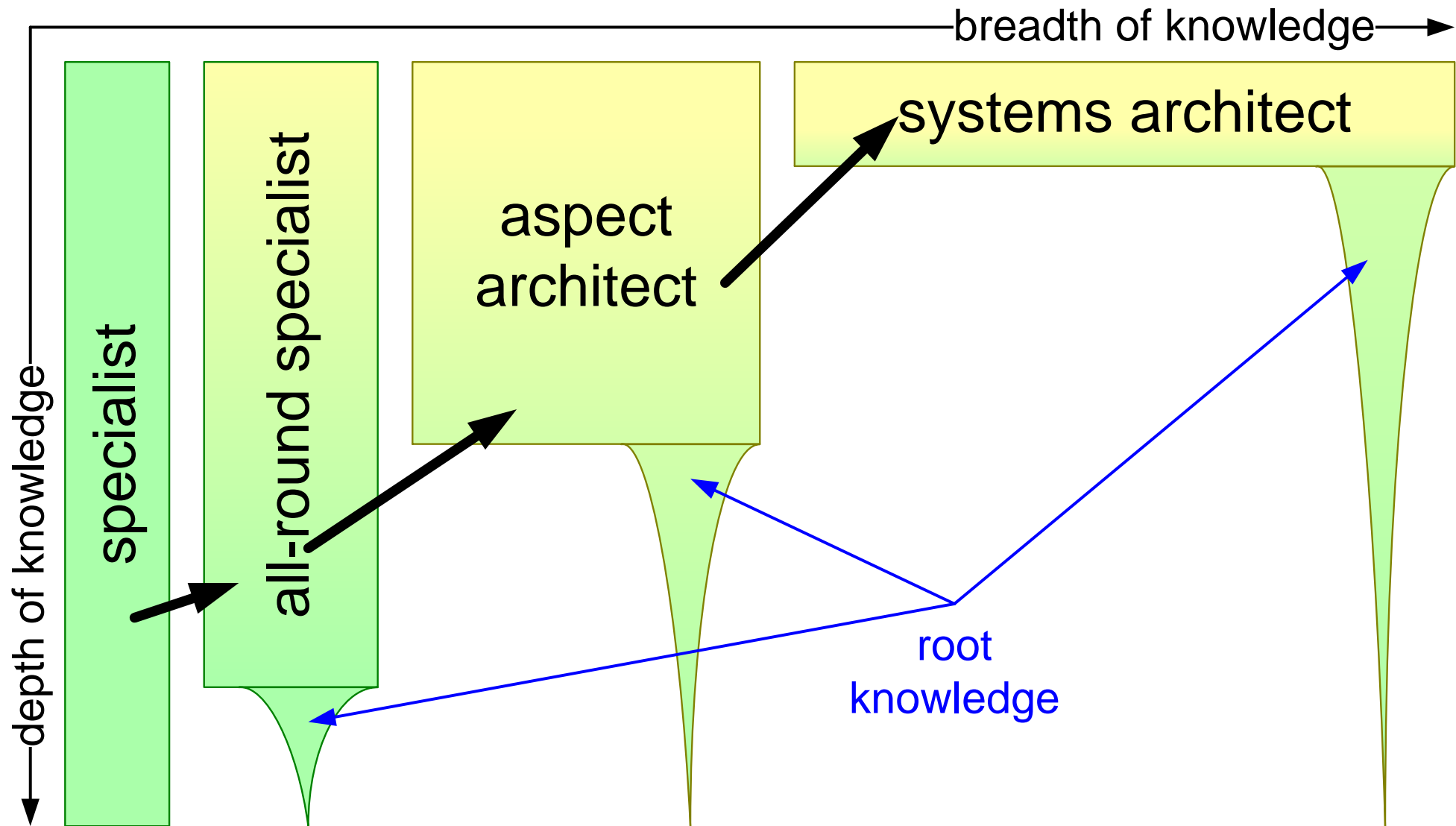
Architecting Interaction Styles

by *Gerrit Muller*   USN-SE

e-mail: gaudisite@gmail.com

www.gaudisite.nl

Abstract

A system architects needs skills to apply different interactions styles, depending on the circumstances. This document discusses the following interaction styles: provocation, facilitation, leading, empathic, interviewing, white board simulation, and judo tactics.

January 22, 2023
status: draft
version: 0.2

provocation — when in an impasse: provoke
effective when used sparsely

facilitation — especially recommended when new in a field:
contribute to the team, while absorbing new knowledge

leading — provide vision and direction, make choices
risk: followers stop to give the needed feedback

empathic — take the viewpoint of the stakeholder
acknowledge the stakeholder's feelings, needs, concerns

interviewing — investigate by asking questions

whiteboard simulation — invite a few engineers and walk through
the system operation step by step

judo tactics — first listen to the stakeholder and then
explain cost and alternative opportunities

Architecting Styles

provocation
when in an impasse: provoke
effective when used sparsely

facilitation
especially recommended when new in a field:
contribute to the team, while absorbing new knowledge

leading
provide vision and direction, make choices
risk: followers stop to give the needed feedback

empathic
take the viewpoint of the stakeholder
acknowledge the stakeholder's feelings, needs, concerns

interviewing
investigate by asking questions

whiteboard simulation
invite a few engineers and walk through
the system operation step by step

judo tactics
first listen to the stakeholder and then
explain cost and alternative opportunities

Exercise Role and Task of the System Architect

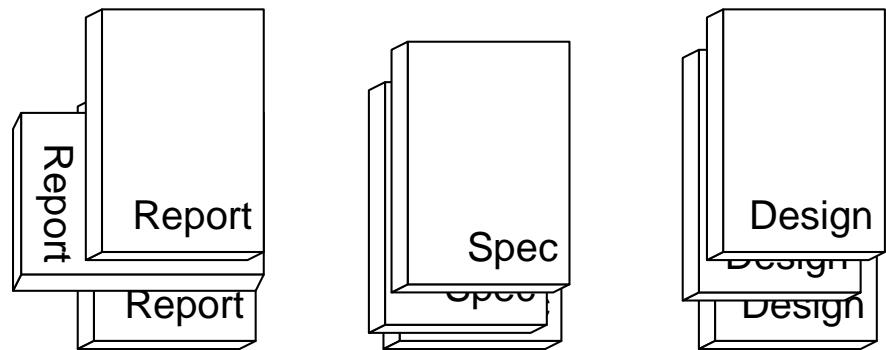Role play with 3 roles and optional observer:

- 1 operational leader (project leader)
- 1 system architect
- 1 marketing manager
- 1 observer (optional)

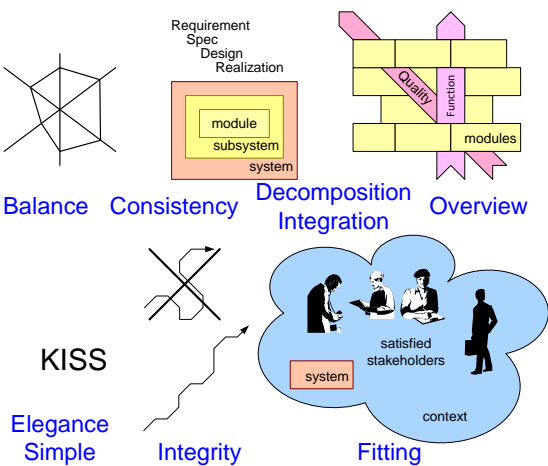Discuss the definition (business relevance, specification, and planning) of a travel e-mail mate.

Present (max. 2 flips) the result and the process (the relation and interaction of the three roles).
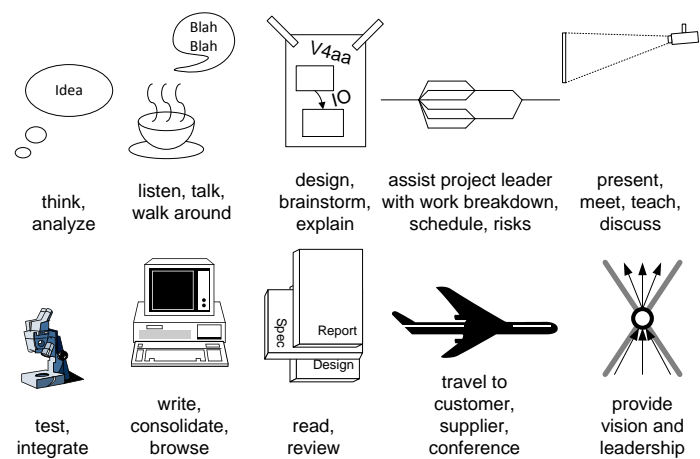
Role and Task of a System Architect

Deliverables

Report
Report
Report

Spec
Spec

Design
Design
Design

Responsibilities

Requirement
Spec
Design
Realization

module
subsystem
system

Quality
Function
modules

Balance   **Consistency**   **Decomposition Integration**   **Overview**

KISS

satisfied stakeholders
system
context

Elegance Simple   **Integrity**   **Fitting**

Daily Activities

Idea

Blah Blah

V4aa

think, analyze

listen, talk, walk around

design, brainstorm, explain

assist project leader with work breakdown, schedule, risks

present, meet, teach, discuss

test, integrate

write, consolidate, browse

read, review

travel to customer, supplier, conference

provide vision and leadership

Spec
Report
Design

From detail to overview

		Quantity per year (order-of-magnitude)	architect time per item
consolidation in deliverables	driving views	10	100 h
	shared issues	10^2	1 h
meetings	touched details	10^4	0.5 – 10 min
informal contacts	seen details	$10^5 - 10^6$	0.1 – 1 sec
sampling scanning	product details	$10^7 - 10^{10}$	
	real-world facts	infinite	

USN   ESI

Personal characteristics of a System Architect

Generalist vs Specialist

Typical growth of a Architect

- root technical knowledge
- generalist technical knowledge
- business, application insight
- process insight
- psychosocial skills
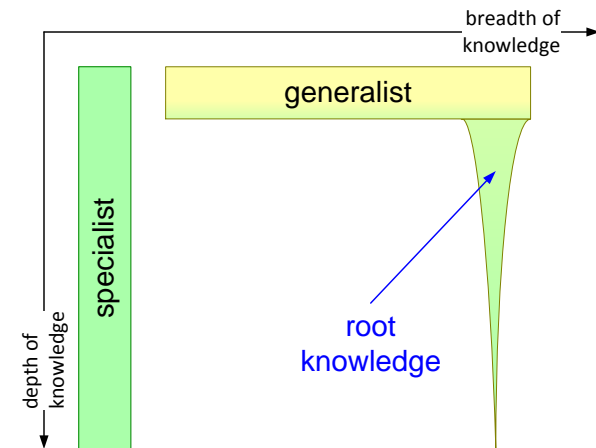
Complementary Roles
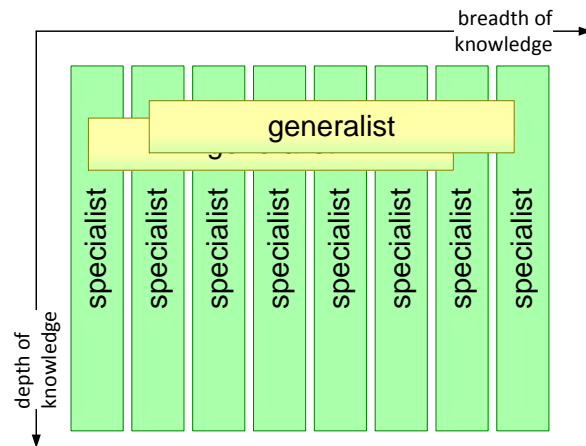
Role Spectrum
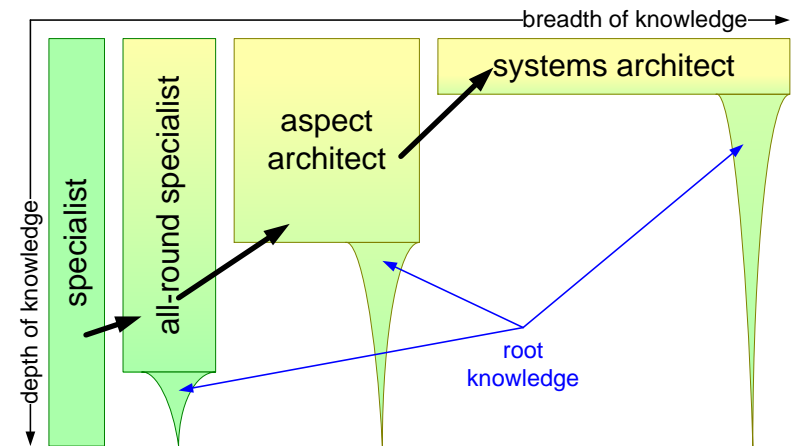
Module Supporting Processes

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

This module addresses supporting processes, for instance documentation, templates, and reviewing.

January 22, 2023
status: draft
version: 1.4

Granularity of Documentation

by *Gerrit Muller*   USN-SE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

The design of documentation is discussed, with emphasis on the requirements, the need for decomposition, the measures needed to maintain overview and criteria for granularity.

January 22, 2023
status: concept
version: 1.2
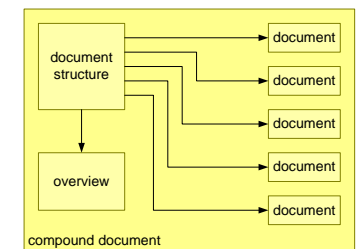
Requirements for the Entire Documentation Structure

Accessibility for the readers

Low threshold for the readers

Low threshold for the authors

Completeness

Consistency

Maintainability

Scalability

Evolvability

Process to ensure the quality of the information

Convenient

easy
fast

viewing

printing

searching

Requirements per Document

High cohesion (within the unit)

Low coupling (outside of the unit)

Accessibility for the readers

Low threshold for the reader

Low threshold for the author

Manageable steps to create, review, and change

Clear responsibilities

Clear position and relation with the context

Well-defined status of the information

Timely availability

Accessibility Requirements

Ease of reading, "juiciness"

High signal-to-noise ratio: information should not be hidden in a sea of words.

Understandability

Reachability in different ways, e.g., by hierarchical or full search

Reachability in a limited number of steps

Responsibility Requirements

single author

limited amount of reviewers

well defined documentation structure

overview specifications at higher aggregation levels

recursive application of structure and overview

delegation of review process

The Stakeholders of a Single Document

Project leader

architect or editor

author

consumer

specification

implementation

producer

context

others

is responsible for time, budget, result

is responsible for technical

interacts with

interacts with

writes

describes

uses

realizes

legend

relation

artifact

stake-holder

USN   ESI
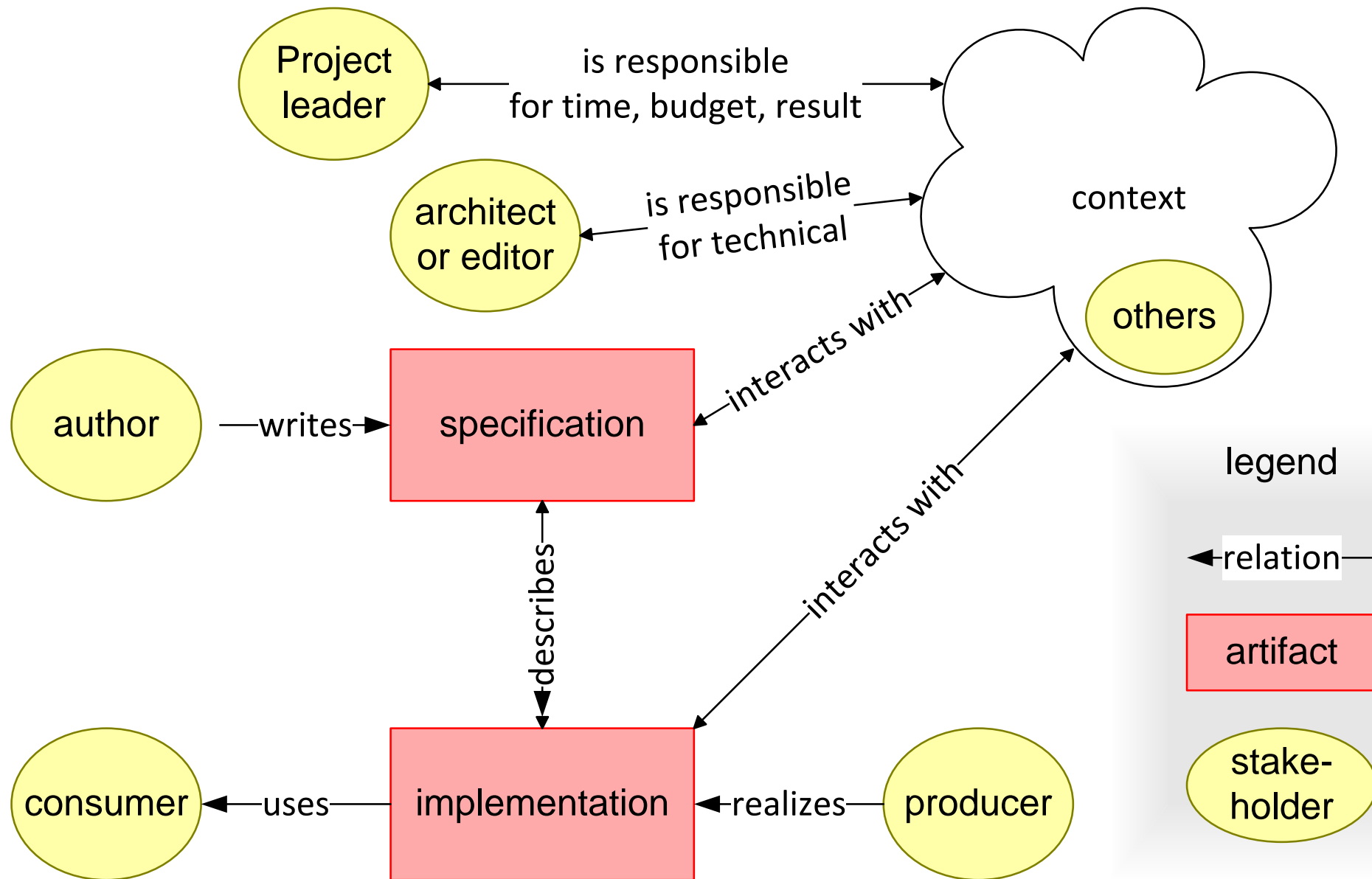
Decomposition of Large Documents
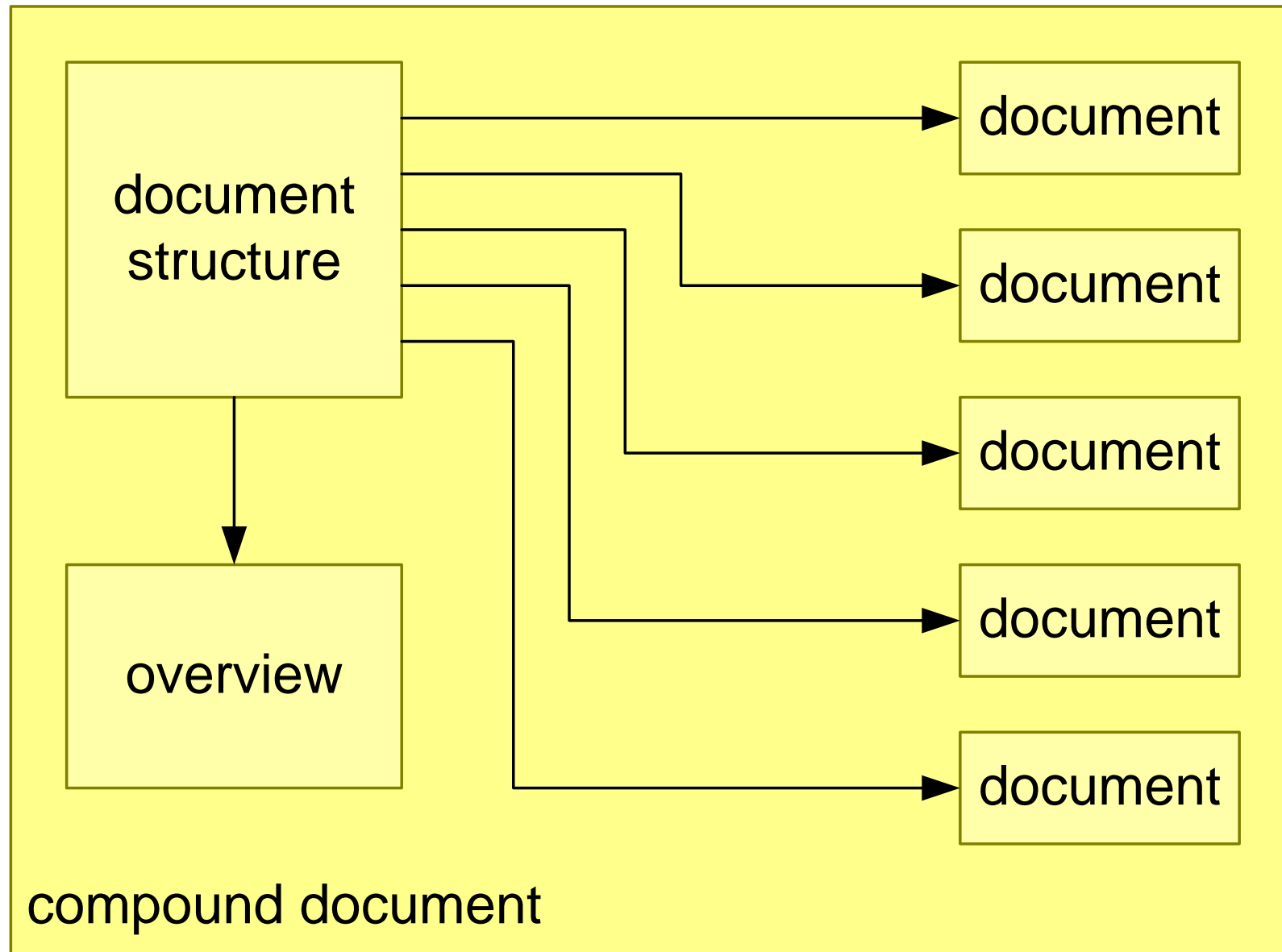
Documentation Tree by Recursive Decomposition
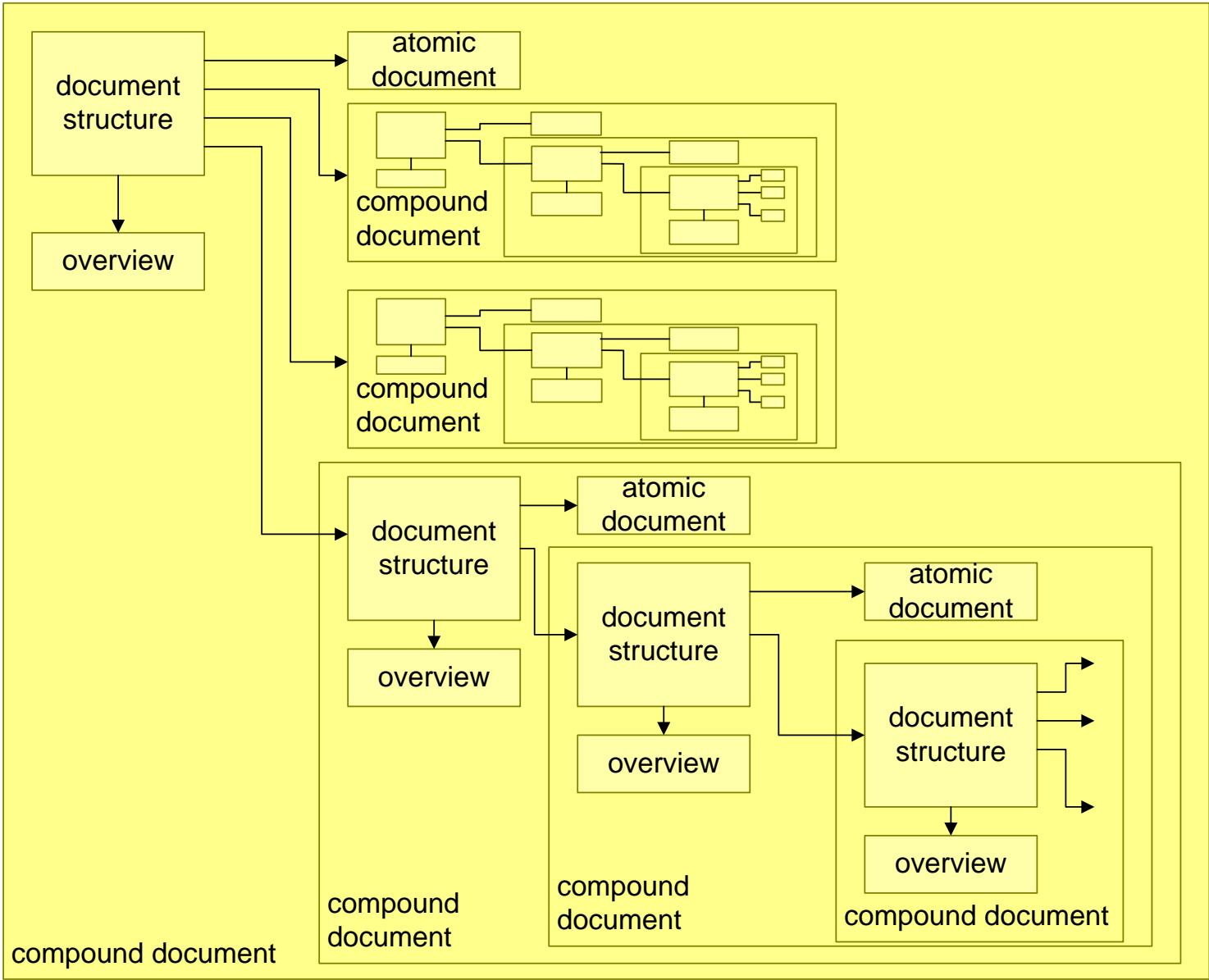
Payload: the Ratio between Content and Overhead

front page

| title
identification
author
distribution
status
review | history
changes | diagrams

tables | 1. aap
2. noot
3. mies

lists

and ca 50%
text |

meta information
max 2 pages

contents
2..18 pages

USN ESI
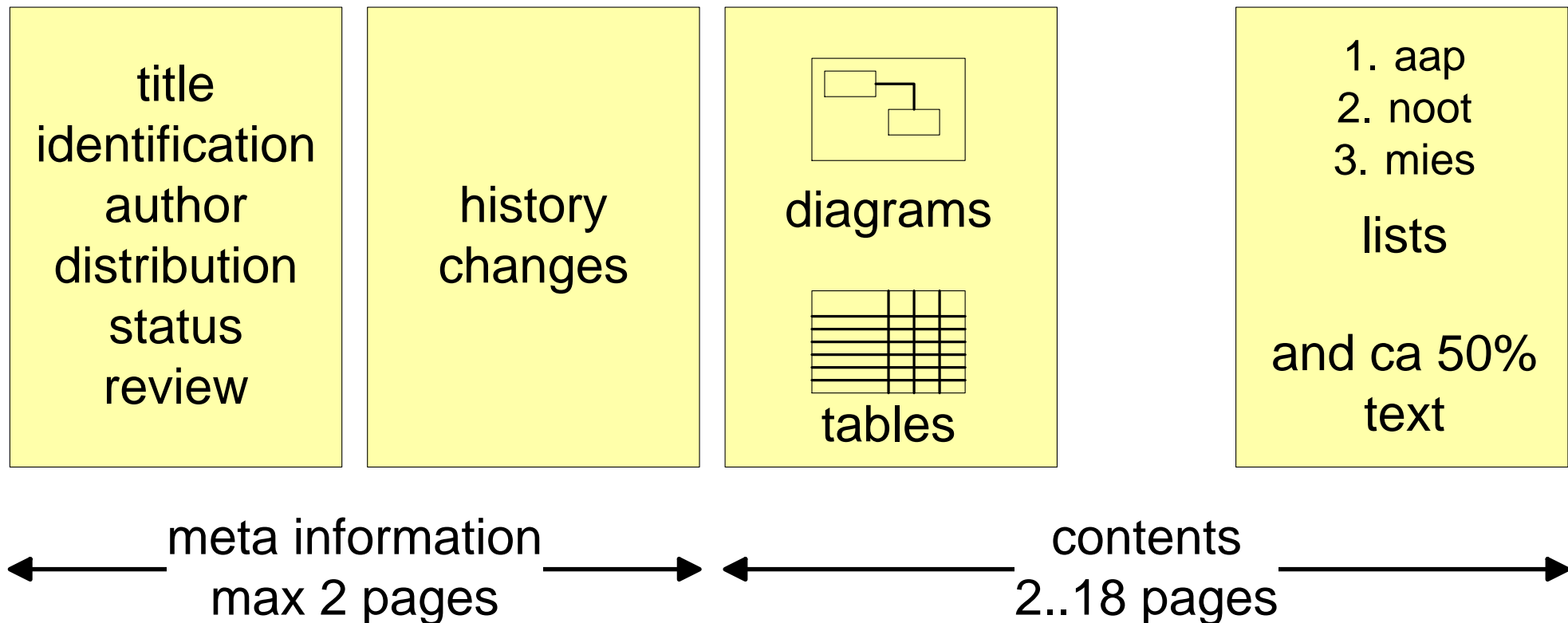
LEAN and A3 Approach to Supporting Processes

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: gaudisite@gmail.com

www.gaudisite.nl

Abstract

LEAN product development is in the process and means area pragmatic. Low tech tools, such as paper, pen and magnets, with very direct interaction are used. For communication the use of single A3-size documents is promoted, because this is a manageable amount of information.

