

Systems Engineering Fundamentals Course

by *Gerrit Muller*

University of South-Eastern Norway-NISE

Abstract

This course touches all fundamentals of Systems Engineering. Topics are programs, projects, strategy and operation, stakeholders and concerns, needs elicitation and requirements management, concept selection, architecting and design, supply chain, risk management, systems integration, verification and validation, deployment, and life cycle.

Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026
status: planned
version: 0.1

logo
TBD

Systems Engineering Fundamentals Course Overview

by *Gerrit Muller* TNO-ESI, University College of South-Eastern Norway

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

Course overview of the course Systems Engineering Fundamentals.

Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026
status: planned
version: 0

logo
TBD

Single page Course Overview

	day 1	day 2	day 3	day 4	day 5
9:00	course intro	system life cycle supporting systems	concept selection	supply chain and logistics	verification and validation
	systems engineering intro				project management
10:00	case discussion	sketch system <i>life cycle</i>	perform <i>concept selection</i>	sketch <i>goods flow</i>	transform sequence into a <i>PERT plan</i>
	company context programs, projects	needs and requirements	architecture and design	risk management	deployment
11:00	Sketch and discuss <i>program</i> and <i>project organization</i>	identify <i>needs</i> and <i>capabilities</i>	dynamic behavior, functionality	assess <i>risks</i>	sketch <i>installation</i> and <i>commissioning</i>
12:00	lunch	lunch	lunch	lunch	lunch
13:00	system development process	reflection and discussion		reflection and discussion	wrap-up
14:00	sketch a <i>typical mission</i> and a <i>scenario</i>	requirements management	reflection and discussion partitioning and interfaces	systems integration	reflection and discussion
15:00	identify <i>stakeholders</i> and <i>concerns</i>	determine 10 <i>SMART KPPs</i> and <i>use case</i>	make <i>system</i> and <i>work breakdown</i>	determine an incremental <i>integration sequence</i>	
16:00	reflection and discussion	reflection and discussion	reflection and discussion	reflection and discussion	

Systems Engineering Fundamentals; Course Material

by *Gerrit Muller* TNO-ESI, University College of South-Eastern Norway

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

Listing the course material for the course Systems Engineering Fundamentals.

Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026
status: planned
version: 0.1

logo
TBD

core

Systems Engineering Fundamentals Introduction

<http://gaudisite.nl/info/SEFintroduction.info.html>

optional

core

Systems Engineering Fundamentals Course Overview

<http://gaudisite.nl/info/SEFOverview.info.html>

optional

Assignments

core

Systems Engineering Fundamentals Assignments

<http://gaudisite.nl/info/SEFassignments.info.html>

optional

Programs, Projects, Process, Organization

core

Project Systems Engineering Introduction; Phasing, Process, Organization

<http://gaudisite.nl/info/ProjectSEintroPPO.info.html>

Module System Architecture Context

<https://gaudisite.nl/ModuleSystemArchitectureContextPaper.pdf>

Products, Projects, and Services; similarities and differences in architecting

<https://gaudisite.nl/ProductsProjectsServicesPaper.pdf>

optional

System Engineering Management Plan (SEMP) DOES ONE SIZE FIT ALL?

Zonnenshain, A., Malotaux, N., Honour, E., Kasser, J., Urio, U., Shabtay, M.,
INCOSE 2009

Systems Engineering Management Plan (SEMP) Technical Content

<https://www.nasa.gov/consortium/SystemsEngineeringManagementPlanTechnicalContent>

core

Systems Engineering Fundamentals Life Cycle

<http://gaudisite.nl/info/SEFlifeCycle.info.html>

Modeling and Analysis: Life Cycle Models

<https://gaudisite.nl/MAlifeCyclePaper.pdf>

optional

SEBoK Life Cycle models

https://www.sebokwiki.org/wiki/Life_Cycle_Models

Needs and Requirements

core

Systems Engineering Fundamentals Needs Elicitation

<http://gaudisite.nl/info/SEFneeds.info.html>

optional

SEBoK Stakeholder Needs and Requirements

https://www.sebokwiki.org/wiki/Stakeholder_Needs_and_Requirements

Requirements Management

core

Systems Engineering Fundamentals Requirements Management

<http://gaudisite.nl/info/SEFrequirements.info.html>

Fundamentals of Requirements Engineering

<https://gaudisite.nl/FundamentalsOfRequirementsPaper.pdf>

optional

Concept Selection

core

Concept Selection, Set Based Design and Late Decision Making

<https://gaudisite.nl/SEFconceptSelectionSlides.pdf>

optional

Concept Selection - Applying Pugh Matrices in the Subsea Processing Domain by Linda Lønmo and Gerrit Muller; INCOSE 2014 in Las Vegas

https://gaudisite.nl/INCOSE2014_Lonmo_Muller_ConceptSelection.pdf

Researching the application of Pugh Matrix in the sub-sea equipment industry by Gerrit Muller, Dag Jostein Klever, Halvard H. Bjørnsen, and Michael Pennotti; CSER 2011 in Los Angeles

https://gaudisite.nl/CSER2011_MullerEtAl_ResearchingPughMatrix.pdf

Visualizing Dynamic Behavior

core

Visualizing Dynamic Behavior

<http://gaudisite.nl/info/VisualizingDynamicBehavior.info.html>

optional

Creating an A3 Architecture Overview; a Case Study in SubSea Systems by Gerrit Muller, Damien Wee, and Martin Moberg; INCOSE 2015 in Seattle, WA, USA

http://gaudisite.nl/INCOSE2015_MullerEtAl_SubseaOverviewA3.pdf

core

Systems Engineering Fundamentals Supply Chain and Logistics

<https://gaudisite.nl/SEFsupplyChainSlides.pdf>

optional

Build to order https://en.wikipedia.org/wiki/Build_to_order

P-D Ratios <https://oldleandude.com/2015/05/27/p-d-ratios/>

core

Systems Engineering Fundamentals Risk Management

<https://gaudisite.nl/SEFriskManagementSlides.pdf>

optional

Failure Mode and Effects Analysis

https://en.wikipedia.org/wiki/Failure_mode_and_effects_analysis

Readiness Levels

core

Course Systems Integration; Readiness Levels

<http://www.gaudisite.nl/info/MSIreadinessLevels.info.html>

optional

From TRL to SRL: The Concept of Systems Readiness Levels

CSER 2006, Brian Sauser et al.

Technology Readiness Levels

https://en.wikipedia.org/wiki/Technology_readiness_level

Systems Integration Process and Positioning

core

Mastering Systems Integration; Process and Positioning

<http://gaudisite.nl/info/MSIprocessAndPositioning.info.html>

optional

SESA /SARCH Module 01, System Architecture Context

<http://gaudisite.nl/info/ModuleSystemArchitectureContext.info.html>

core

Course Systems Integration; Project Management

<http://gaudisite.nl/info/MSIprojectManagement.info.html>

optional

Combating Uncertainty in the Workflow of Systems Engineering Projects

INCOSE 2013, Barry Papke and Rick Dove

Verification and Validation Terminology

core

Course Systems Integration; Terminology

<http://www.gaudisite.nl/info/MSITerminology.info.html>

optional

Understanding Objective Evidence: (What It Is and What It Definitely Is Not),
by Denise Dion

http://www.eduquest.net/Advisories/EduQuest%20Advisory_ObjectiveEvidence.pdf

List of Cognitive Biases, Wikipedia:

https://en.wikipedia.org/wiki/List_of_cognitive_biases

Systems Engineering Fundamentals Assignments

by *Gerrit Muller* TNO-ESI, USN-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

All assignments of the course Systems Engineering Fundamentals.

Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026
status: planned
version: 0

logo
TBD

Propose a Non-Lethal Urban Crowd Controller

Sketch the **system-of-interest**

Sketch some of the **environment** the system will be operating in

Sketch some of the **system internals**

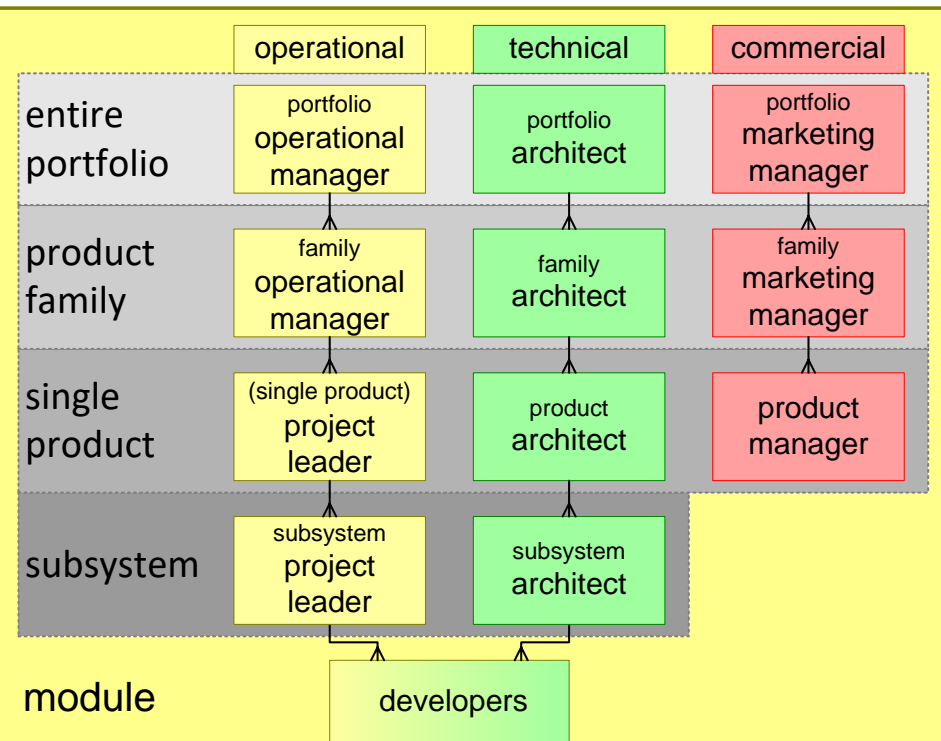
Draw the **system boundary**

Map the Operational Organization

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**



Sketch Mission and Scenario

Sketch
a *typical mission*
and a specific *scenario*.

The scenario needs to be highly specific:

- numbers (how much, how far, how accurate)
- names (where, who)
- circumstances (when, where)
- actions (what, how)

Identify Stakeholders and Concerns

Brainstorm **stakeholders**

Brainstorm for each stakeholder the **concerns**

Elaborate concerns in 5 to 10 words, make them more specific

Use the **mission** and **scenario** for inspiration

Sketch the System Life Cycle

Sketch the system ***life cycle***

from idea until decommissioning and recycling.

Identify **stakeholders** per phase or activity

Identify Needs and Capabilities

Identify *stakeholder needs*

in terms of *capabilities*.

Capabilities typically are *functions*

with *quantifiable characteristics*

Use the mission, scenario, and stakeholder analysis for inspiration

Determine Key Performance Parameters and Use Case

Determine 5 to 10 **Key Performance Parameters** (KPP) of the System




Quantify these KPPs

Define the KPPs roughly, using a **Use Case**

Perform a Concept Selection

Make a **decision matrix** for one of the **concept selections**.

- define at least 3 concepts
- define 7 to 10 criteria for selection
- score the concepts against the criteria, for example using a scale from 1 to 5: 1 = very poor, 5 = very good
- recommend a concept with a rationale

	concept 1 	concept 2 	concept 3 
critterion 1	vulnerable MTRR 1	robust 30 hours 3	robust access 5
critterion n	reusing platform 4	reusing platform 4	resource shortage 2
			best, because ...

Model the Dynamic Behavior of the System.

Focus on the Dynamic Behavior that relates to the KPP.

Visualize the Dynamic Behavior with various sketches, diagrams, or graphs (see Visualizing Dynamic Behavior for inspiration).

Make a ***system breakdown***

in subsystems and subsystems

and a ***work breakdown structure***

to assist in organizing the project

Sketch the Goods Flow

sketch the **goods flow**

from (sub) **suppliers**

via **assembly** and **test**

to **customer site**,

deployment,

and **maintenance**

Assess *risks*

- *feasibility* of achieving *KPPs*
- *fitness for purpose* in customer context
- *integration configurations* and *testware*
- *supplier* and *logistics* status
- *technology readiness*
- *development* and *resource* status

Determine *probability* and *severity* per risk

Determine an Incremental Integration Sequence

Determine an incremental ***integration sequence*** to build confidence in the KPP ASAP.

Strive for about 6 main increments.

Reason starting at the end result and then backward in time.

For each increment determine its prerequisites in terms of parts, interfaces, functions, and performance levels.

Transform Sequence into a PERT Plan

Transform the integration sequence and the planning from the other perspectives into a **PERT-plan**.

A PERT-plan focuses on **activities** and their mutual **relations**; the logic of the plan. Time and resources are secondary information.

Sketch an Installation and Commissioning

Sketch an *installation*
and *commissioning*

Systems Engineering Fundamentals Introduction

by *Gerrit Muller* TNO-ESI, University College of South-Eastern Norway

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

This presentation introduces the ideas behind the course Systems Engineering Fundamentals.

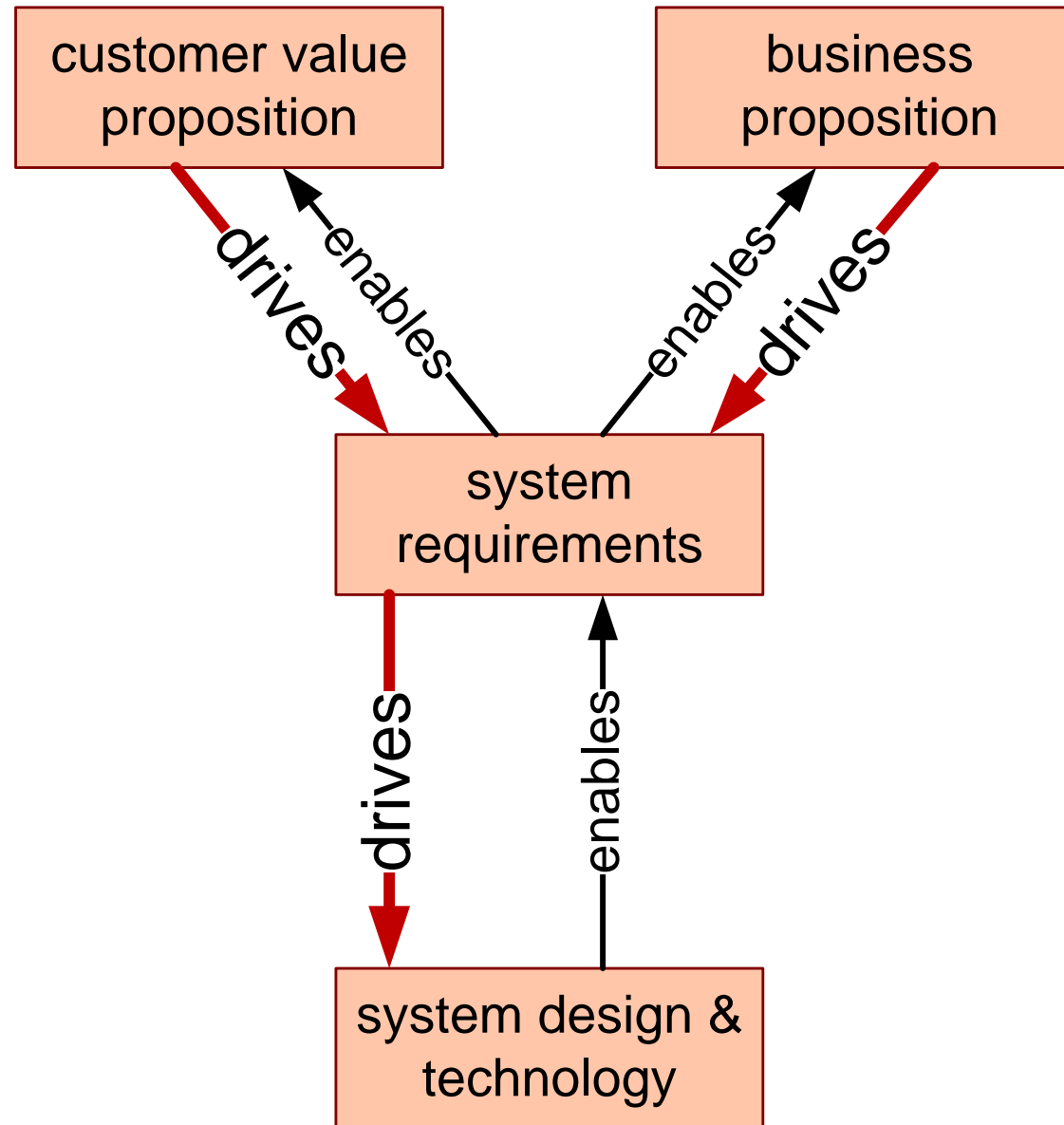
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

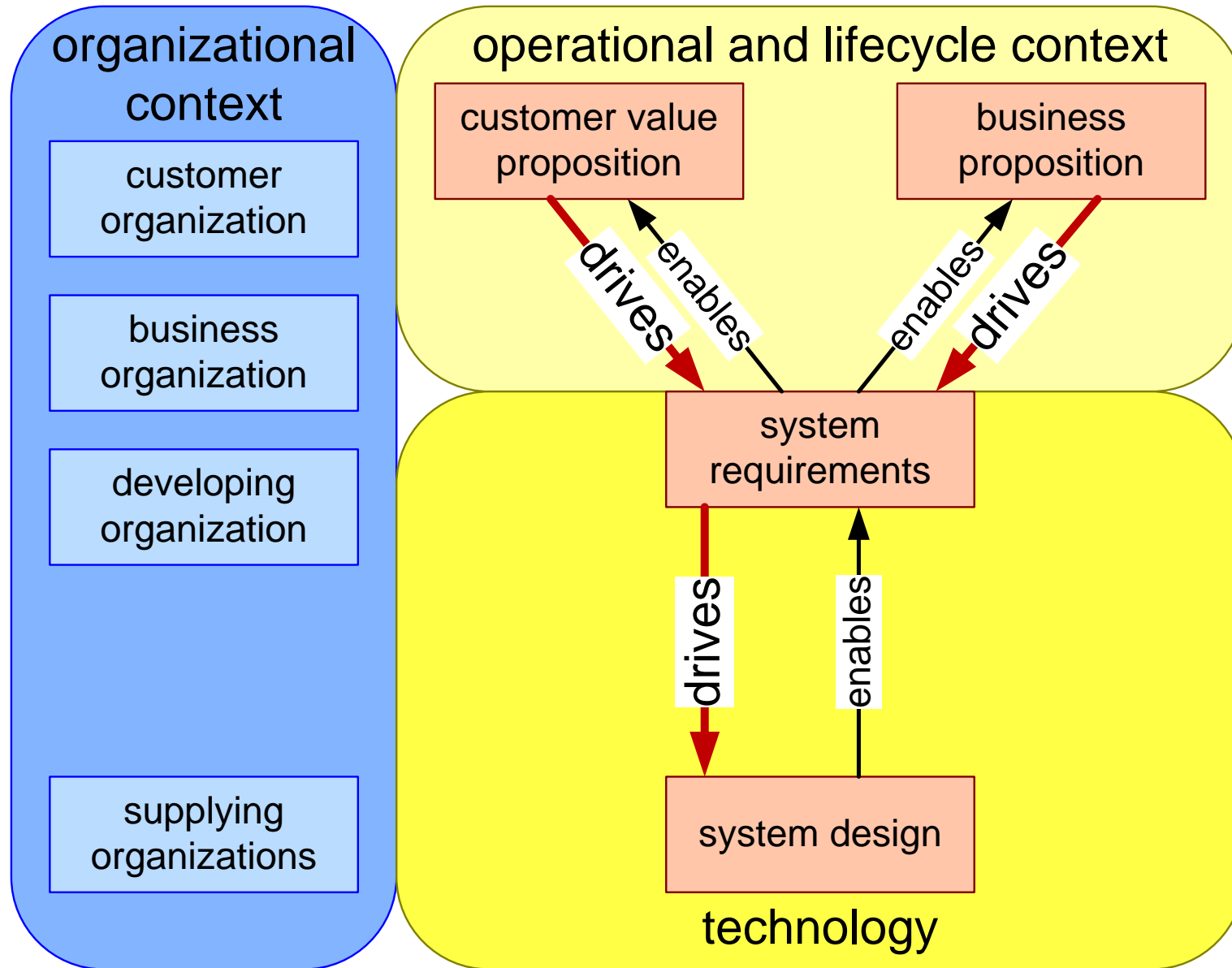
February 8, 2026
status: planned
version: 0

logo
TBD

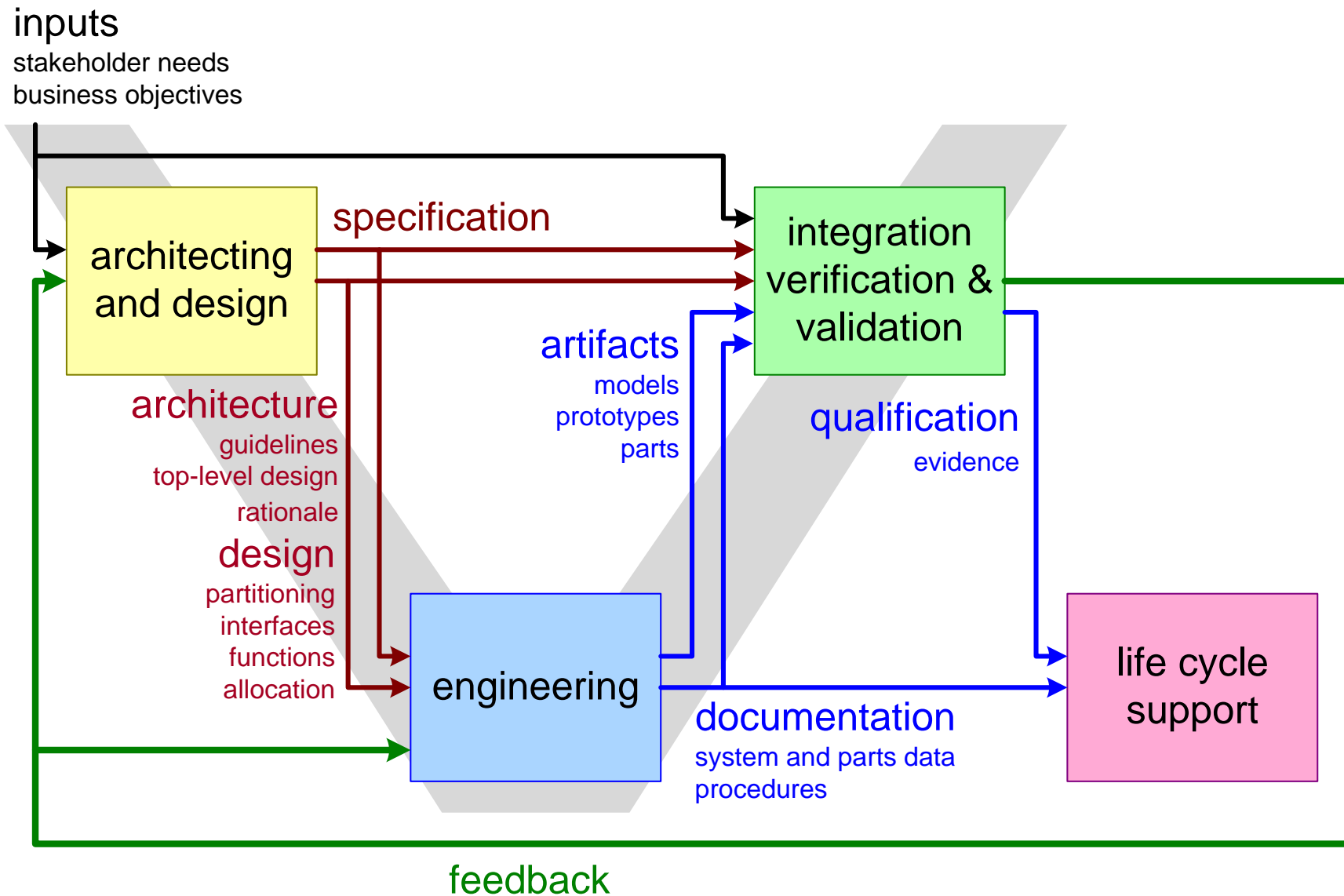
Architecture Top View



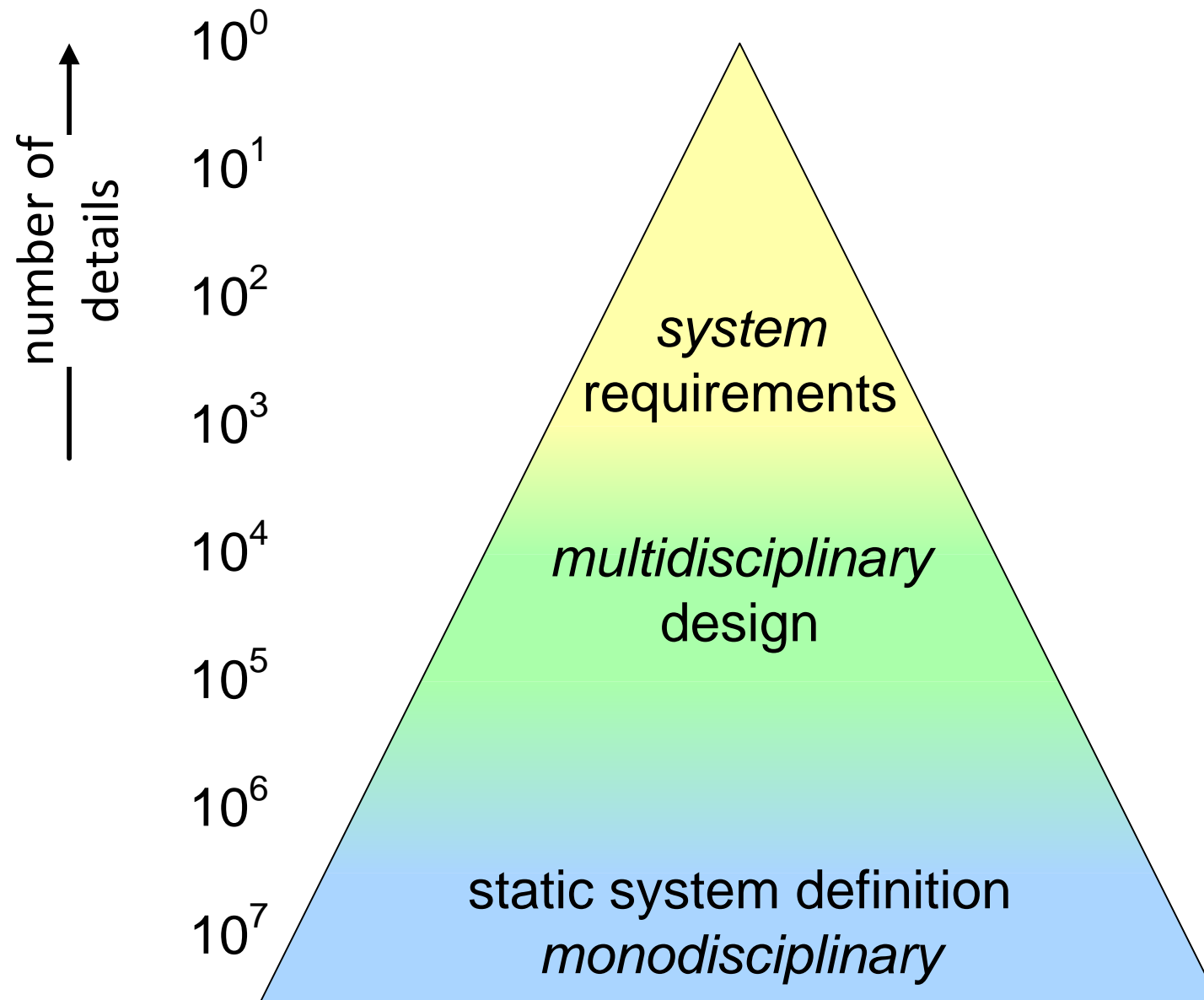
Architecting Playing Field



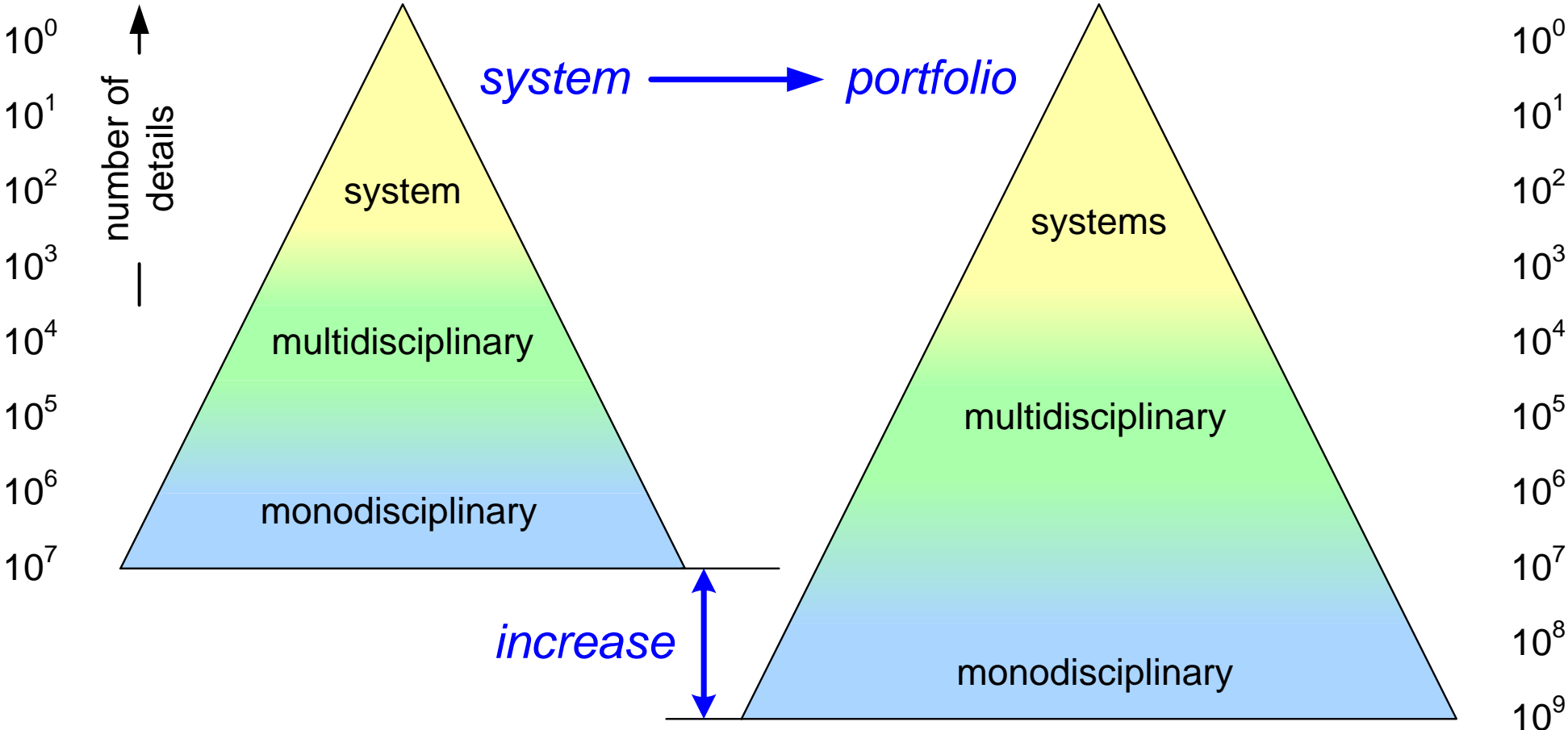
Simplified Systems Engineering Model



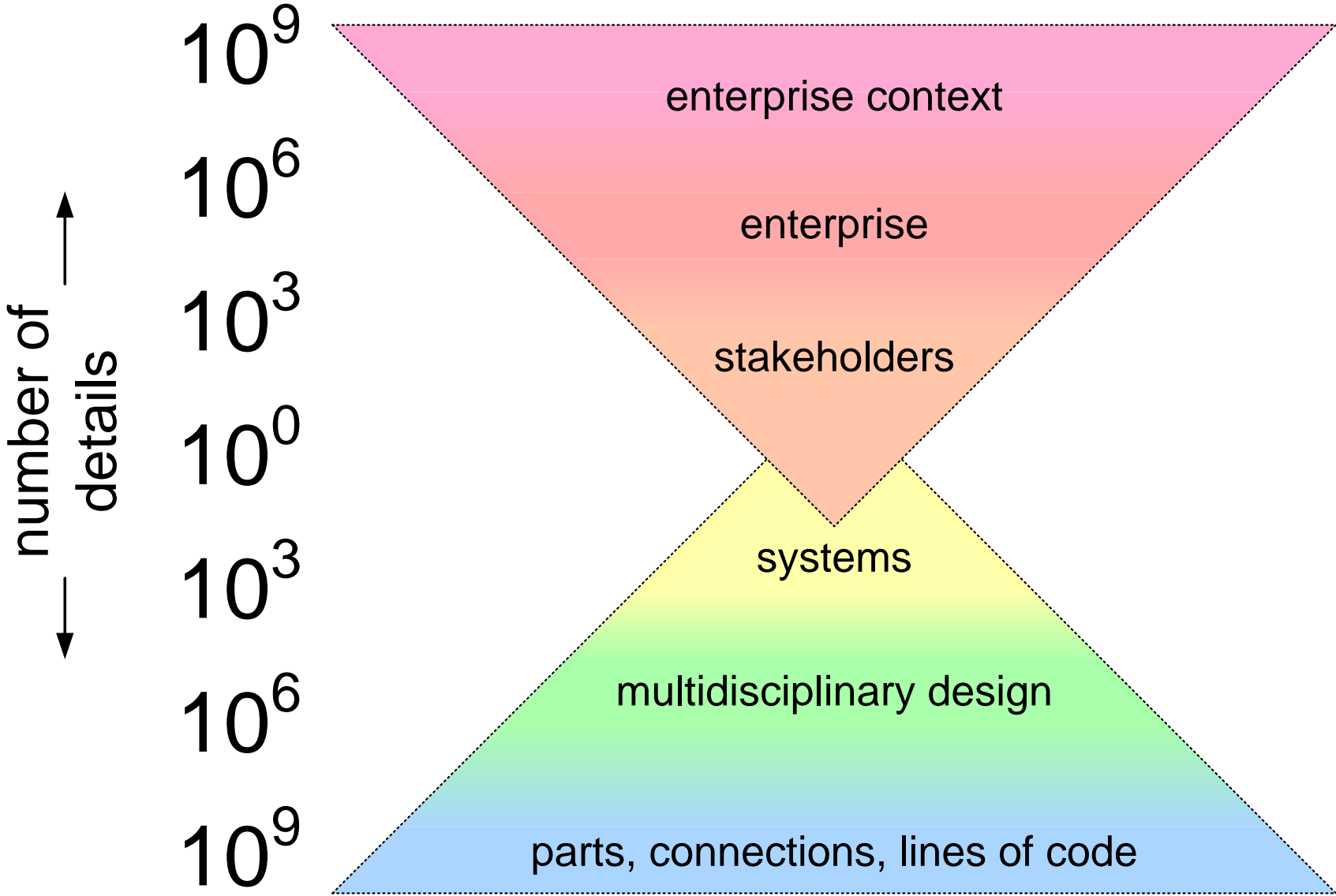
Level of Abstraction Single System

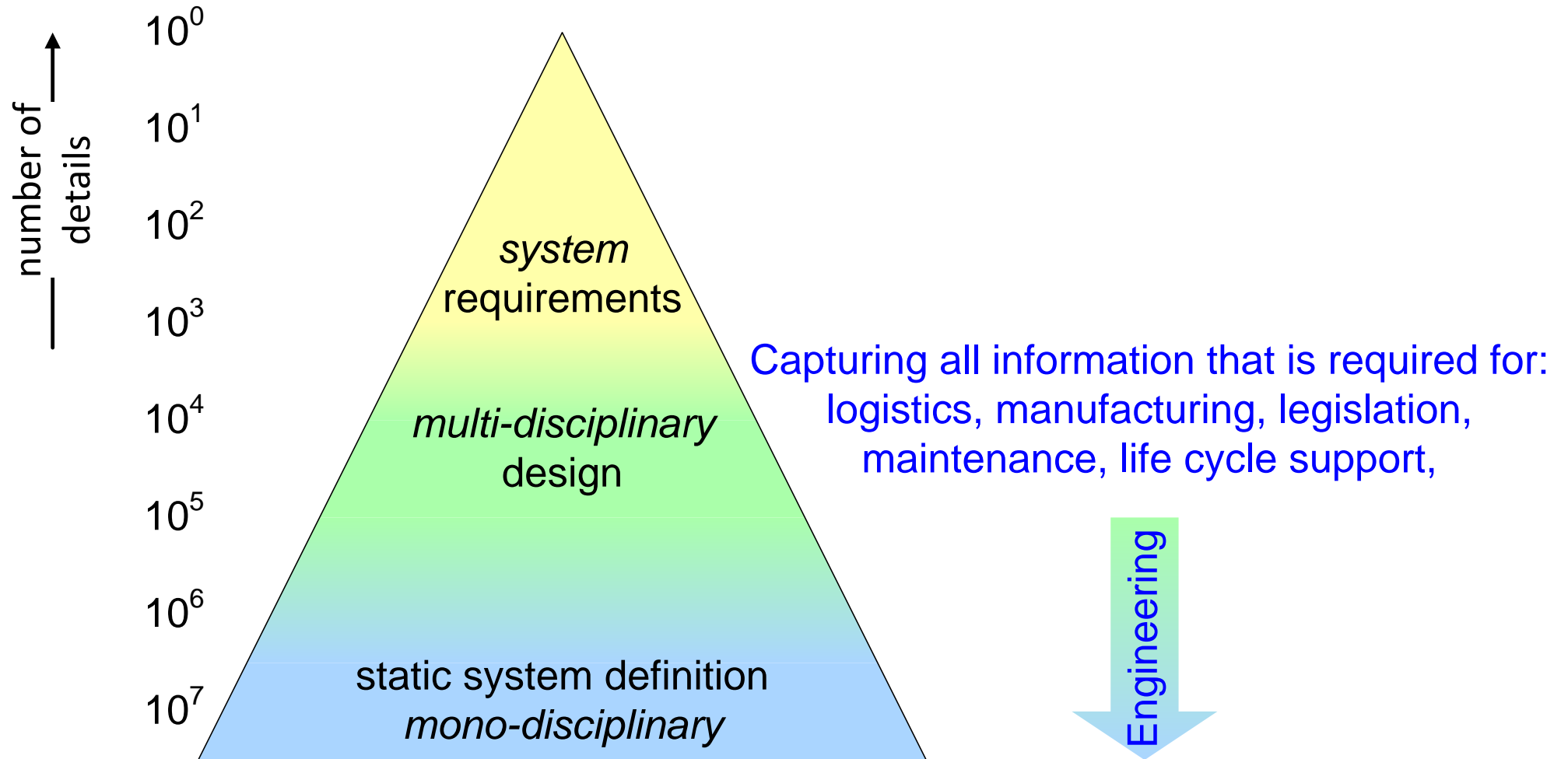


From system to Product Family or Portfolio

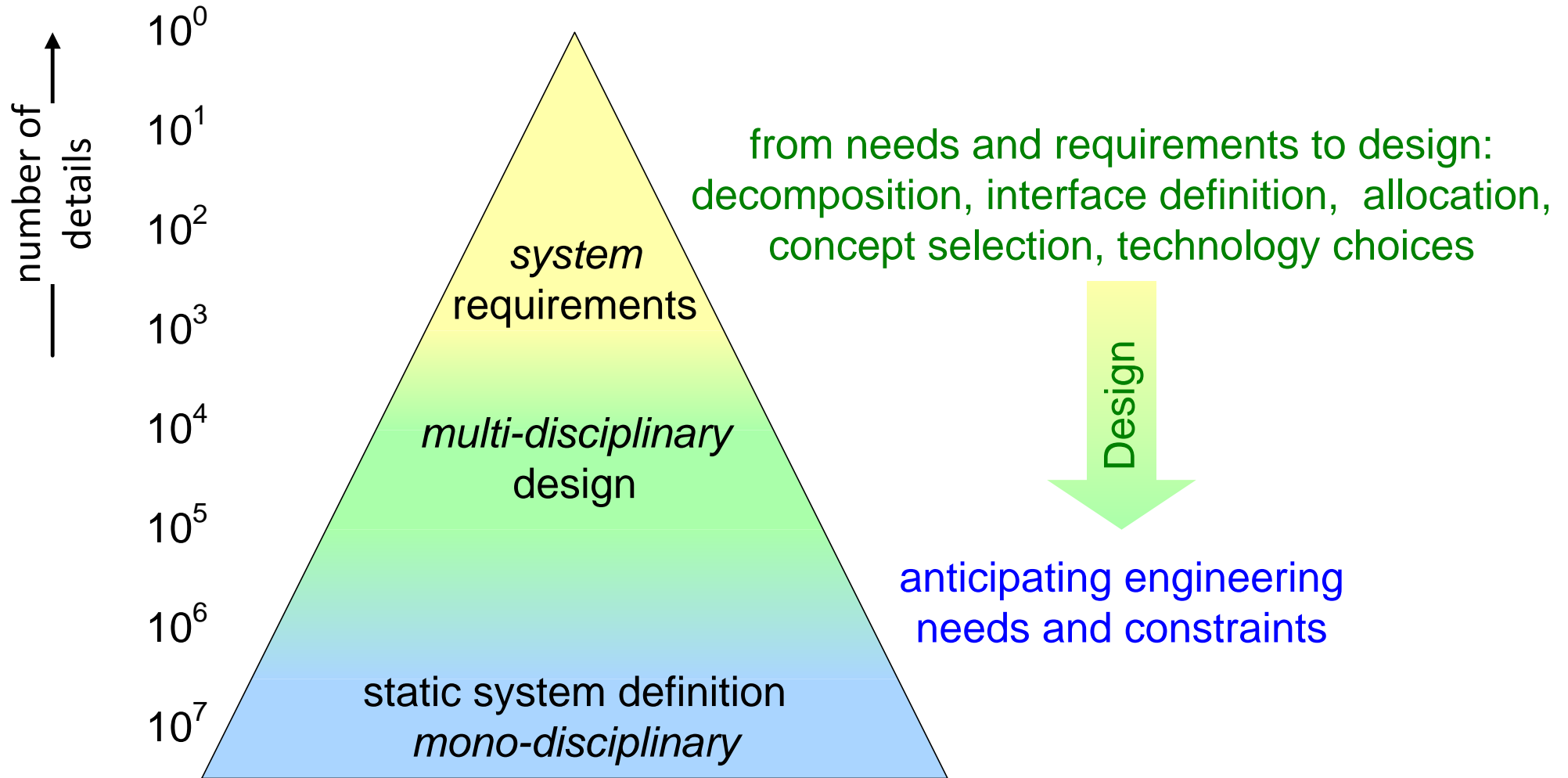


Product Family in Context

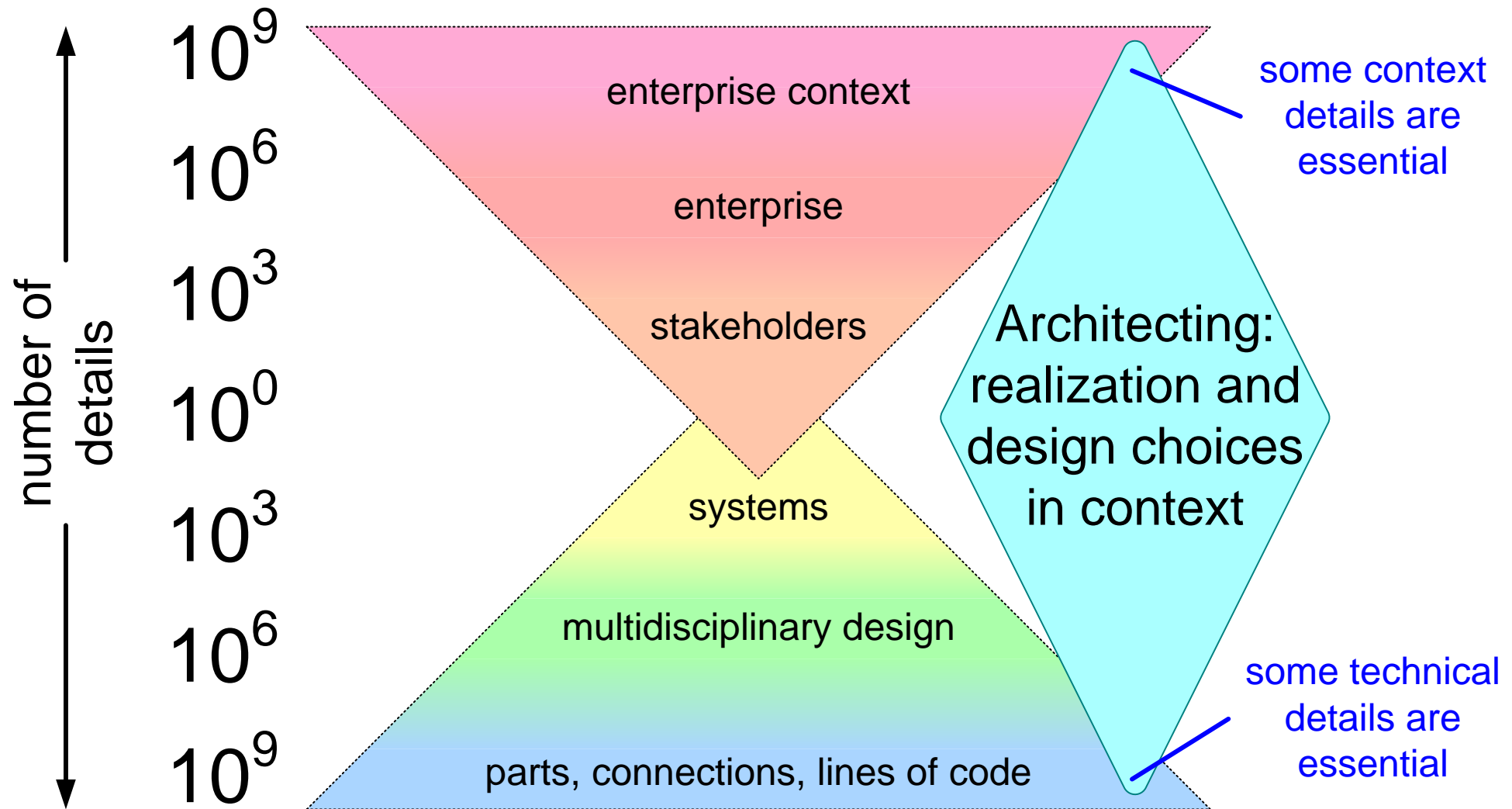




Design



Architecting



Project Systems Engineering Introduction; Phasing, Process, Organization

by *Gerrit Muller* USN-SE

e-mail: gaudisite@gmail.com

www.gaudisite.nl

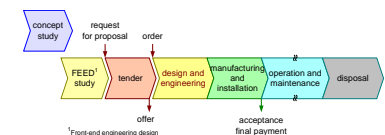
Abstract

The fundamental concepts and approach to project oriented Systems Engineering are explained. We look at project phasing, phase transition, processes, and organization.

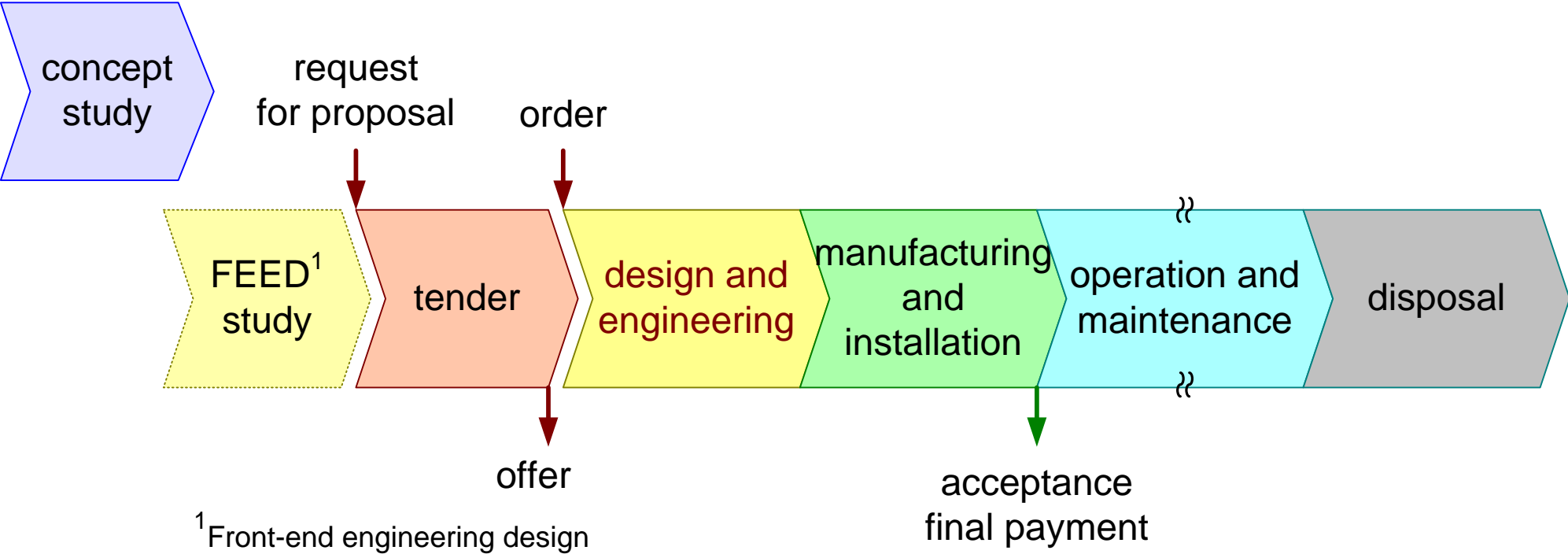
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

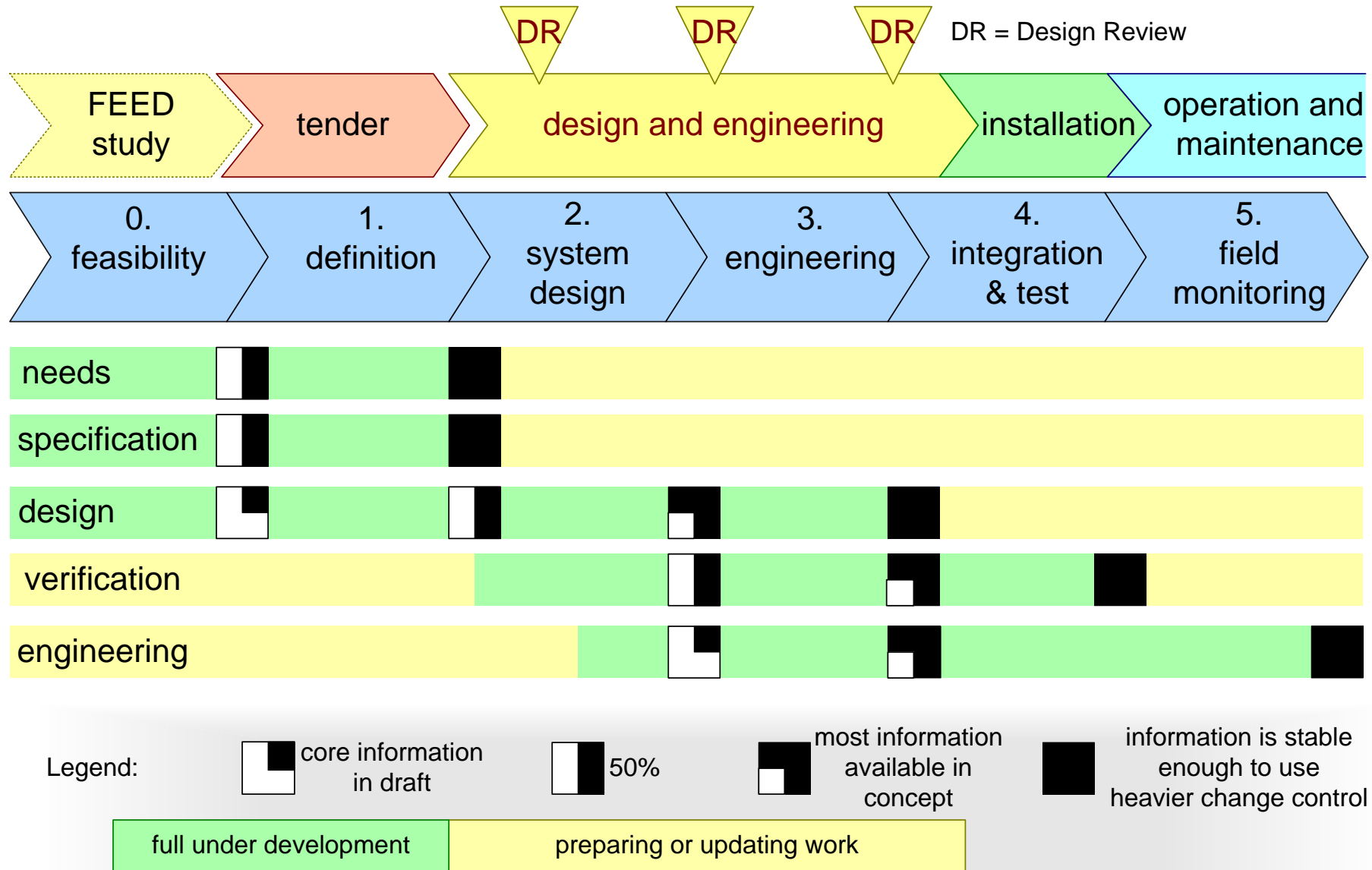
February 8, 2026
status: preliminary
draft
version: 0.2