January 22, 2023
status: draft
version: 0.1
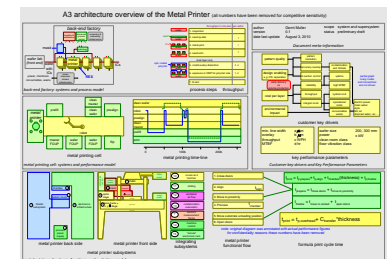
Characteristics of LEAN

A holistic, systems approach to product development including people, processes, and technology.

Multi-disciplinary from the early start, with a drive to be fact based.

Customer understanding as the the starting point.

Continuous improvement and learning as cultural value.

Small distance between engineers and real systems, including manufacturing, sales and service and the system of interest.

Example of A3 Architecture Overview

A3 architecture overview of the Metal Printer (all numbers have been removed for competitive sensitivity)

back-end factory: systems and process model

back-end factory

logistics & automation · advanced process control · computing & networking infrastructure

metrology

mask

wafer fab (front end)

wafer with ICs

power, chemicals consumables, waste

power chemicals climate infrastructure

metal printer

REX

ICs

process steps — throughput

	process steps	throughput in minutes per wafer
wafer	1. inspection	
seed — wafer	2. seed sputter	1
Cu — wafer	3. metal print	t
wafer	4. seed etch	1
	dual layer only	
spin coated polymer — wafer	5. coat/develop dielectrics	3..4
wafer	6. exposure or CMP for polymer vias	1..2
	7. E-test	

Document meta-information

author	Gerrit Muller	scope	system and supersystem
version	0.1	status	preliminary draft
date last update	August 3, 2010		

metal printing cell: systems and performance model

metal printer

clean master · clean wafer · prealign

prefill

robot

master FOUP · wafer FOUP · wafer FOUP · flip
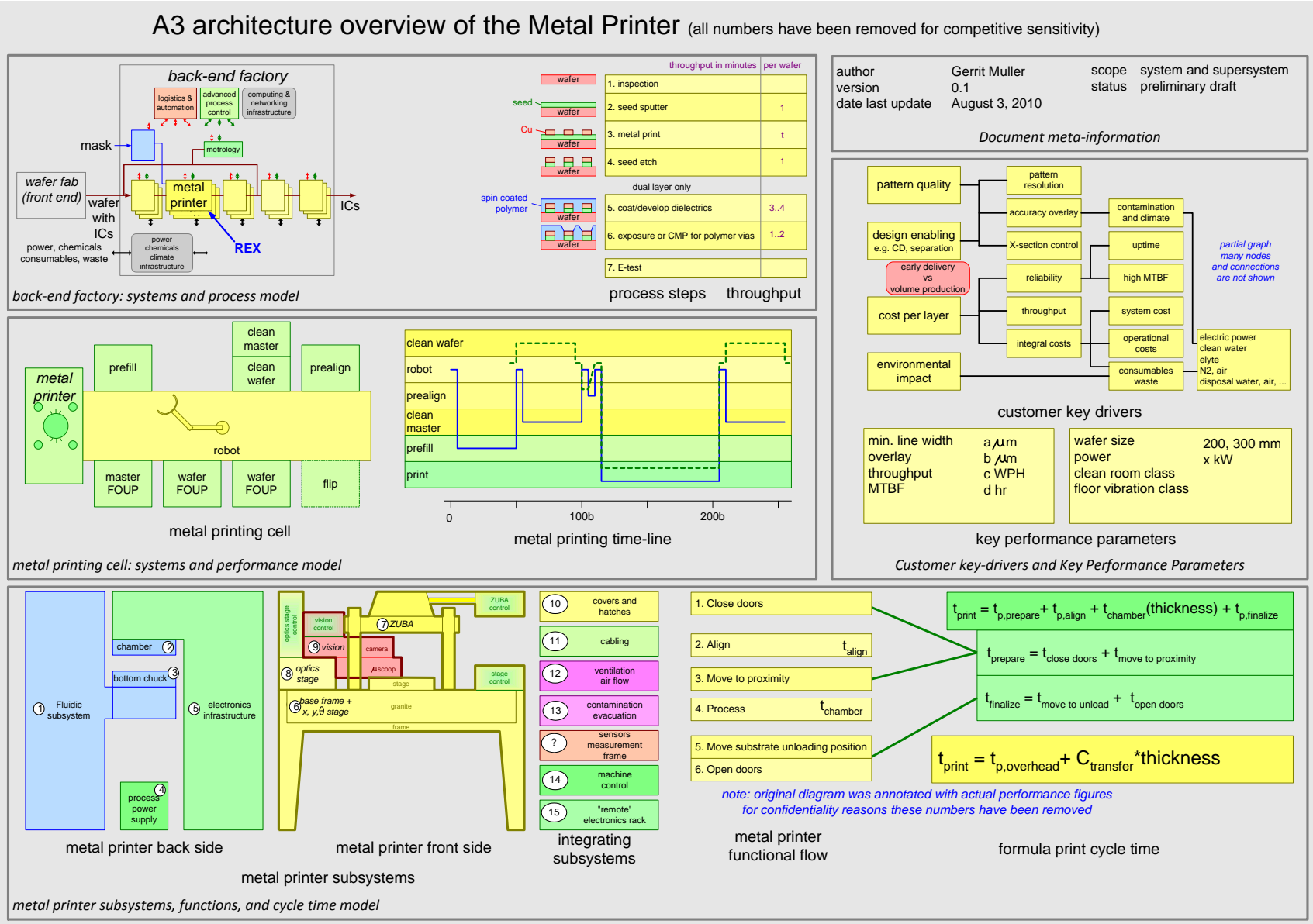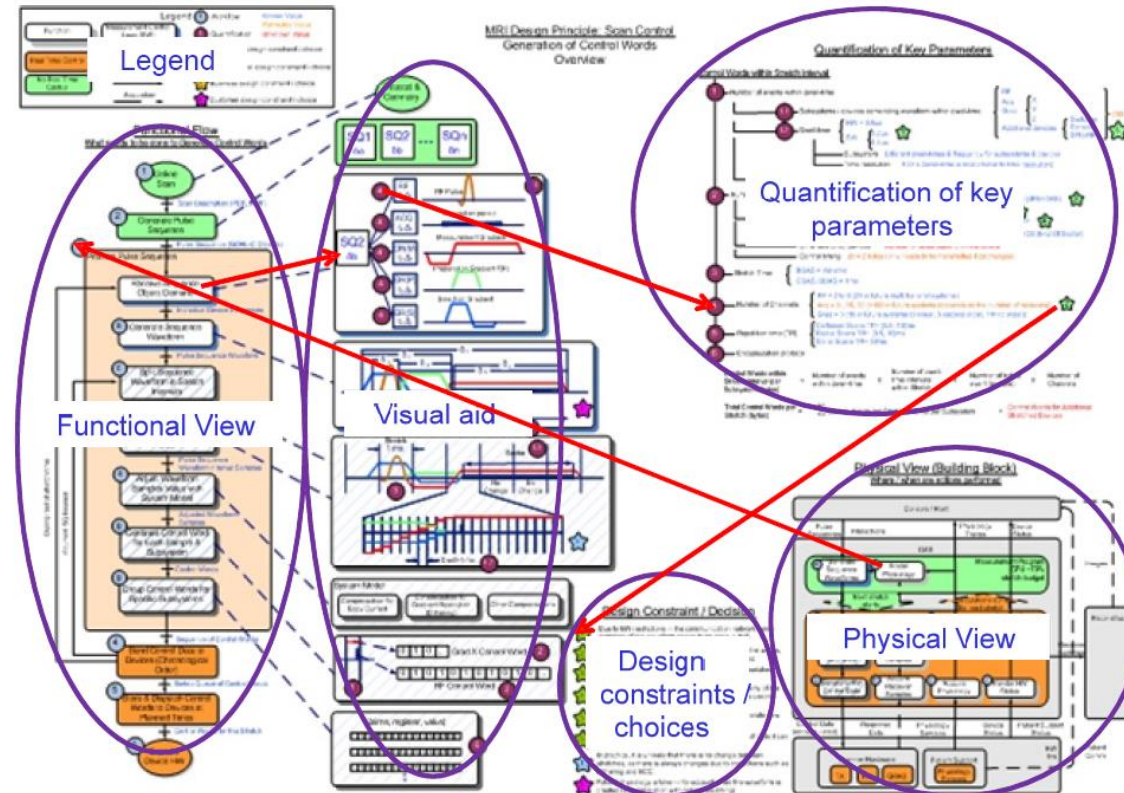
metal printing cell

metal printing time-line: clean wafer, robot, prealign, clean master, prefill, print

0   100b   200b

Customer key-drivers and Key Performance Parameters

customer key drivers

- pattern quality → pattern resolution
- design enabling e.g. CD, separation → accuracy overlay → contamination and climate
- early delivery vs volume production → X-section control → uptime
- reliability → high MTBF
- cost per layer → throughput → system cost
- integral costs → operational costs
- environmental impact → consumables waste

electric power, clean water, elyte, N2, air, disposal water, air, ...

partial graph many nodes and connections are not shown

key performance parameters

min. line width	a µm	wafer size	200, 300 mm
overlay	b µm	power	x kW
throughput	c WPH	clean room class	
MTBF	d hr	floor vibration class	

metal printer subsystems, functions, and cycle time model

metal printer back side

① Fluidic subsystem · ② chamber · ③ bottom chuck · ④ process power supply · ⑤ electronics infrastructure

metal printer front side

optics stage control · vision control · ⑦ ZUBA · ZUBA control · ⑨ vision · camera · ⑧ optics stage · µscoop · stage control · ⑥ base frame + x, y, θ stage · granite · frame

metal printer subsystems

integrating subsystems

⑩	covers and hatches
⑪	cabling
⑫	ventilation air flow
⑬	contamination evacuation
?	sensors measurement frame
⑭	machine control
⑮	"remote" electronics rack

metal printer functional flow

1. Close doors
2. Align — t_{align}
3. Move to proximity
4. Process — $t_{chamber}$
5. Move substrate unloading position
6. Open doors

formula print cycle time

$$t_{print} = t_{p,prepare} + t_{p,align} + t_{chamber}(thickness) + t_{p,finalize}$$

$$t_{prepare} = t_{close\ doors} + t_{move\ to\ proximity}$$

$$t_{finalize} = t_{move\ to\ unload} + t_{open\ doors}$$

$$t_{print} = t_{p,overhead} + C_{transfer} * thickness$$

note: original diagram was annotated with actual performance figures for confidentiality reasons these numbers have been removed

multiple related views   quantifications

one topic
per A3

Legend

Quantification of key
parameters

Functional View   Visual aid

capture
"hot" topics

Design
constraints /
choices

Physical View

source: PhD thesis Daniel Borches http://doc.utwente.nl/75284/

digestable   practical
(size limitation)   close to stakeholder experience

Light Weight Review Process

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

A light weight review process is described that can be used for documents made during product creation. This review process is focused on improving the contents of specifications as early as possible. The process is light weight to increase the likelihood that it is performed *de facto* instead of *pro forma*.
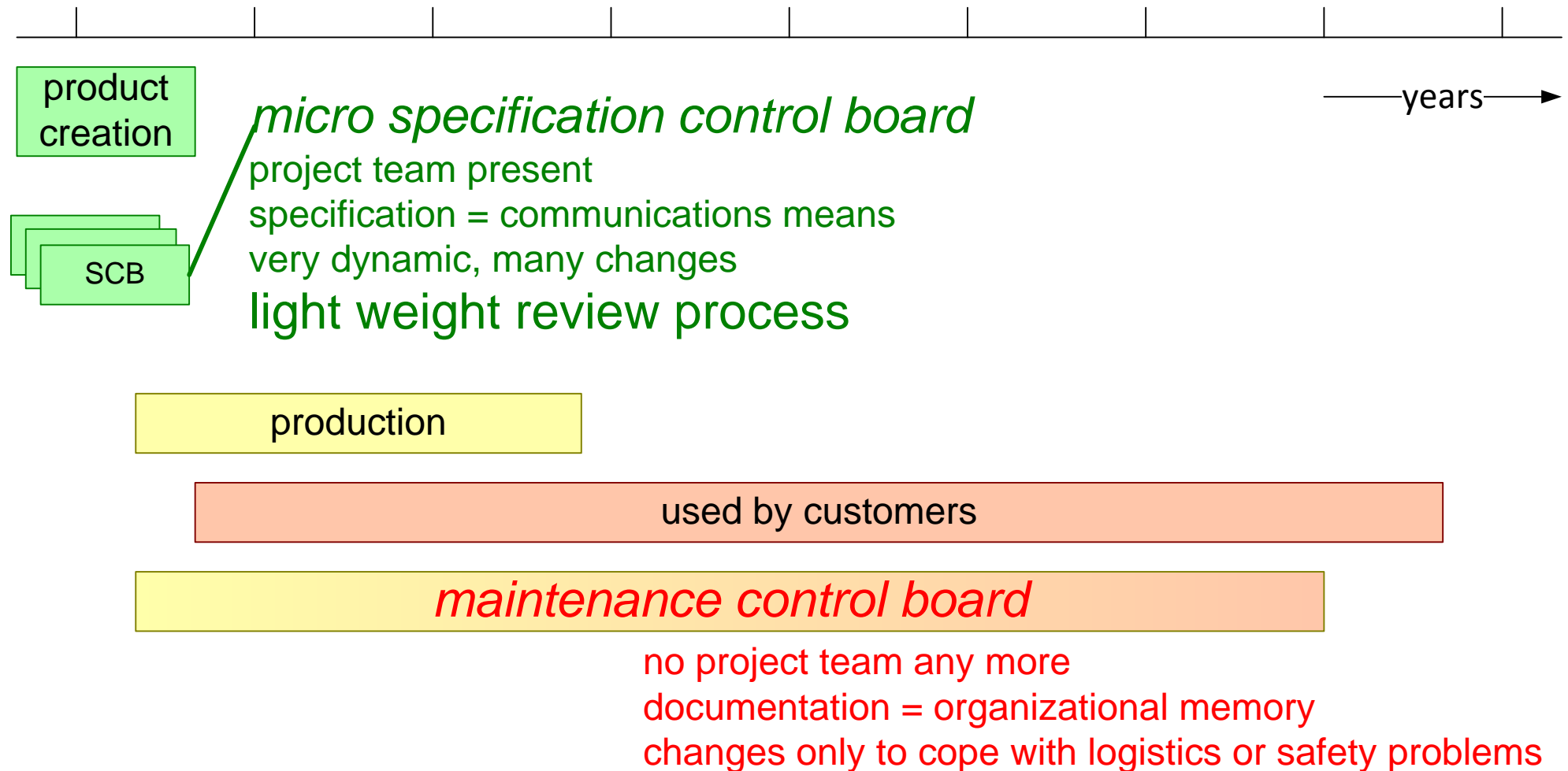
January 22, 2023
status: preliminary
draft
version: 0
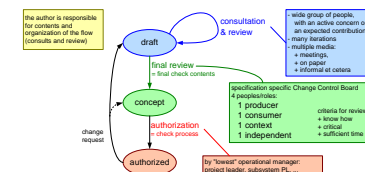
Product Life Cycle and Change Management

product creation

SCB

micro specification control board

project team present
specification = communications means
very dynamic, many changes
light weight review process

production

used by customers

maintenance control board

no project team any more
documentation = organizational memory
changes only to cope with logistics or safety problems

years

USN   ESI

Light Weight Specification Review Process

the author is responsible
for contents and
organization of the flow
(consults and review)

draft

**consultation
& review**

- wide group of people,
  with an active concern or
  an expected contribution;
- many iterations
- multiple media:
  + meetings,
  + on paper
  + informal et cetera

final review
= final check contents

concept

specification specific Change Control Board
4 peoples/roles:

    1 producer
    1 consumer
    1 context
    1 independent

criteria for reviewers:
+ know how
+ critical
+ sufficient time

change
request

authorization
= check process

authorized

by "lowest" operational manager:
project leader, subsystem PL, ...

USN ESI

Template How To

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

The introduction of a new process (way of working) is quite often implemented by supplying ready-to-go tools and templates. This implementation mainly serves the purpose of a smooth introduction of the new process.

Unfortunately the benefits of templates are often cancelled by unforeseen side-effects, such as unintended application, inflexibility, and so on. This intermezzo gives hints to avoid the **Template Trap**, so that templates can be used more effectively to support introduction of new processes.

January 22, 2023
status: draft
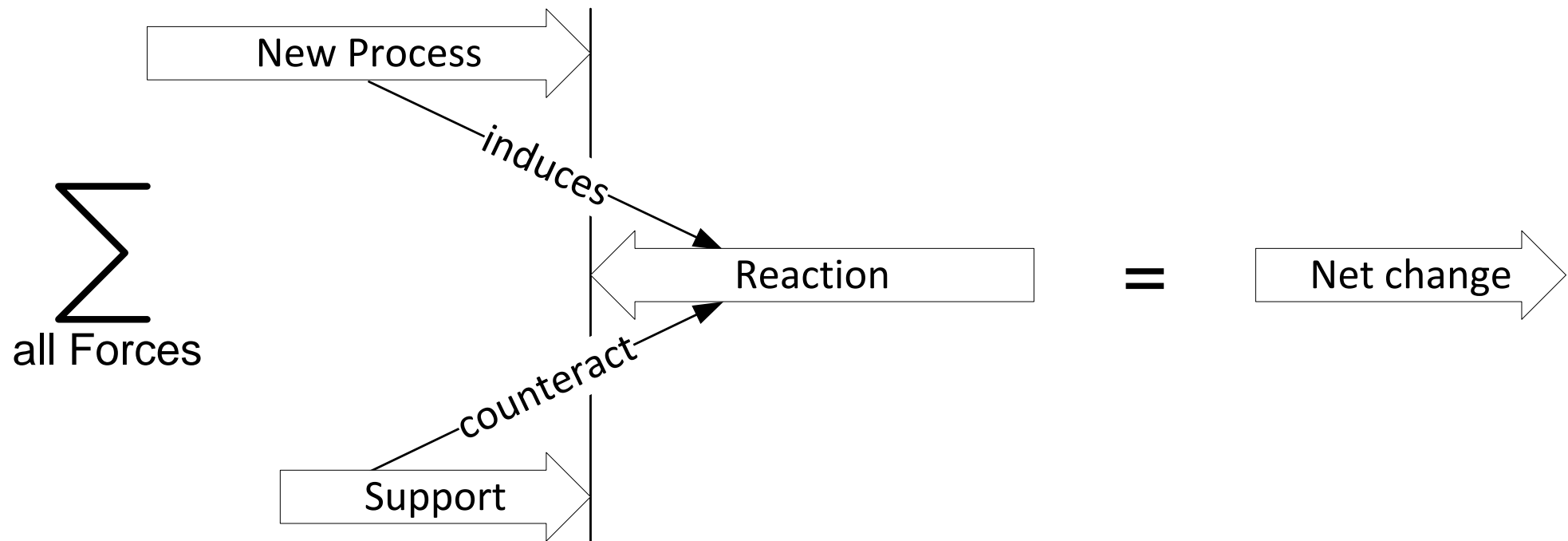version: 1.6
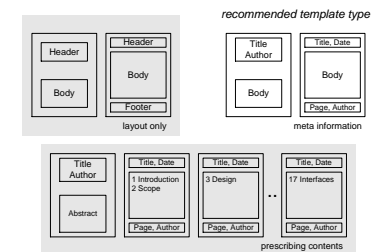
Rationale for Templates

- Low threshold to apply a (new) process   (1)

- Low effort to apply a (new) process   (2)

- No need to know low level implementation details   (3)

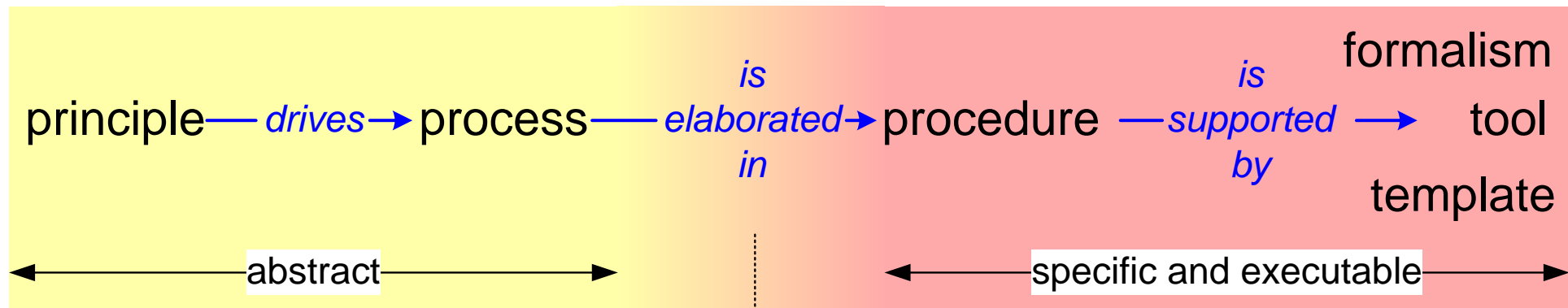- Means to consolidate and reuse experiences   (4)

Bogus Arguments for Templates

- Obtain a uniform look (5)

- Force the application of a (new) process (6)

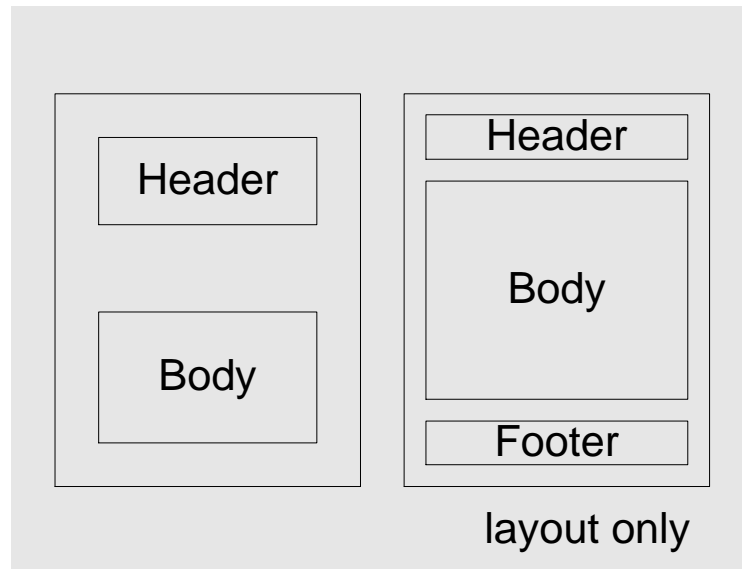- Control the way a new process is applied (7)
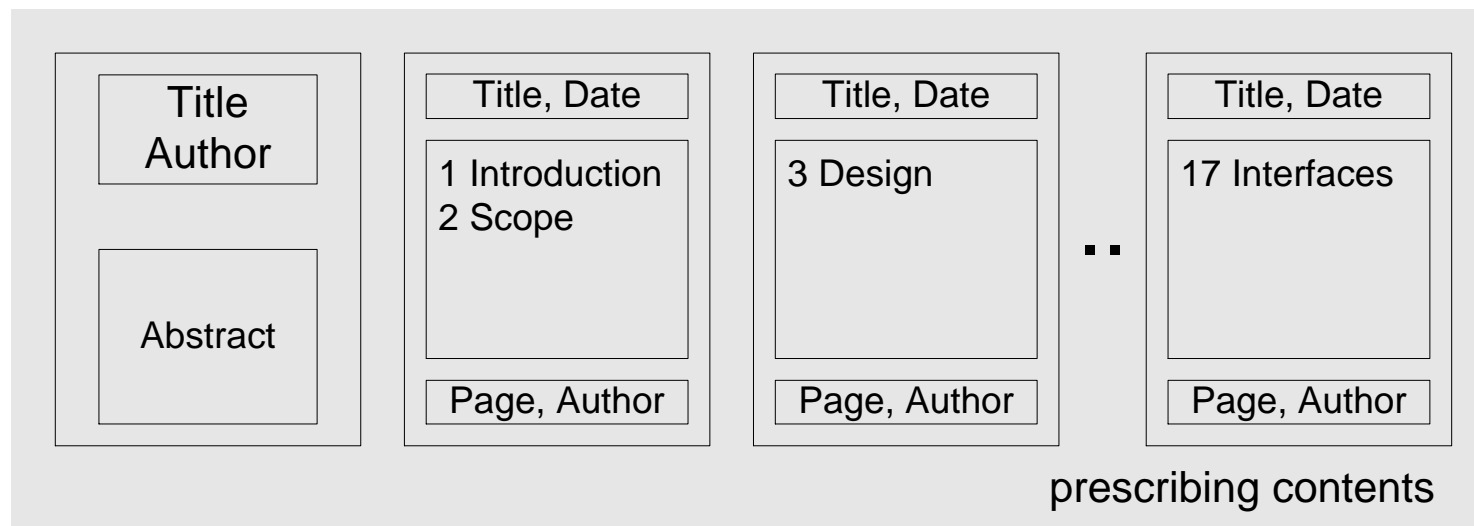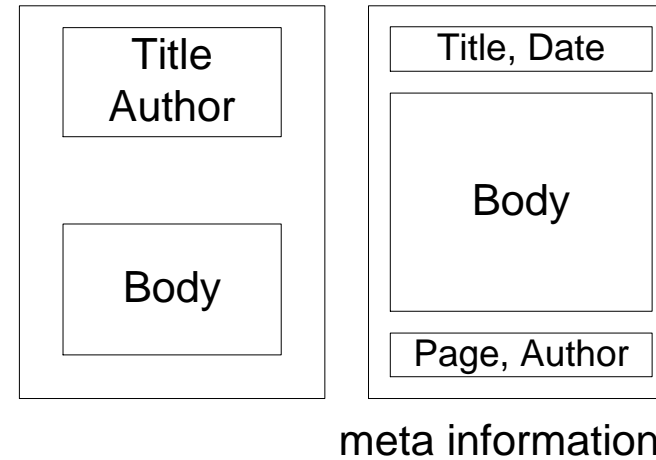
Forces of Change: Action = - Reaction

$$\sum_{\text{all Forces}}$$

New Process

induces

Reaction

counteract

Support

= Net change

Template as Support for Process

principle ——*drives*→ process ——*is elaborated in*→ procedure ——*is supported by*→ formalism / tool / template

←——— abstract ———→   ←——— specific and executable ———→

Types of Templates

recommended template type

Header

Body

Header
Body
Footer

layout only

Title
Author

Body

Title, Date

Body

Page, Author

meta information

Title
Author

Abstract

Title, Date

1 Introduction
2 Scope

Page, Author

Title, Date

3 Design

Page, Author

..

Title, Date

17 Interfaces

Page, Author

prescribing contents

Recommendation

template type	context knowhow	value
layout only	no	low
meta information	process	high
prescribing content	process and domain	constraining

- Use templates for meta-information.

- Use checklists for structure and contents.