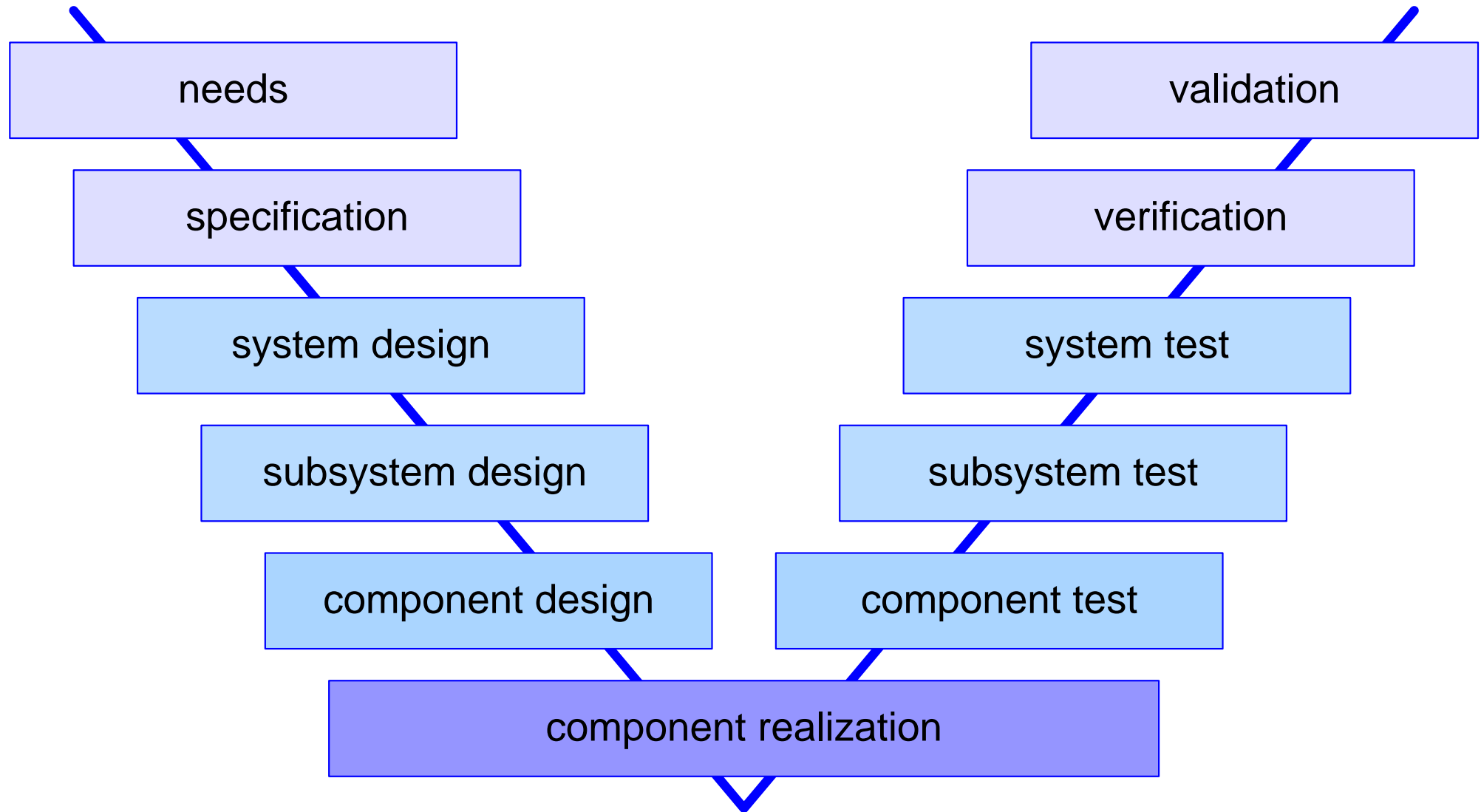
Project Life Cycle



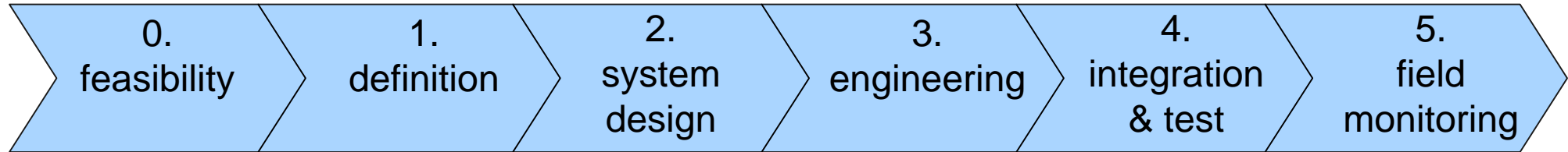
Phased Project Approach



V-Model



All Business Functions Participate



sales

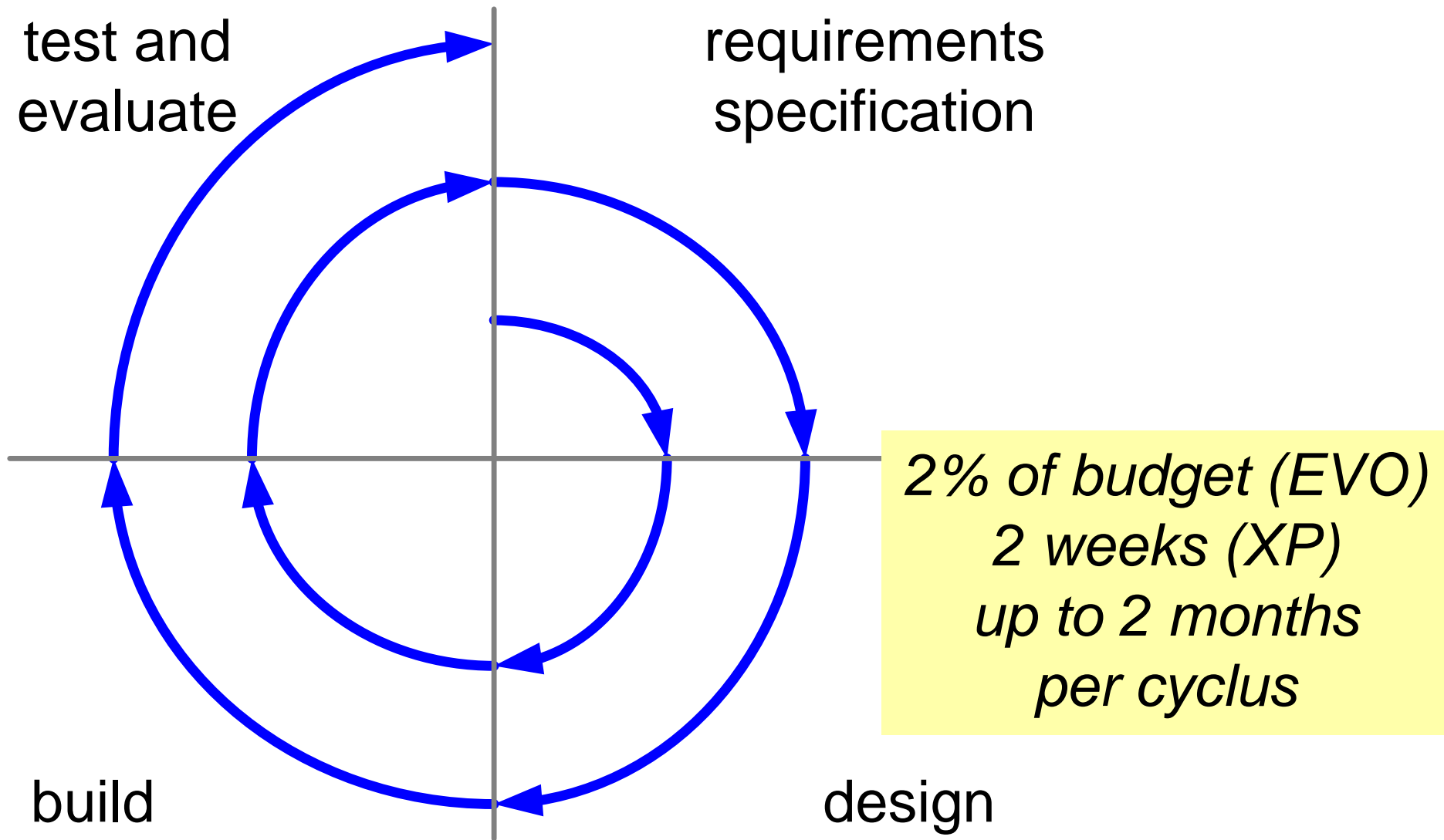
logistics

production

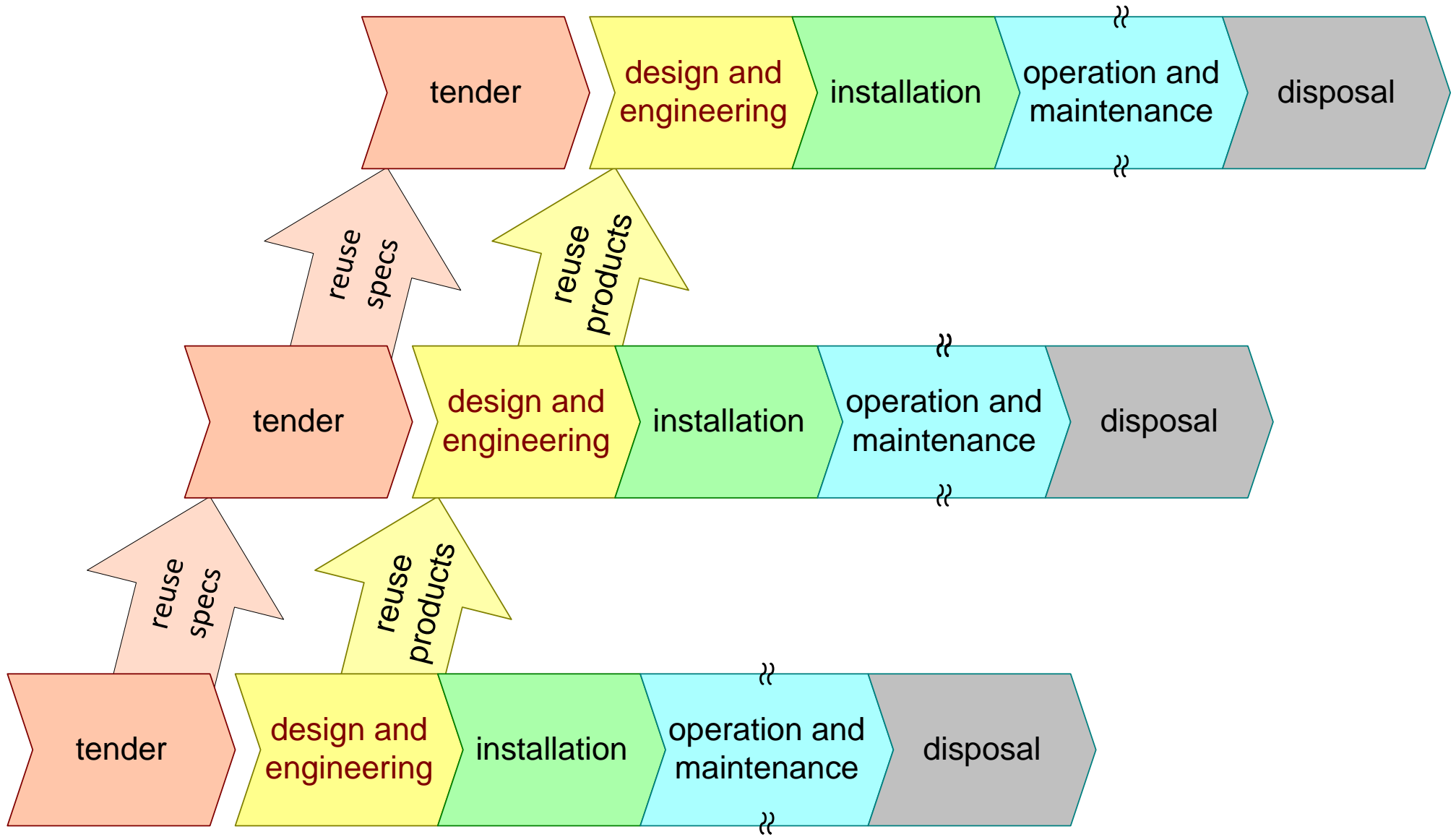
service

development & engineering: marketing, project management, design

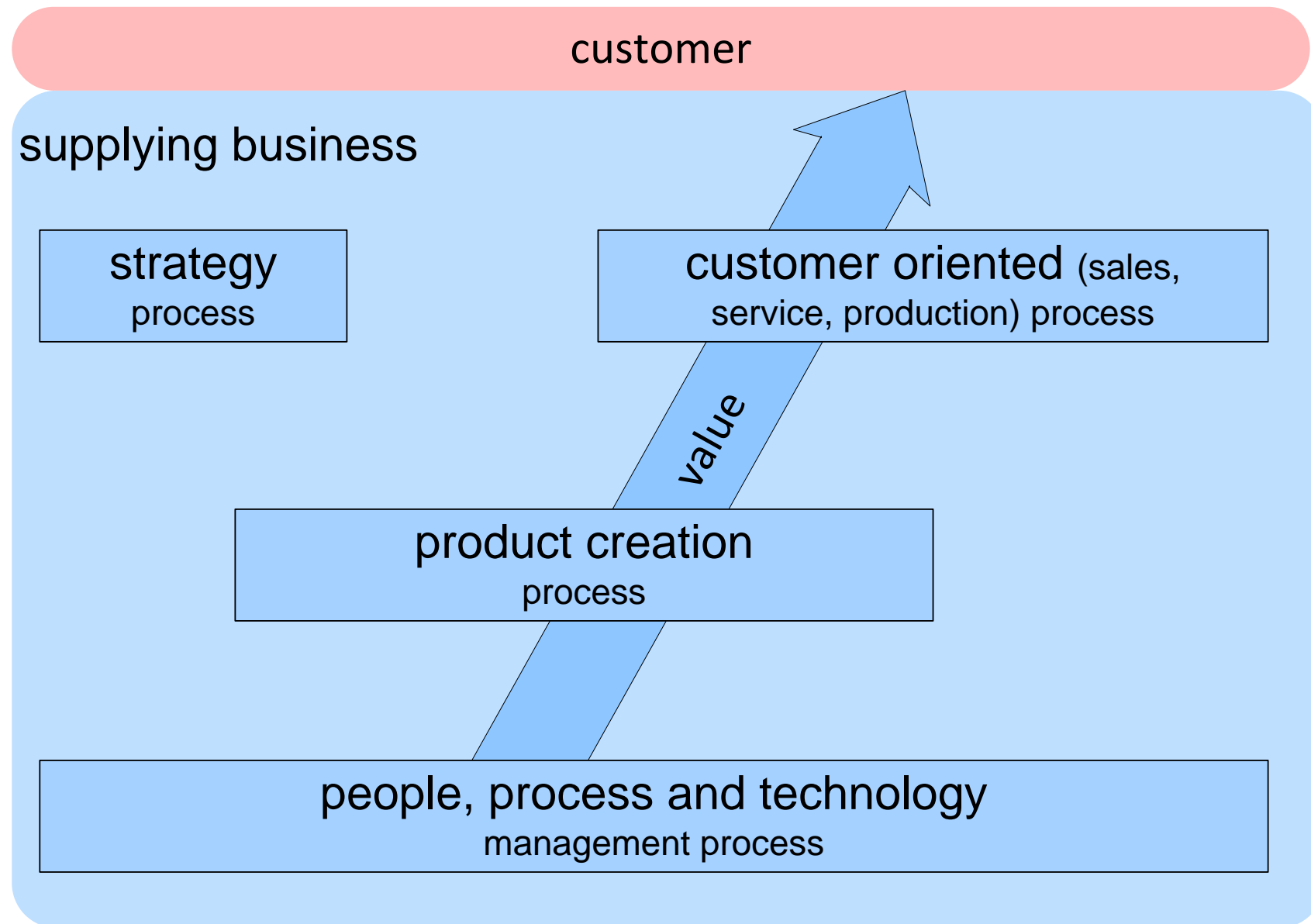
Evolutionary PCP model



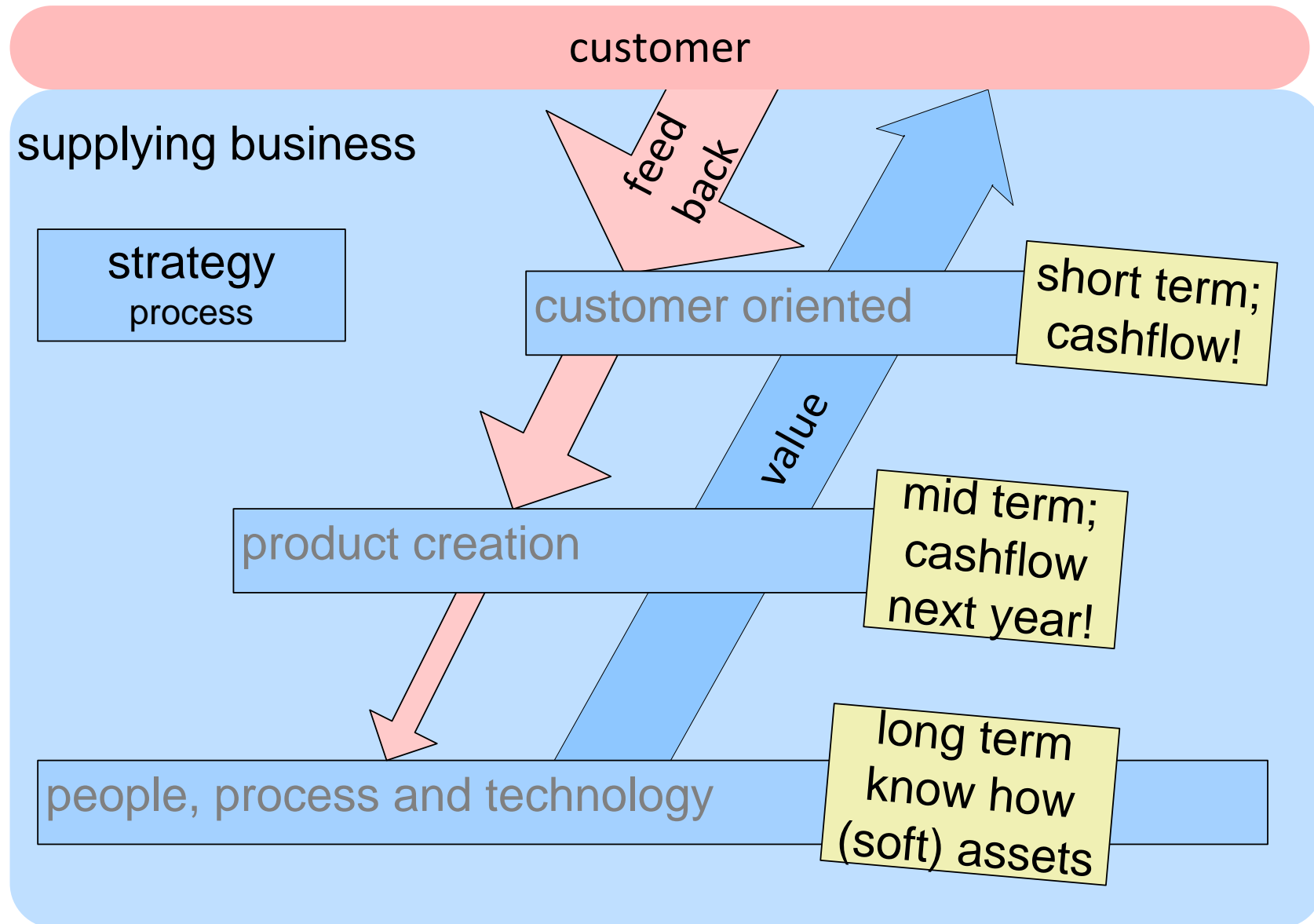
Reuse and Products



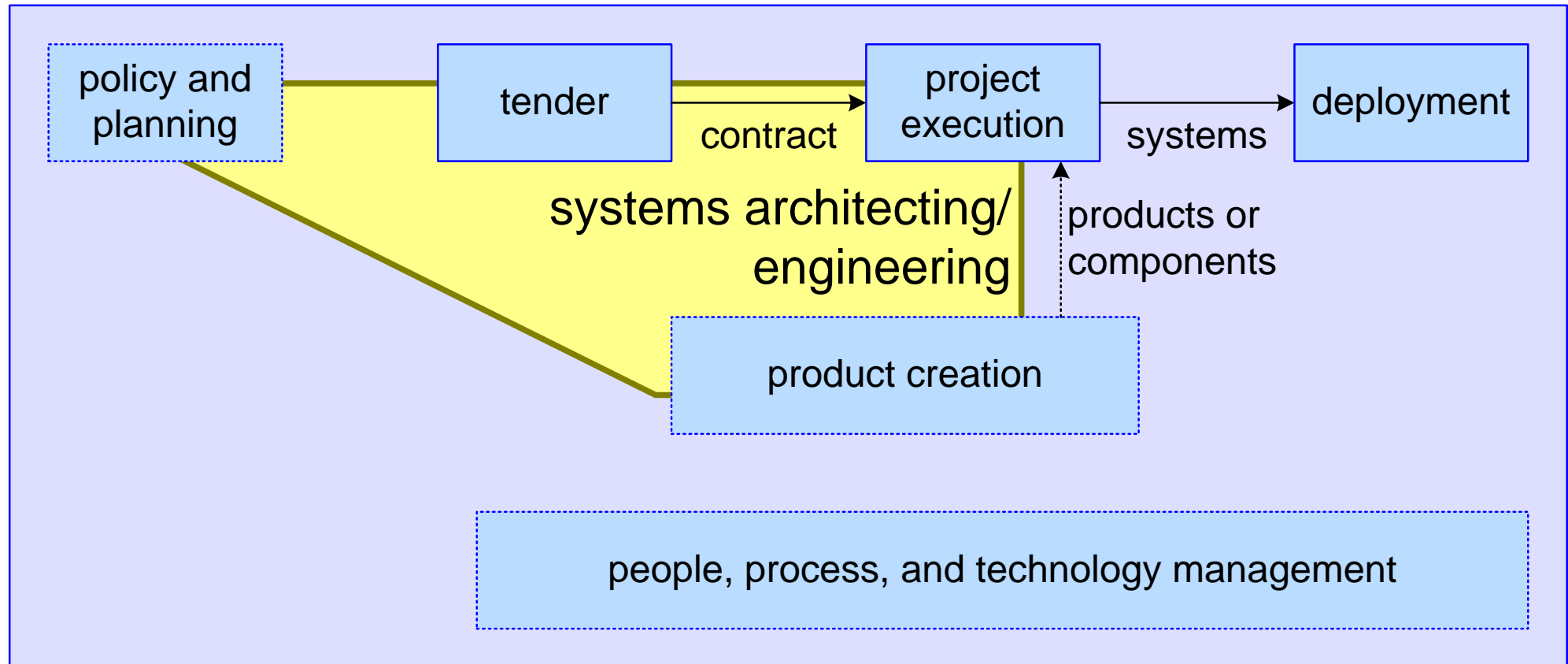
Simplified Process View



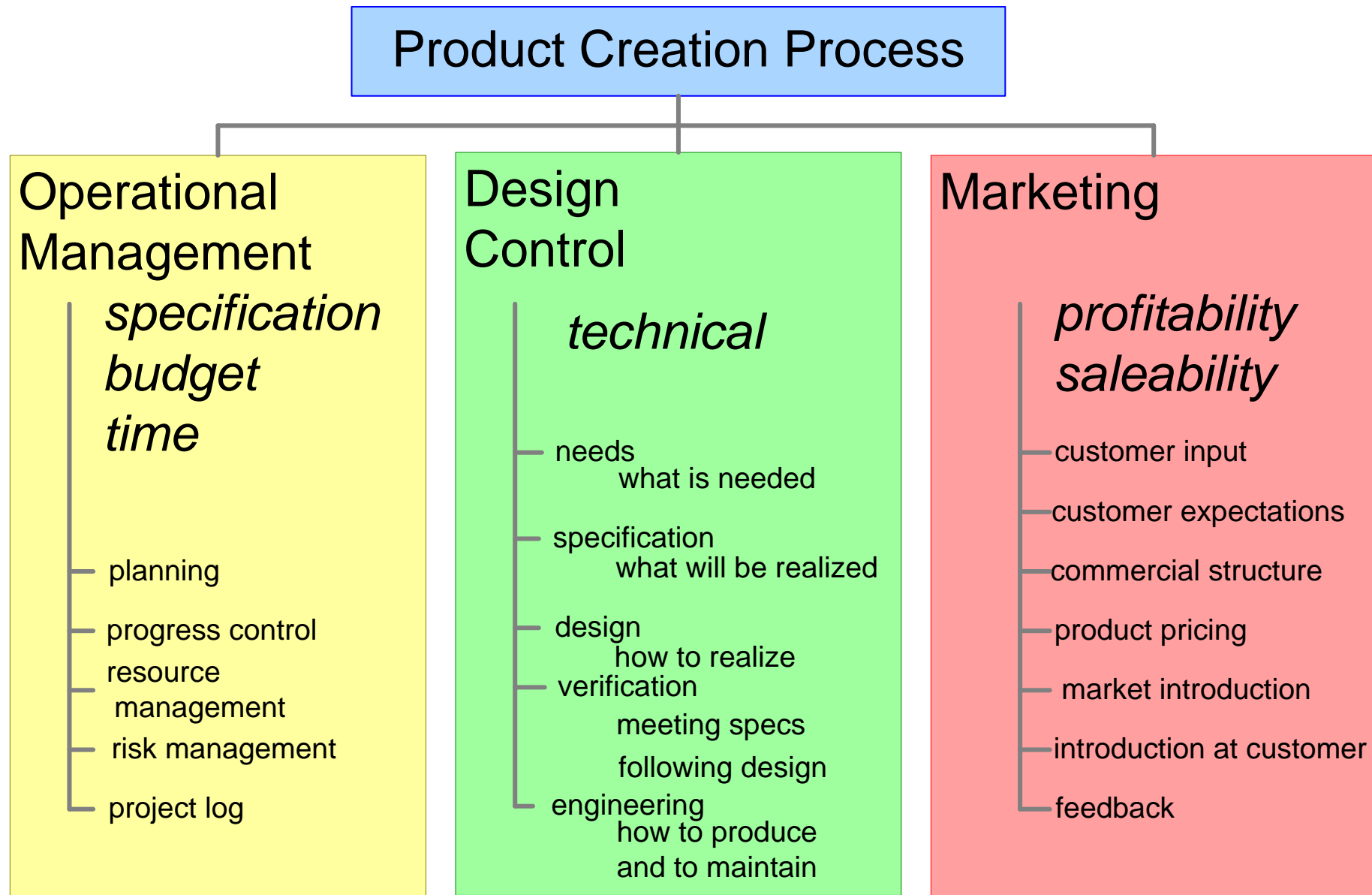
Simplified Process; Money and Feedback



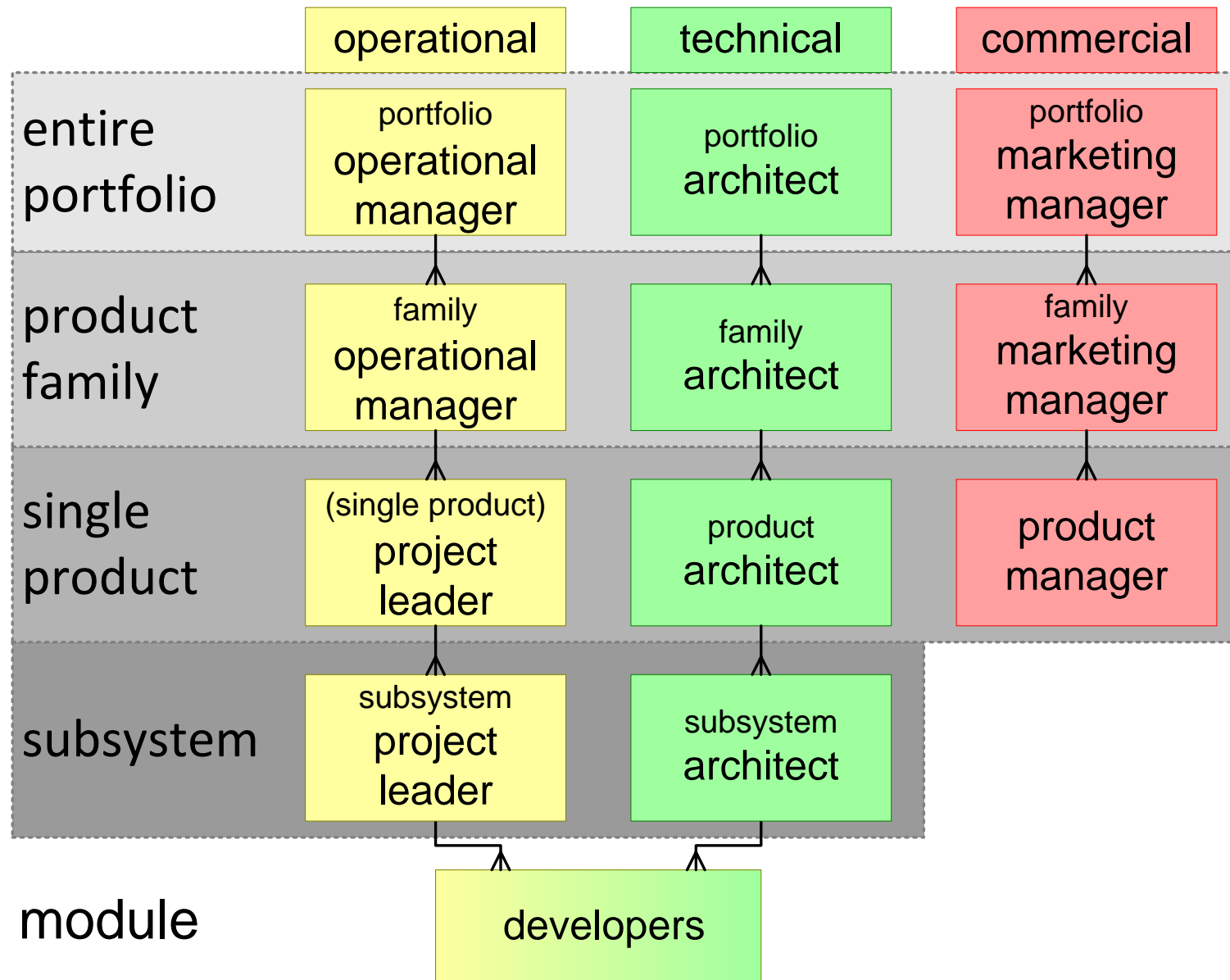
Simplified process diagram for project business