Template Development

Templates are an optimization of the Copy Paste Modify pattern:

- Look for a similar problem

- Copy its implementation

- Modify the copy to fulfil the new requirements

Spiral model: Use before Re-use

Extract template

Implement document

Evaluate

Use
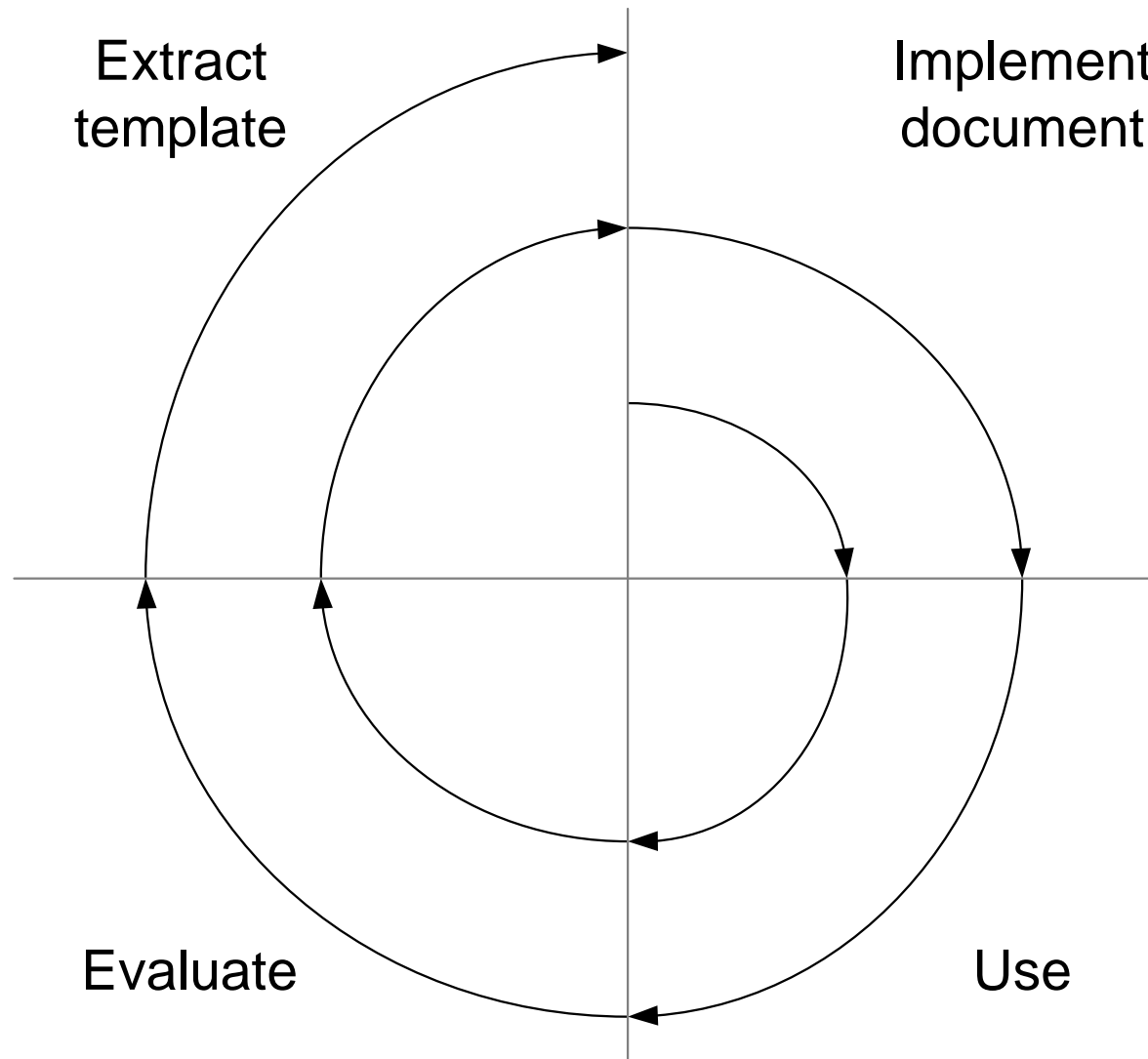
Example Guidelines Meta Information(1)

Mandatory per page:

- Author

- Title

- Status

- Version

- Date of last update

- Unique Identification

- Business Unit

- Page number

Example Guidelines Meta Information(2)

Mandatory per document:

- Distribution (Notification) list

- Reviewers and commentators

- Document scope (Product family, Product, Subsystem, Module as far as applicable)

- Change history

Example Guidelines Meta Information(3)

Recommended Practice:

- Short statement on frontpage stating what is expected from the addressed recipients, for example:

  - Please send comments before february 29, this document will be reviewed on that date

  - This document is authorized, changes are only applied via a change request

- See Granularity of Documentation [**?**] for guidelines for modularization and contents

Template Pitfalls

- Author follows template instead of considering the purpose of the document.

- Template is too complex.

- There is an unmanageable number of variants.

- Mandatory use of templates results in:
  - no innovation of templates (= no learning)

  - no common sense in deployment

  - strong dependency on templates

Recommendation:
- Enforce the procedure *(what)*

- Provide the template *(how)* as supporting means.

Summary

- Templates support (new) processes

- Use templates for layout and meta information support

- Do not use templates for documents structure or contents

- Stimulate evolution of templates, keep them alive

- Keep templates simple

- Standardize on **what** (process or procedure), not on **how** (tool and template)

- Provide (mandatory) guidelines and recommended practices

- Provide templates as a supportive choice, don't force people to use templates

System Integration How-To

by *Gerrit Muller*   USN-SE

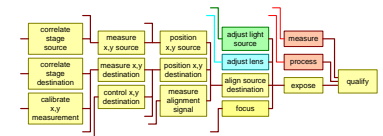e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

In this document we will discuss the full integration flow. We will discuss the goal of integration, the relation between integration and testing, what is integration and how to integrate, an approach to integration, scheduling and dealing with disruptive events, roles and responsibilities, configuration management aspects, and typical order of integration problems occurring in real life.
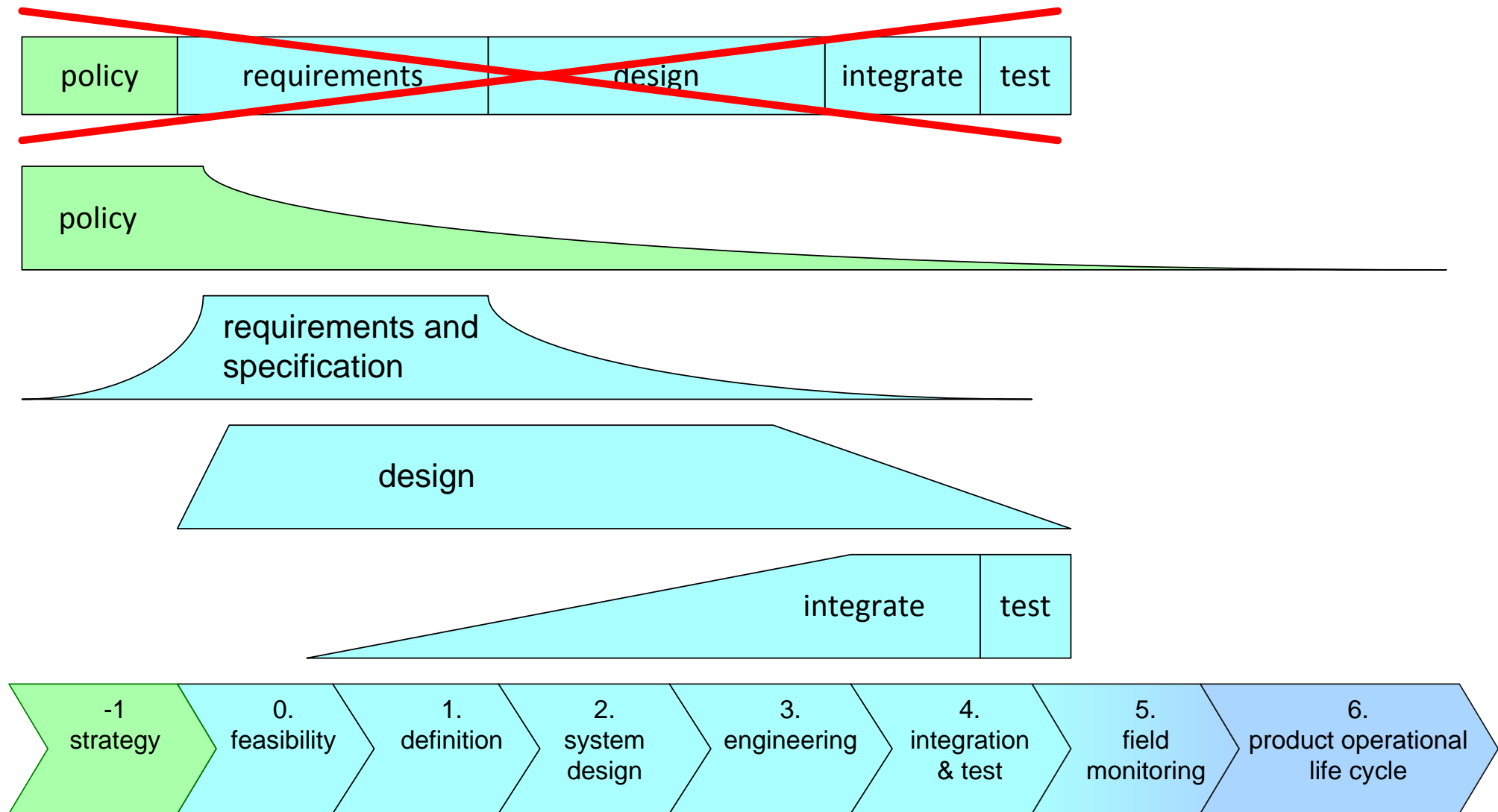
January 22, 2023
status: concept
version: 0.2

Typical Concurrent Product Creation Process

policy | requirements | design | integrate | test

policy

requirements and specification

design

integrate | test

-1 strategy | 0. feasibility | 1. definition | 2. system design | 3. engineering | 4. integration & test | 5. field monitoring | 6. product operational life cycle

Zooming in on Integration and Tests

integrate | system test | alpha test | beta test | gamma test

0. feasibility | 1. definition | 2. system design | 3. engineering | 4. integration & test | 5. field monitoring | 6. product operational life cycle
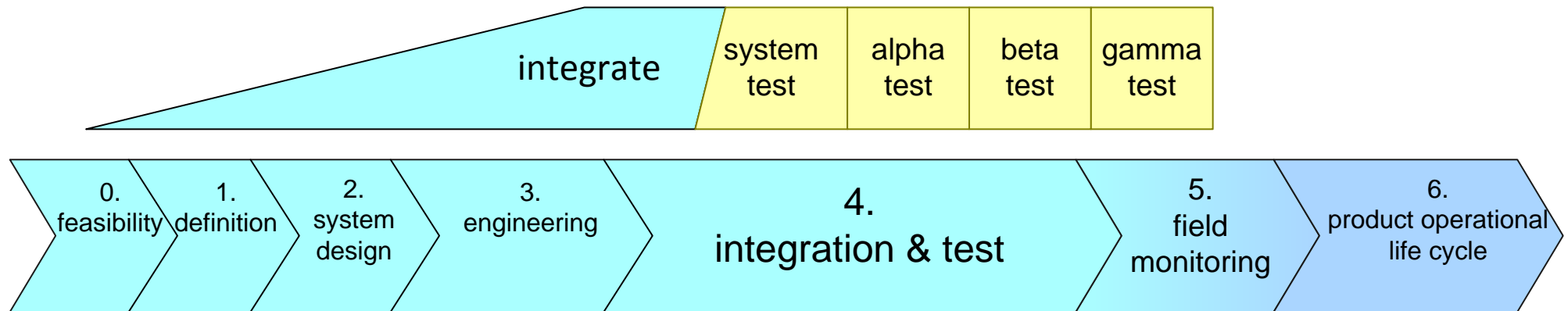
Integration Takes Place in a Bottom-up Fashion

integrate

alpha test

component

subsystem

system function

product

context
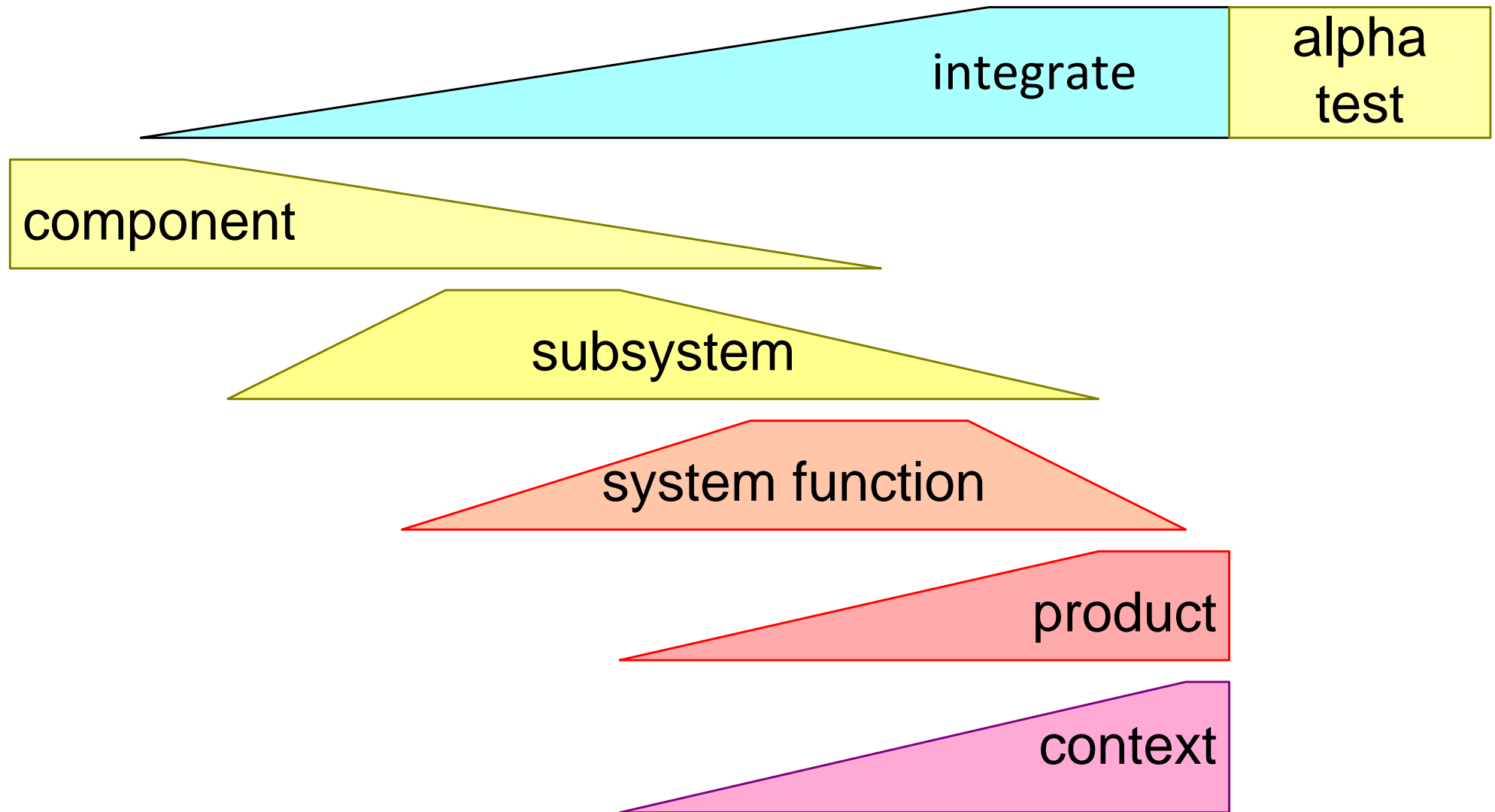
Transition from Previous System to New System

existing   new

adopt existing base SW

new application

SW dev system

test and refine application

application integration

integrate and refine application

existing base system

SW for new HW subsystem

2 partial systems for SW testing

test SW for new HW subsystem

SW dev system

new subsystem integration

2 existing base systems

new HW subsystem

integrate subsystem

existing base system

test HW subsystem

integrated system

adopt existing base SW

new base system

test new base system

integrate HW system

integrate system

new base systems

time

USN   ESI

Alternatives to Integrate a Subsystem Early in the Project

← spectrum →

physical reality — **physical simulated** — **complex virtual** — **simple**

virtual environment

simulated subsystems | stubs

physical environment

(modified) existing subsystems

(prototype) new subsystems

to-be-integrated subsystem

USN   ESI
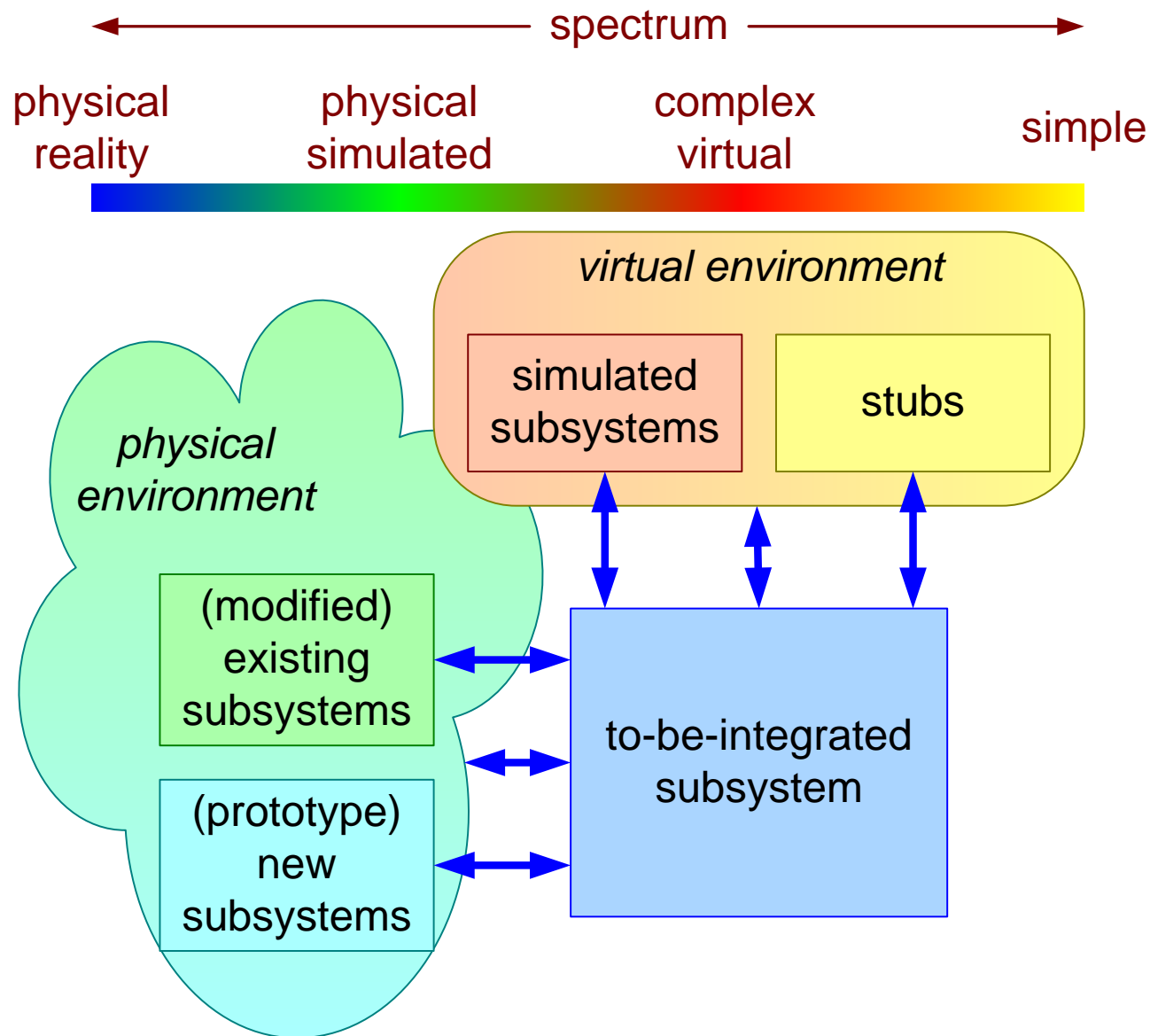
Stepwise Integration Approach

1	Determine most critical system performance parameters.
2	Identify subsystems and functions involved in these parameters.
3	Work towards integration configurations along these chains of subsystems and functions.
4	Show system performance parameter as early as possible; start with showing "typical" system performance.
5	Show "worst-case" and "boundary" system performance.
6	Rework manual integration tests in steps into automated regression tests.
7	Monitor regression results with human-driven analysis.
8	Integrate the chains: show system performance of different parameters simultaneously on the same system.

Order of Functions Required for the IQ of a Waferstepper
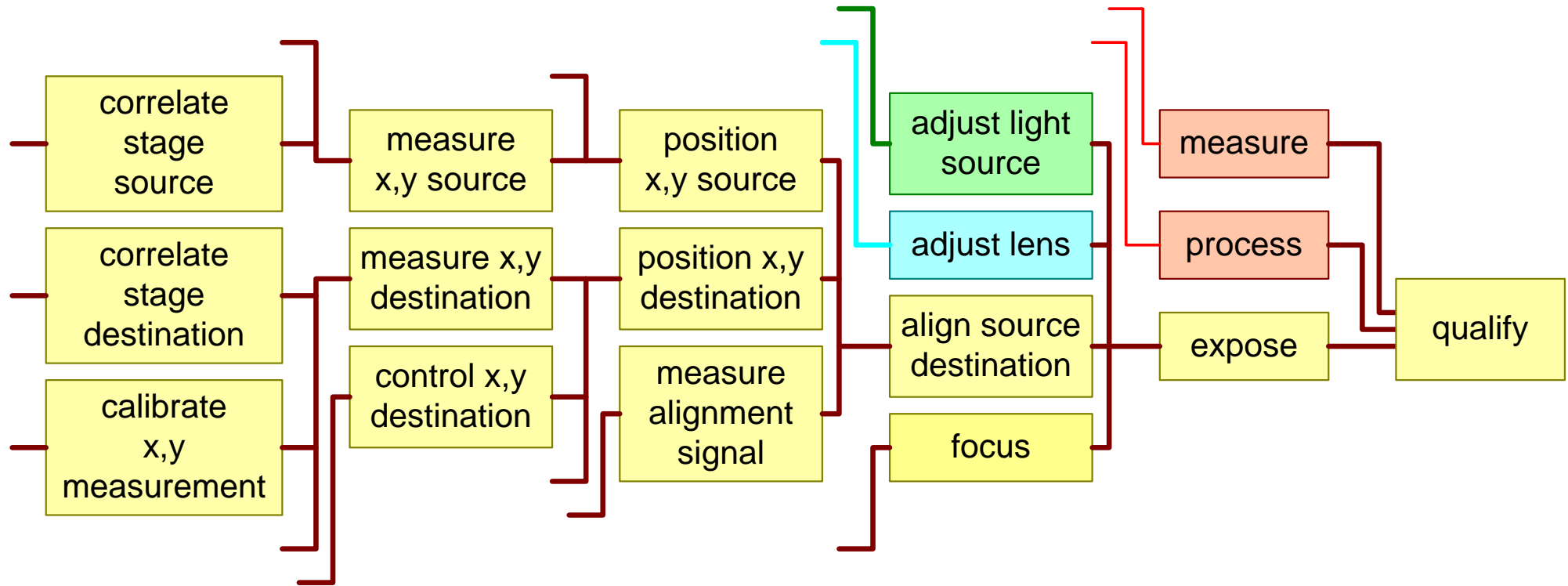
Roles and Responsibilities During the Integration Process

project leader

organization
resources
schedule
budget

**systems architect/
engineer/integrator**
system requirements
design inputs
test specification
schedule rationale
troubleshooting
participate in test

system tester

test
troubleshooting
report

**logistics and
administrative support**
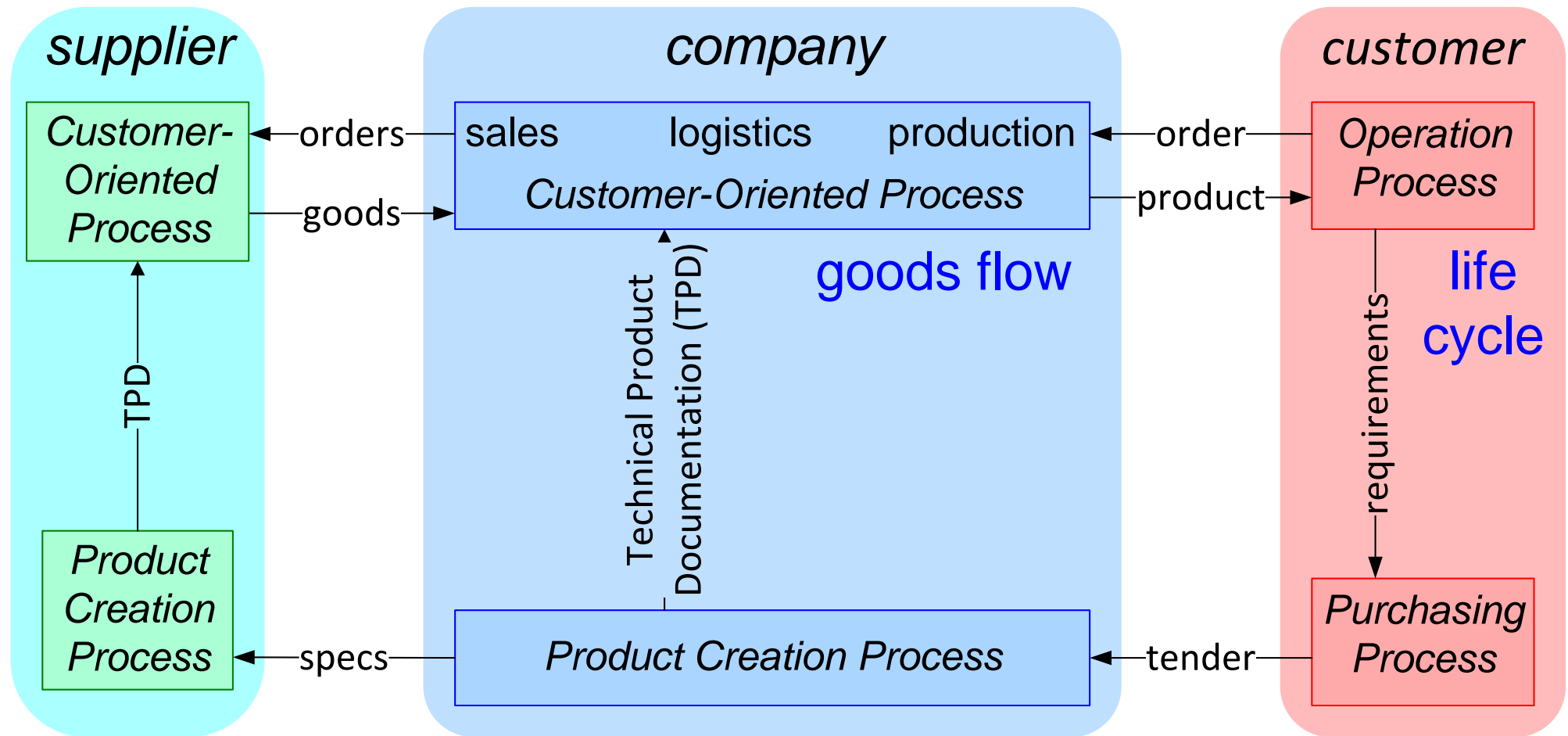configuration
orders
administration

engineers
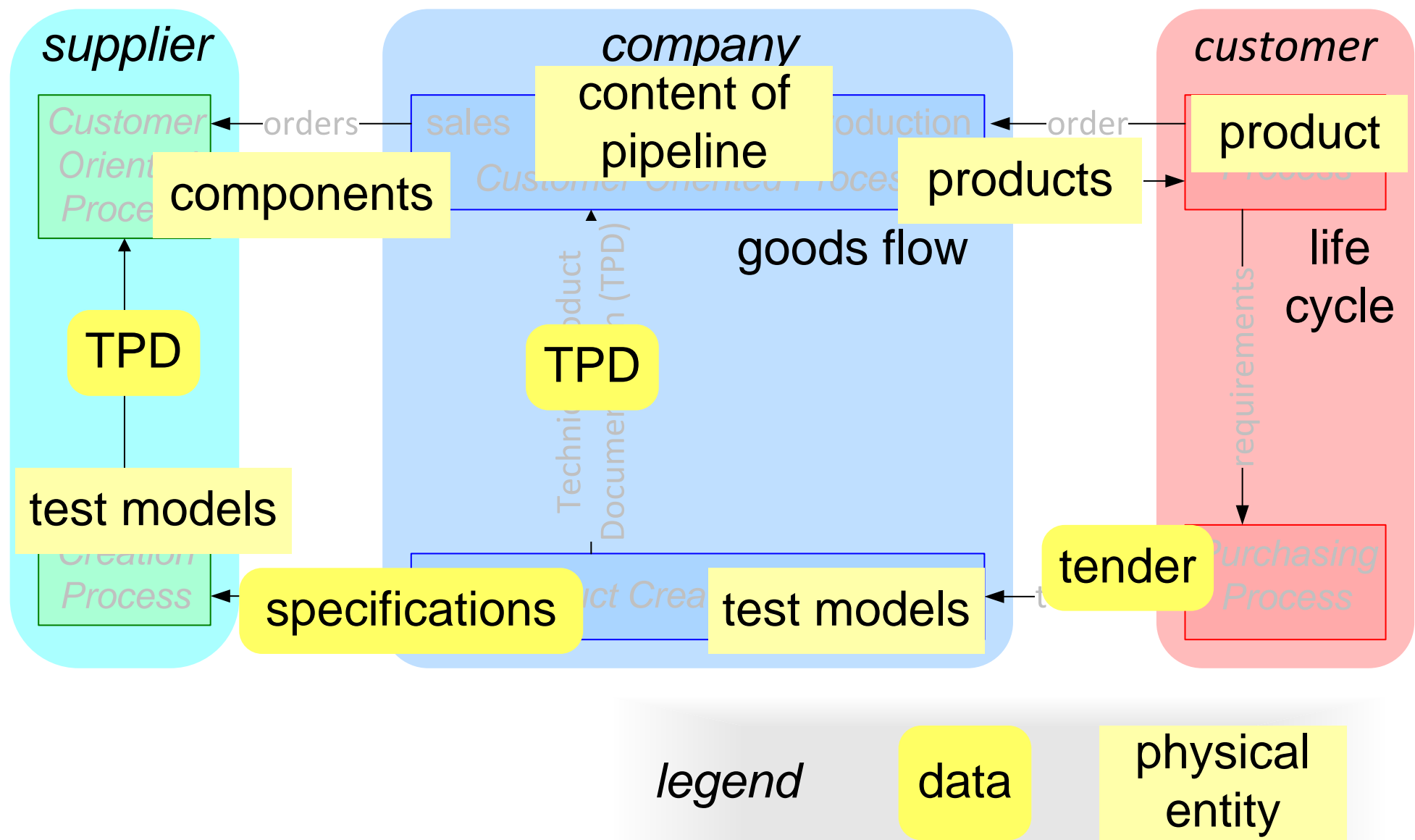
design
component test
troubleshooting
participate in test

machine owner

maintain test model
support test

USN ESI

Simplified Process Diagram

Configuration Management Entities

legend — data — physical entity

Typical Order of Integration Problems

1. The (sub)system does not build.

2. The (sub)system does not function.

3. Interface errors.

4. The (sub)system is too slow.

5. Problems with the main performance parameter, such as image quality.

6. The (sub)system is not reliable.

Exercise Documentation

Make a design for the documentation structure of the case, take into account a.o.:

- target audience per documentation module

- lifecycle

- author

- size (budget)

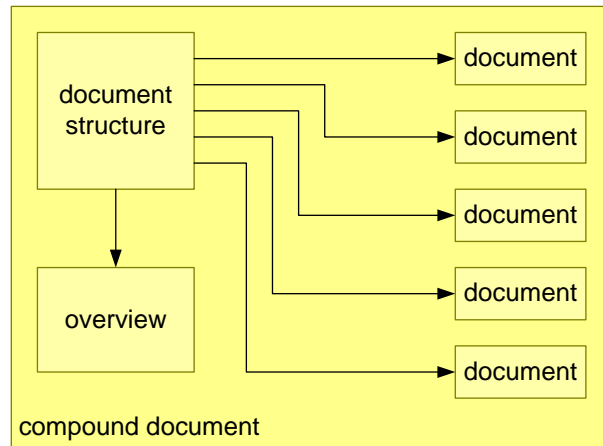Present (max 1 flip) the proposed documentation structure and the rationale.