Decomposition of the Product Creation Process

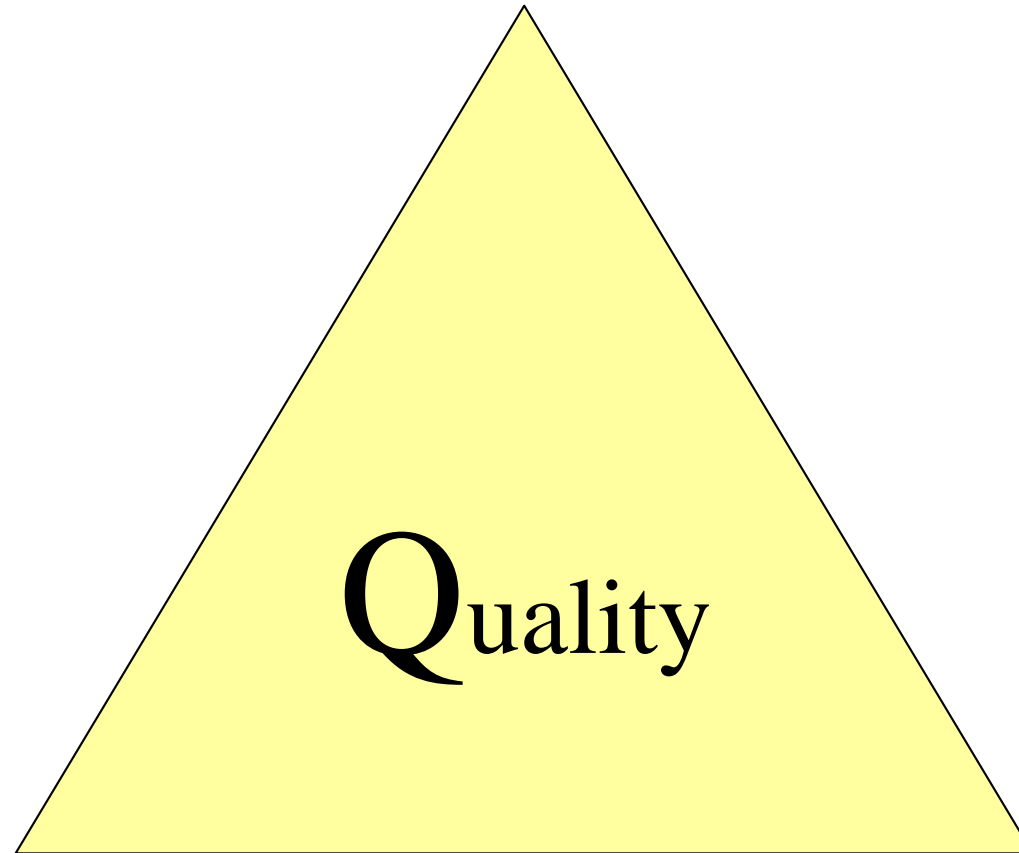


Operational Organization of the PCP



Prime Responsibilities of the Operational Leader

Specification



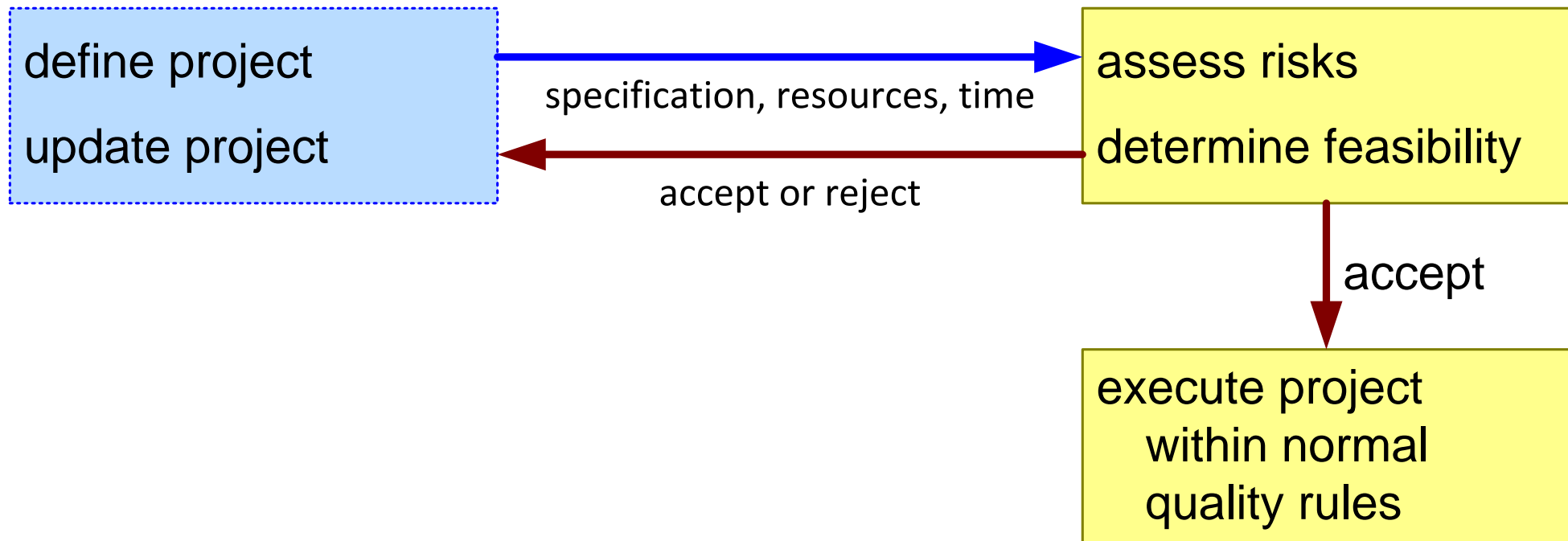
Resources

Time

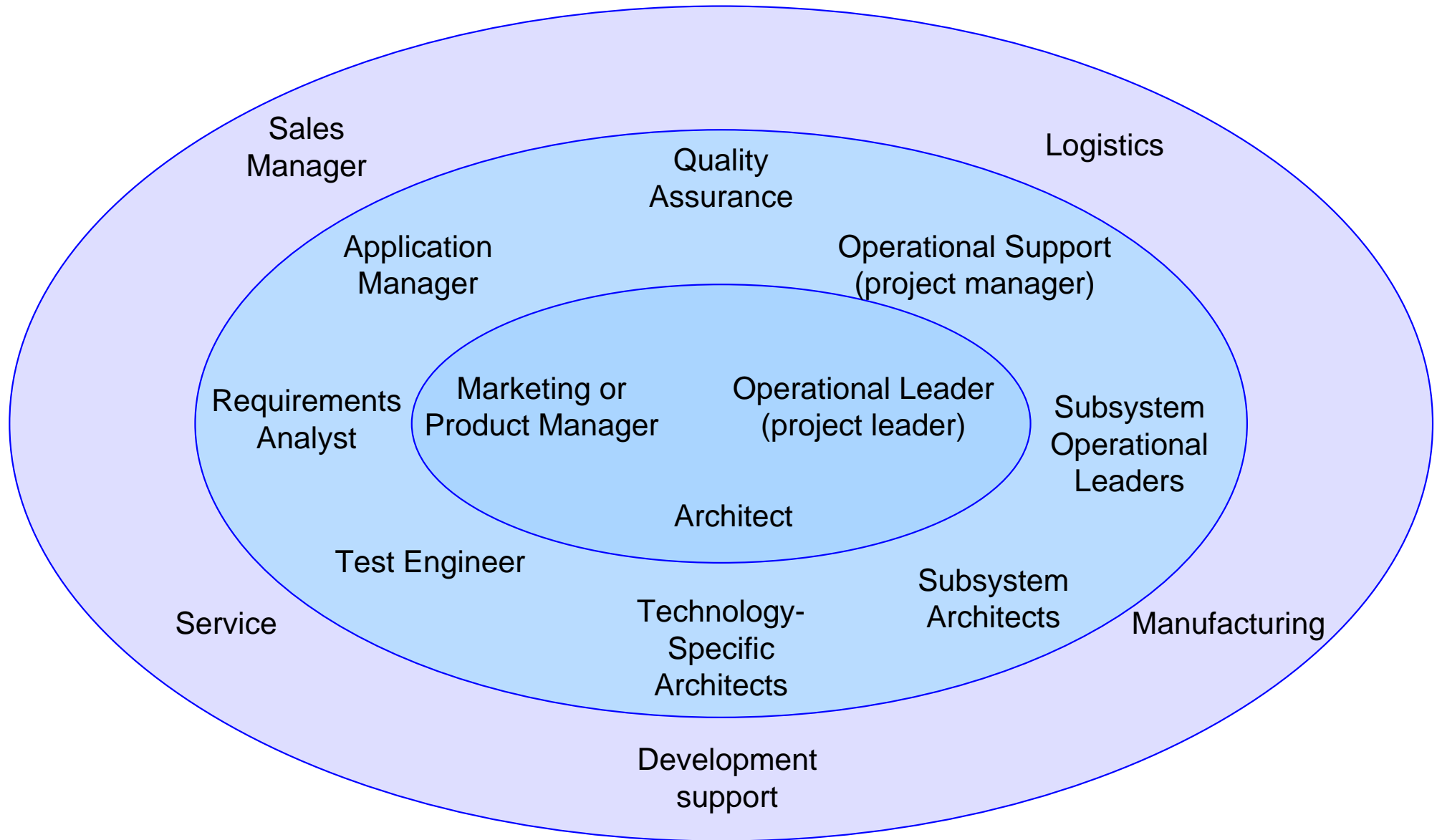
The Rules of the Operational Game

business management

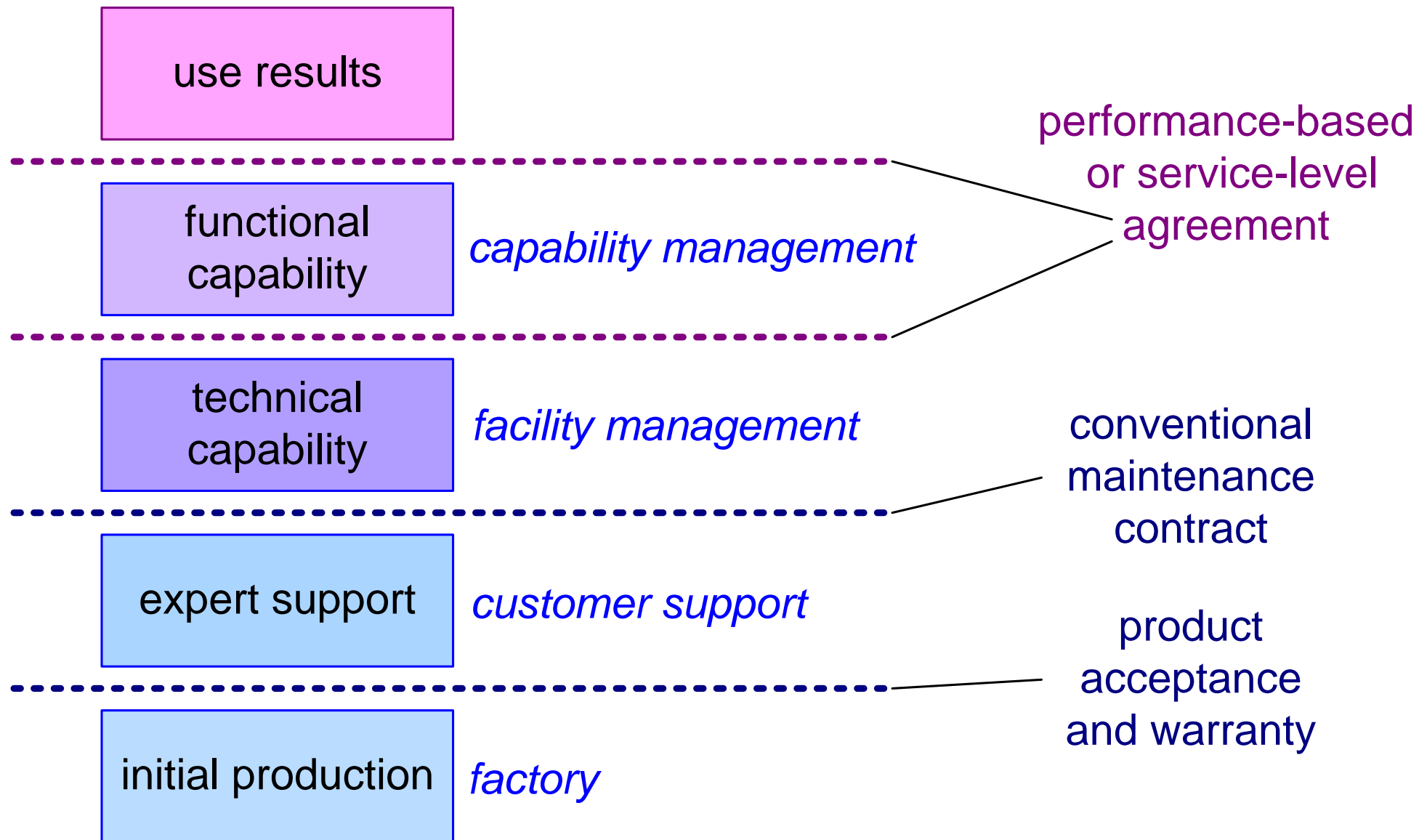
project leader



Operational Teams



What Service Level to Deliver?



Systems Engineering Management Plan (SEMP)

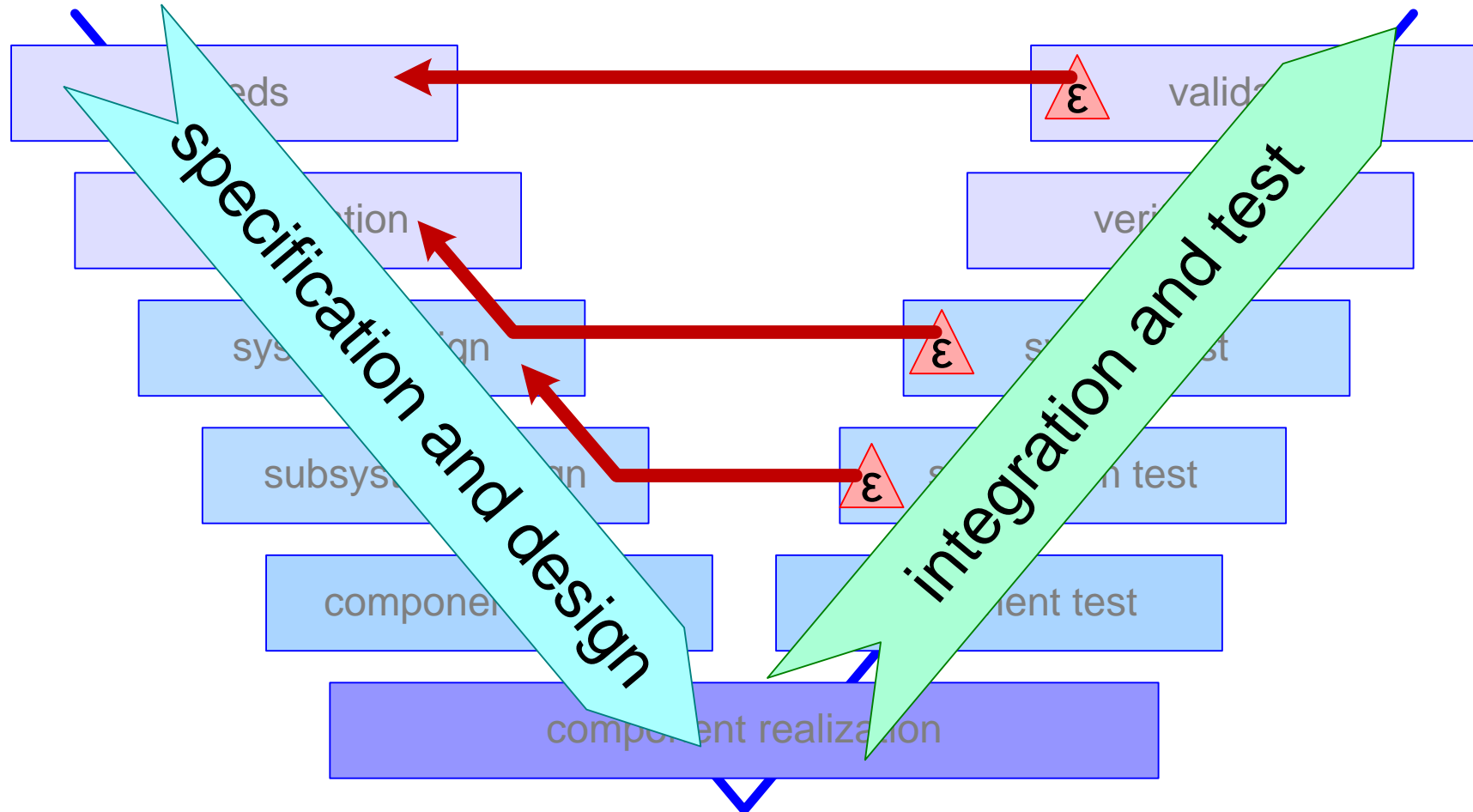
How the project will perform the systems engineering process:

- main events and activities
- roles and responsibilities
- work products
- procedures and standards

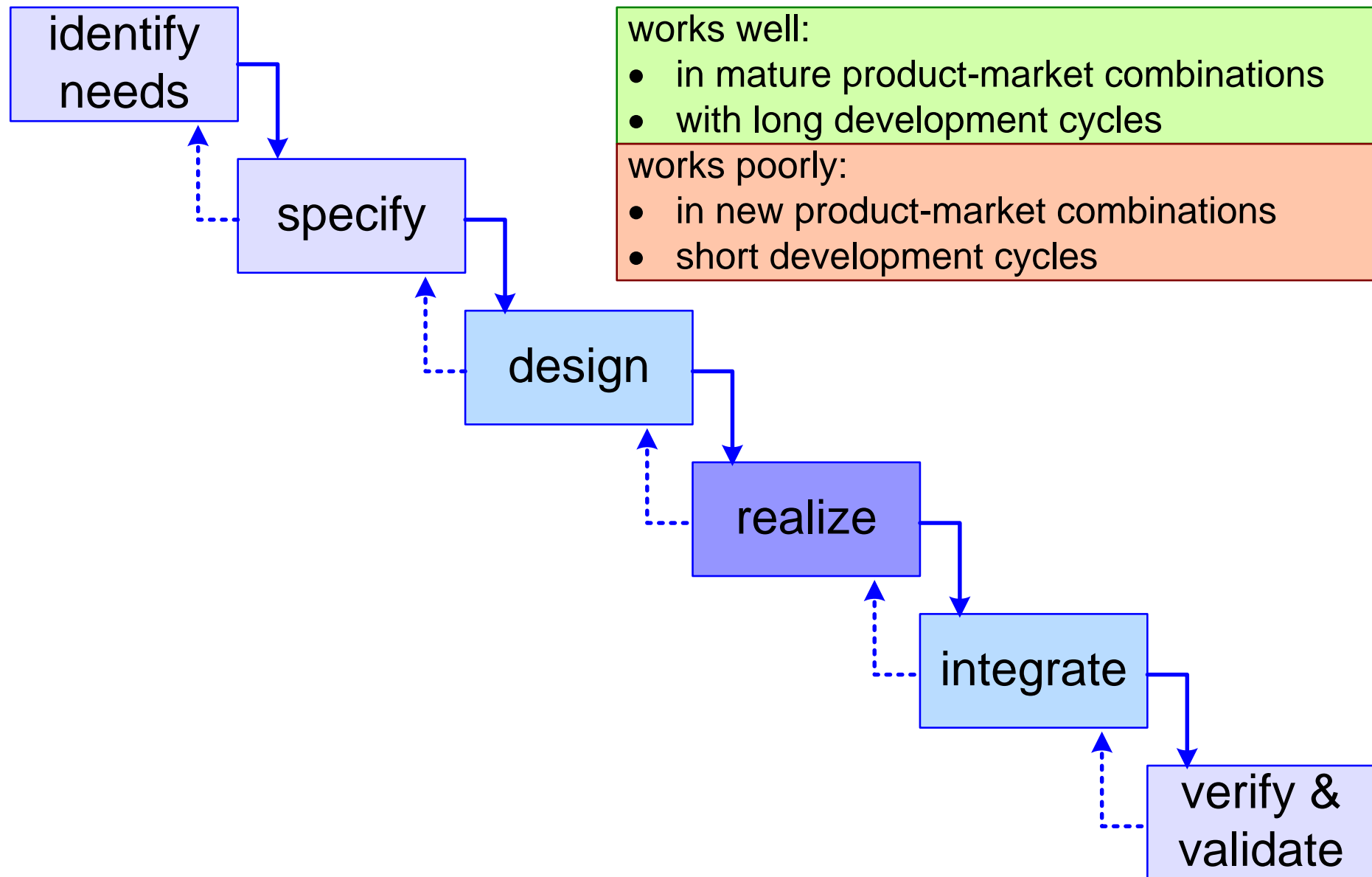
Bridge between project management and engineering (NASA 2016)

Most Problems are Found Late

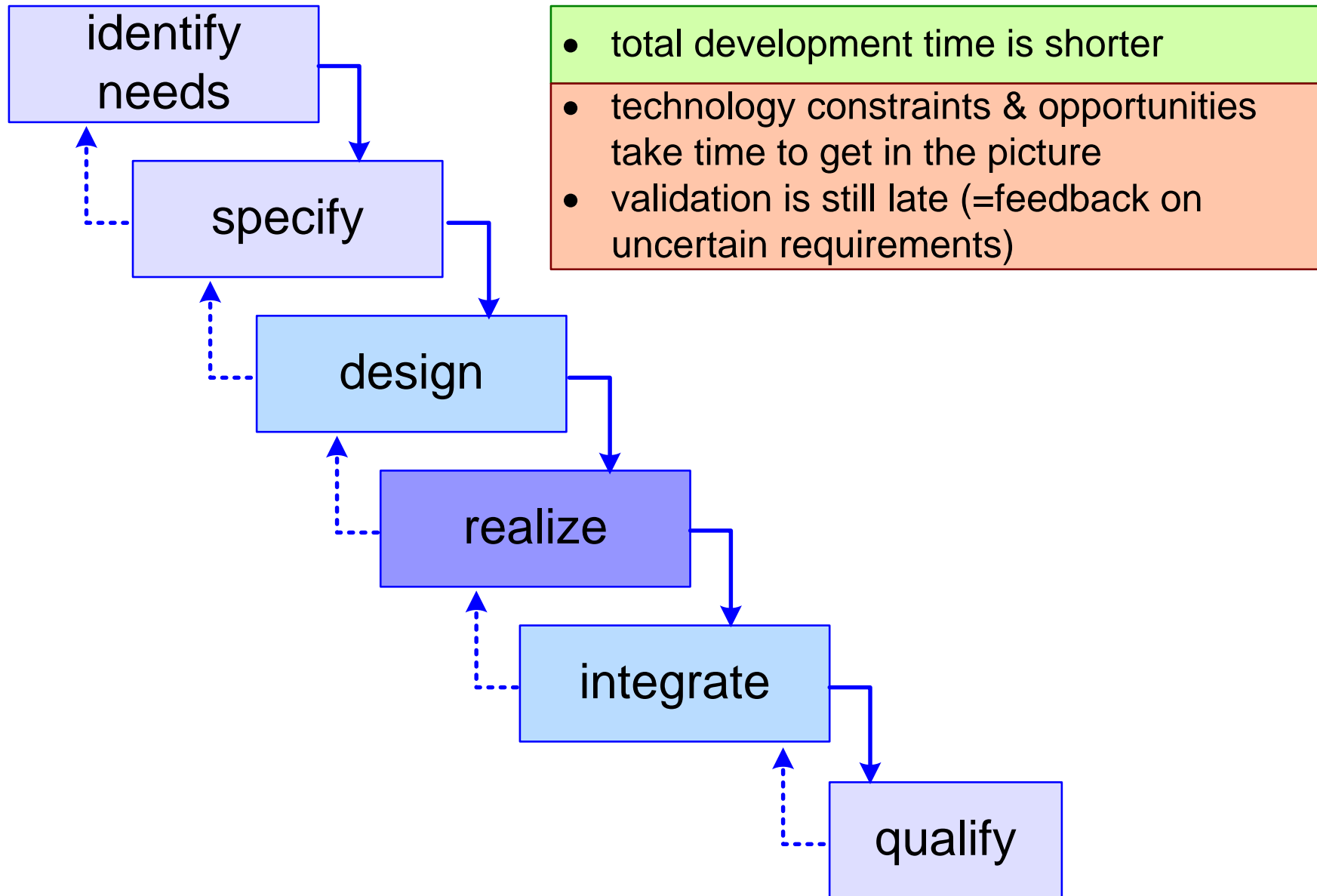
failures found during integration and test can be traced back to unknowns, unforeseens, and wrong assumptions



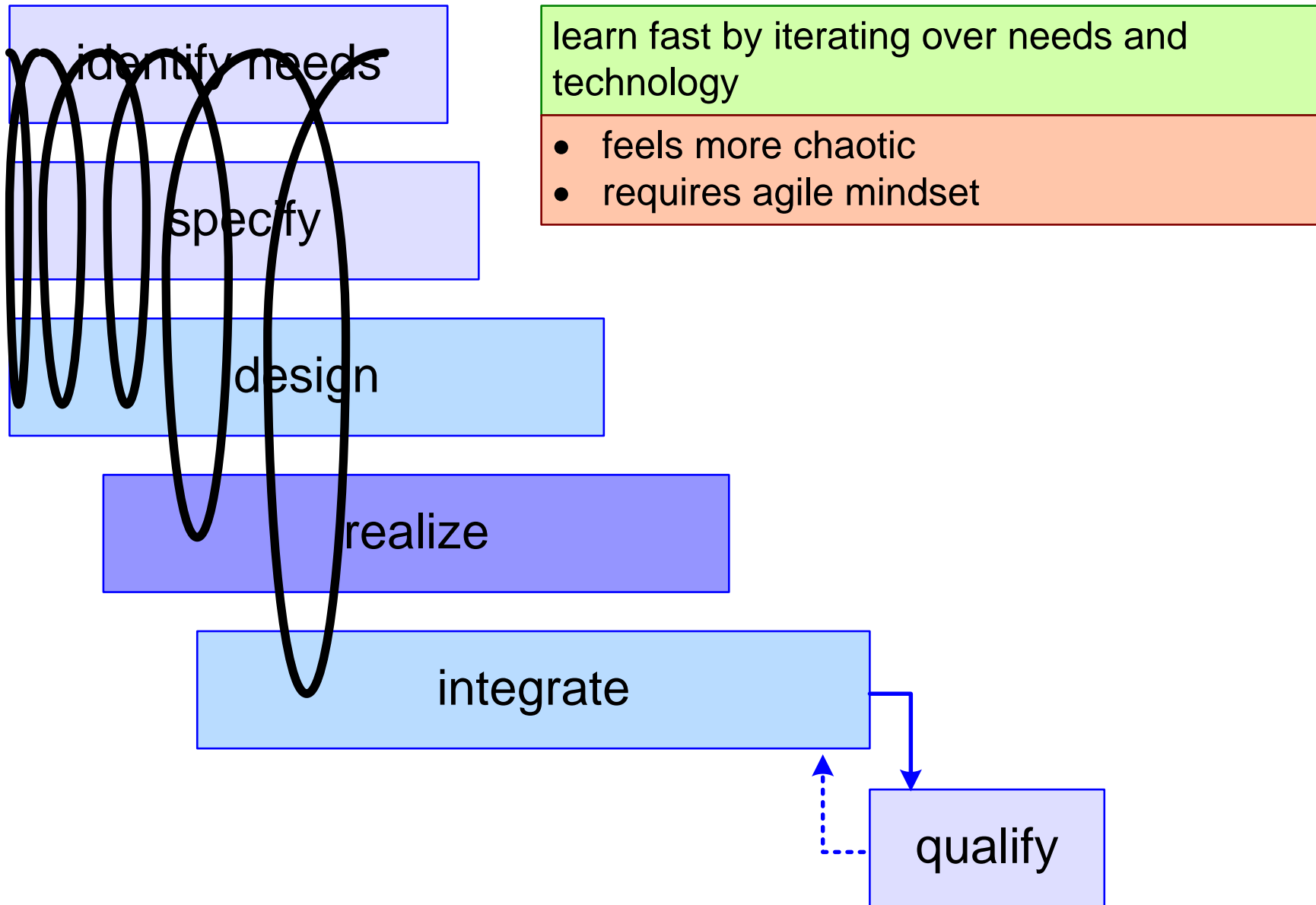
Waterfall model



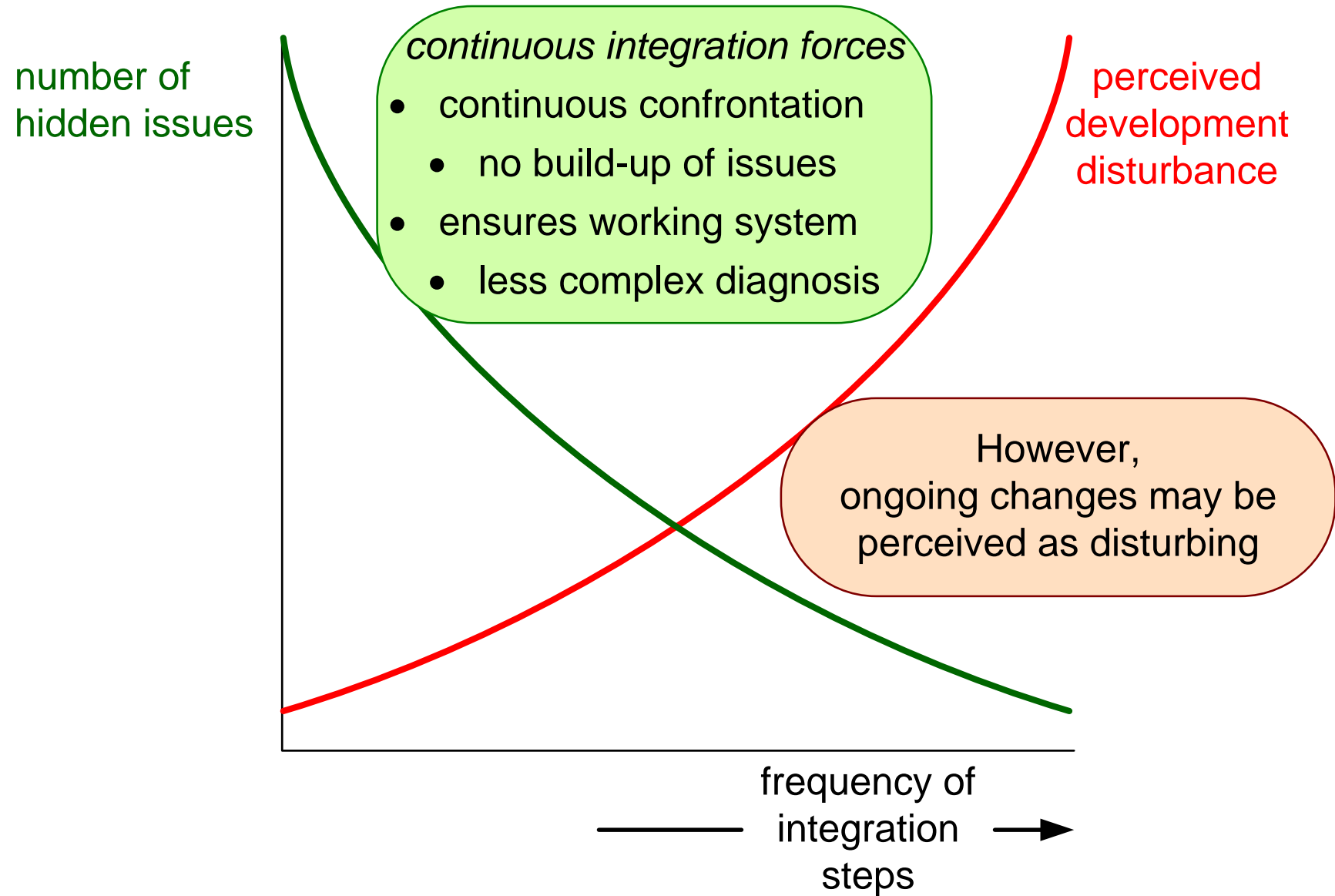
Concurrent Engineering



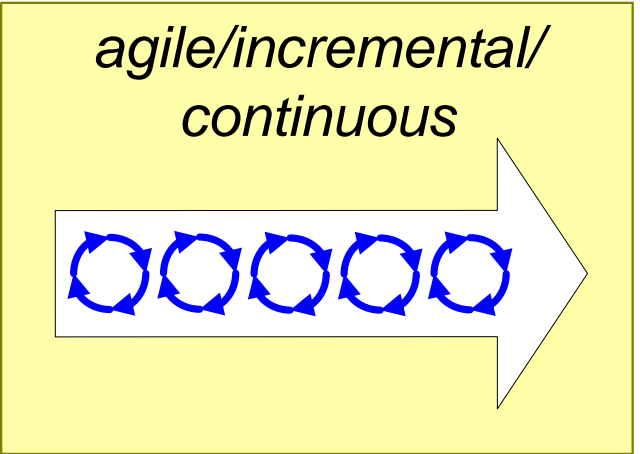
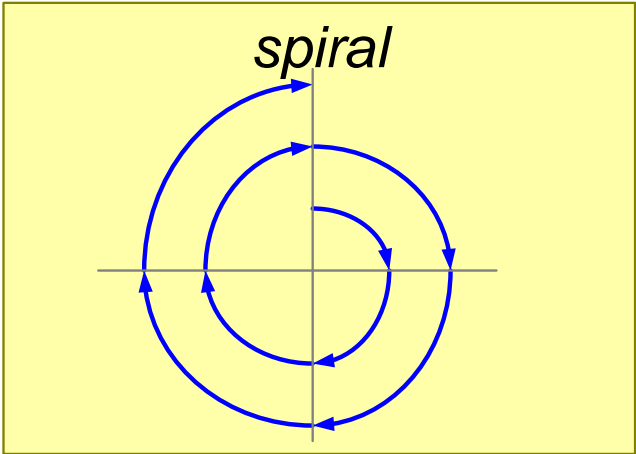
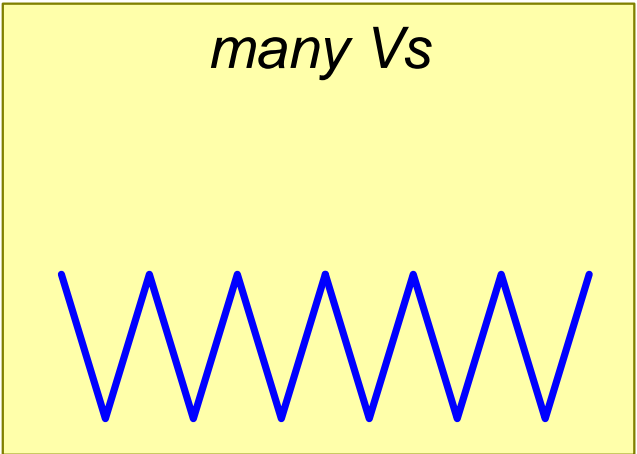
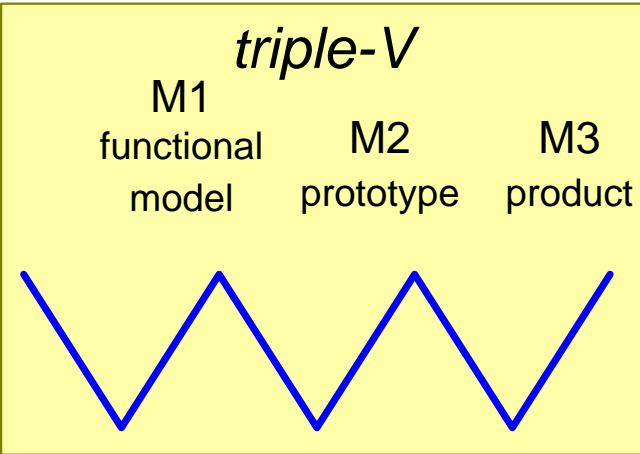
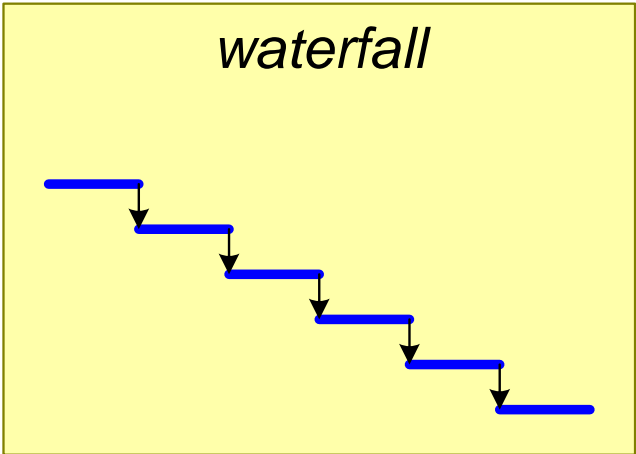
Iterative Approach



Continuous Integration



Development Processes From Waterfall to Agile



and all kinds of hybrids

Systems Engineering Fundamentals Life Cycle

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

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

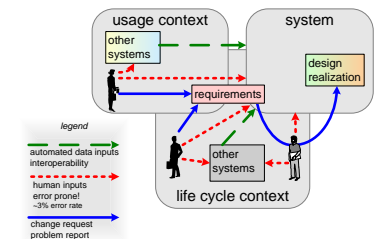
Abstract

Products and enterprises evolve over time. This presentation explores the impact of these changes on the system and on the business by making (small and simple) models of life cycle aspects.

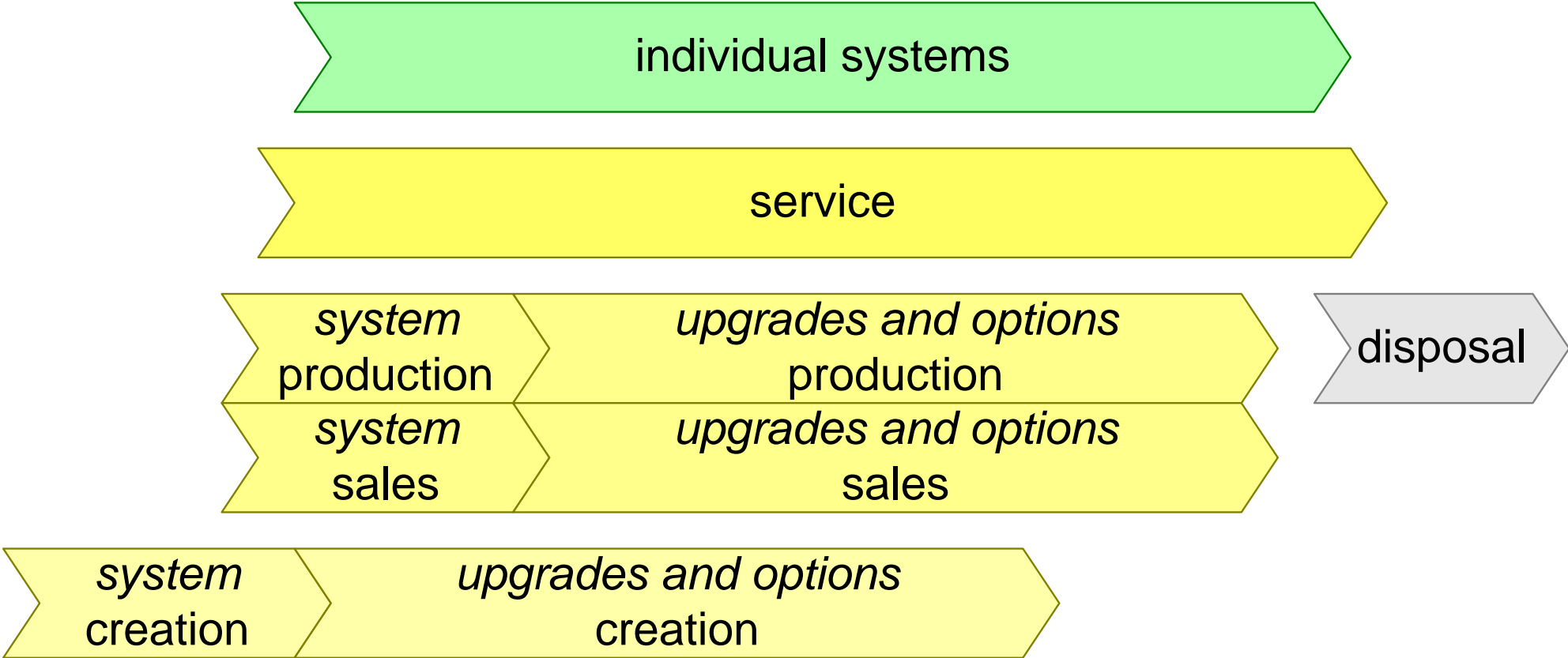
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

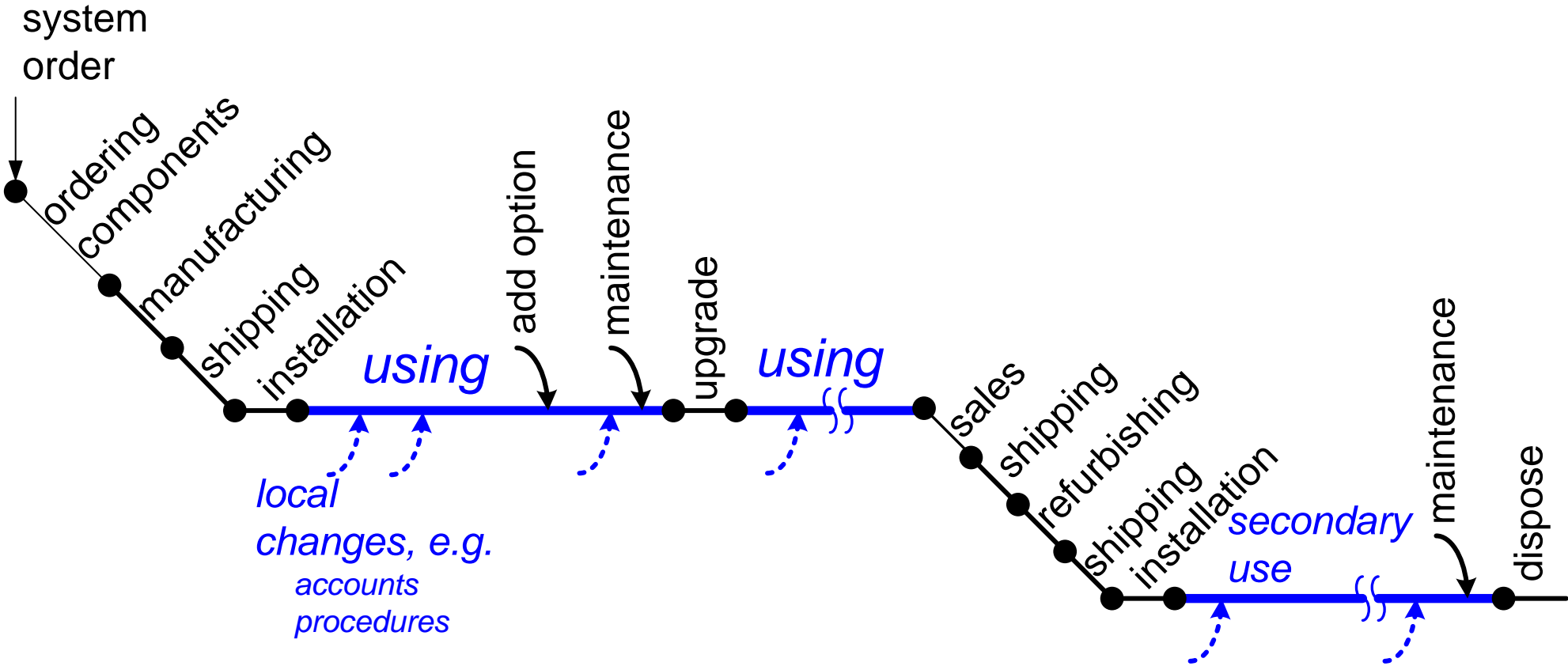
February 8, 2026
status: preliminary
draft
version: 0



Product Related Life Cycles



System Life Cycle



Approach to Life Cycle Modeling

Identify potential life cycle changes and sources		
Characterize time aspect of changes	how often how fast	
Determine required effort	amount type	
Determine impact of change on system and context	performance reliability	} see reasoning
Analyse risks	business	

Systems Engineering Fundamentals Needs Elicitation

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

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

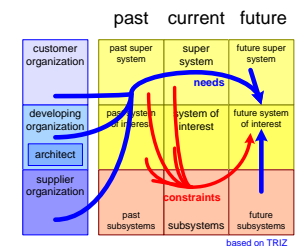
Abstract

This presentation uses the TRIZ 9 Windows diagram as framework to think about needs elicitation.

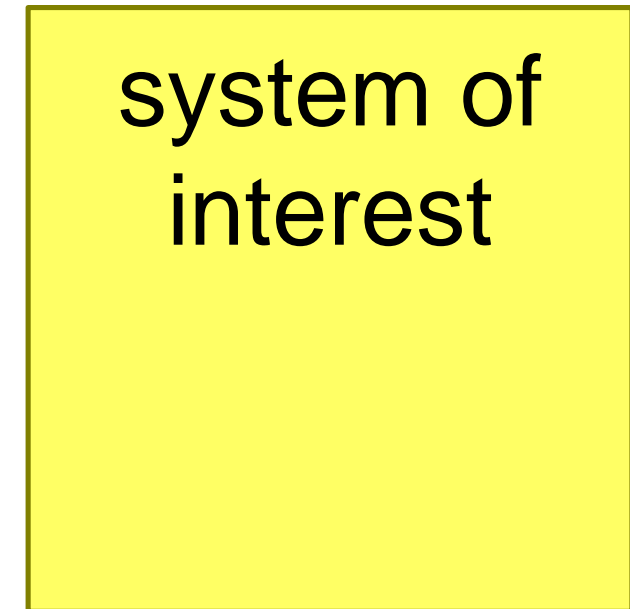
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

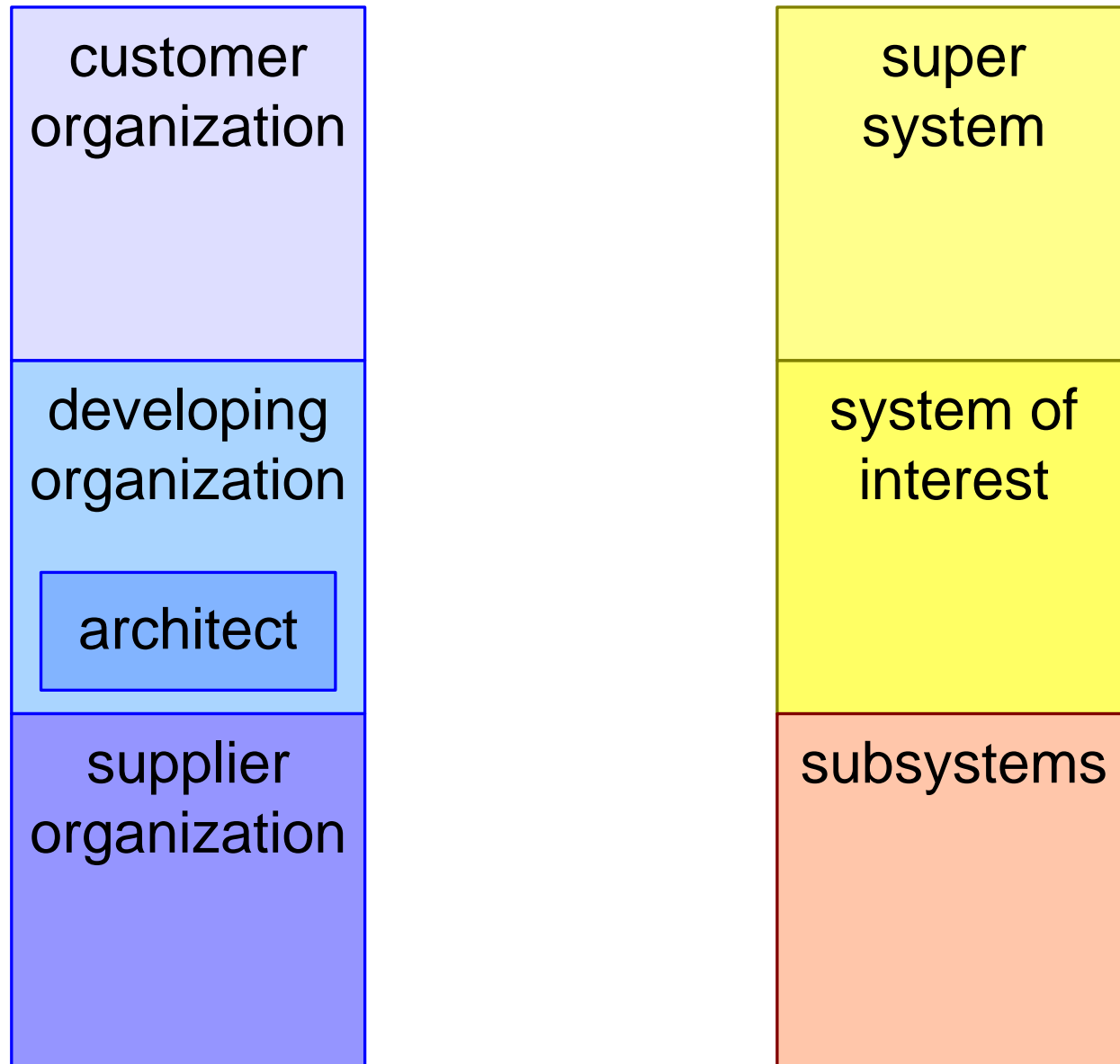
February 8, 2026
status: preliminary
draft
version: 0



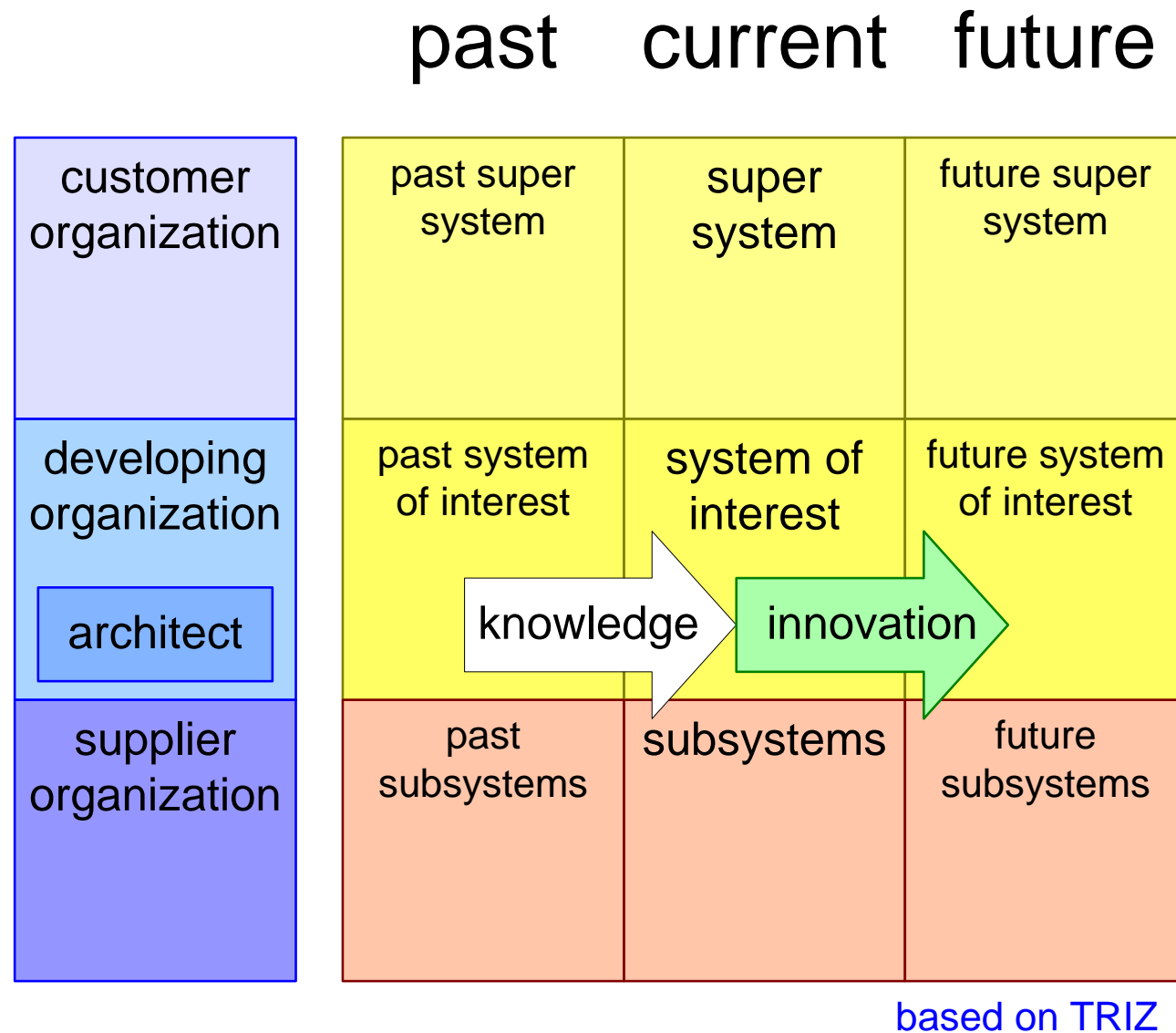
Our Primary Interest



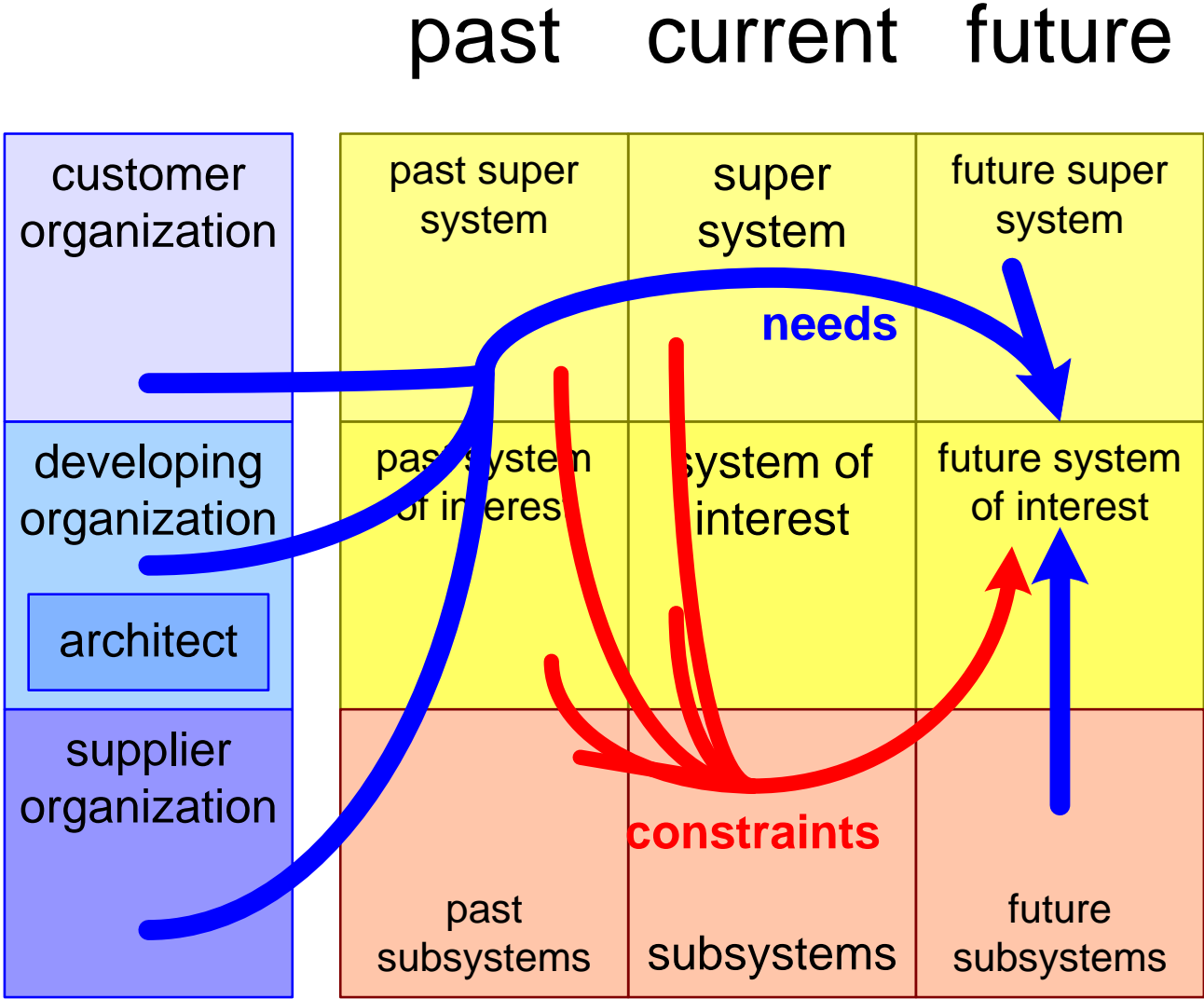
Context, Zoom-out and Zoom-in



Adding the Time Dimension



Sources of Needs and Constraints



based on TRIZ

Systems Engineering Fundamentals Requirements Management

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

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. Needs and requirements prescribe *what* rather than *how*.