Documentation

Requirements Entire Documentation

Accessibility for the readers

Low threshold for the readers

Low threshold for the authors

Completeness

Consistency

Maintainability

Scalability

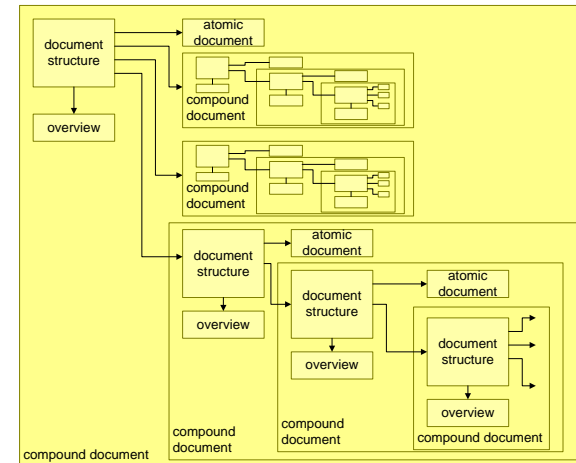Evolvability

Process to ensure the quality of the information

Requirements per Document

High cohesion (within the unit)

Low coupling (outside of the unit)

Accessibility for the readers

Low threshold for the reader

Low threshold for the author

Manageable steps to create, review, and change

Clear responsibilities

Clear position and relation with the context

Well-defined status of the information
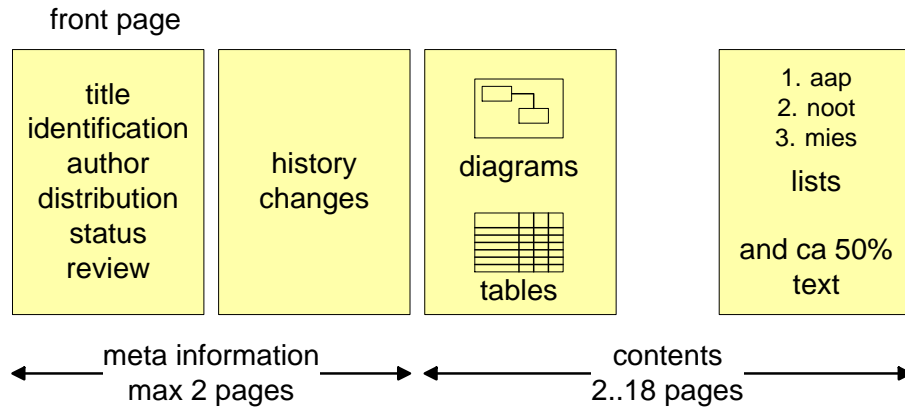
Timely availability

Decompose Large Documents
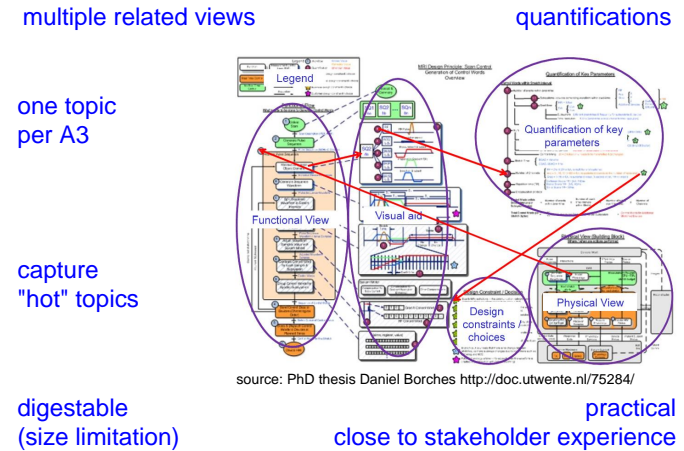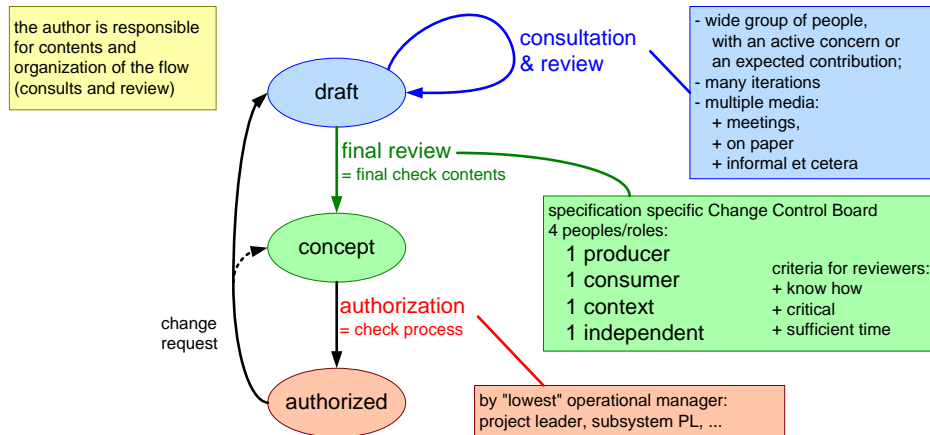

Recursive Decomposition

version: 0.2
January 22, 2023

Documentation

Maximize Payload

front page

title identification author distribution status review	history changes	diagrams tables	1. aap 2. noot 3. mies lists and ca 50% text

← meta information →
max 2 pages

← contents →
2..18 pages

A3s

multiple related views   quantifications

one topic
per A3

capture
"hot" topics

source: PhD thesis Daniel Borches http://doc.utwente.nl/75284/

digestable   practical
(size limitation)   close to stakeholder experience

Light Weight Review

the author is responsible
for contents and
organization of the flow
(consults and review)

draft

consultation
& review

- wide group of people,
   with an active concern or
   an expected contribution;
- many iterations
- multiple media:
   + meetings,
   + on paper
   + informal et cetera

final review
= final check contents

concept

specification specific Change Control Board
4 peoples/roles:
   1 producer
   1 consumer   criteria for reviewers:
   1 context   + know how
   1 independent   + critical
                       + sufficient time

change
request

authorization
= check process

authorized

by "lowest" operational manager:
project leader, subsystem PL, ...

intentionally left blank

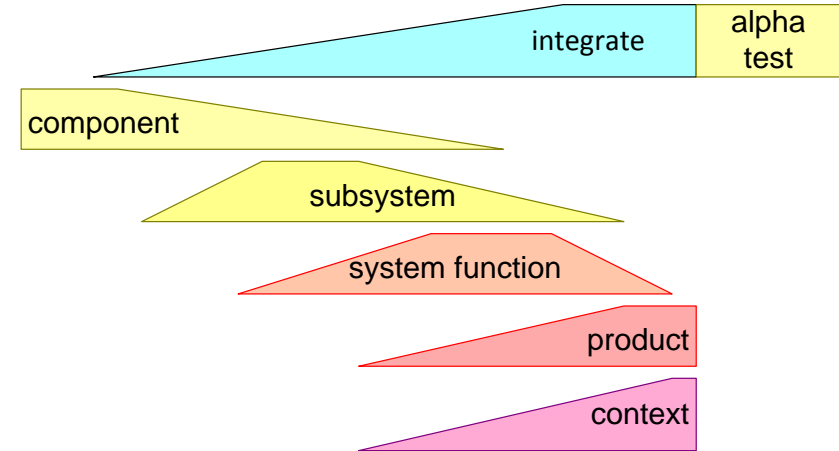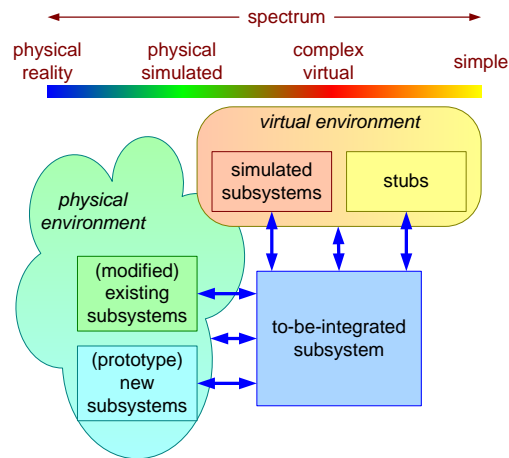USN   ESI

Systems Integration

Integration Starts at Feasibility
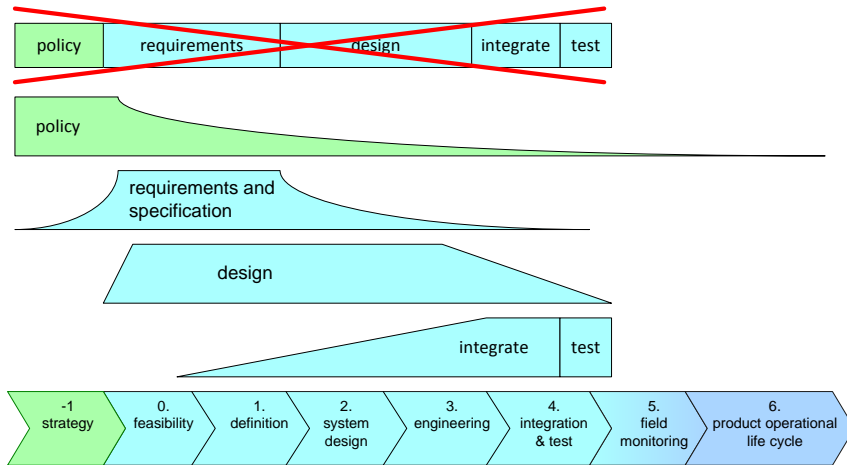
Bottom-up

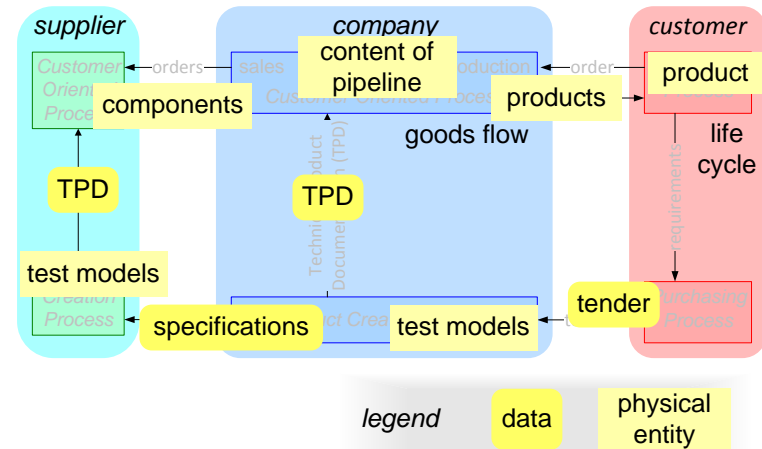## Alternatives for Early Integration

Propagation of Configuration Issues

Module Requirements

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

This module addresses requirements: What are requirements? How to find, select, and consolidate requirements?

January 22, 2023
status: concept
version: 1.4

Fundamentals of Requirements Engineering

by *Gerrit Muller*   USN-SE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

Requirements engineering is one of the systems engineering pillars. In this document we discuss the fundamentals of systems engineering, such as the transformation of needs into specification, the need to prescribe *what* rather than *how*, and the requirements when writing requirements.

January 22, 2023
status: concept
version: 0.1
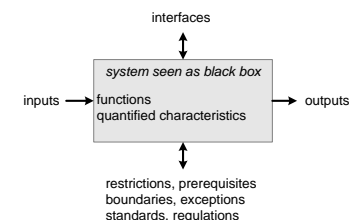
Definition of "Requirement"

Requirements describing the needs of the customer: **Customer Needs**

Requirements describing the characteristics of the final resulting system (product): **System (Product) Specification**

The **requirements management process** recursively applies this definition for every level of decomposition.

Requirements describing the needs of the company itself over the life cycle: **Life Cycle Needs**

Flow of Requirements

What

  choices
  trade-offs
  negotiations

What

How

What What What

How How How

customer needs:
  What is needed by the customer?

product specification:
  What are we going to realize?

system design:
  How are we going to realize the product?

What are the subsystems we will realize?

How will the subsystems be realized?

What What What

How How How

What What What

How How How

up to "atomic" components

USN ESI

System as a Black Box

interfaces

$\updownarrow$

system seen as black box

functions
quantified characteristics

inputs $\longrightarrow$

$\longrightarrow$ outputs

$\updownarrow$

restrictions, prerequisites
boundaries, exceptions
standards, regulations
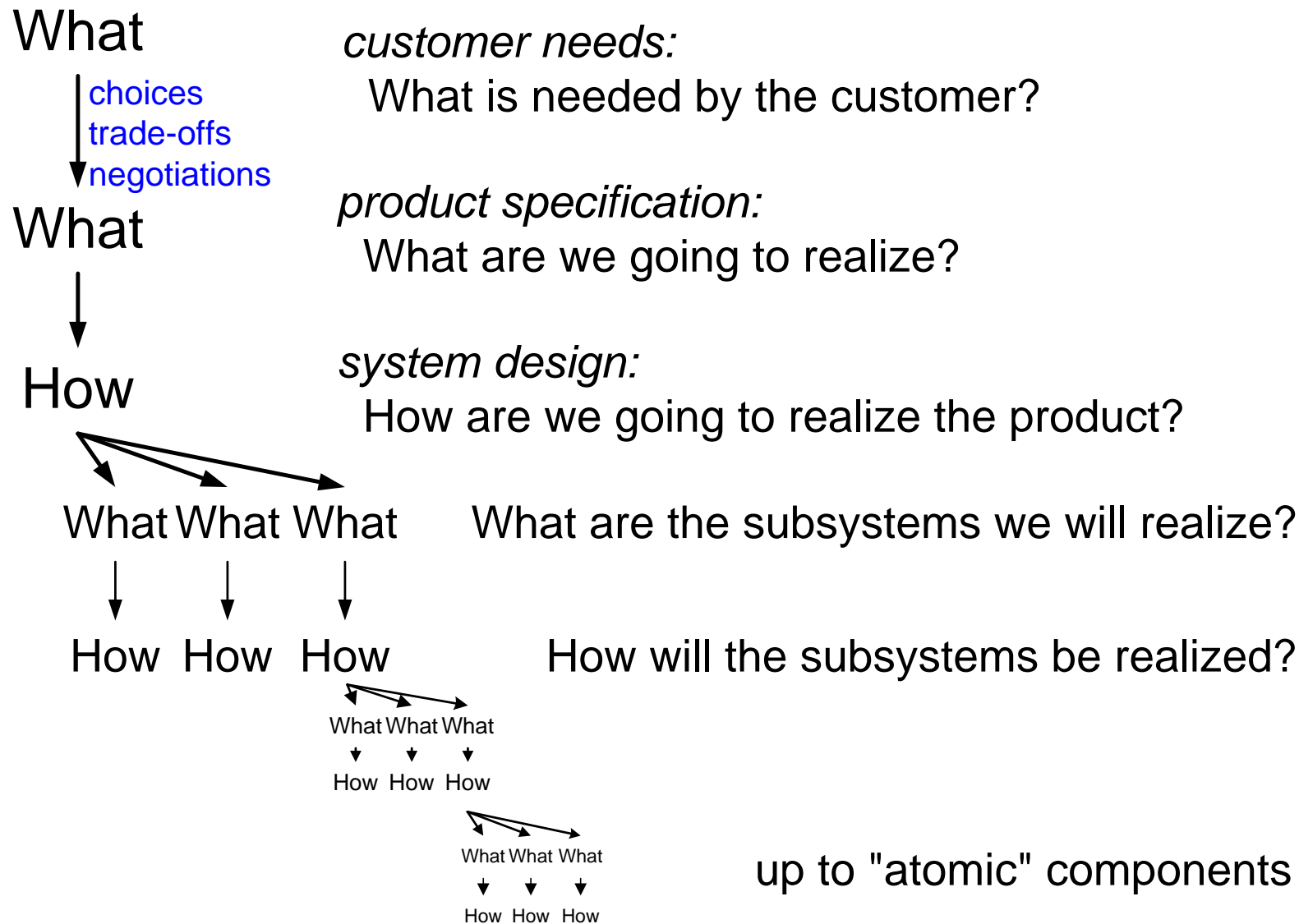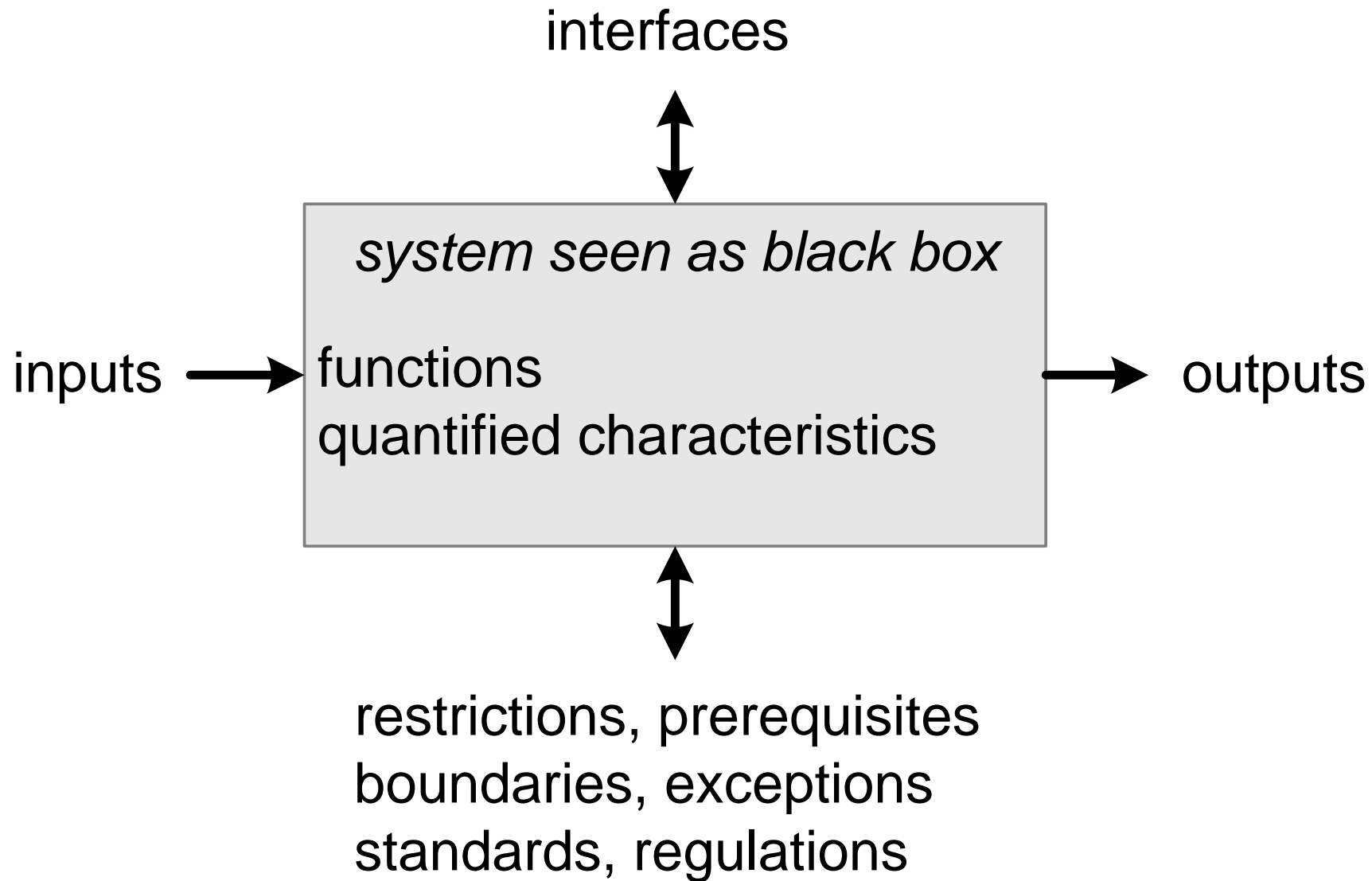
Stakeholders w.r.t. Requirements

customer
(purchaser, decision maker, user, operator, maintainer)

company

Policy and Planning
(business, marketing, operational managers)

Customer-Oriented Process
(sales, service, production, logistics)

Product Creation Process
(project leader, product manager, engineers, suppliers)

People, Process, and Technology management process
(capability managers, technology suppliers)

The "Formal" Requirements for Requirements

Specific

Unambiguous

Verifiable

Quantifiable

Measurable

Complete

Traceable

The Requirements to Enable Human Use

Accessible

Understandable

Low threshold

Short introduction to basic "CAFCR" model

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`
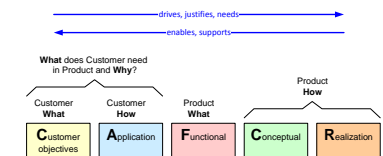
`www.gaudisite.nl`

Abstract

The basic "CAFCR" reference model is described, which is used to describe a system in relation to its context. The main stakeholder in the context is the customer. The question "Who is the customer?" is addressed.

January 22, 2023
status: draft
version: 0.4

The "CAFCR" model

drives, justifies, needs →

← enables, supports

What does Customer need in Product and **Why**?

Customer
What

Customer
How

Product
What

Product
How

| **C**ustomer objectives | **A**pplication | **F**unctional | **C**onceptual | **R**ealization |

USN ESI

Integrating CAFCR

What does Customer need
in Product and **Why**?

Product
How

Customer **What**	Customer **How**	Product **What**		
Customer objectives	**A**pplication	**F**unctional	**C**onceptual	**R**ealization

context
understanding
intention
objective
driven

oppor-
tunities
constraint
awareness
knowledge
based

USN ESI

CAFCR can be applied recursively

Consumer

Drives

Enables

Customer's Customer Business

Drives

Enables

Customer Business

Drives

Value Chain
larger scope has smaller influence on architecture

Enables

System (producer)

USN ESI
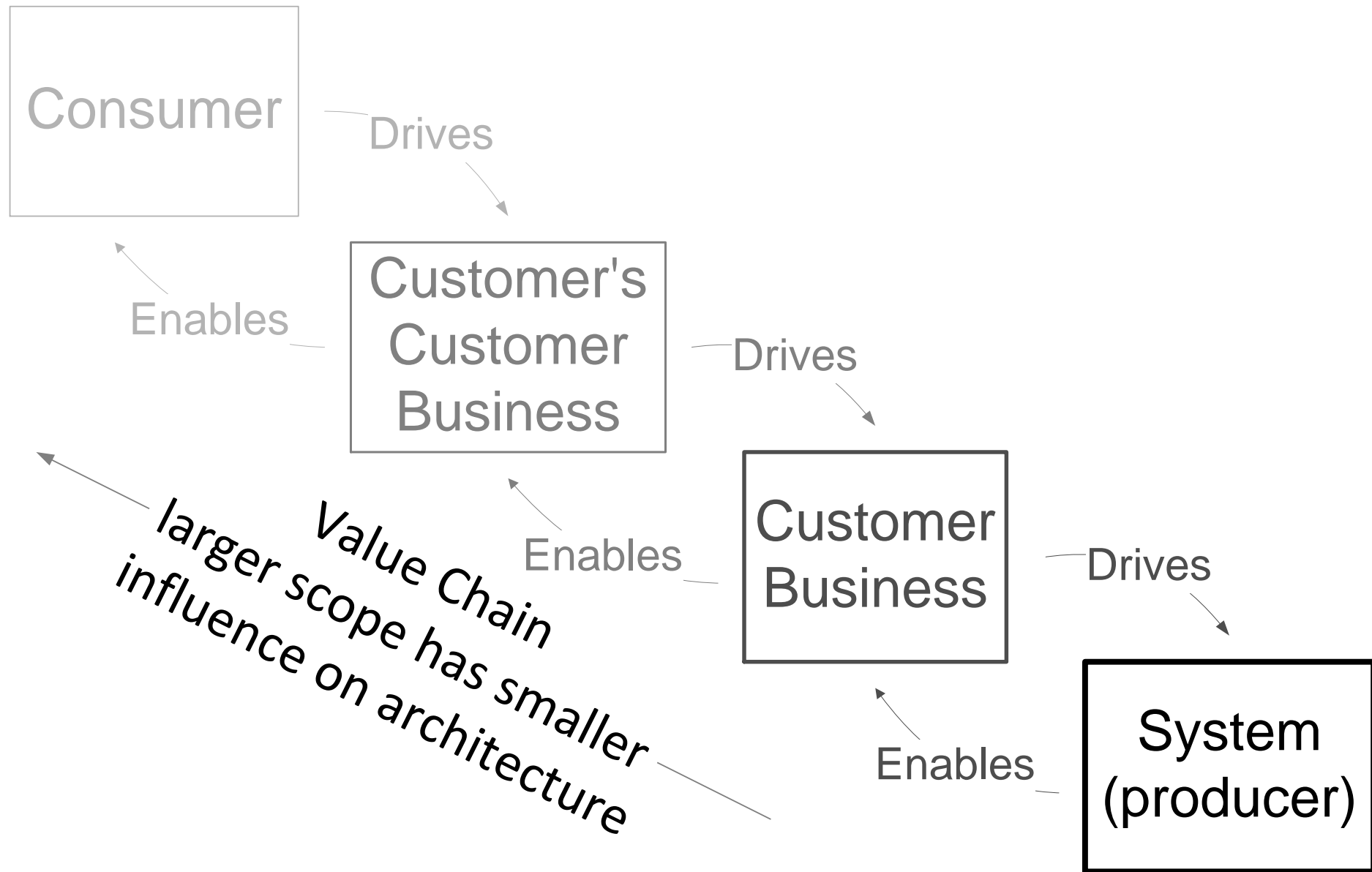
Market segmentation

segmentation axis	examples
geographical	USA, UK, Germany, Japan, China
business model	profit, non profit
economics	high end versus cost constrained
consumers	youth, elderly
outlet	retailer, provider, OEM, consumer direct

Example of a small buying organization

CFO   CIO

CMO   CEO   CTO

decision maker(s)

purchaser

*Who is **the** customer?*

department head

user

maintainer

operator

CEO: Chief Executive Officer
CFO: Chief Financial Officer
CIO: Chief Information Officer
CMO: Chief Marketing Officer
CTO: Chief Technology Officer

USN   ESI
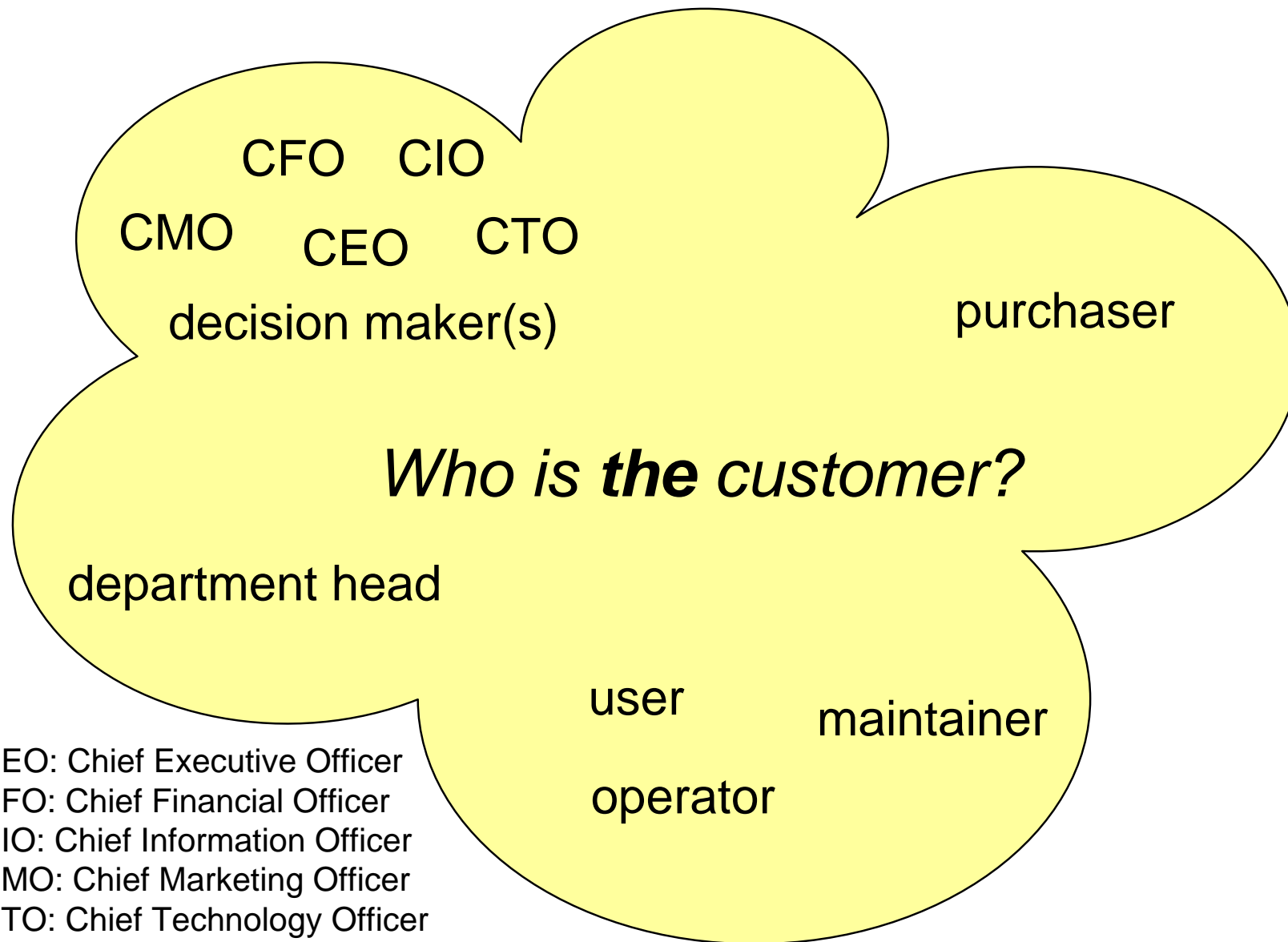
CAFCR+ model; Life Cycle View

Customer objectives

Application

Functional

Conceptual

Realization

operations
maintenance
upgrades

Life cycle

development
manufacturing
installation

sales, service, logistics, production, R&D

USN   ESI
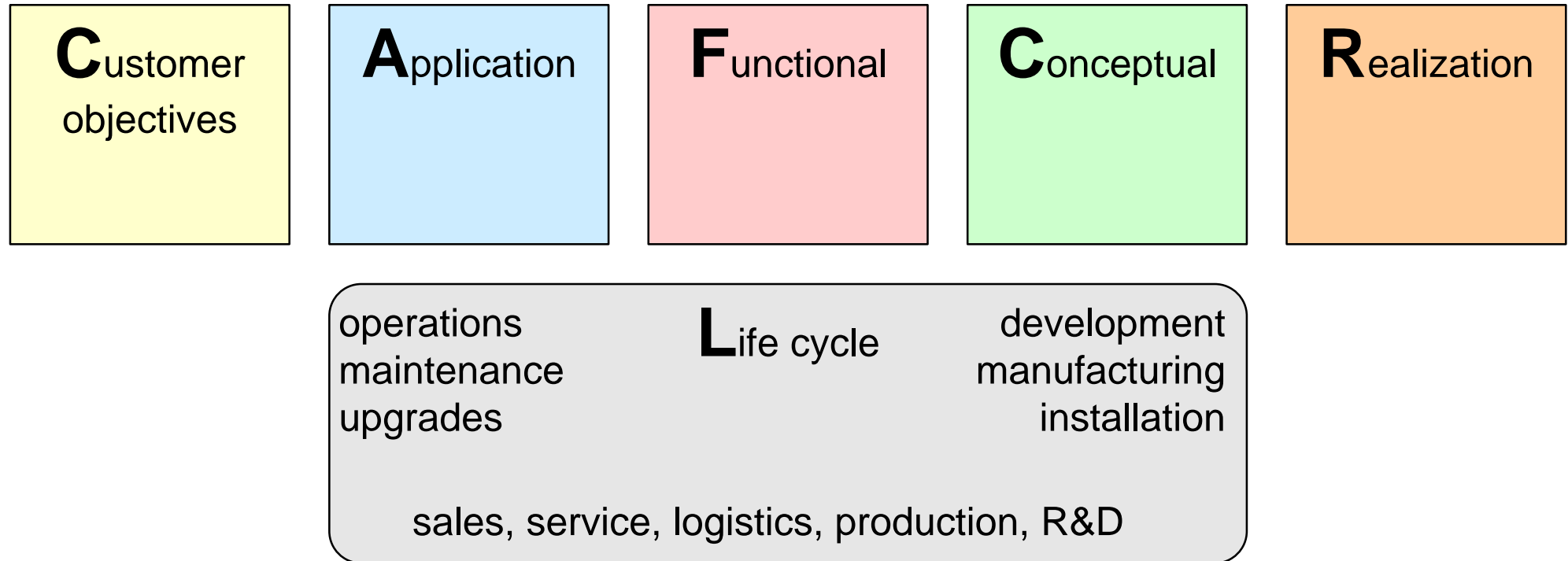
Key Drivers How To

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

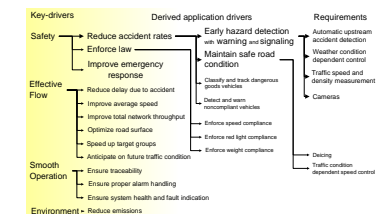**Abstract**

The notion of "business key drivers" is introduced and a method is described to link these key drivers to the product specification.
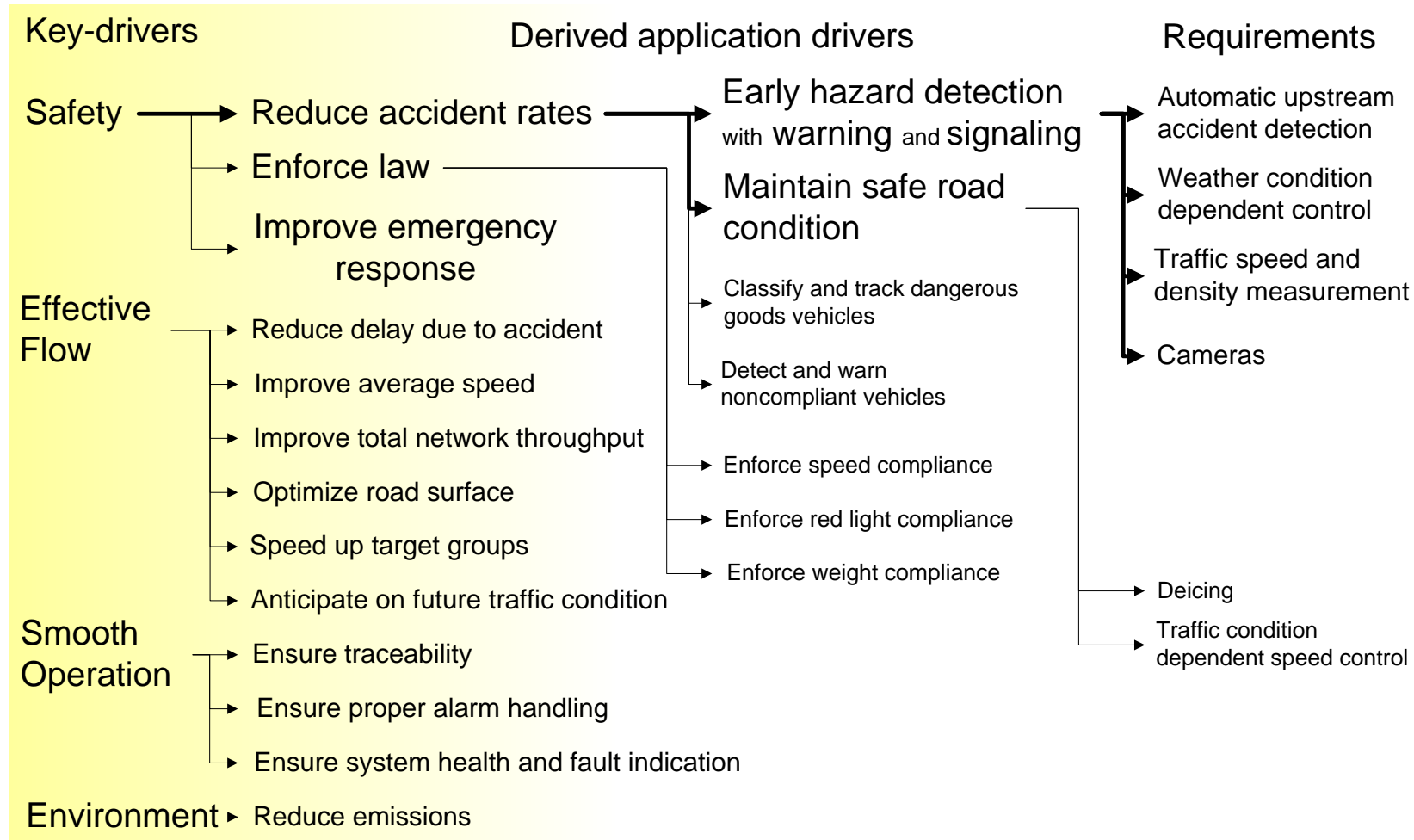
January 22, 2023
status: draft
version: 0.2

Example Motorway Management Analysis

Key-drivers

Safety → Reduce accident rates
- Enforce law
- Improve emergency response

Effective Flow
- Reduce delay due to accident
- Improve average speed
- Improve total network throughput
- Optimize road surface
- Speed up target groups
- Anticipate on future traffic condition

Smooth Operation
- Ensure traceability
- Ensure proper alarm handling
- Ensure system health and fault indication

Environment ▸ Reduce emissions

Derived application drivers

Early hazard detection with **warning** and **signaling**

Maintain safe road condition

- Classify and track dangerous goods vehicles
- Detect and warn noncompliant vehicles

- Enforce speed compliance
- Enforce red light compliance
- Enforce weight compliance

Requirements

- Automatic upstream accident detection
- Weather condition dependent control
- Traffic speed and density measurement
- Cameras

- Deicing
- Traffic condition dependent speed control

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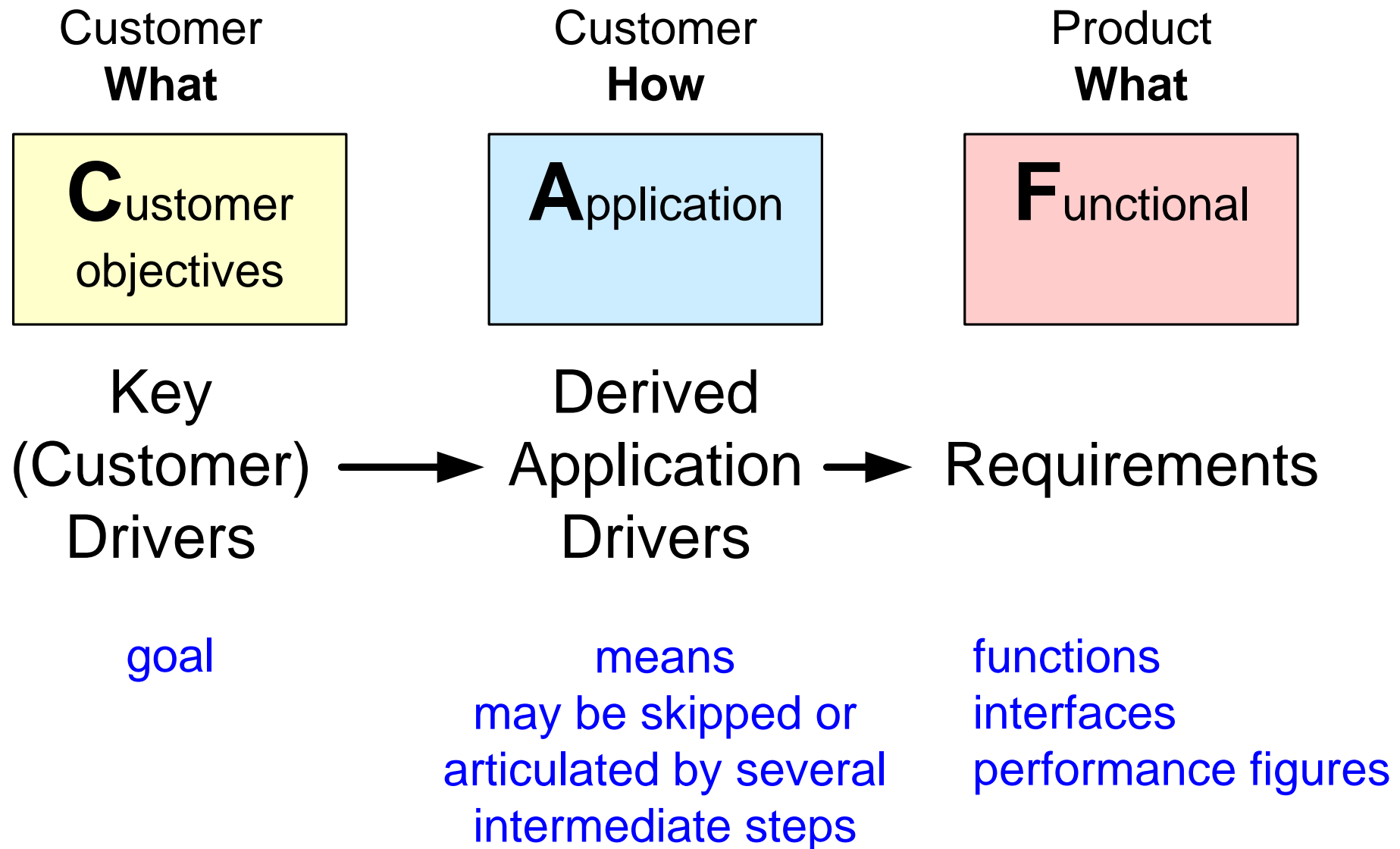
USN ESI

Method to create Key Driver Graph

• Define the scope specific.	in terms of **stakeholder** or **market segments**
• Acquire and analyze facts	extract **facts** from the **product specification** and ask **why questions** about the **specification** of **existing products**.
• Build a graph of relations between drivers and requirements by means of brainstorming and discussions	where **requirements** may have **multiple drivers**
• Obtain feedback	discuss with **customers**, **observe** their **reactions**
• Iterate many times	**increased understanding** often triggers the **move** of issues from **driver** to **requirement** or vice versa and **rephrasing**

USN ESI

Recommendation for the Definition of Key Drivers

- Limit the number of key-drivers   minimal **3**, maximal **6**

- Don't leave out the obvious key-drivers   for instance the well-known **main function** of the product

- Use short names, recognized by the customer.

- Use market-/customer- specific names, no generic names   for instance replace "ease of use" by "minimal number of actions for experienced users", or "efficiency" by "integral cost per patient"

- Do not worry about the exact boundary between Customer Objective and Application   create clear **goal means** relations

Transformation of Key Drivers into Requirements

Customer	Customer	Product
What	**How**	**What**

Customer objectives	**A**pplication	**F**unctional

Key
(Customer)
Drivers $\longrightarrow$ Derived
Application
Drivers $\longrightarrow$ Requirements

goal

means
may be skipped or
articulated by several
intermediate steps

functions
interfaces
performance figures

USN ESI

Requirements Elicitation and Selection

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`
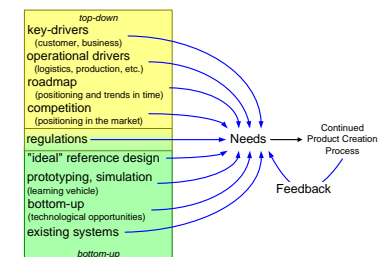
`www.gaudisite.nl`

Abstract

An elicitation method for needs is described using many different viewpoints. A selection process with a coarse and a fine selection is described to reduce the specification to an acceptable and feasible subset.
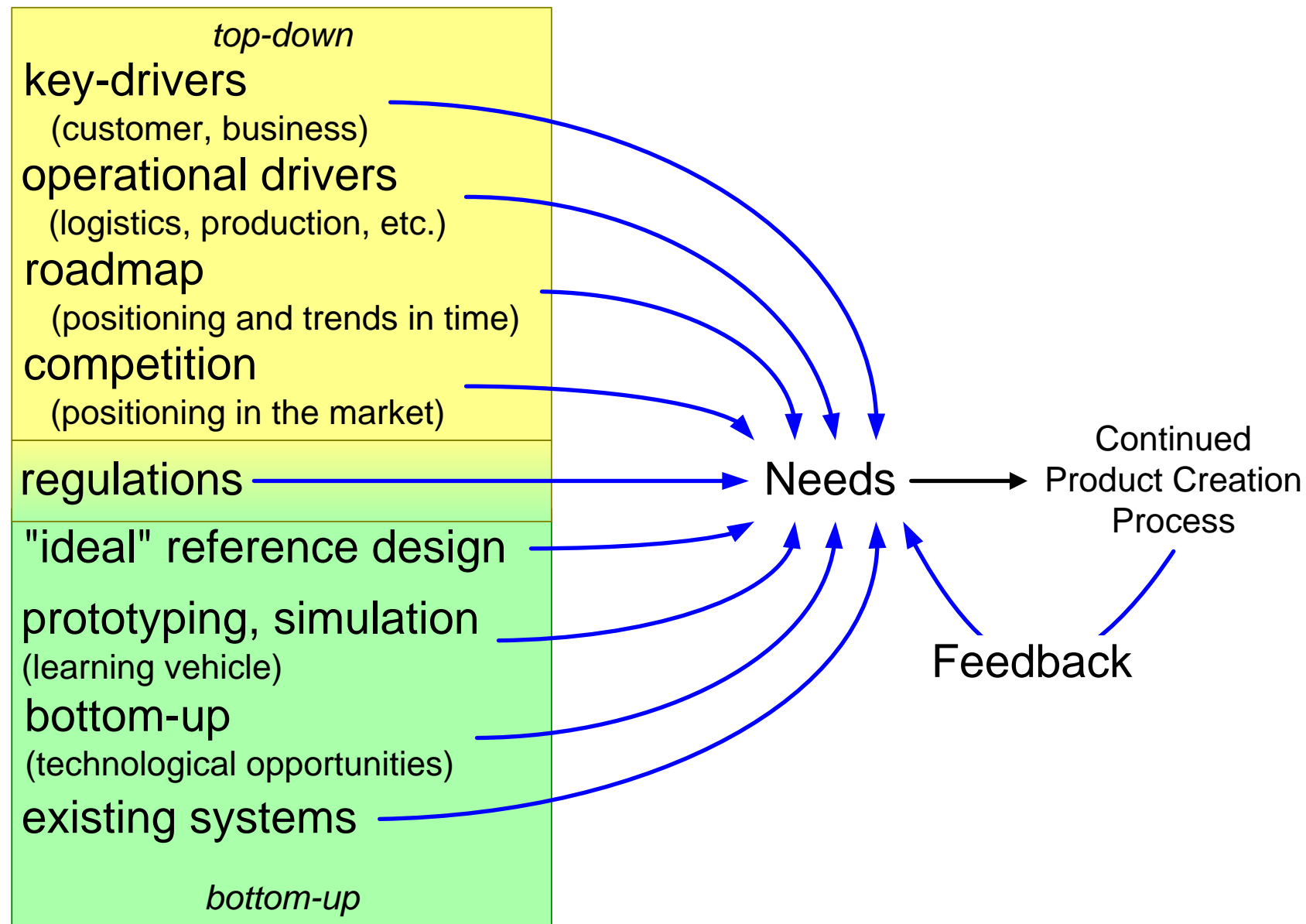
January 22, 2023
status: draft
version: 0

Complementary Viewpoints to Capture Requirements

top-down

key-drivers
(customer, business)

operational drivers
(logistics, production, etc.)

roadmap
(positioning and trends in time)

competition
(positioning in the market)

regulations

"ideal" reference design

prototyping, simulation
(learning vehicle)

bottom-up
(technological opportunities)

existing systems

bottom-up

Needs → Continued Product Creation Process

Feedback

USN ESI

Requirement Selection Process

strategy

roadmap   competition

customer needs →

operational needs →

selection process

→ product specification

→ need characterization

→ requirement phasing

Technology, People, Process
costs and constraints
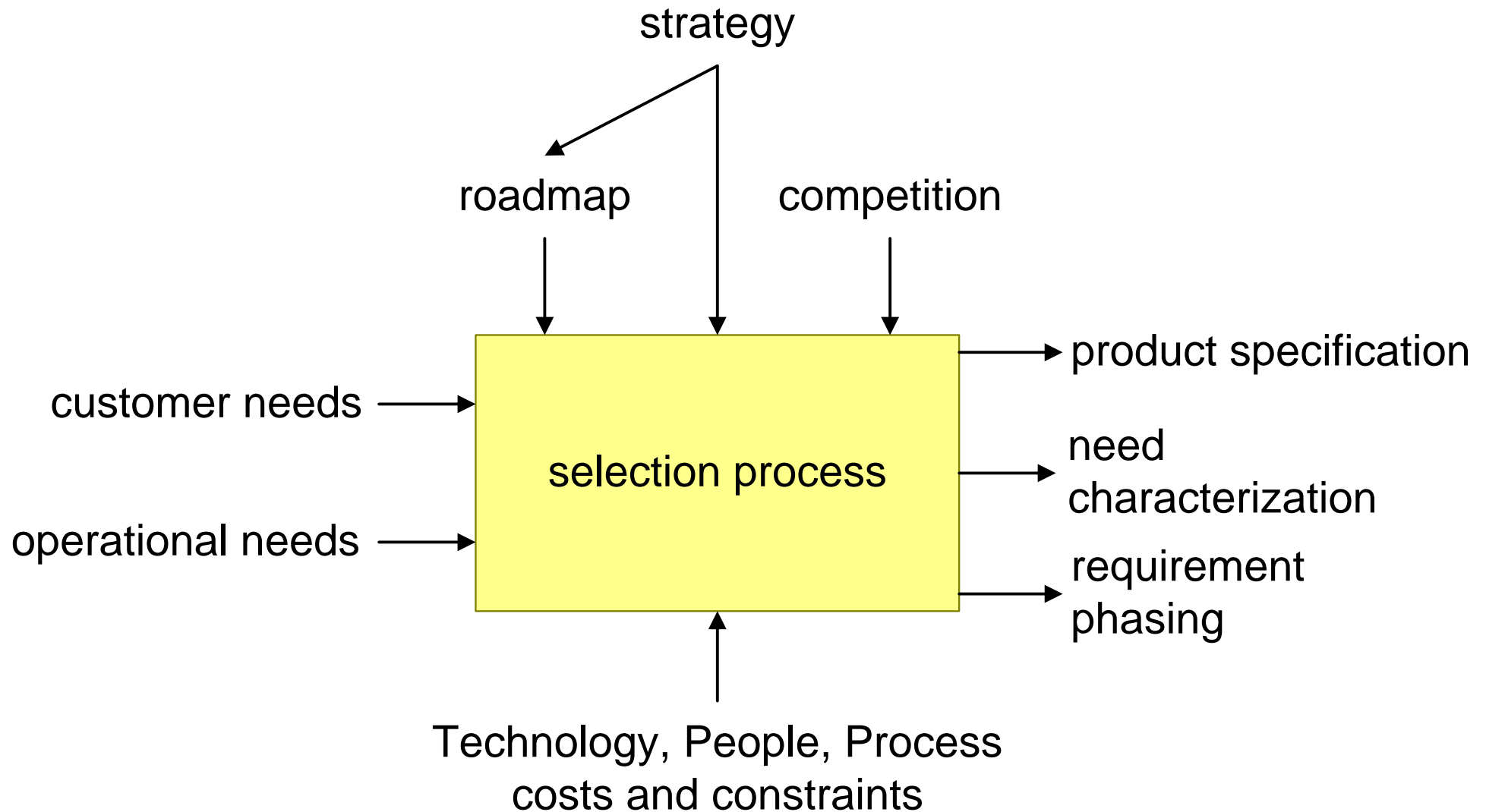
Simple Qualification Method
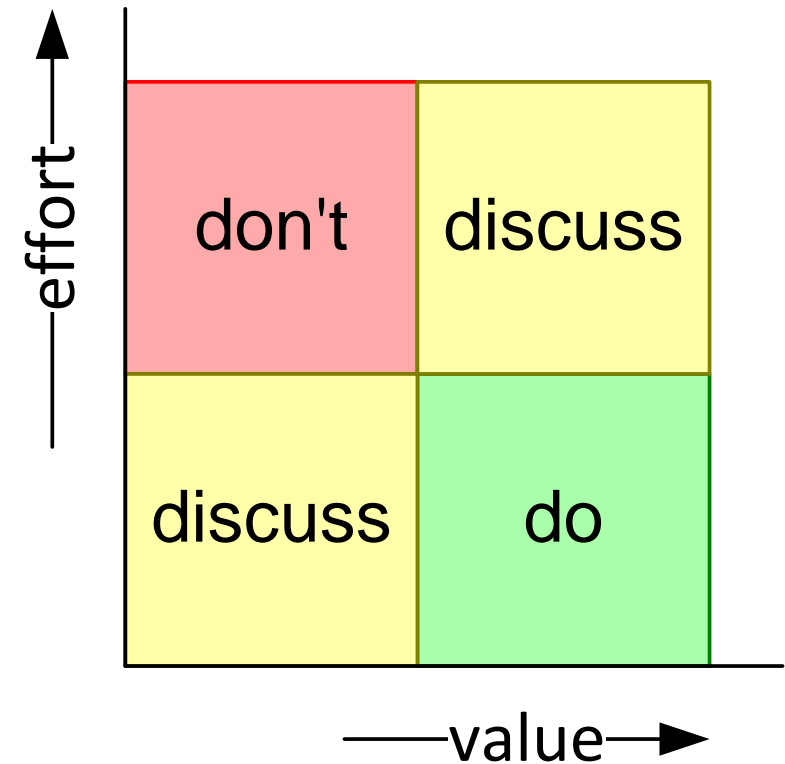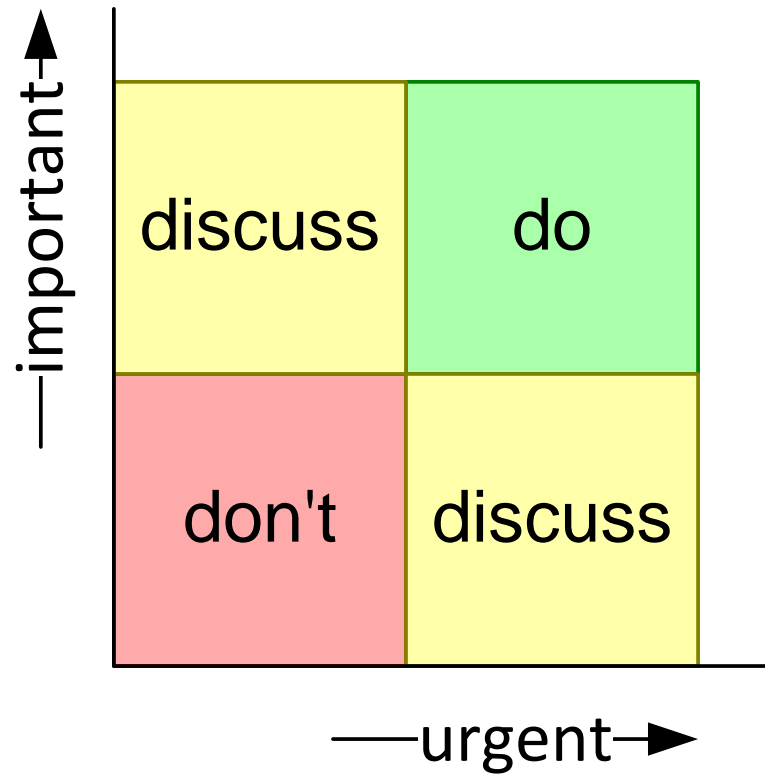
Examples of Quantifiable Aspects

- Value for the customer

- (dis)satisfaction level for the customer

- Selling value (How much is the customer willing to pay?)