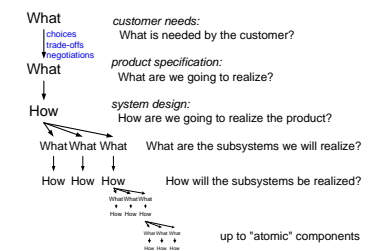
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026

status: draft

version: 0



Definition of “Requirement”

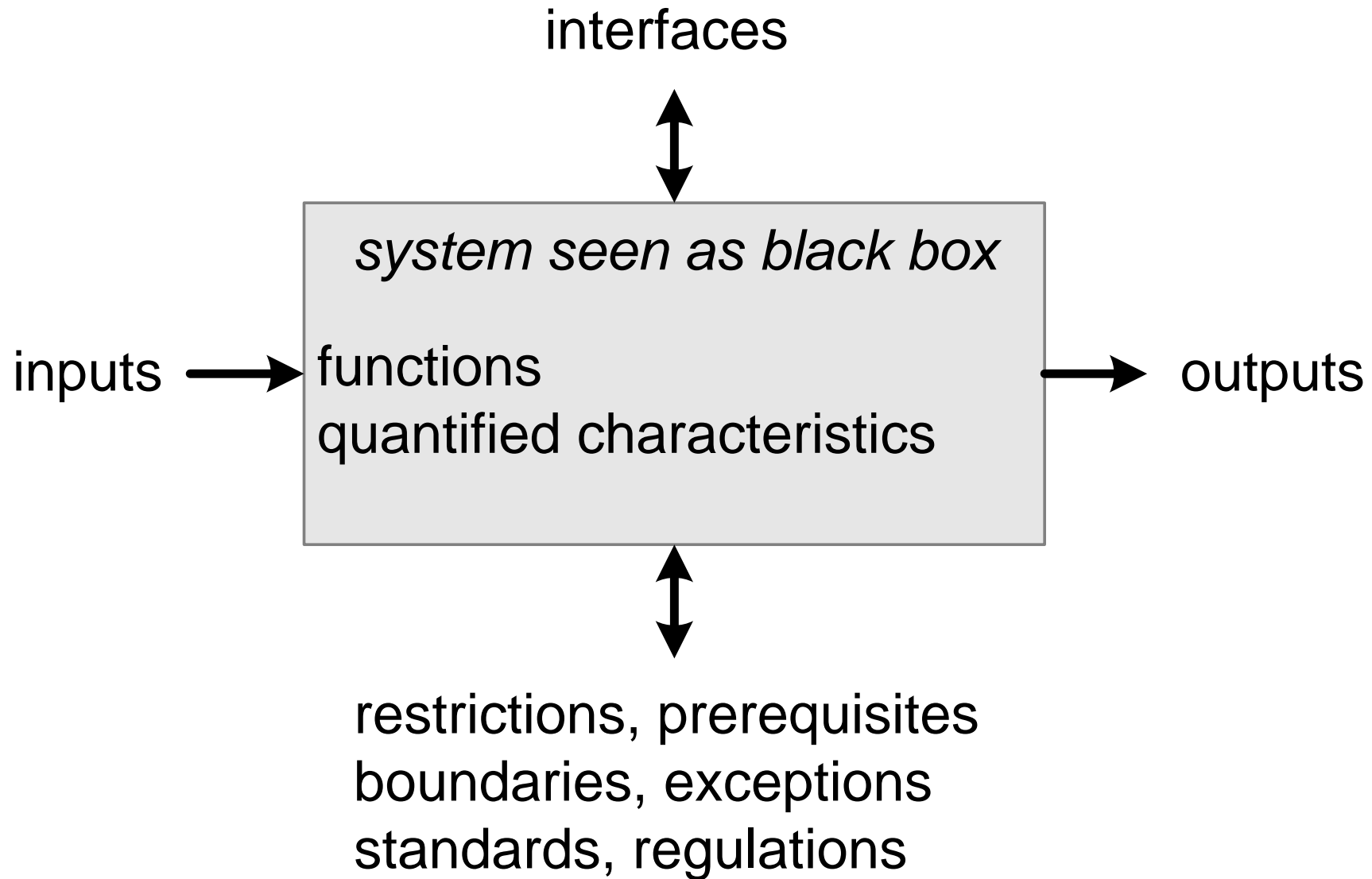
Requirements describing the needs of the customer:
Customer Needs

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***

System as a Black Box



Good Requirements are “SMART”

- **S**pecific quantified
- **M**easurable verifiable
- **A**chievable (Attainable, Action oriented, Acceptable, Agreed-upon, Accountable)
- **R**ealistic (Relevant, Result-Oriented)
- **T**ime-bounded (Timely , Tangible, Traceable)

Specific Requirements have Specific Circumstances

Typical Use Case

- What is the user typically doing with the system in the system context
- Quantify the operation and context in this typical case

Other Use Cases

- Operational variants
- Boundary behavior
- Exceptional cases

these use cases >> SysML use cases

Concept Selection, Set Based Design and Late Decision Making

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

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

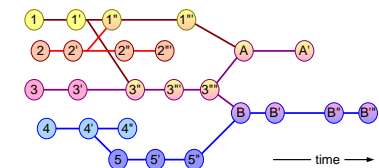
Abstract

We discuss a systems design approach where several design options are maintained concurrently. In LEAN Product Development this is called set-based design. Conventional systems engineering also promotes the concurrent evaluation of multiple concepts, the so-called concept selection. Finally, LEAN product development advocates to keep options open as long as feasible; the so-called late decision making.

Distribution

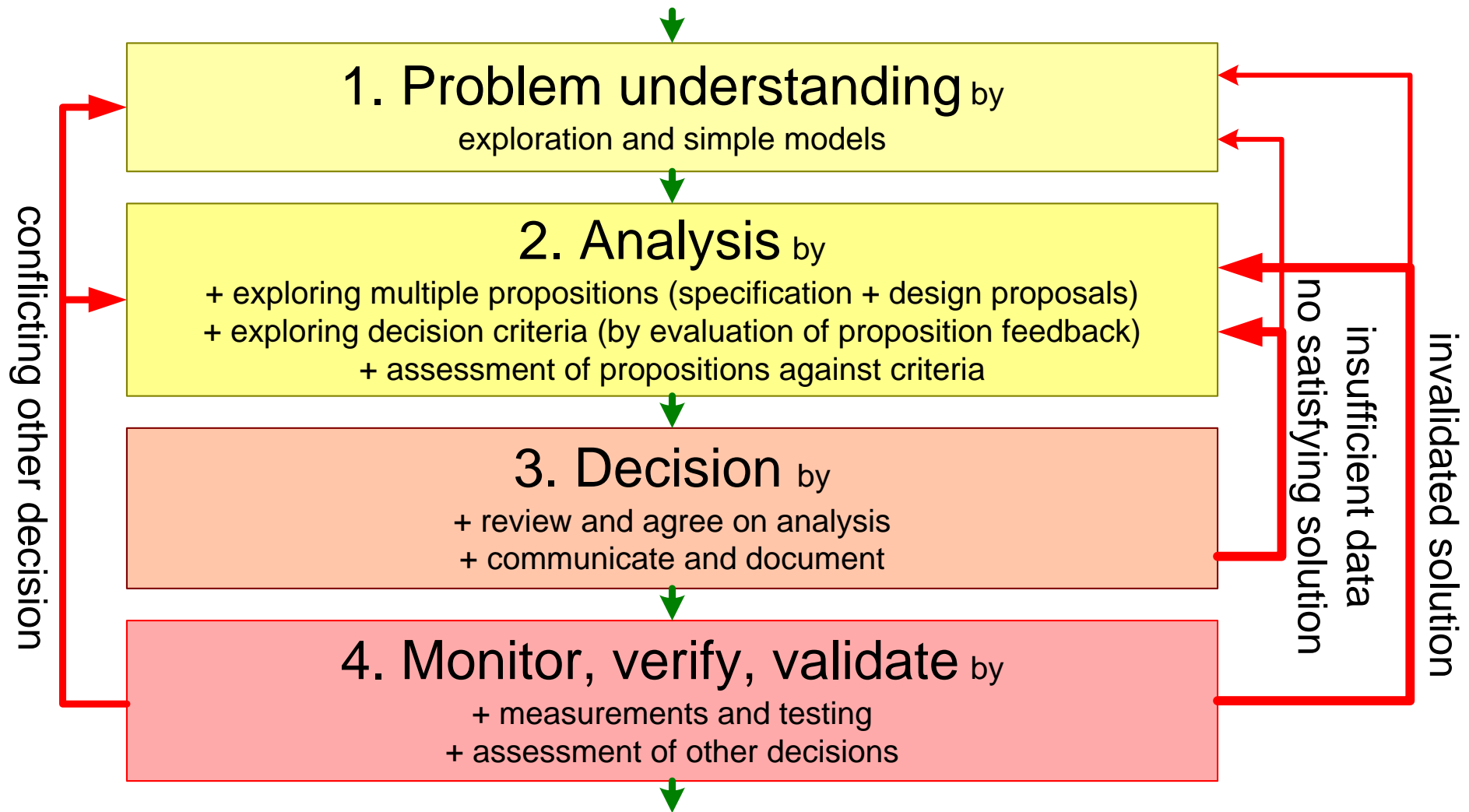
This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026
status: planned
version: 0



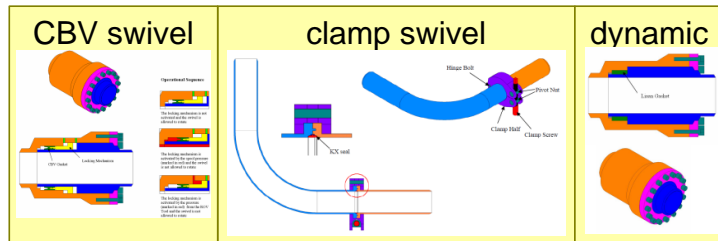
Problem Solving Approach

vague problem statement



Examples of Pugh Matrix Application

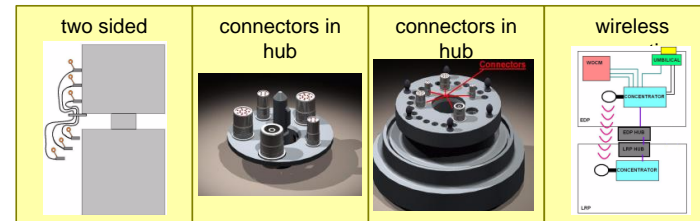
Swivel concept selection



evaluation criteria	weight	CBV		clamp		dynamic		
Maturity	10	5	50	2	20	2	50	
Development level								
Cost	20	4	80	2	40	5	100	
Hardware cost								
Development cost		5	100	2	40	2	40	
Design robustness	25							
Design life								
swivel cycles		5	125	3	75	3	75	
pressure cycles		5	125	4	100	5	125	
Pressure range								
internal		4	100	4	100	4	100	
external		2	50	5	125	2	50	
Temperature range		4	100	4	100	4	100	
Installation	20							
Initial installatio/retrieval			2	40	3	60	4	80
Connection/disconnection		2	40	4	80	5	100	
Operation	25							
Swivel resistance			1	25	4	100	5	125
Spool Length Short			1	25	4	100	5	125
Spool Length Long			3	75	5	125	5	125
Hub loads			2	50	4	100	5	125
Σ points			985		1165		1290	

from master paper Halvard Bjørnsen, 2009

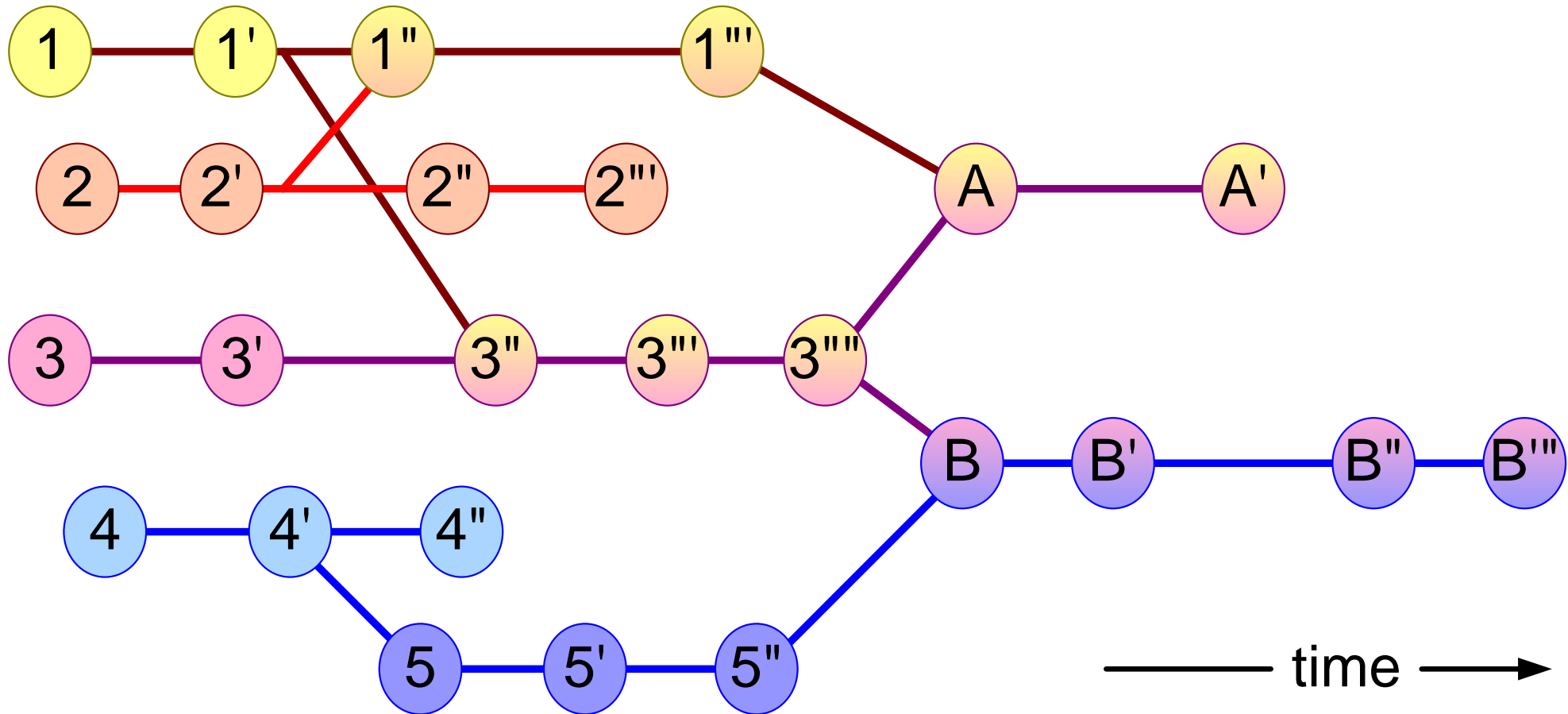
EDP-LRP connection



Evaluation Criteria	Score	Concepts			
		1	2	3	4
Time to connect					
Need for ROV		-	+	+	+
Design		-	+	+	+
Robustness					
Connector design		-	S	S	+
Number of parts		-	-	+	+
Handle roll-off		+	-	S	+
Influence other		+	S	-	S
Redundancy					
Design		+	-	-	S
Interchangeability		+	-	-	-
Cost					
HW cost		-	-	-	-
Manufacturing cost		S	S	-	S
Engineering cost		+	-	S	-
Service cost		-	+	+	+
Maturity		-	-	S	+
Σ -		7	7	5	3
Σ S		1	3	4	3
Σ +		5	3	4	7
Pos.		3	4	2	1

from master paper Dag Jostein Klever, 2009

Evolution of Design Options



Evolving multiple concepts increases insight and understanding
(LEAN product development: set-based design, SE: Pugh matrix)

Articulation of criteria sharpens evaluation

The discussion about the Pugh matrix is more valuable than final
bottomline summation

Delaying decisions may help to keep options (Lean Product
Development: late decision making, finance: real options)

Systems Engineering Fundamentals Architecture and Design

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

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

This presentation explains the fundamentals behind Architecture and Design, such as conceptual and functional design, partitioning, interfaces, and allocation.

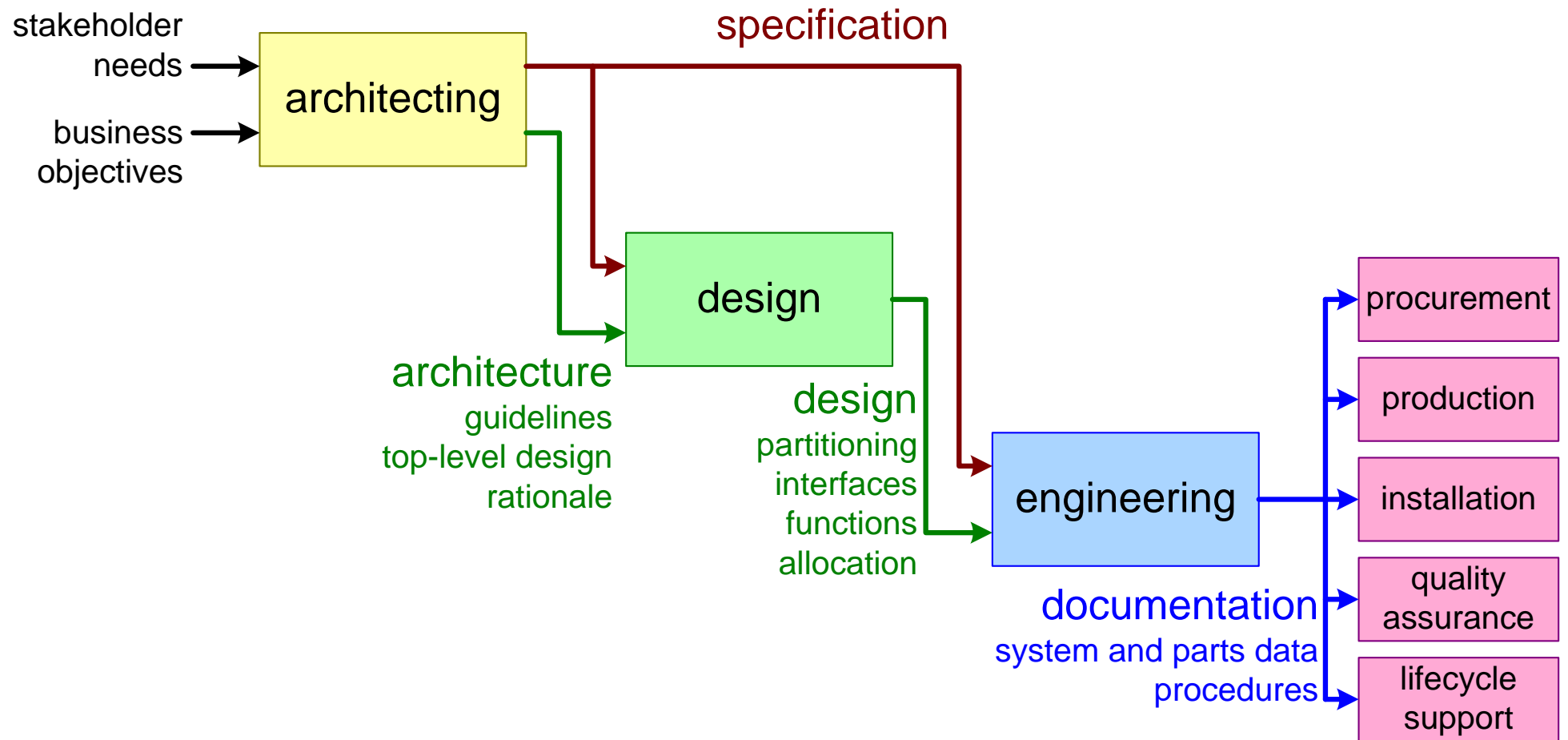
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026
status: planned
version: 0

logo
TBD

Architecture and Design



Visualizing Dynamic Behavior

by *Gerrit Muller* [TNO-ESI, University of South-Eastern Norway]

e-mail: gaudisite@gmail.com

www.gaudisite.nl

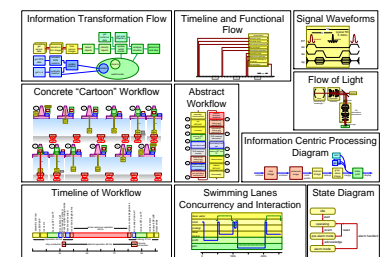
Abstract

Dynamic behavior manifests itself in many ways. Architects need multiple complementary visualizations to capture dynamic behavior effectively. Examples are capturing information, material, or energy flow, state, time, interaction, or communication.

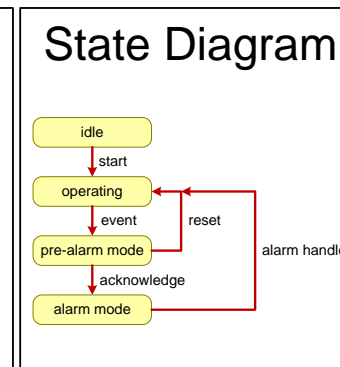
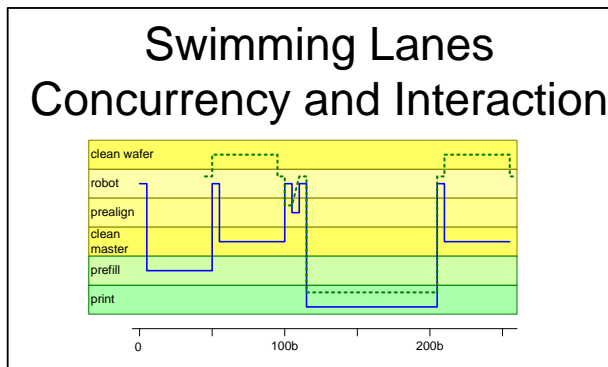
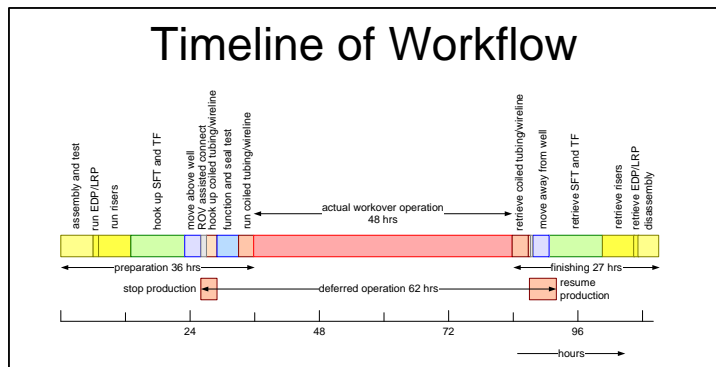
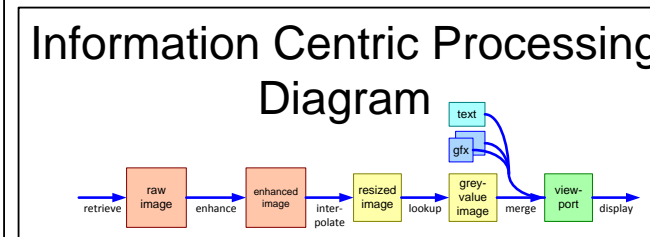
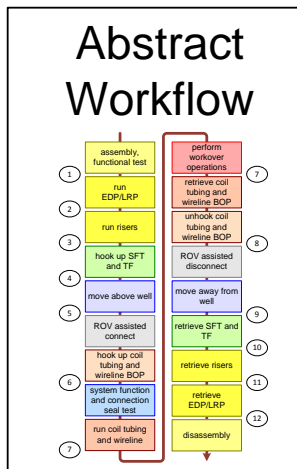
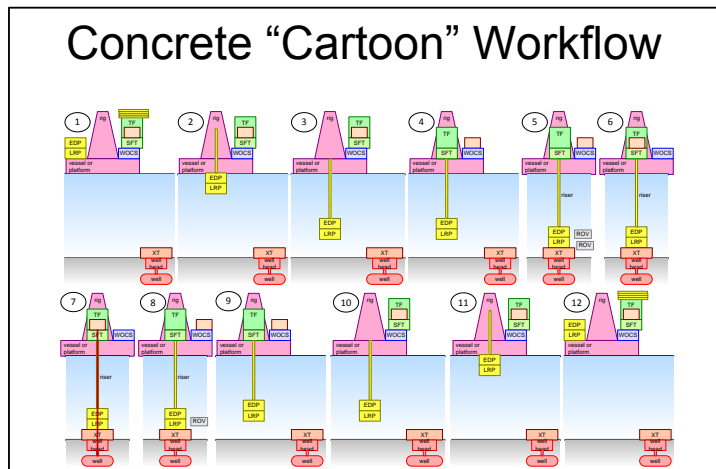
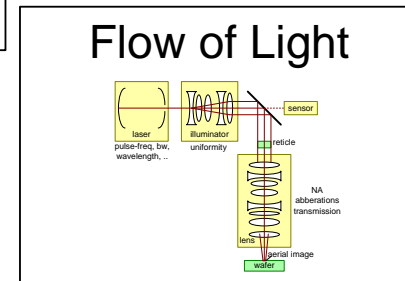
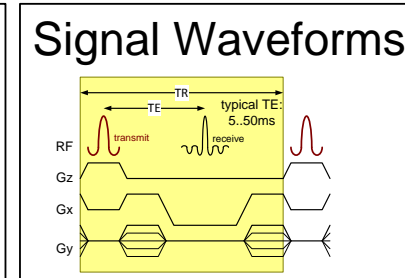
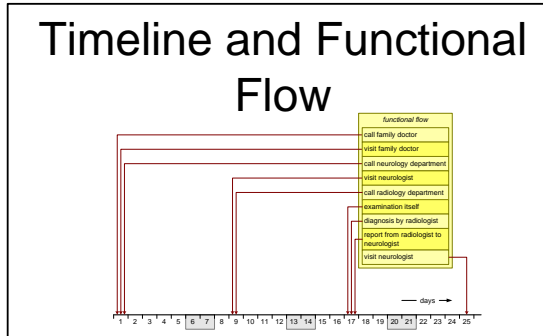
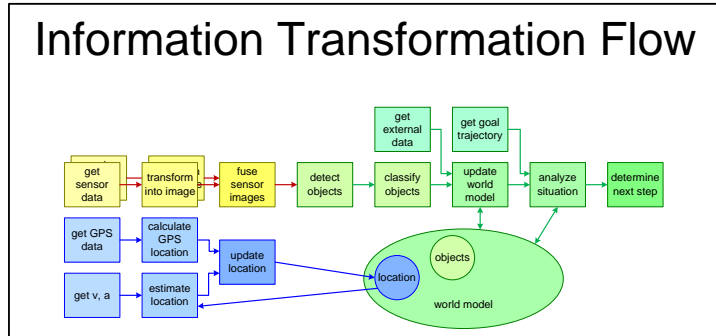
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

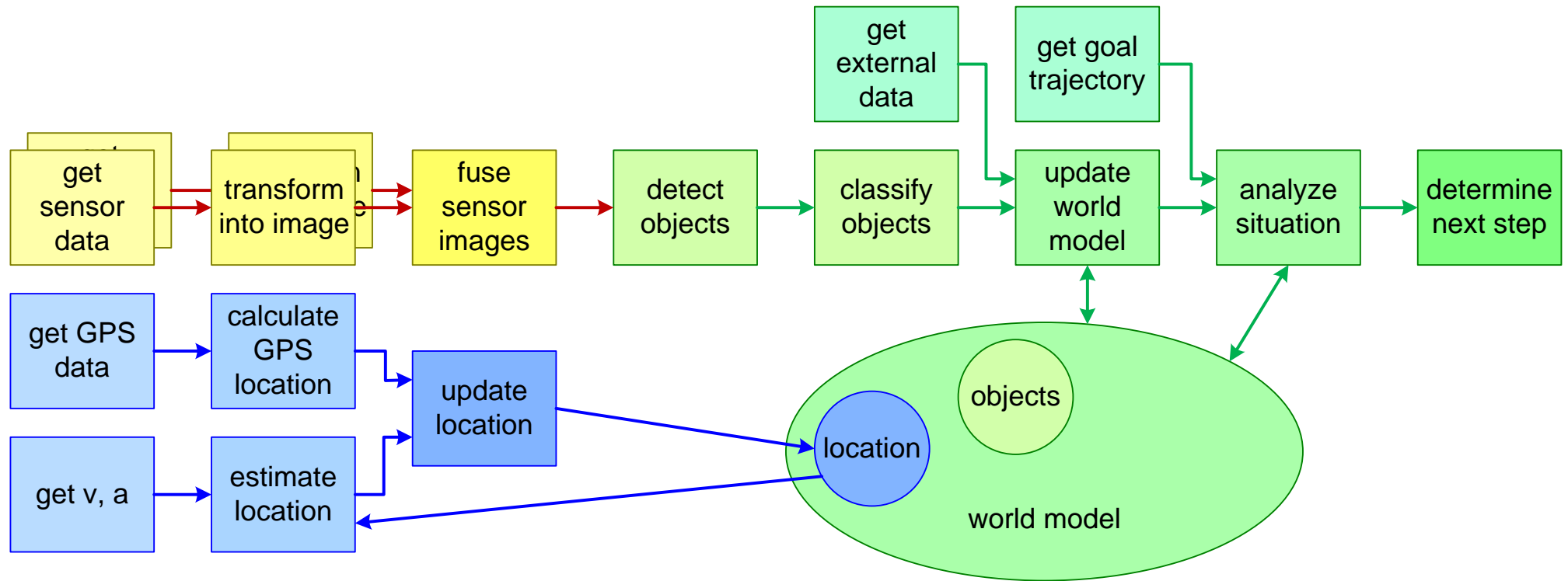
February 8, 2026
status: preliminary
draft
version: 0