- Level of differentiation w.r.t. the competition

- Impact on the market share

- Impact on the profit margin

Use relative scale, e.g. 1..5 1=low value, 5 -high value

Ask several knowledgeable people to score

Discussion provides insight (don't fall in spreadsheet trap)

Exercise Requirements Capturing

- Determine the key drivers for one particular product family.
- Translate these drivers into application drivers and derive from them the requirements.

Needs and Requirements
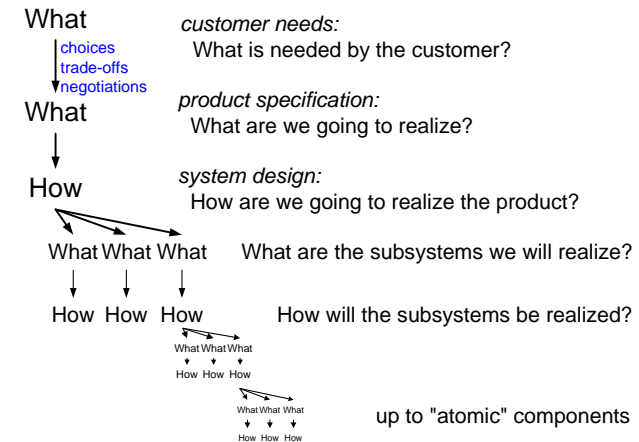
Needs, Specification, Requirements

Requirements describing the needs of the customer:
Customer Needs

Requirements describing the characteristics of the final resulting system (product): **_System (Product) Specification_**

The **_requirements management process_** recursively applies this definition for every level of decomposition.

Requirements describing the needs of the company itself over the life cycle: **_Life Cycle Needs_**

Flow of Requirements

What

choices
trade-offs
negotiations

What

How

What What What

How How How

What What What
How How How

What What What
How How How

customer needs:
What is needed by the customer?

product specification:
What are we going to realize?

system design:
How are we going to realize the product?

What are the subsystems we will realize?

How will the subsystems be realized?

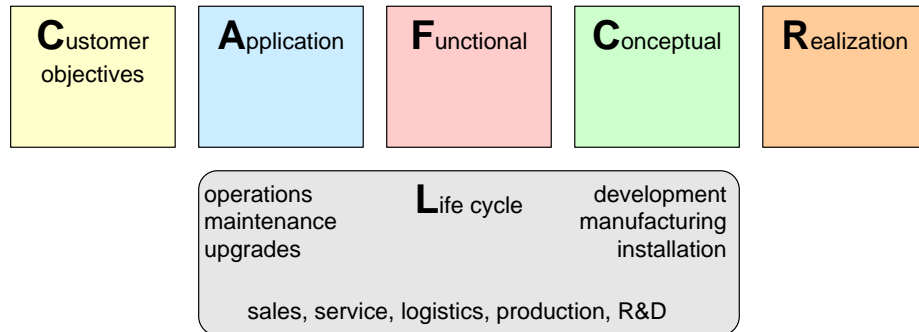up to "atomic" components

Requirements for Requirements

Specific

Unambiguous

Verifiable

Quantifiable

Measurable

Complete

Traceable

Enable Human Use

Accessible
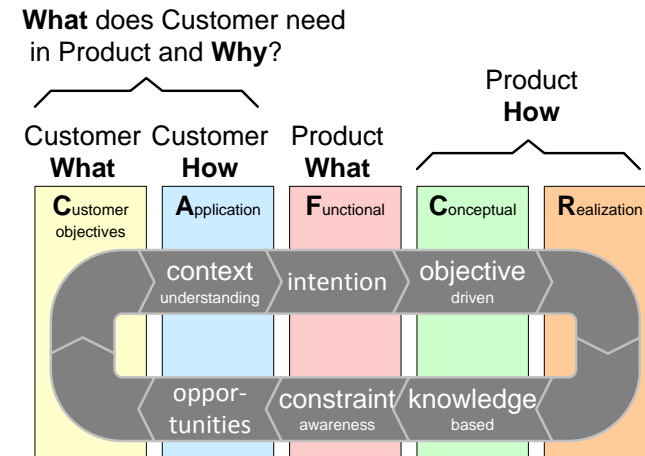
Understandable

Low threshold

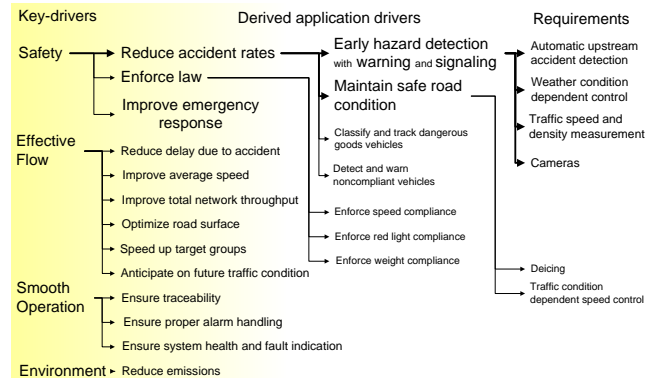USN ESI

CAFCR, Customer Key Driver Graph
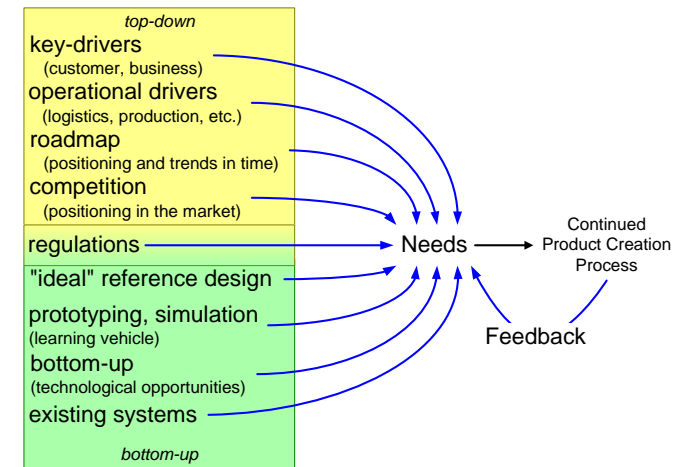
CAFCR+ Model

Customer objectives	**A**pplication	**F**unctional	**C**onceptual	**R**ealization

operations
maintenance
upgrades   **L**ife cycle   development
                                                manufacturing
                                                installation

sales, service, logistics, production, R&D

Iterate over Views

What does Customer need in Product and **Why**?

Customer **What** · Customer **How** · Product **What** · Product **How**

Example Key Driver Graph

Key-drivers · Derived application drivers · Requirements

Safety → Reduce accident rates → Early hazard detection with warning and signaling → Automatic upstream accident detection

Enforce law → Maintain safe road condition → Weather condition dependent control

Improve emergency response → Traffic speed and density measurement

Effective Flow → Reduce delay due to accident → Classify and track dangerous goods vehicles → Cameras

Improve average speed → Detect and warn noncompliant vehicles

Improve total network throughput

Optimize road surface → Enforce speed compliance

Speed up target groups → Enforce red light compliance → Deicing

Anticipate on future traffic condition → Enforce weight compliance → Traffic condition dependent speed control

Smooth Operation → Ensure traceability

Ensure proper alarm handling

Ensure system health and fault indication

Environment → Reduce emissions

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

Complementary Viewpoints

top-down

key-drivers (customer, business)
operational drivers (logistics, production, etc.)
roadmap (positioning and trends in time)
competition (positioning in the market)
regulations
"ideal" reference design
prototyping, simulation (learning vehicle)
bottom-up (technological opportunities)
existing systems

bottom-up

→ Needs → Continued Product Creation Process

Feedback

USN   ESI

Module Roadmapping

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

This module addresses roadmapping.

January 22, 2023
status: draft
version: 1.2

Roadmapping

by *Gerrit Muller*   USN-SE
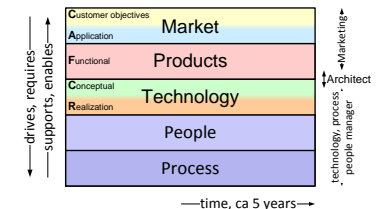
e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

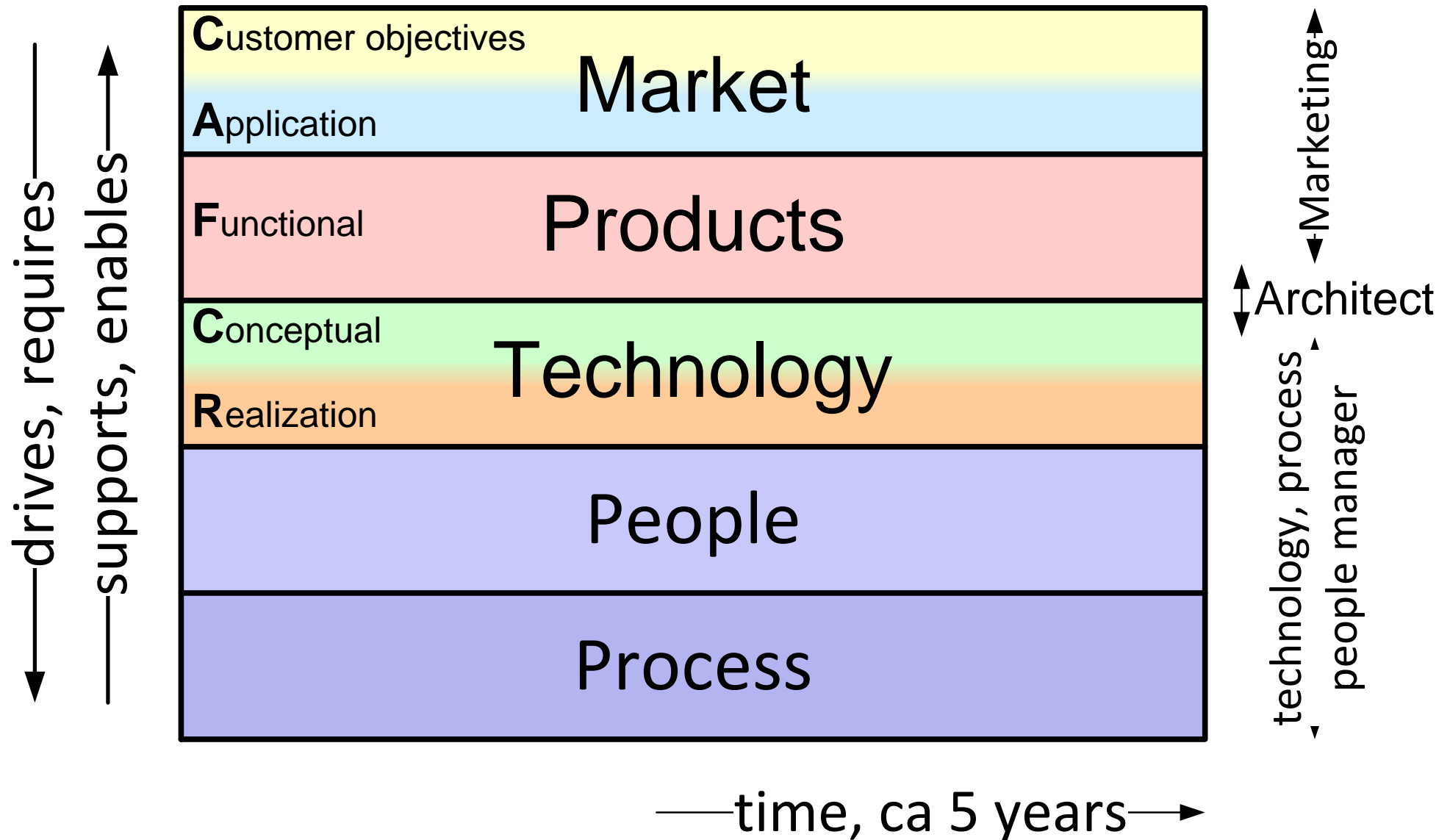This article describes what a roadmap is, how to create and maintain a roadmap, the involvement of the stakeholders, and criteria for the structure of a roadmap.
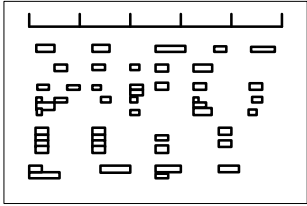
January 22, 2023
status: concept
version: 2.0

The Roadmap Integrates Five Views
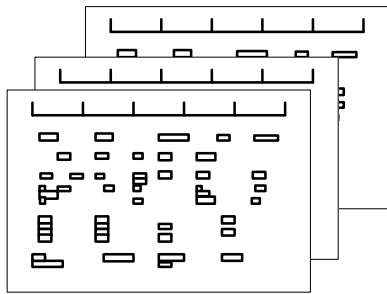
Granularity of Roadmap Material

Top-level roadmap

Single page
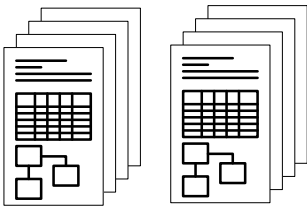
Poster

part of many presentations

Supporting roadmaps

Single page
per view
or per driver

Poster

part of many presentations

Supporting reports

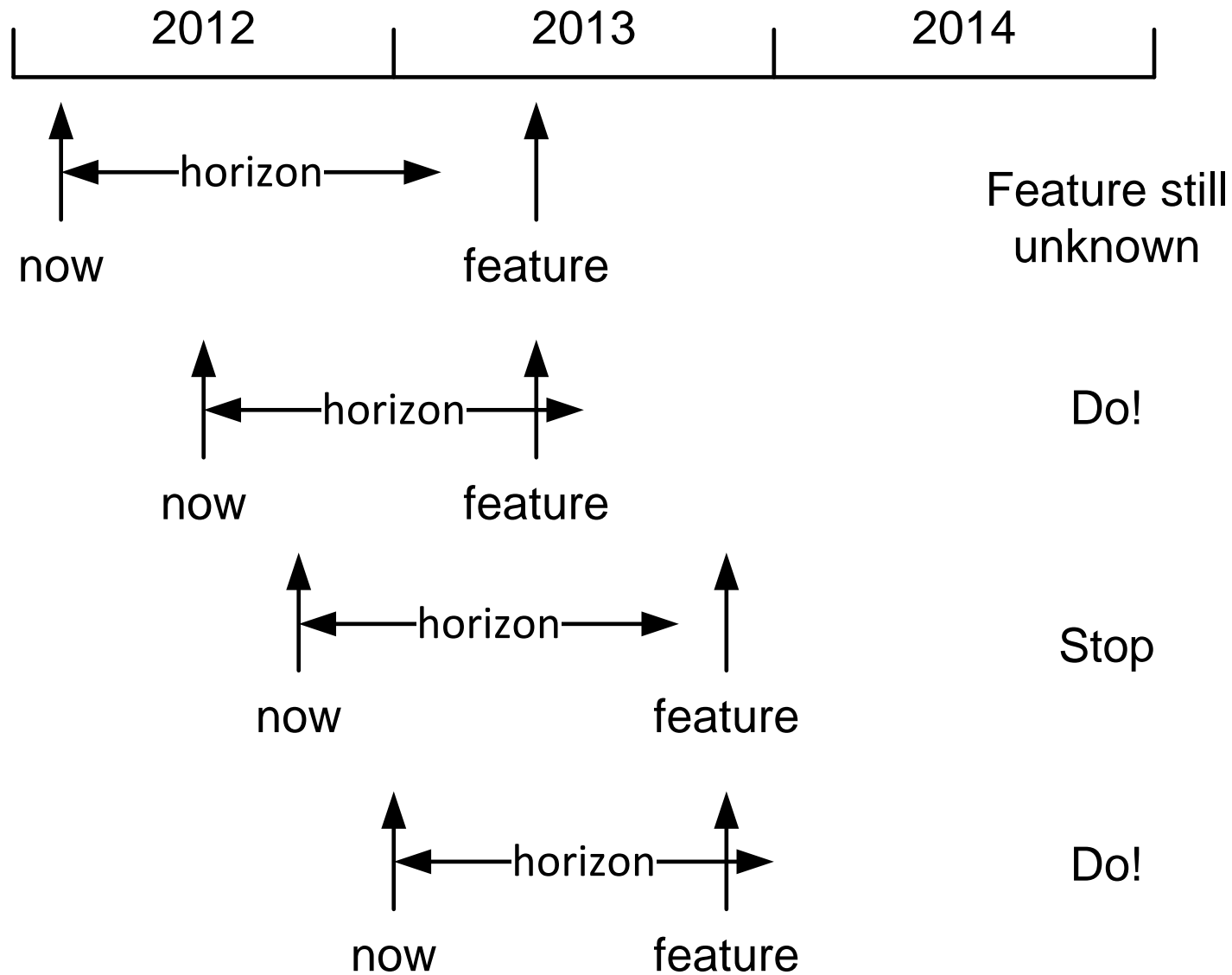Document
per relevant
subject

Frequent changes in product policy

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

Diverging activities of teams

Missed market opportunities

Management with a Limited Horizon

Management with a Broader Time Perspective

2012   2013   2104

now   feature   Preparation by 0.5 person

now   feature   Work with 1.5 persons

now   feature   Continue with 0.5 person

now   feature   Work with 1.5 persons

legend

number of people allocated

time
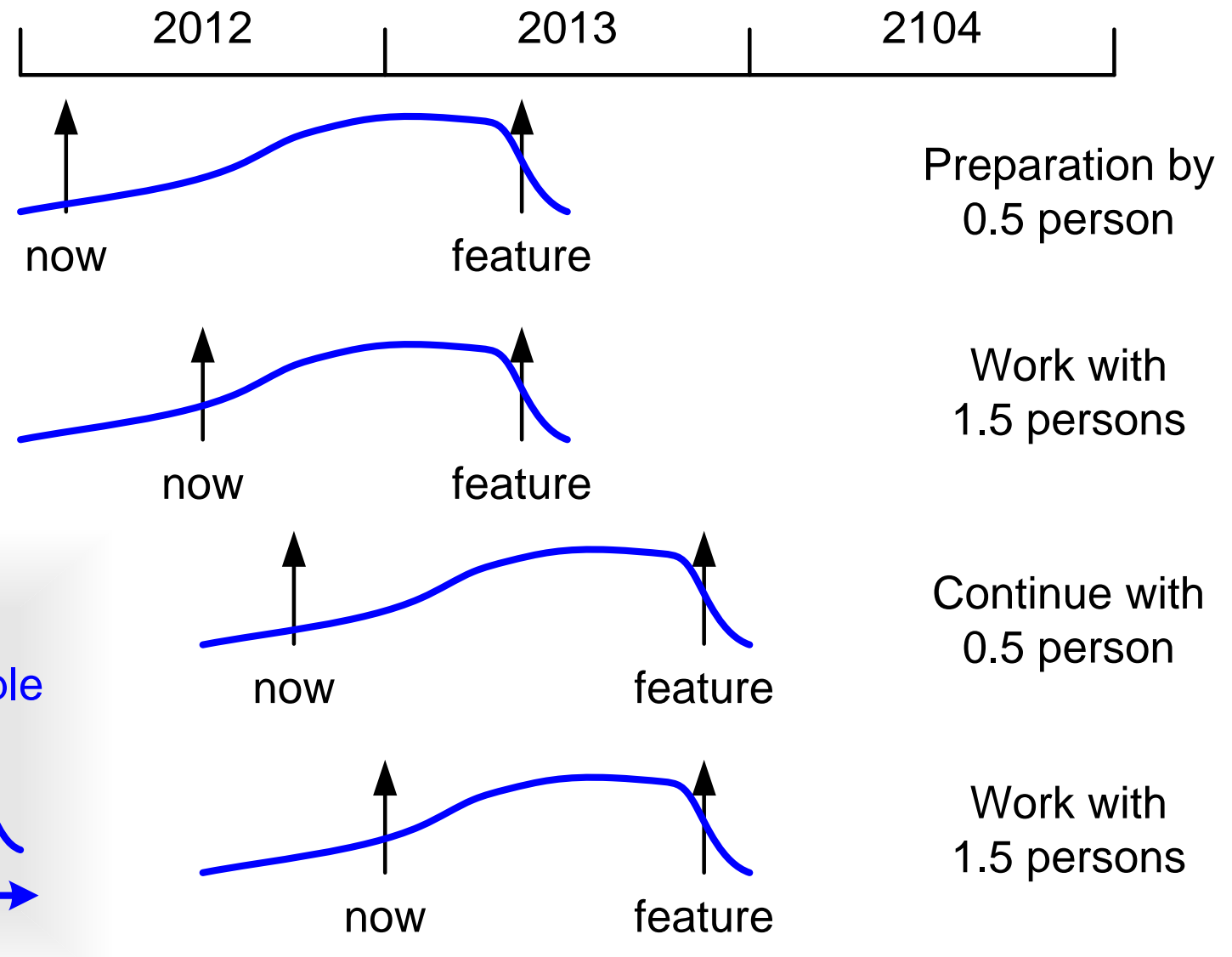
Creation or Update of Roadmap in Burst Mode
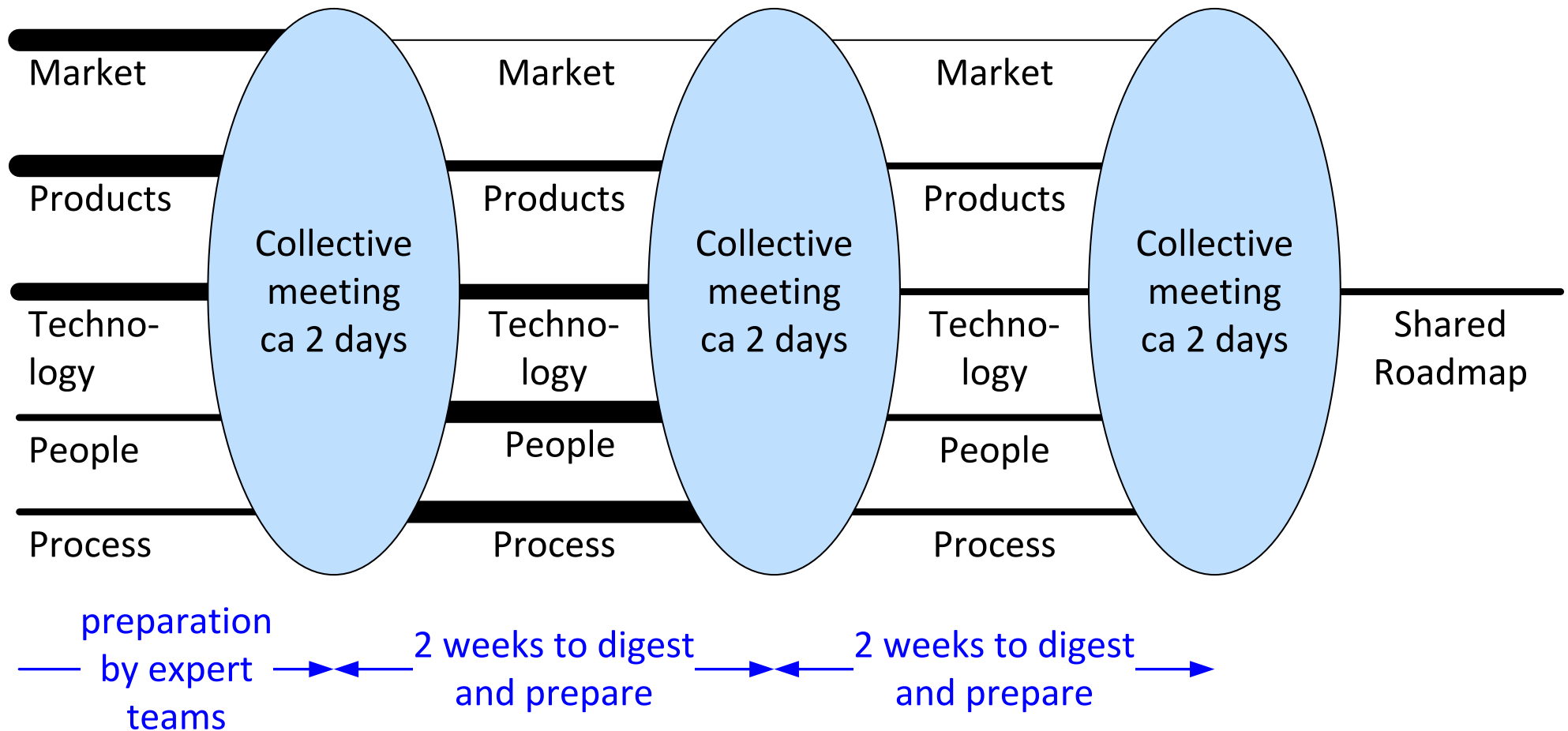
Typical Stakeholders of a Roadmap

business manager overall enterprise responsible

marketing manager(s)

discipline or line managers

people, process, and technology manager(s)

operational manager(s) project or program managers

architect(s)

Target of the First Session

Shared vision on market

First iteration of possible products as an answer to the market

Share technology status, as starting point for technology roadmap

Explore people and technology status, to identify main issues

Target of the Second Session

Obtaining a shared vision on the desired technology roadmap

Sharing the people and process issues required for the products defined in the first iteration

Analyzing a few scenarios for products, technologies, people, and process

The Roadmap Update Visualized in Time

Market: What is needed by the customers?

Products: How to package technologies into products to fulfill market needs?

Technology: What technological trends are relevant? What technologies are needed?

People: What kind of and how many people are required to realize the products and technologies?

Process: What processes are required to let these people realize the products and technologies?

time →

USN ESI

From Roadmap to Detailed Plans
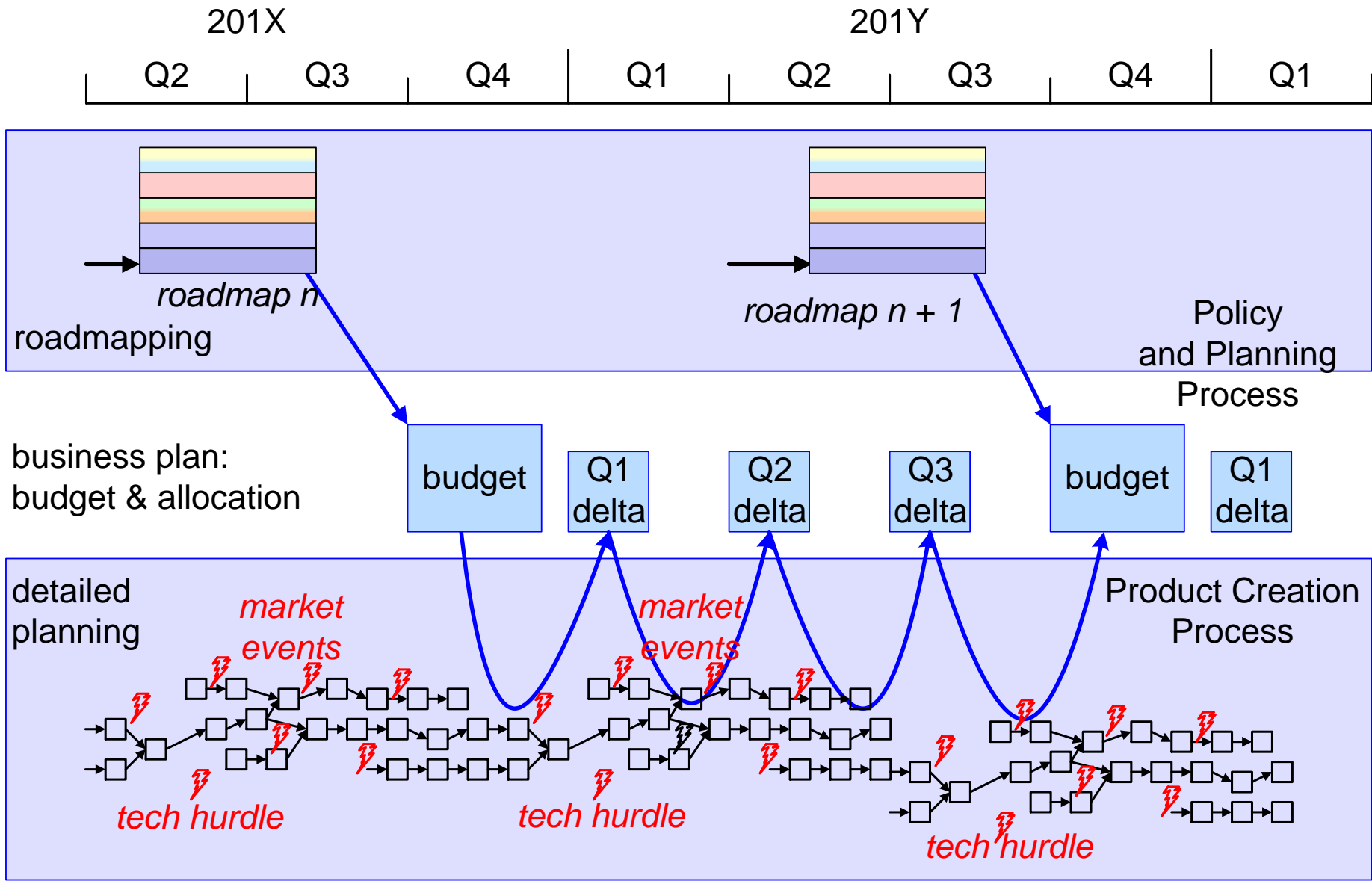
3-Tier Approach

	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

Roadmap Essentials

Selection of most important or relevant issues

Key drivers as a means to structure the roadmap

Nothing is certain; ambiguity is normal

Use facts whenever possible

Don't panic in case of impossibilities

USN ESI

Requirements for a Good Roadmap

Recognizable issues for all stakeholders

Clear positioning in time; uncertainty can be visualized

The main events (enabling or constraining) must be present

Limited amount of information to maintain the overview

Sources of Facts

Market analysis reports

number of customers, market size, competition, trends

Installed base

change requests, problem reports, historical data

Manufacturing (statistical process control)

statistical process control

Suppliers (roadmaps, historical data)

roadmaps, historical data

Internal reports (technology studies, simulations)

technology studies, simulations

Causes for Overestimation

Quantization effects of small activities (the amount of time is rounded to manweeks/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)

Lack of pragmatism (technical ambition is not too bad during the roadmap process, as long as it does not pre-empt a healthy decision)

Too many bells and whistles without business or customer value

USN ESI

Market Product Life Cycle Consequences for Architecting

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: gaudisite@gmail.com

www.gaudisite.nl

Abstract

The lifecycle of a product category in the market determines many aspects of the architecting approach. The lifecycle consists typical of 4 phases: infancy, adolesence, mature and aging.

A discontinuity in market success is seen in the transition from one phase to the next phase. The explanation given is that the phases differ in characterictics and require different approaches. The right approach for one phase is sub optimal for the next phase. A set of characteristics per phase is given and the consequences for architecting are discussed.

January 22, 2023
status: concept
version: 1.2
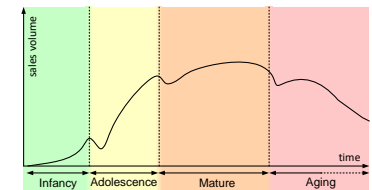
Ideal Bathtub Curve

Market Product Life Cycle Phases in Practice
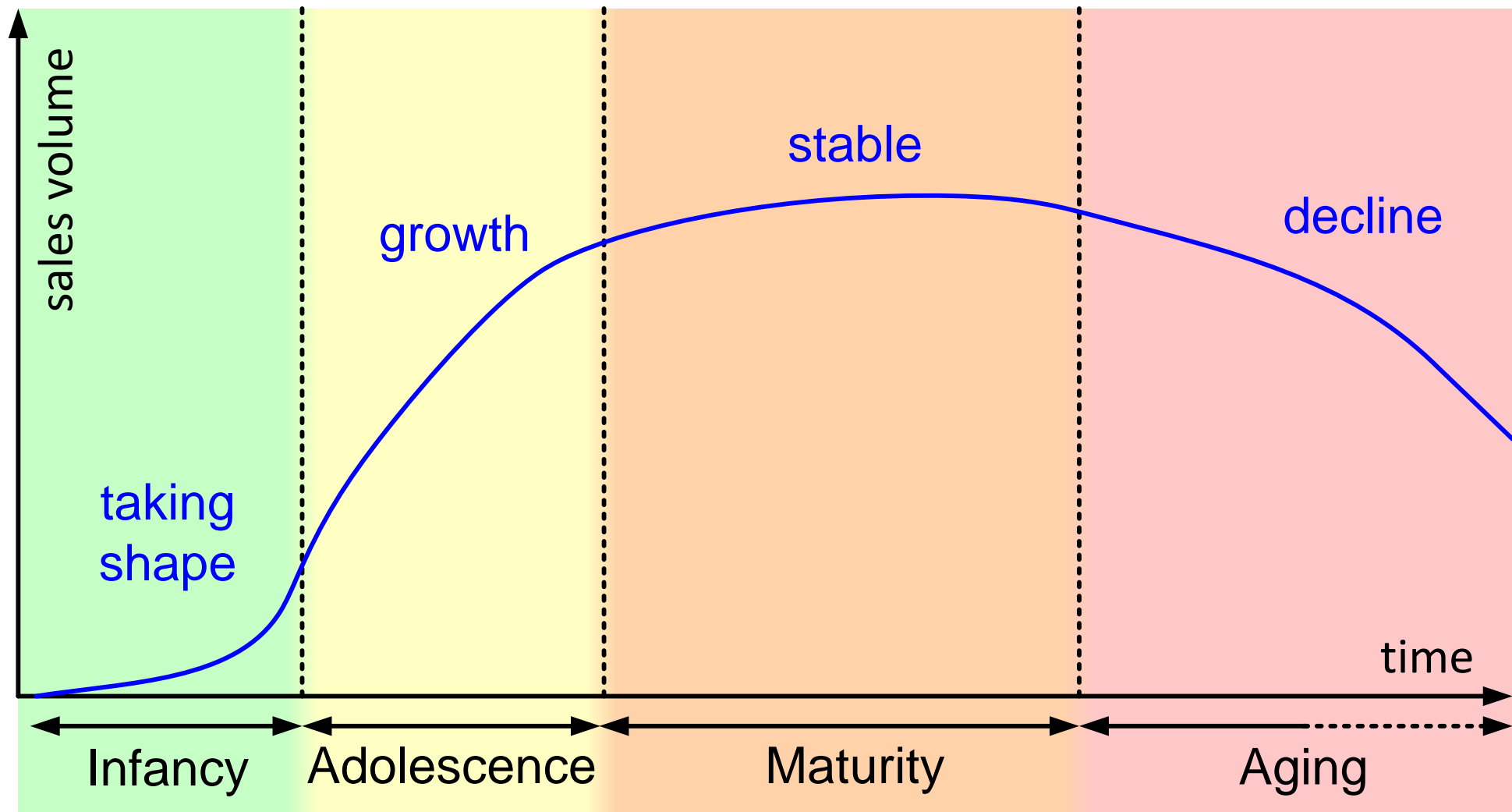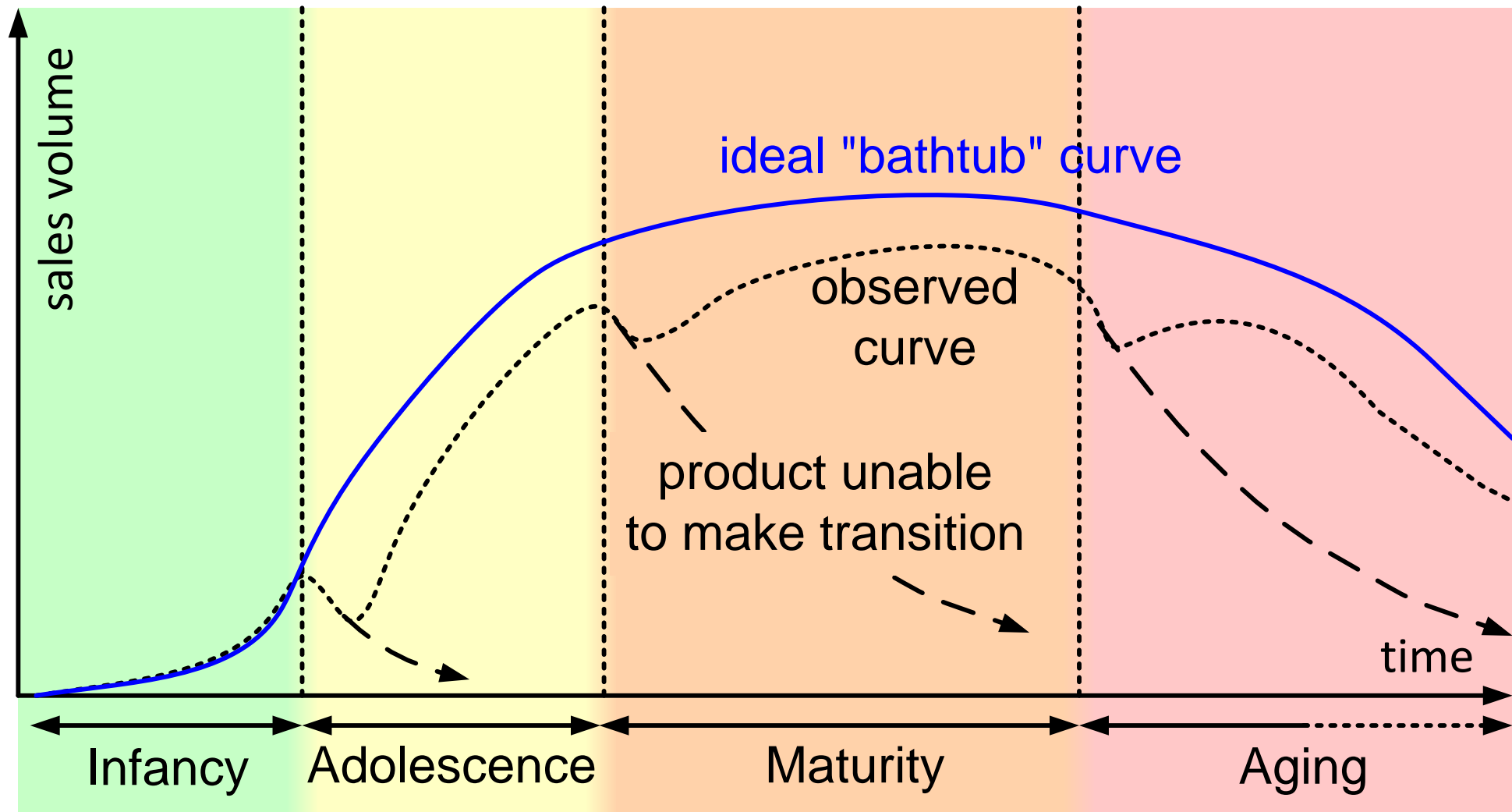
Examples of Product Classes on the Curve
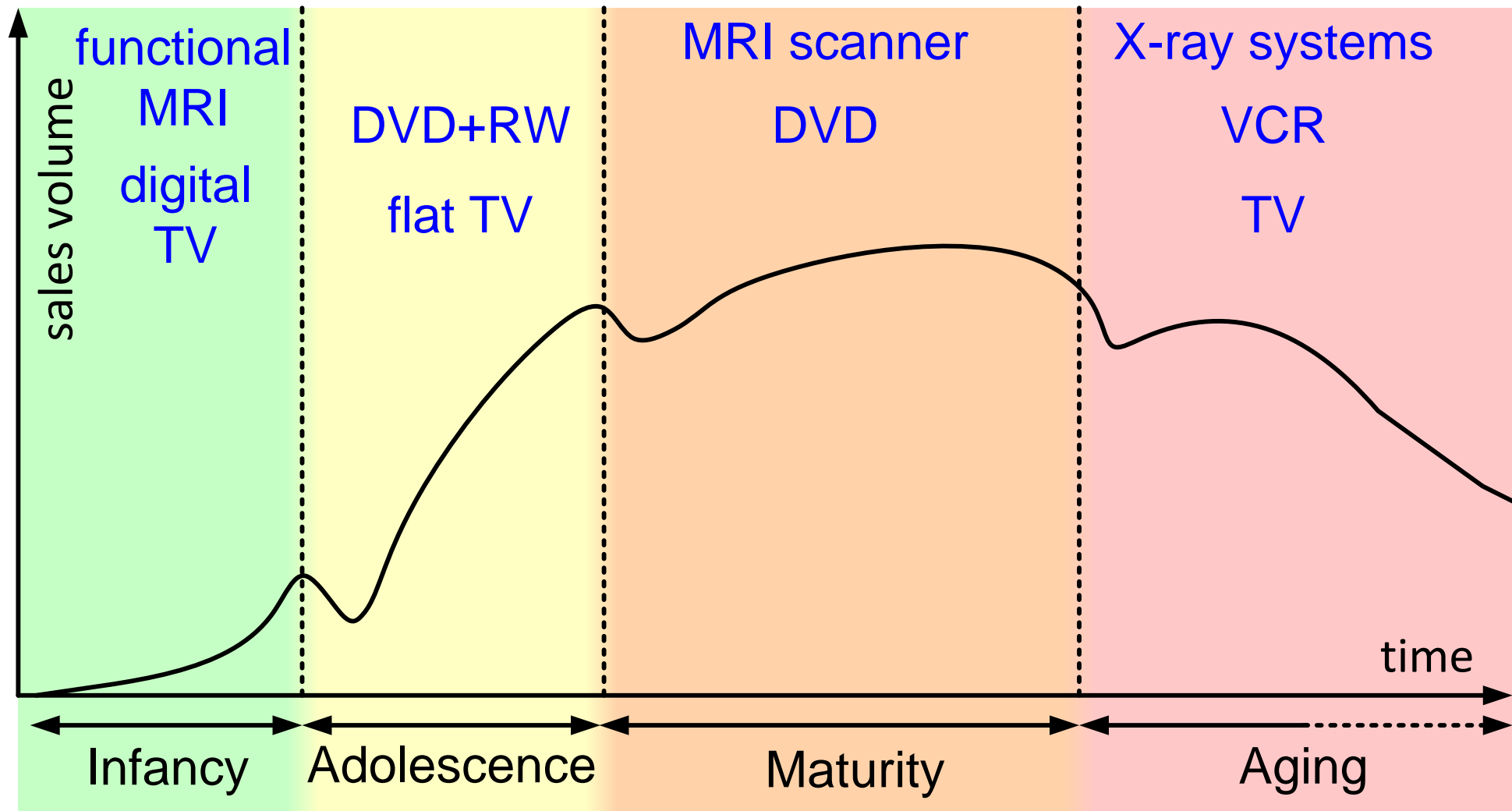
Attributes per Phase

	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

USN   ESI

From Market, Product, Technology to People, Process

The diagram shows the following flow:

Market / Products / Technology box with layers:
- **C**ustomer objectives — Market
- **A**pplication
- **F**unctional — Products
- **C**onceptual — Technology
- **R**ealization

homework arrow points from the Market/Products/Technology box to the tables.

Stacked tables labeled **system**, **electronics**, **software**:

software	2002 actual	2003	2004	2005	2006	2007
Orion	1	2 / 3	3 / 5	5 / 6	10 / 8	20 / 10
Gemini	2	10 / 10	30 / 20	50 / 40	40 / 20	30 / 10
Scorpion	54	80 / 70	60 / 50	30 / 20	20 / 5	15 / 5
research	4	10 / 10	11 / 11	12 / 12	13 / 13	15 / 15
maintenance	22	30 / 40	35 / 50	40 / 50	42 / 55	44 / 55
total	83	132 / 133	139 / 136	137 / 128	125 / 101	124 / 95

estimate by program manager
estimate by people manager

feedback arrow points from the tables back to the Market/Products/Technology box.

after iteration arrow points from the tables to the **People / Process** box.

People

Process

USN ESI
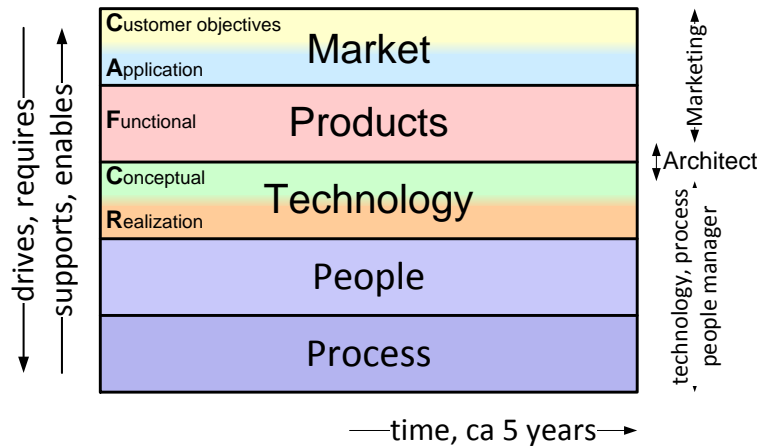
Summary of strategy process
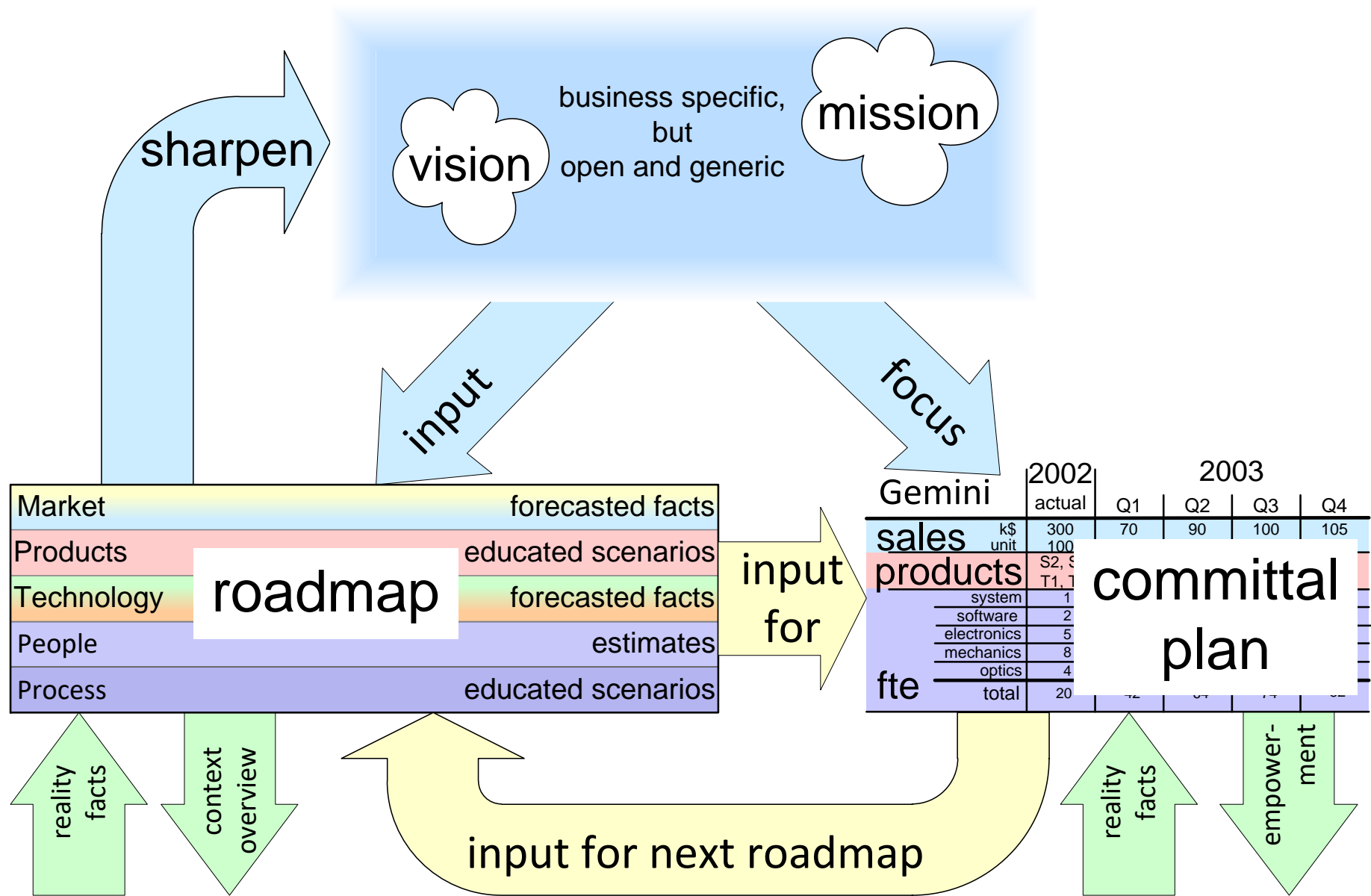
Exercise Roadmapping

Make a roadmap on the basis of what you know at this moment, or what you perceive as the "shared expectation".

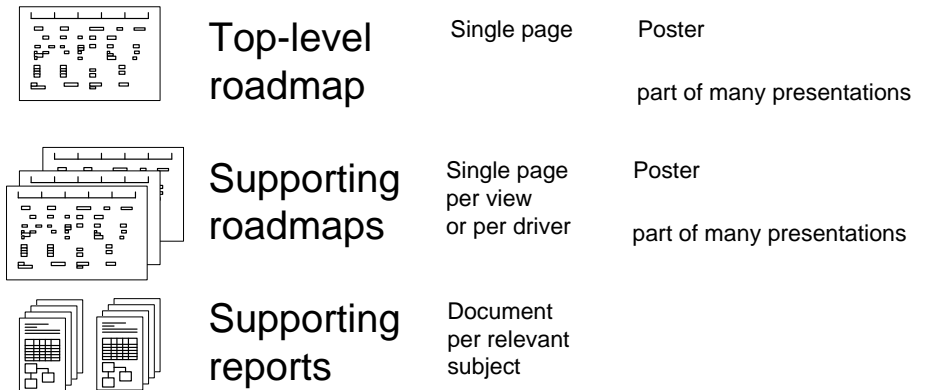Try to fill in as many views (market, products, technology, people and process) as possible.

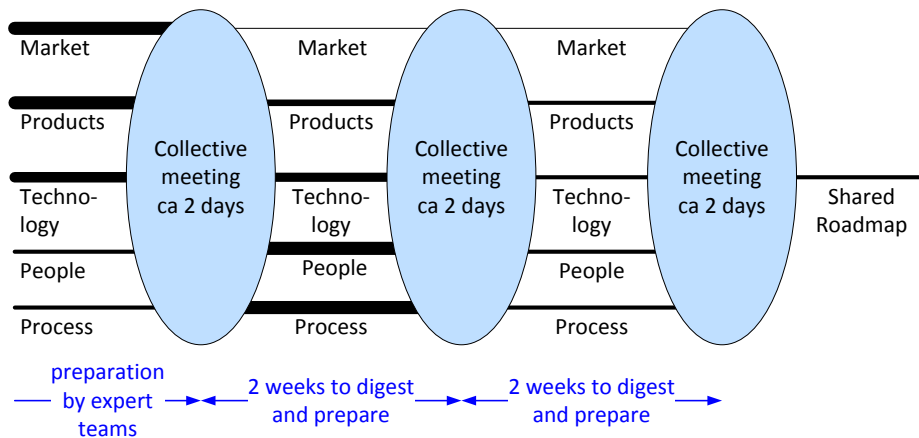Present an overview by minimizing the contents to the most essential data.

Roadmap Creation

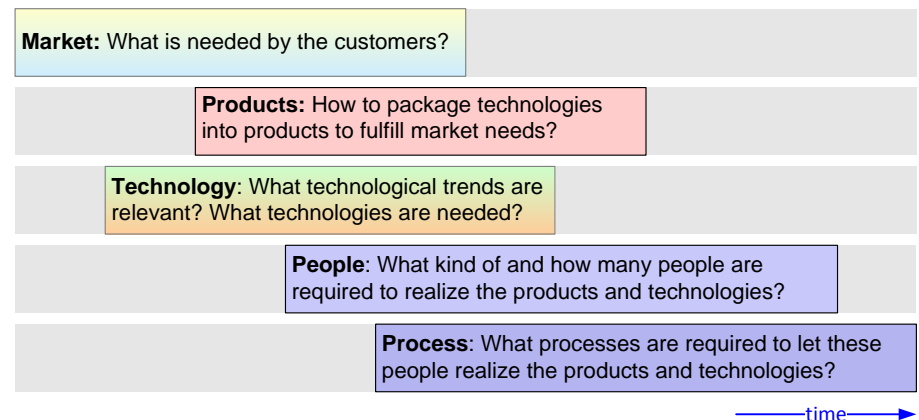## The Roadmap Integrates Five Views

Customer objectives
Application
Functional
Conceptual
Realization

Market
Products
Technology
People
Process

drives, requires — supports, enables

Marketing
Architect
technology, process, people manager

time, ca 5 years

Multiple Levels

Top-level roadmap — Single page — Poster — part of many presentations

Supporting roadmaps — Single page per view or per driver — Poster — part of many presentations

Supporting reports — Document per relevant subject

Creation in Teams

Market
Products
Techno-logy
People
Process

Collective meeting ca 2 days

Market
Products
Techno-logy
People
Process

Collective meeting ca 2 days

Market
Products
Techno-logy
People
Process

Collective meeting ca 2 days

Shared Roadmap

preparation by expert teams

2 weeks to digest and prepare

2 weeks to digest and prepare

Order of Creation

Market: What is needed by the customers?

Products: How to package technologies into products to fulfill market needs?

Technology: What technological trends are relevant? What technologies are needed?

People: What kind of and how many people are required to realize the products and technologies?

Process: What processes are required to let these people realize the products and technologies?

time

USN   ESI

Time Horizons

201X   201Y
| Q2 | Q3 | Q4 | Q1 | Q2 | Q3 | Q4 | Q1 |

roadmap n   *roadmap n + 1*   Policy
roadmapping   and Planning
                                                   Process

business plan: budget   Q1   Q2   Q3   budget Q1
budget & allocation   delta delta delta   delta

detailed   *market*   *market*   Product Creation
planning   *events*   *events*   Process

tech hurdle   *tech hurdle*   *tech hurdle*

Life Cycle Transitions

sales volume

ideal "bathtub" curve

observed
curve

product unable
to make transition

time

| Infancy | Adolescence | Maturity | Aging |

People and Process

Customer objectives
Application   Market
Functional
                     Products
Conceptual
Realization   Technology

homework

feedback

People

Process

after iteration

intentionally left blank

software	2002 actual	2003	2004	2005	2006	2007
Orion	1	2	3	5	10	20
		3	5	6	8	10
Gemini	2	10	30	50	40	30
		10	20	40	20	10
Scorpion	54	80	60	30	20	15
		70	50	20	5	5
research	4	10	11	12	13	15
		10	11	12	13	15
maintenance	22	30	35	40	42	44
		40	50	50	55	55
total	83	132	139	137	125	124
		133	136	128	101	95

estimate by program manager
estimate by people manager

USN ESI

Module Human Resource Management

by *Gerrit Muller*   University of South-Eastern Norway-NISE

e-mail: gaudisite@gmail.com

www.gaudisite.nl

Abstract

The module Human Resource Management addresses the HRM aspects of systems architects, such as the profile of an architect, selection, education, appraisal and motivation.

January 22, 2023
status: planned
version: 1.0

logo
TBD

Function Profiles; The Sheep with Seven Legs

by *Gerrit Muller*   USN-SE

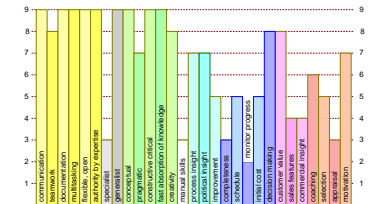e-mail: `gaudisite@gmail.com`

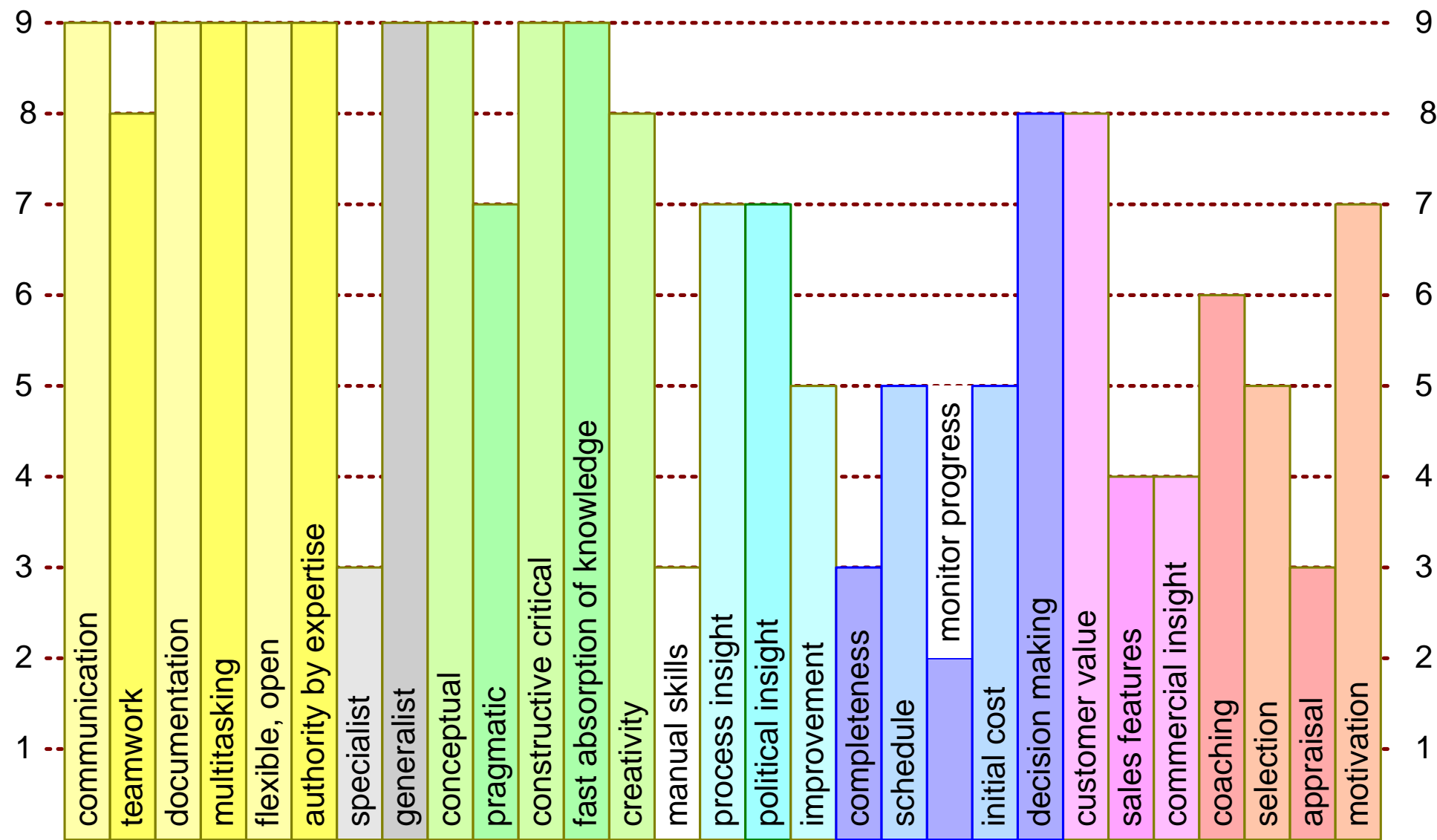`www.gaudisite.nl`

Abstract

The profile of a system architect is quantified for a large list of system architect related characteristics. For comparison the function profiles of related functions are given as well. This profile is based on personal observations and experience.