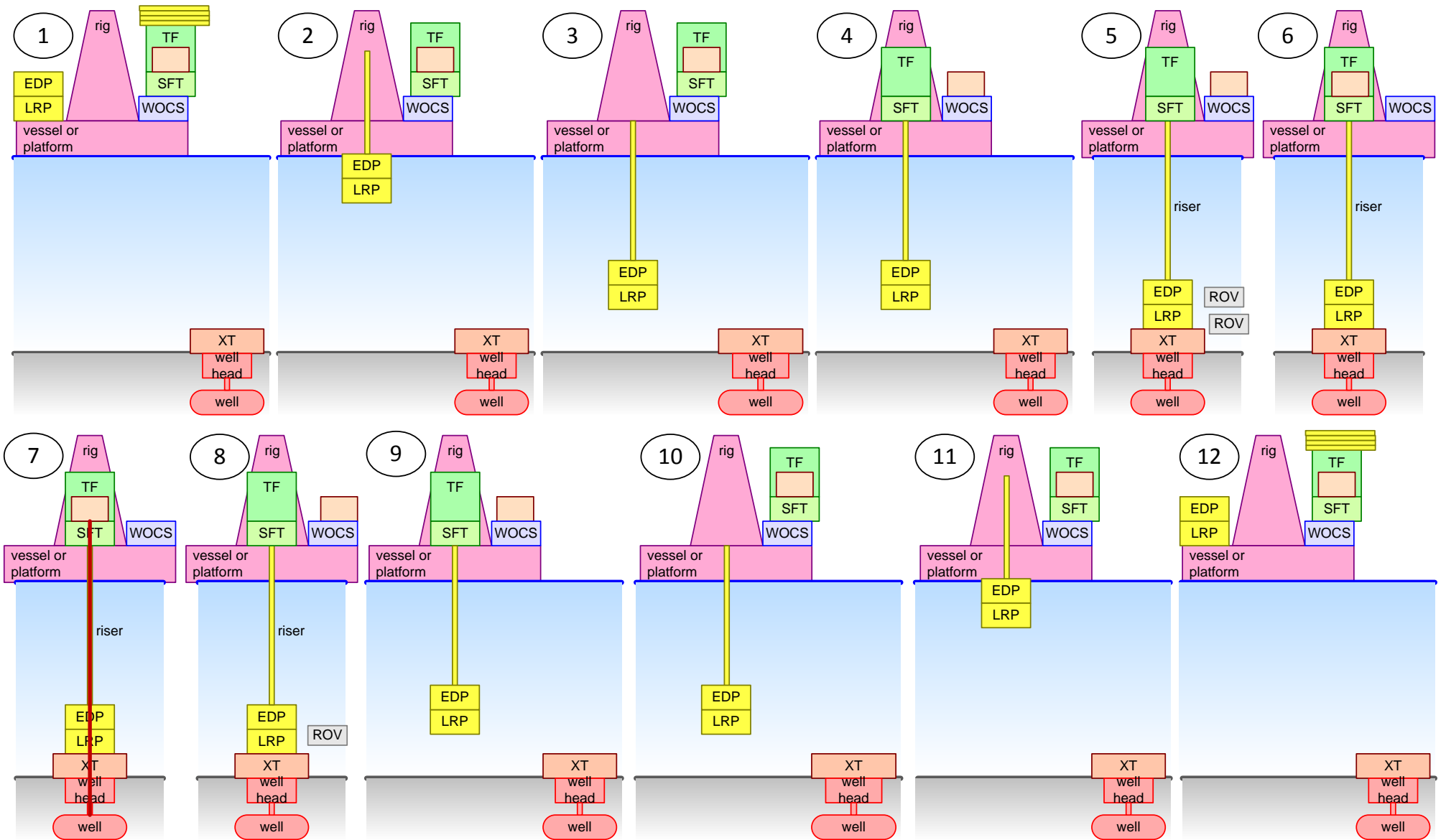
Overview of Visualizations of Dynamic Behavior



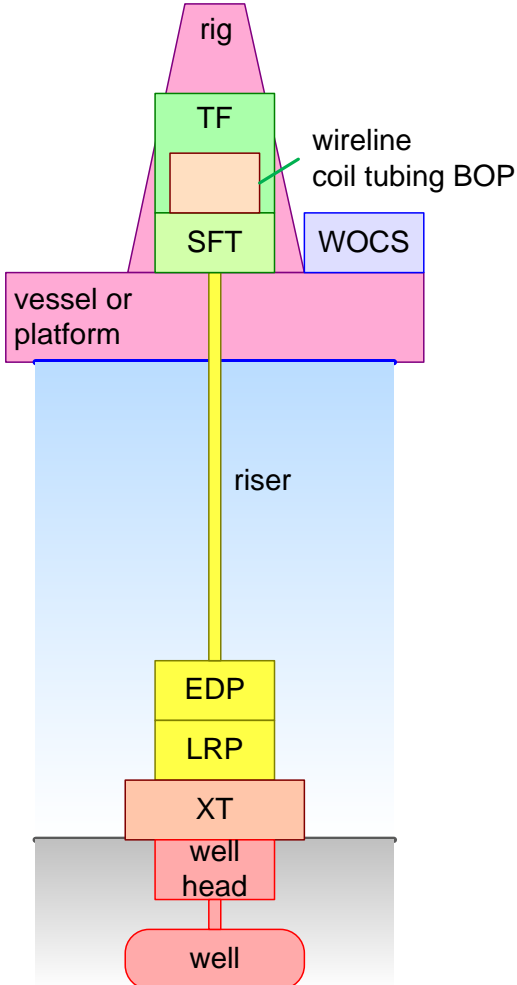
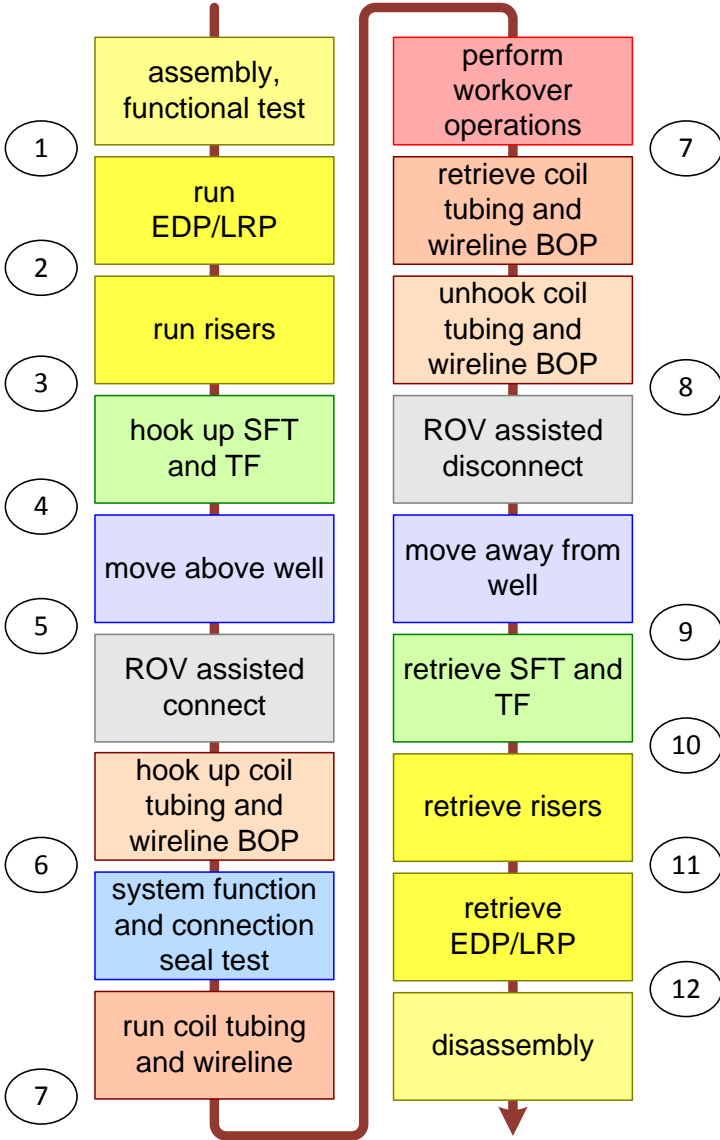
Example Functional Model of Information Flow



"Cartoon" Workflow



Workflow as Functional Model



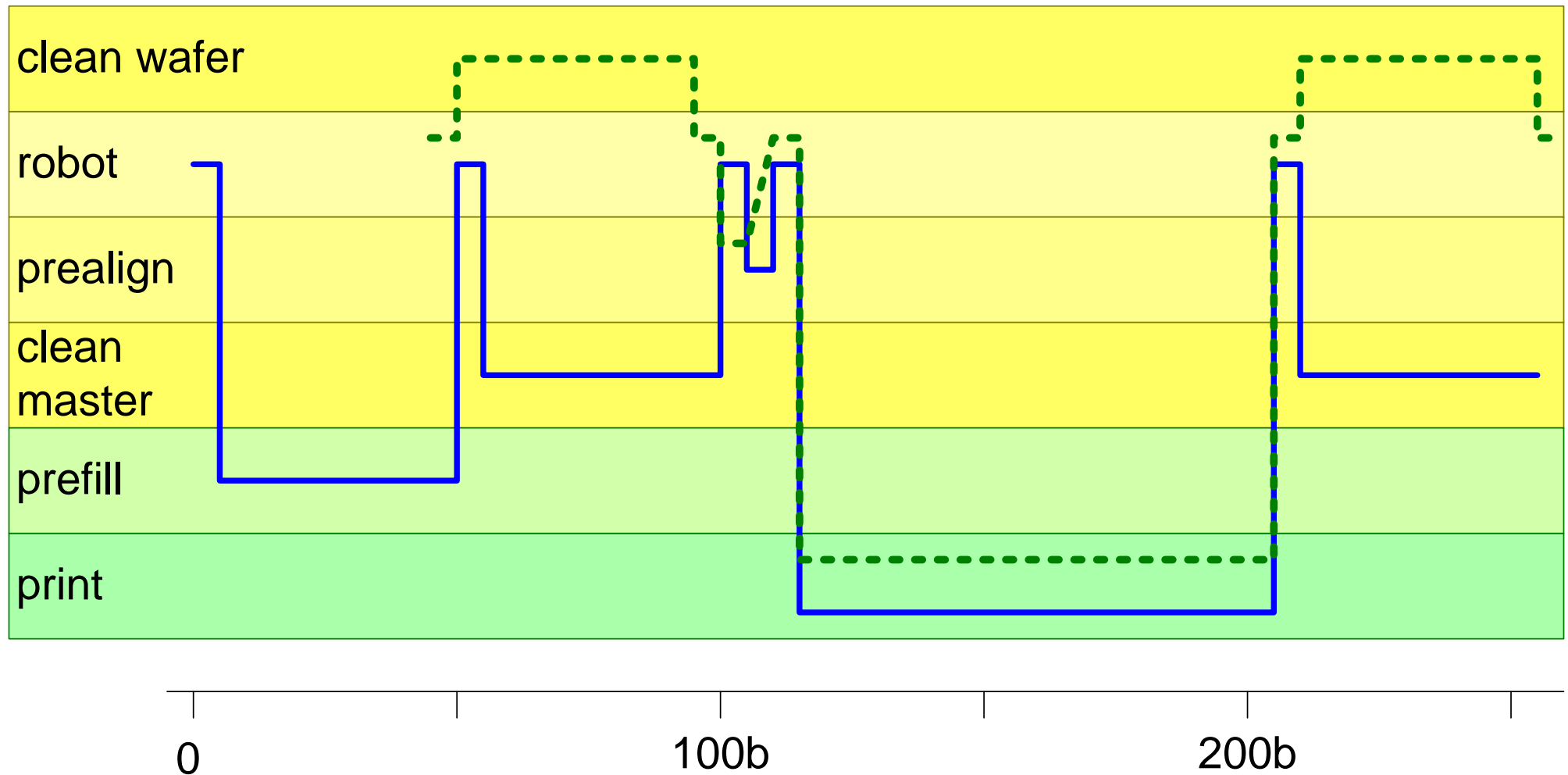
Workflow as Timeline

assumptions:

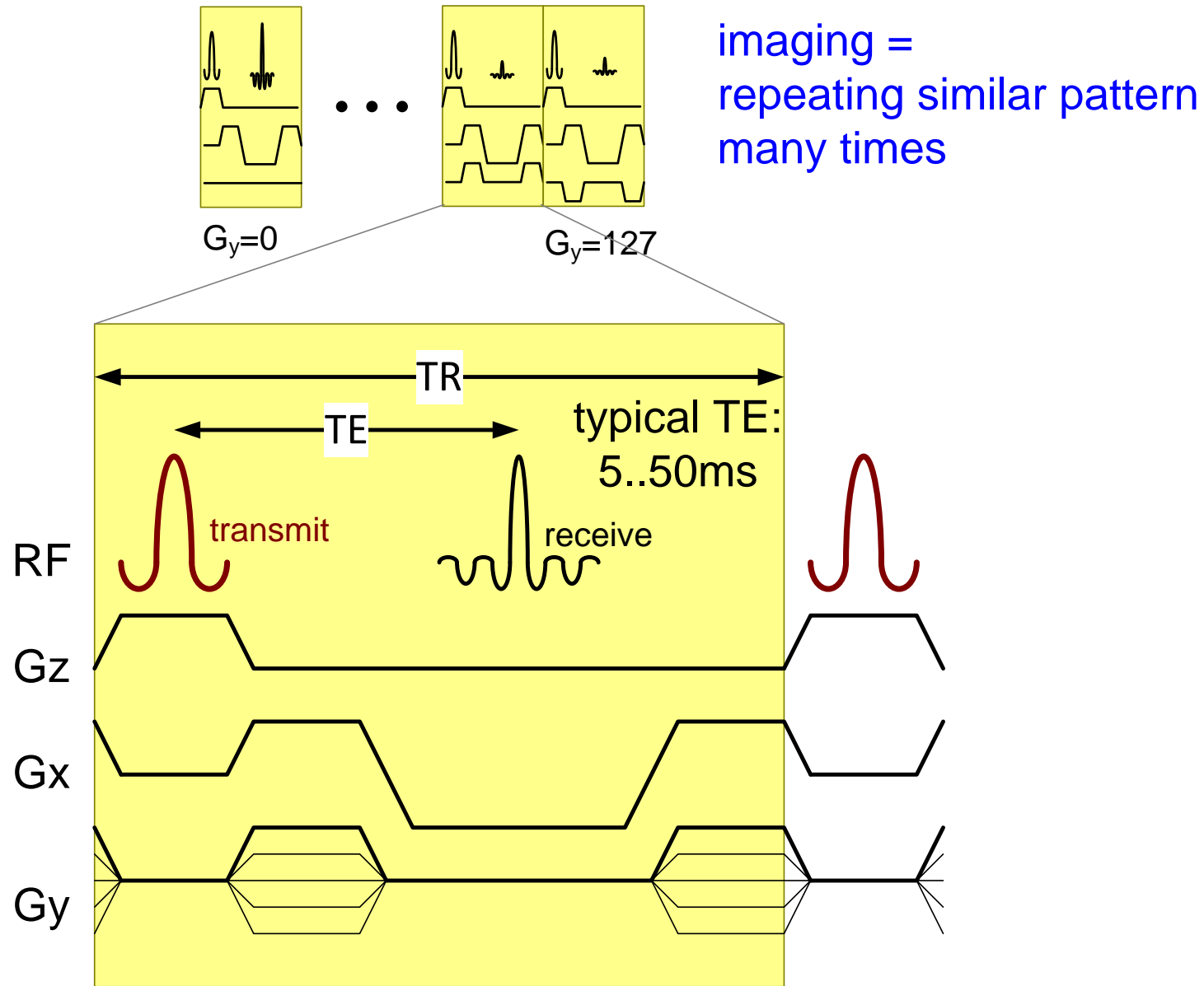
running and retrieving risers: 50m/hr
 running and retrieving coiled tubing/wireline: 100m/hr
 depth: 300m



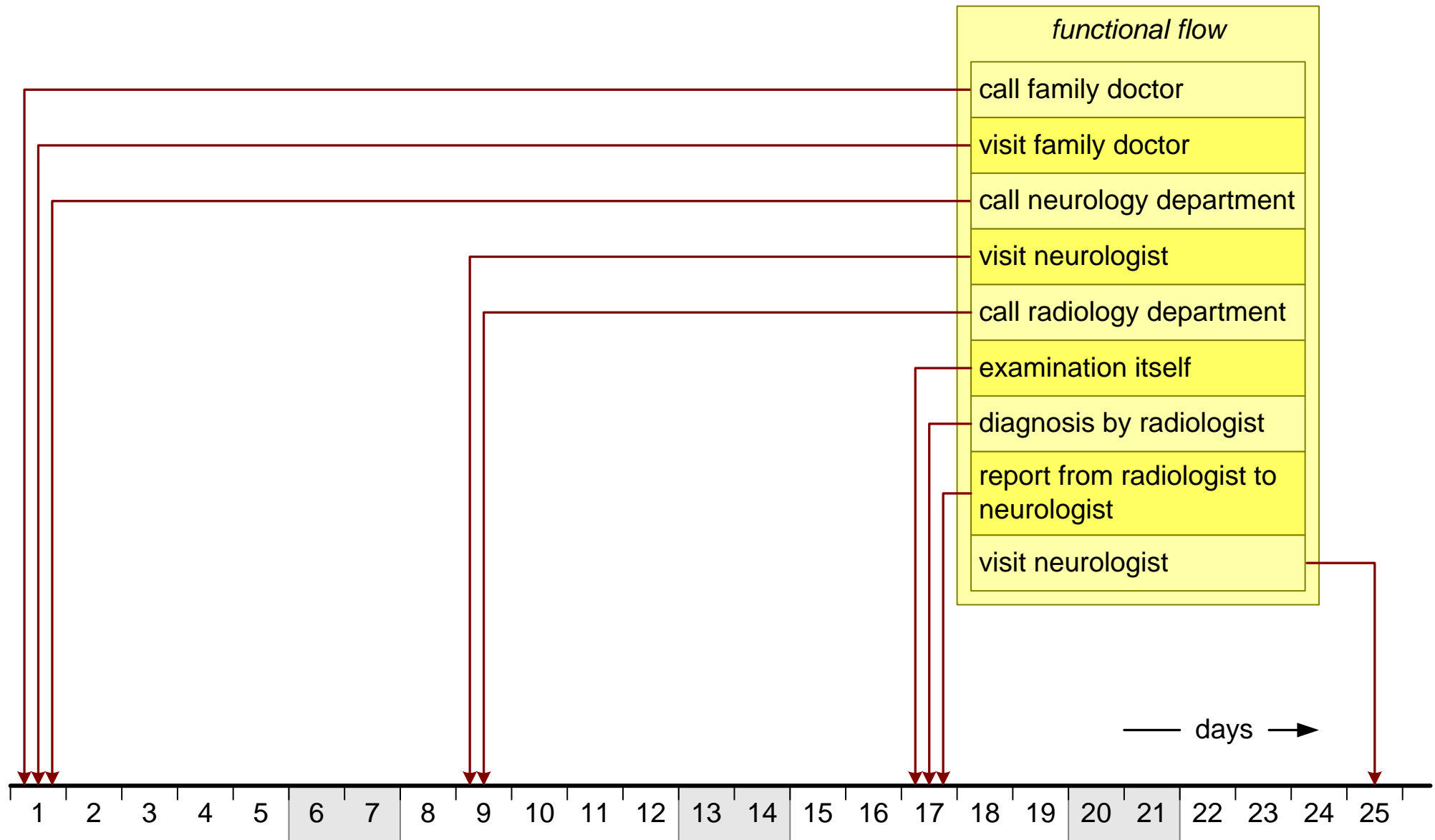
Swimming Lane Example



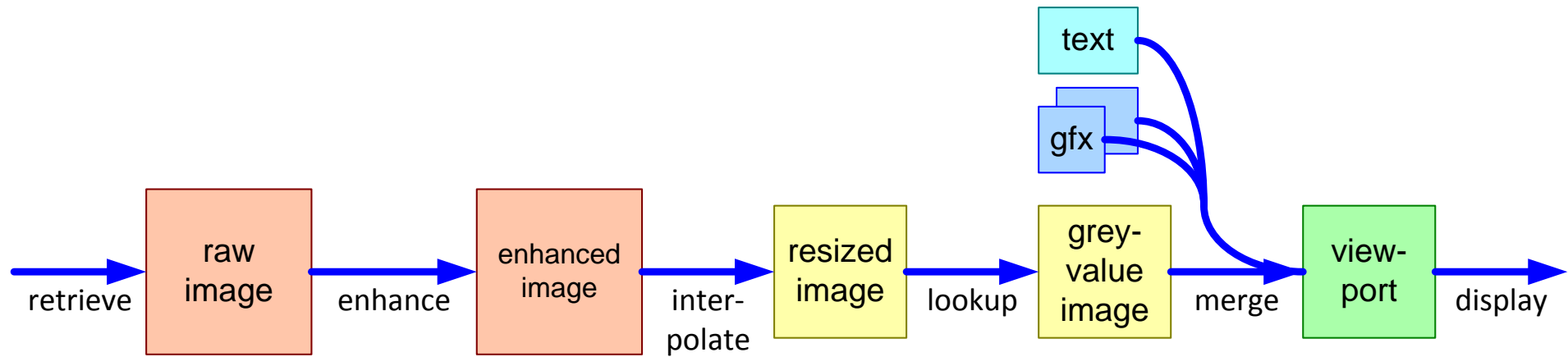
Example Signal Waveforms



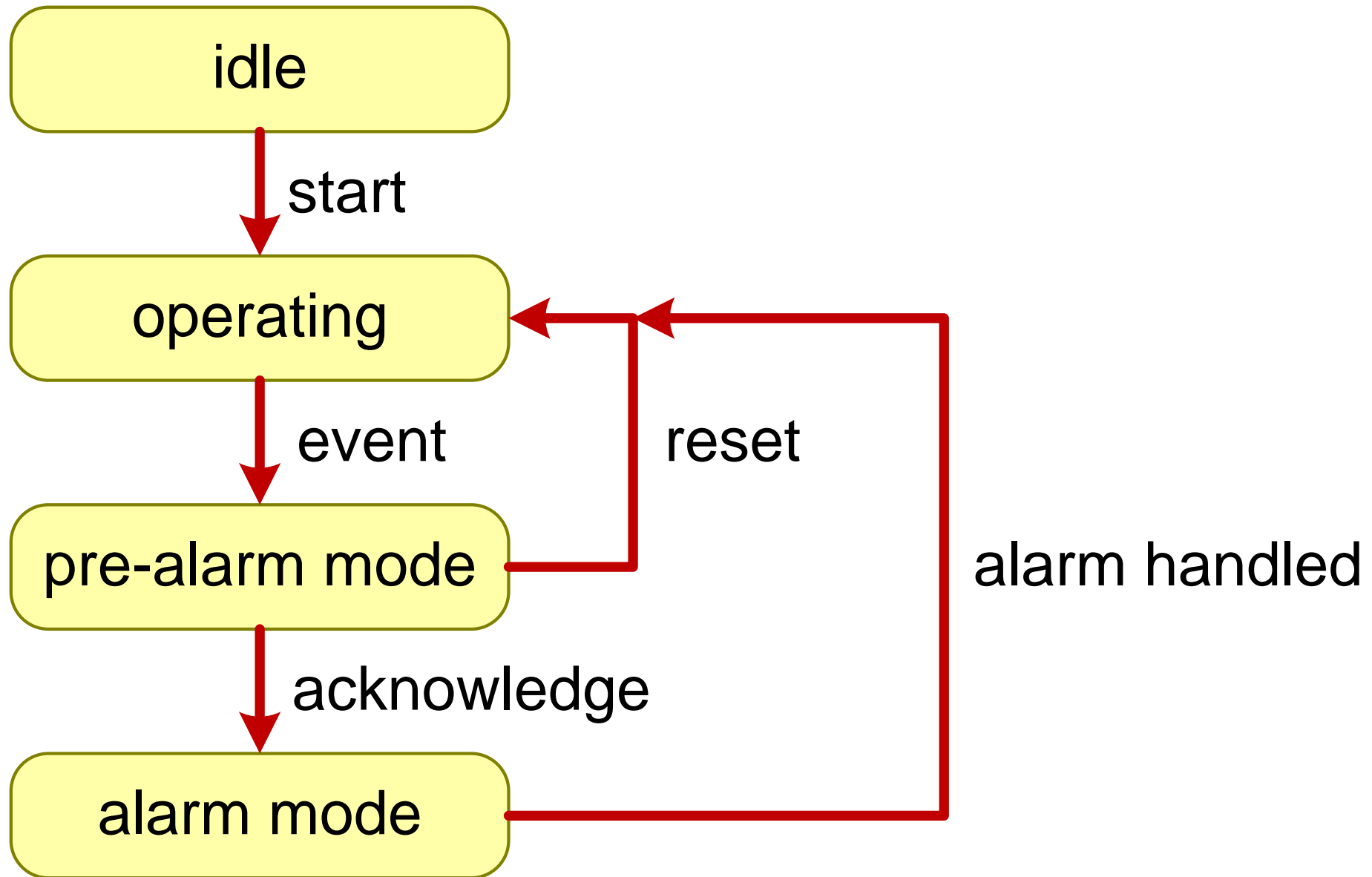
Example Time Line with Functional Model



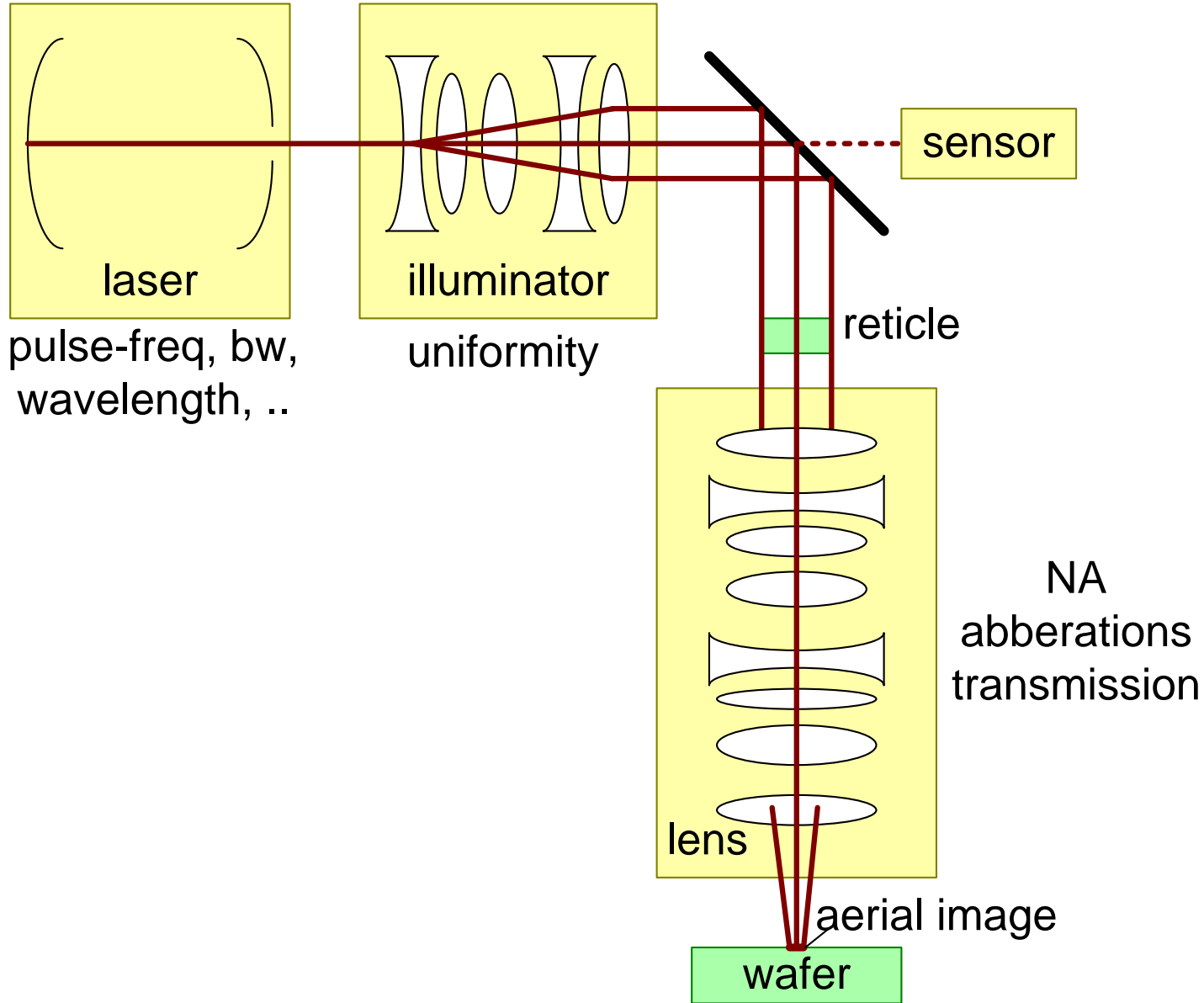
Information Centric Processing Diagram



Example State Diagram



Flow of Light (Physics)



Dynamic Behavior is Multi-Dimensional

How does the system work and operate?