January 22, 2023
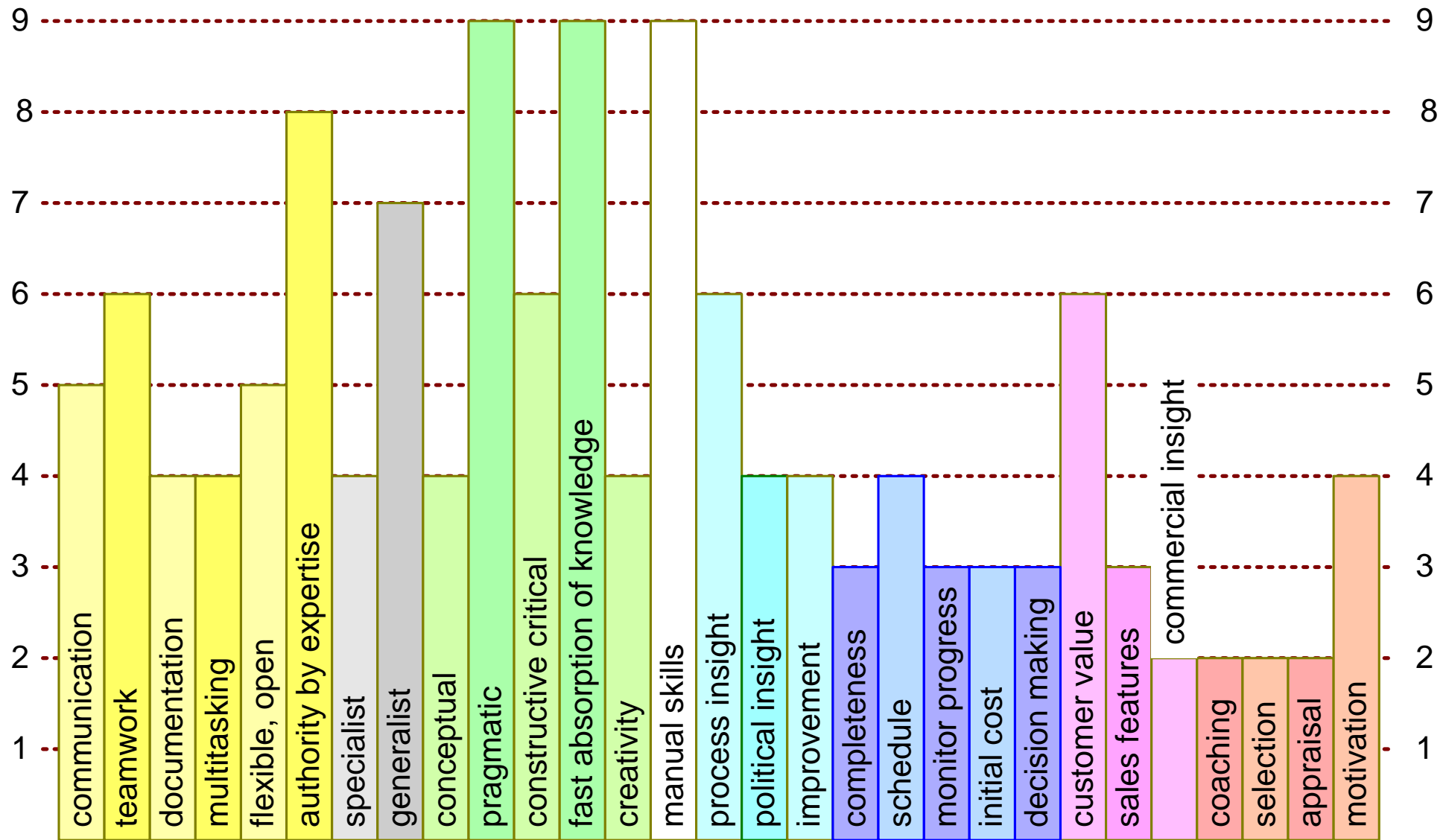status: concept
version: 1.0

System Architect

Test Engineer

Developer

A bar chart displaying Developer function profile values. Categories and values:

- communication: 6
- teamwork: 8
- documentation: 7
- multitasking: 4
- flexible, open: 6
- authority by expertise: 6
- specialist: 9
- generalist: 3
- conceptual: 6
- pragmatic: 8
- constructive critical: 6
- fast absorption of knowledge: 5
- creativity: 9
- manual skills: 7
- process insight: 6
- political insight: 5
- improvement: 4
- completeness: 7
- schedule: 7
- monitor progress: 6
- initial cost: 8
- decision making: 4
- customer value: 5
- sales features: 3
- commercial insight: 2
- coaching: 2
- selection: 3
- appraisal: 2
- motivation: 2

USN ESI

Operational Leader

Line Manager

Chart values (left to right):
- communication: 6
- teamwork: 6
- documentation: 4
- multitasking: 4
- flexible, open: 5
- authority by expertise: 5
- specialist: 7
- generalist: 4
- conceptual: 6
- pragmatic: 5
- constructive critical: 5
- creativity: 4
- manual skills: 2
- fast absorption of knowledge: (axis label)
- process insight: 9
- political insight: 6
- improvement: 9
- completeness: 4
- schedule: 4
- monitor progress: 4
- initial cost: 4
- decision making: 4
- customer value: 5
- sales features: 3
- commercial insight: 3
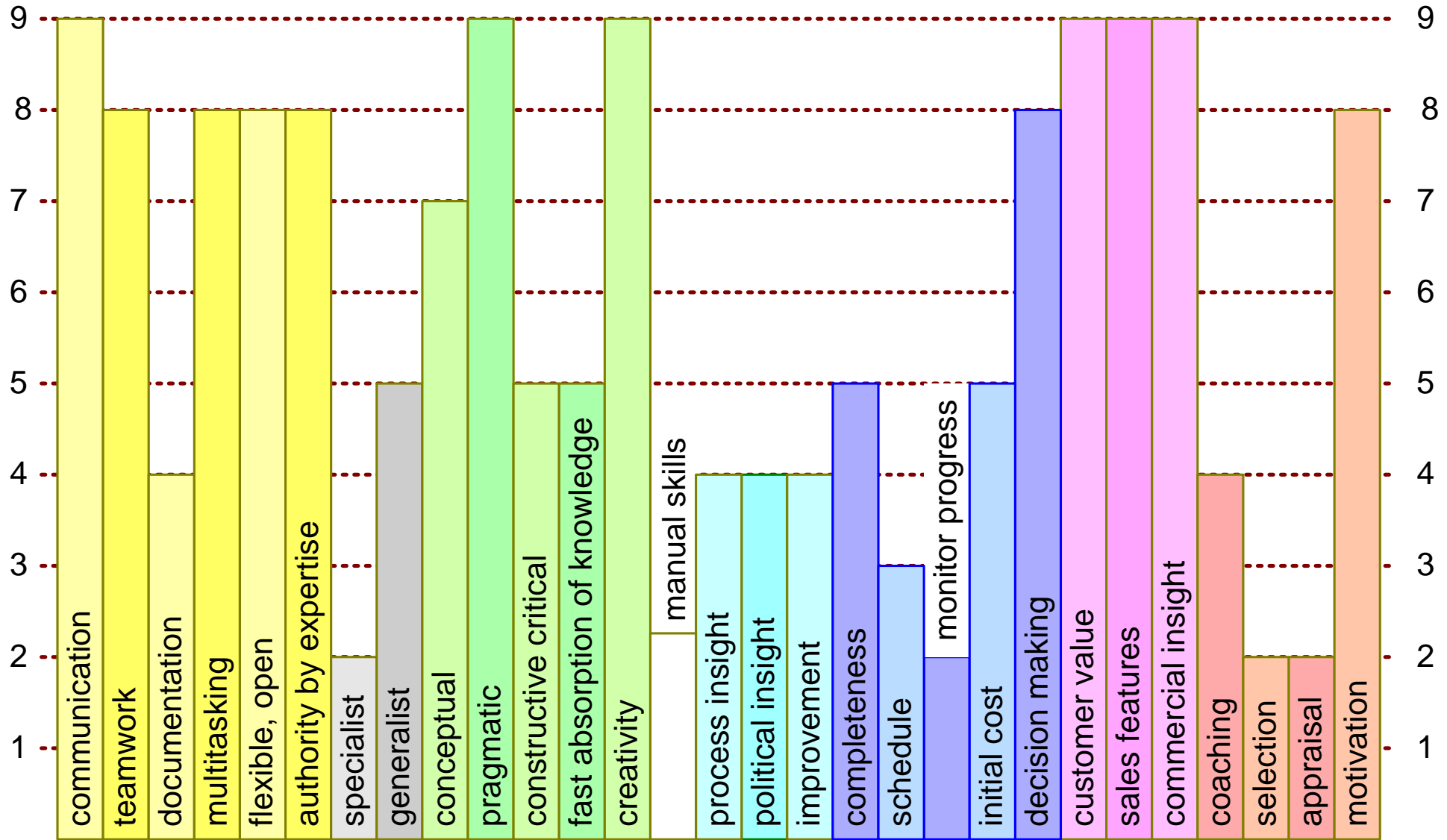- coaching: 9
- selection: 9
- appraisal: 9
- motivation: 9

Commercial Manager

The numbers behind the bars

	communication	teamwork	documentation	multitasking	flexible, open	authority by expertise	specialist	generalist	conceptual	pragmatic	constructive critical	fast absorption of knowledge	creativity	manual skills	process insight	politicsal insight	improvement	completeness	schedule	monitor progress	initial cost	decision making	customer value	sales features	commercial insight	coaching	selection	appraisal	motivation
systems architect	9	8	9	9	9	9	3	9	9	7	9	9	8	3	7	7	5	3	5	2	5	8	8	4	4	6	5	3	7
test engineer	5	6	4	4	5	8	4	7	4	9	6	9	4	9	6	4	4	3	4	3	3	3	6	3	2	2	2	2	4
developer	6	8	7	4	6	6	9	3	6	8	6	5	9	7	6	5	4	7	7	6	8	4	5	3	2	2	3	2	2
operational leader	8	8	4	9	5	4	2	6	4	9	5	5	5	3	9	9	5	9	9	9	9	9	7	5	3	5	6	6	8
line manager	6	6	4	4	5	5	7	4	6	5	5	4	4	2	9	6	9	4	4	4	4	4	5	3	3	9	9	9	9
commercial manager	9	8	4	8	8	8	2	5	7	9	5	5	9	2	4	4	4	5	5	2	5	8	9	9	9	4	2	2	8

How to appraise or assess an architect?

by *Gerrit Muller*   University of South-Eastern Norway-NISE

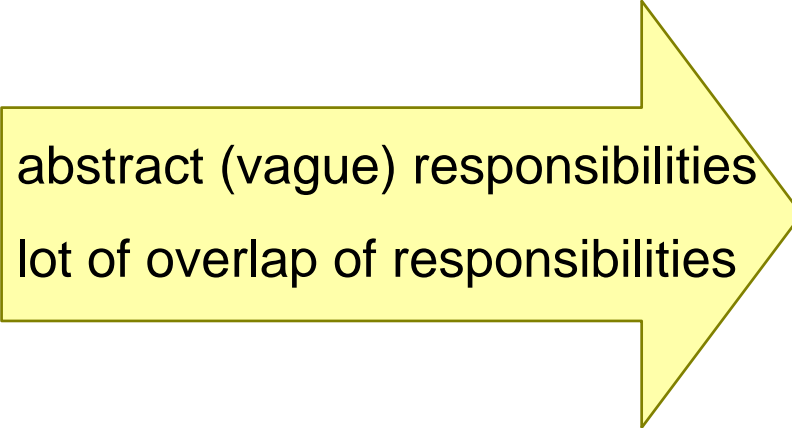e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

The appraisal of system architect is handicapped by the vague and abstract responsibilities of the system architect. The success criterions for architecting are discussed. An approach to "measure" or assess the architect is described.

January 22, 2023
status: planned
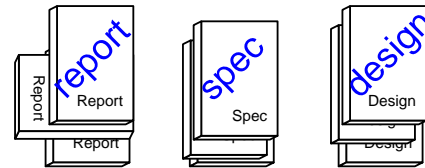version: 0.1

Problem statement

abstract (vague) responsibilities

lot of overlap of responsibilities

- difficult to define yardstick

- difficult to measure

- difficult to compare

- difficult to certify

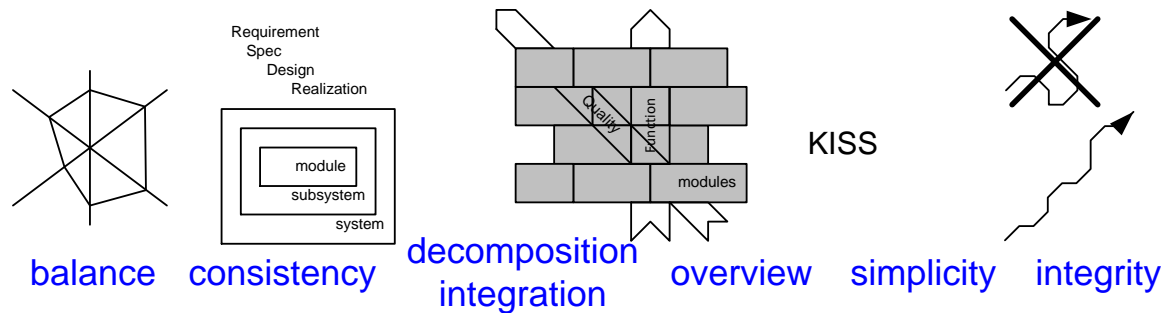- difficult to translate in (financial) consequences

How to assess an architect?

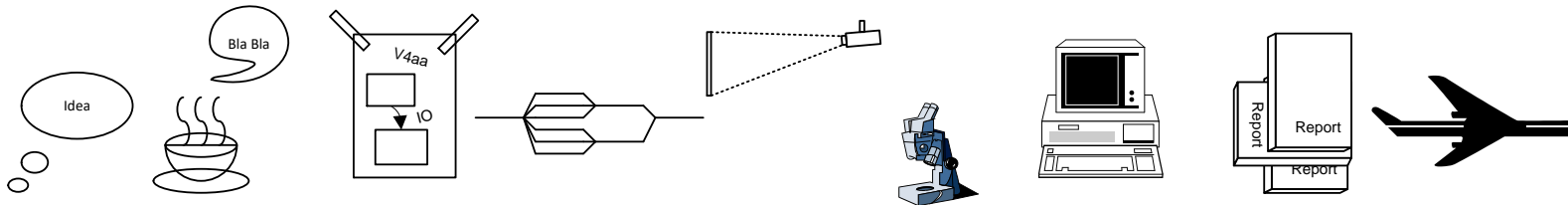USN ESI

Tangible deliverables based upon many invisible activities

report

spec

design

Deliverables

paperwork only

balance   consistency   decomposition integration   overview   simplicity   integrity

KISS

**Responsibilities**

abstract and qualitative

many very detailed

Activities

necessary but invisible

thinking, talking, discussing, scheduling, presenting, measuring, writing, reviewing, visiting customers
analyzing, listening, brainstorming, supporting, teaching, testing, reading, visiting trade-shows
simulating, communicating, troubleshooting, selling, integrating, browsing, consolidating, visiting suppliers

Criterions for successful architecting

How to appraise or assess an architect?
Gerrit Muller

version: 0.1
January 22, 2023
HTAsuccessfulArchitecting
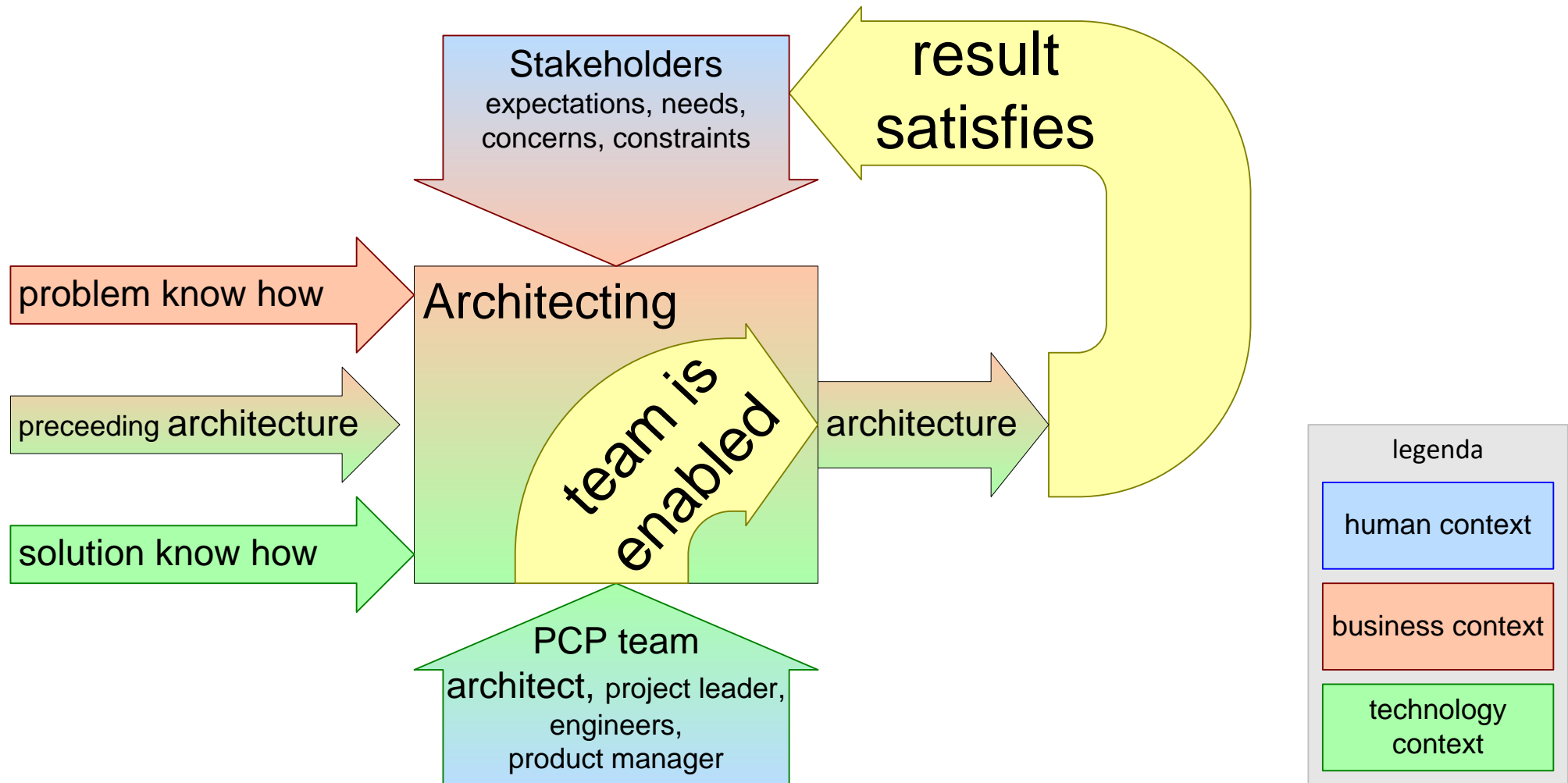
Yardsticks for architect assessment

formalized expectations

function appraisal system,
f.i. from Hay Management Consultants

impact
scope of control
freedom of thinking

job description

deliverables
timing

career development plan

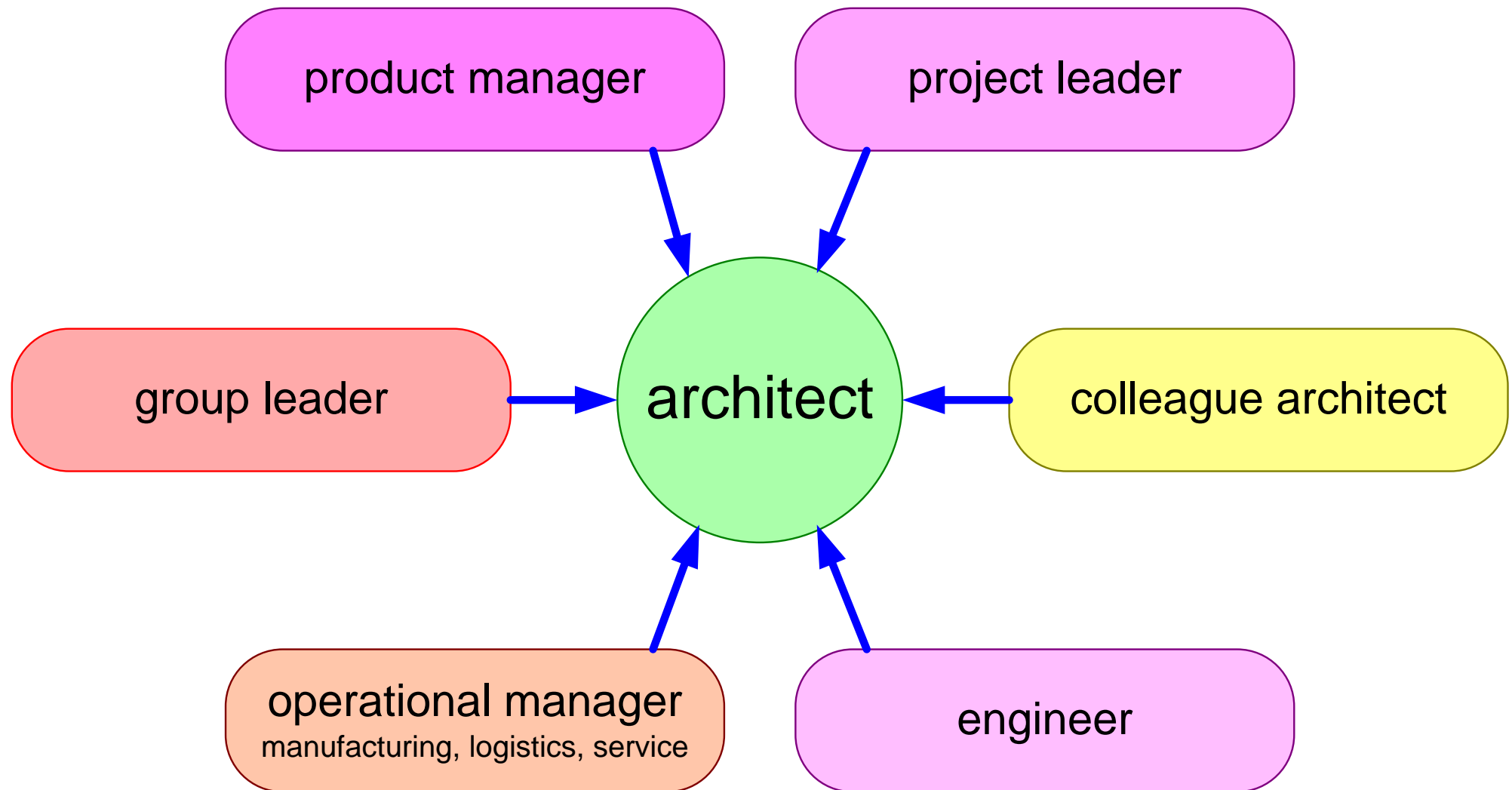skills
know how

actual architect performance

architecture fitness

sales turnover
business success
market continuity

internal *stakeholder* satisfaction

contribution
deliverables
timing
skills
know how

USN   ESI

360 degree assessment

product manager

project leader

group leader

architect

colleague architect

operational manager
manufacturing, logistics, service

engineer

Ranking as trigger for discussions

value for the company

very high

low

potential

Jim Green
(family architect)

The Boss
(business manager)

John Brown
(product architect)

Joe Go
(project leader)

Yo Nerd
(SW engineer)

Se Nior
(chief designer)

Ju Nior
(product architect)

D. Blackhat
(product architect)
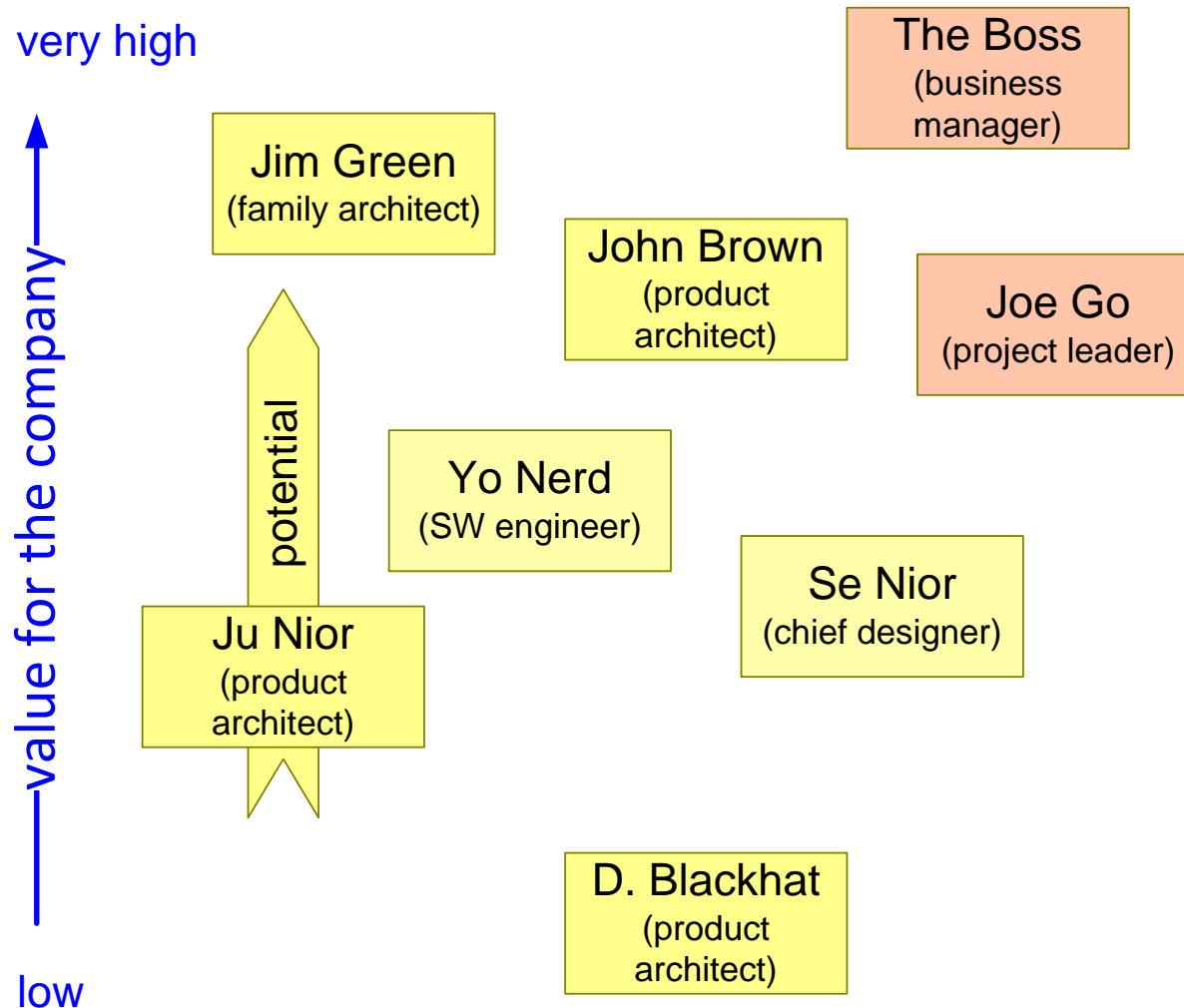
1 ask for ranking

2 ask for justification (why ...?)

3 clarify criterions

4 iterate ranking and justification

USN ESI