Functions describe *what* rather than *how*.

Functions are *verbs*.

Input-Process-Output paradigm.

Multiple kinds of flows:

physical (e.g. hydrocarbons, goods, energy)

information (e.g. measurements, signals)

control

Time, events, cause and effect

Concurrency, synchronization, communication

multi-dimensional
information and
dynamic behavior

Systems Engineering Fundamentals Partitioning and Interfaces

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

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

The presentation explains fundamental concepts of and approach to system partitioning .

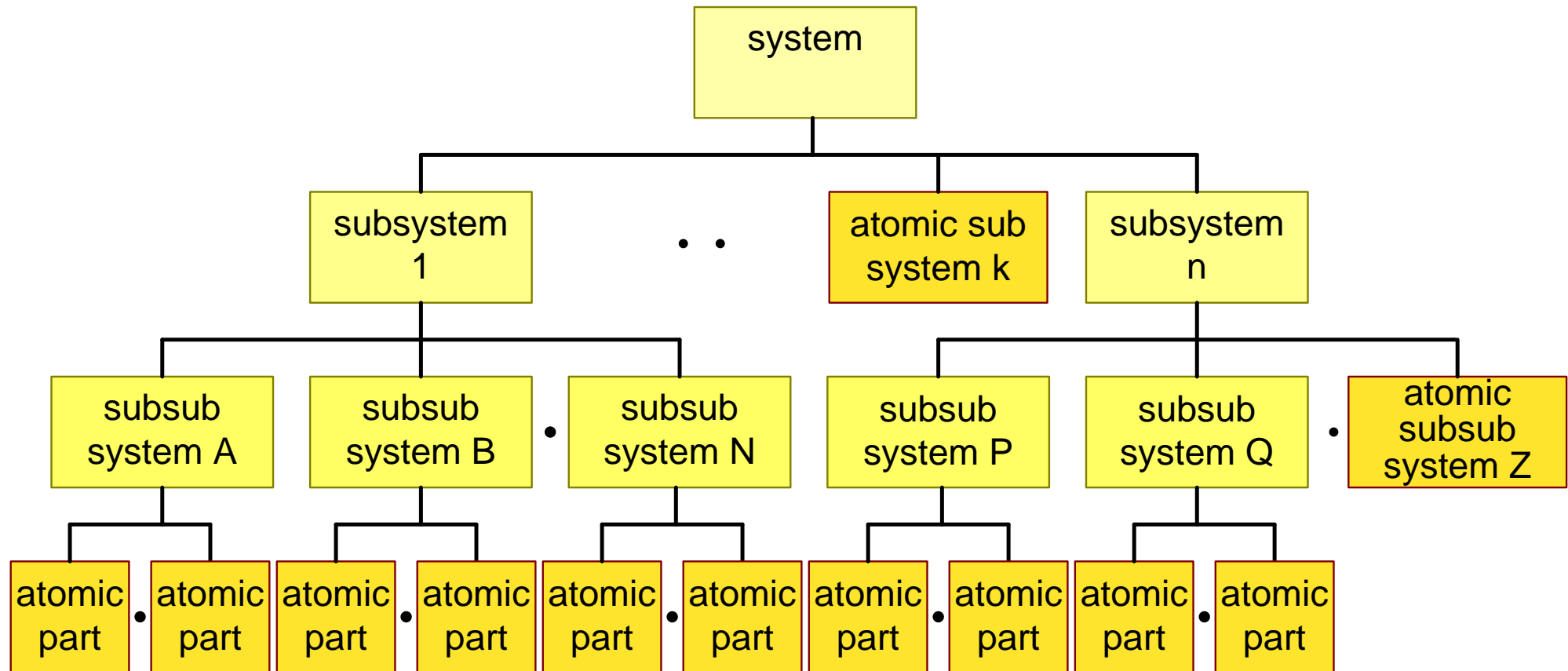
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

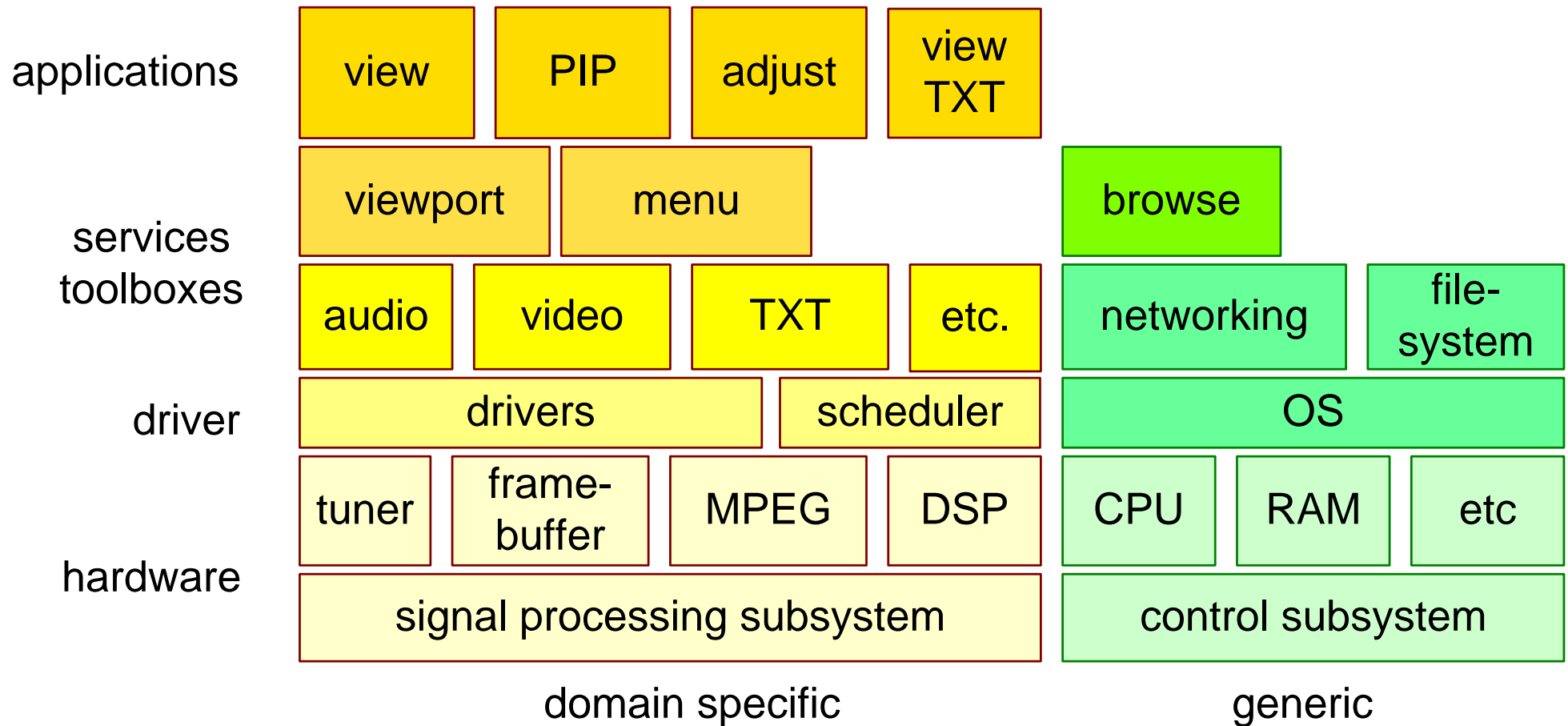
February 8, 2026
status: planned
version: 0

logo
TBD

Partitioning is Applied Recursively



Software plus Hardware Decomposition



the part is cohesive

functionality and technology belongs together

the coupling with other parts is minimal

minimize interfaces

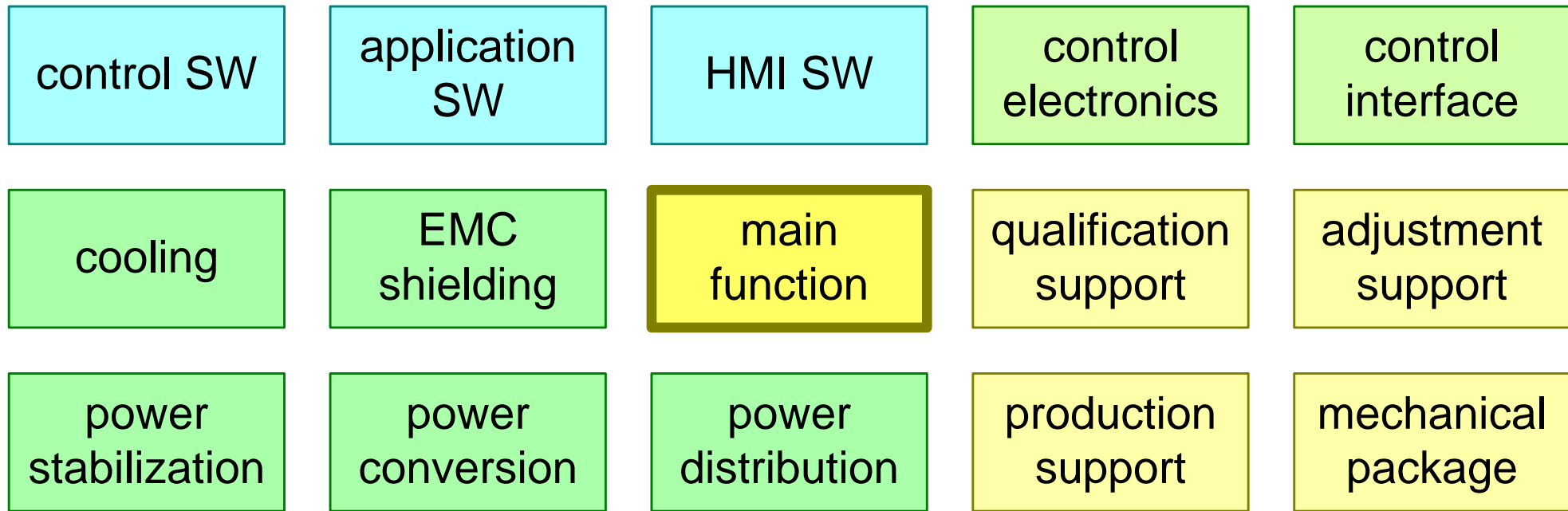
the part is self-sustained for production and qualification

can be in conflict with cost or space requirements

clear ownership of part

e.g. one department or supplier

How much self-sustained?



How self sustained should a part be?

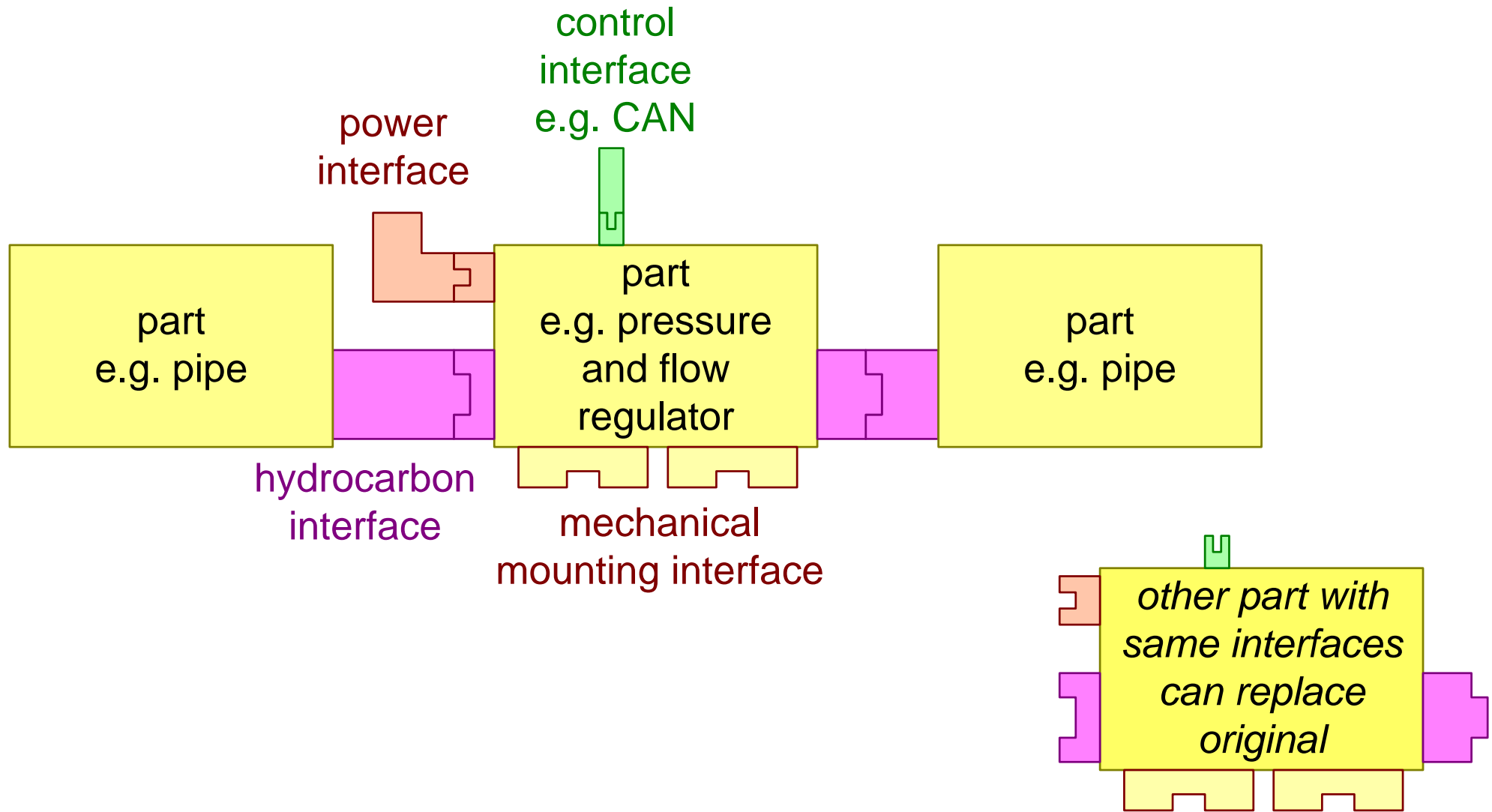
trade-off:

cost/speed/space
optimization



logistics/lifecycle/production
flexibility
clarity

Decoupling via Interfaces



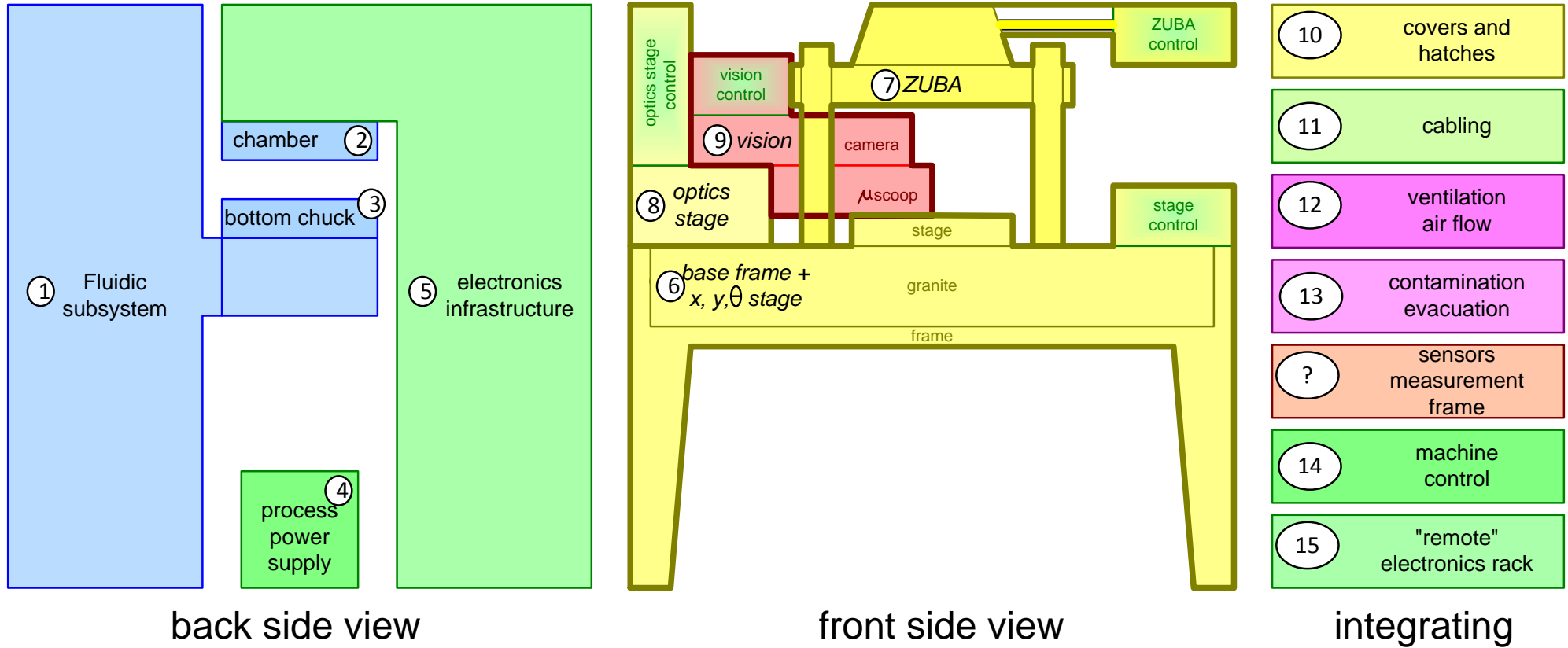
The Ideal Modularity

System is composed

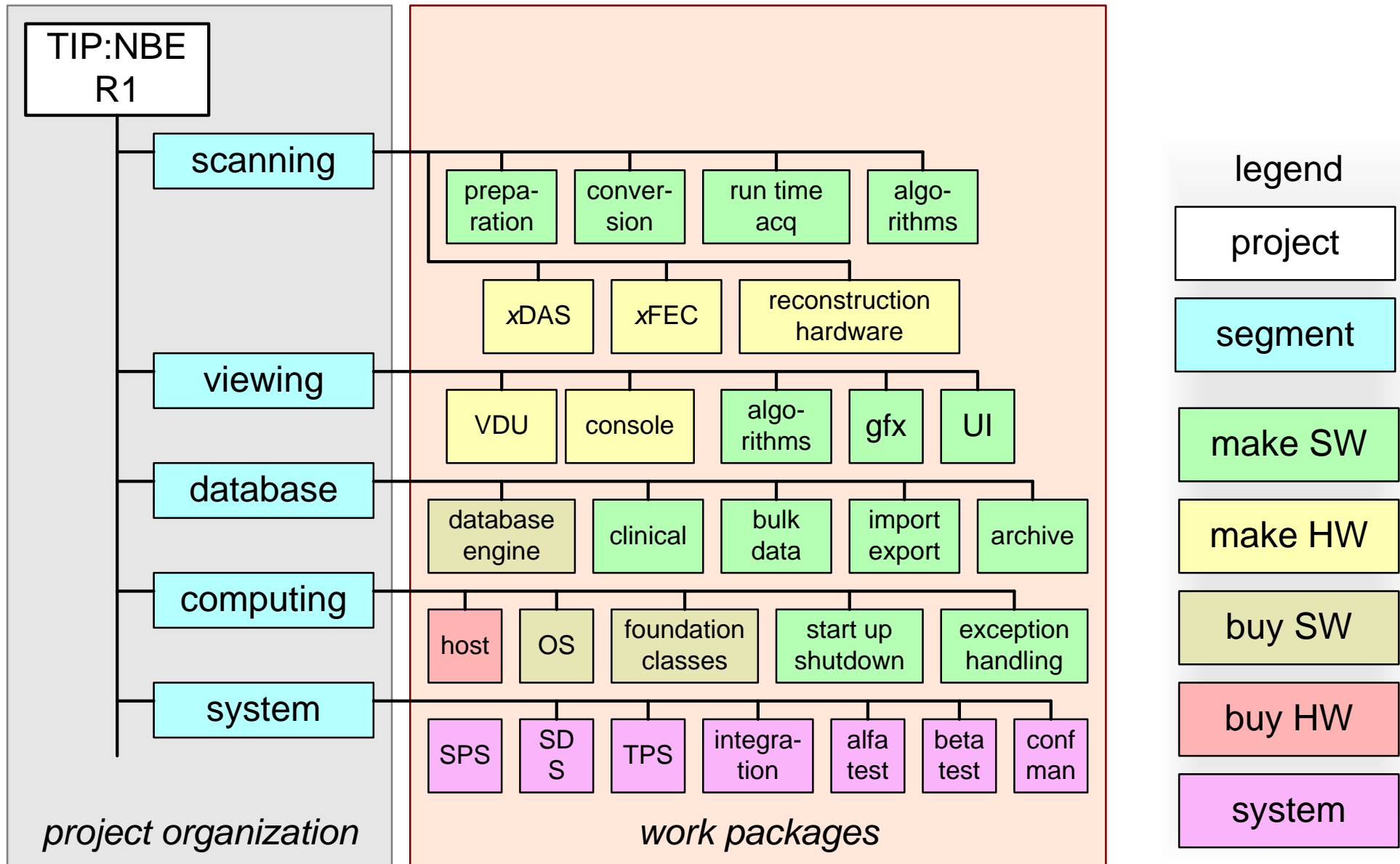
by using standard interfaces

limited catalogue of variants (e.g. cost performance points)

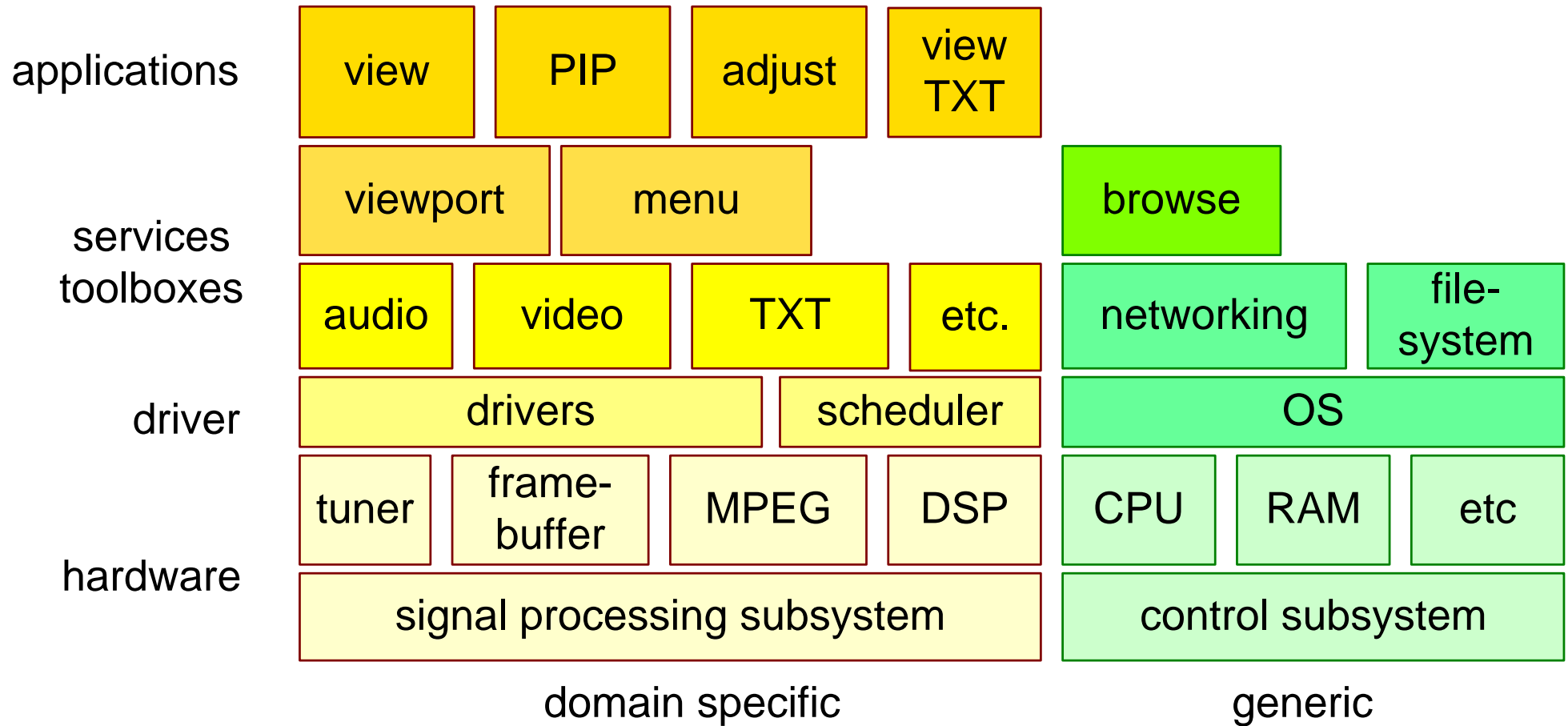
Example Physical Decomposition and Visualization



Example Work Breakdown Structure



Example SW plus HW Decomposition



Systems Engineering Fundamentals Supply Chain and Logistics

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

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

The supply chain dominates the economic viability of systems. Developing a system and its business requires the design of the supply chain.

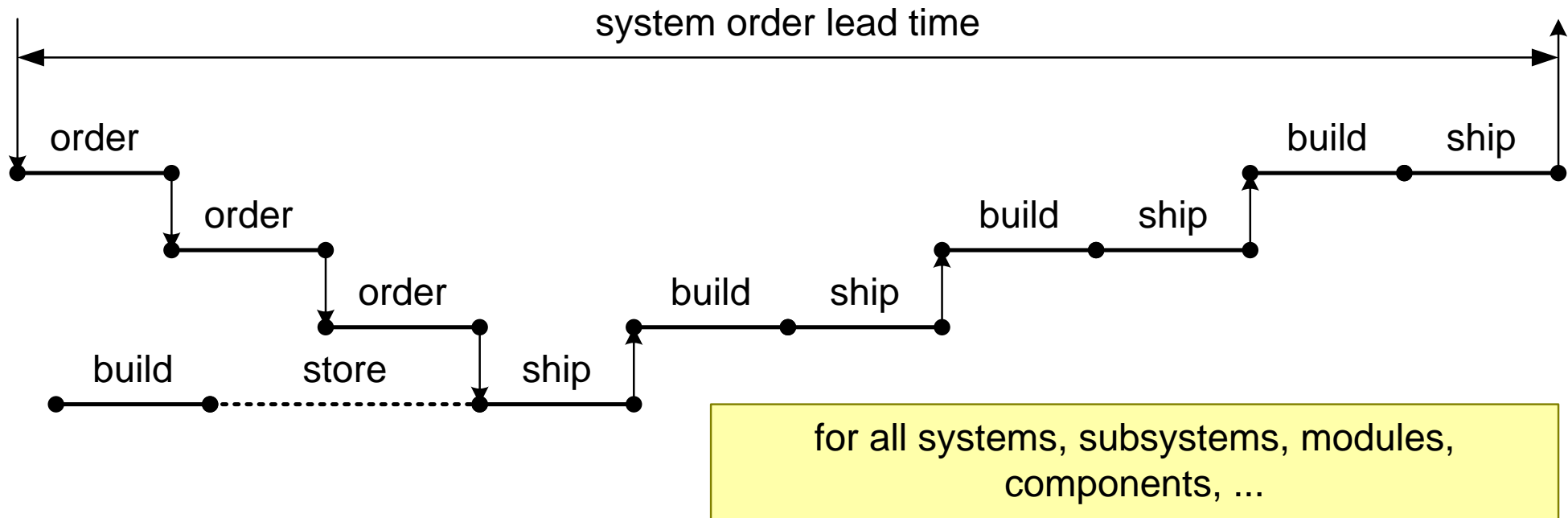
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026
status: planned
version: 0

logo
TBD

System Order Lead Time



Considerations for Designing the Supply Chain

- **Flow** of goods; stock = cost
- Produce Delivery Ratio < 1
 - Facilitate **demand-driven** goods flow.
 - Forecasting causes stocks and risks of obsolescence or underrun)
- **Risk** management
 - supplier dependency (2nd supplier policy)
- Production and service **life time**
- **Traceability** of configurations and versions

Systems Engineering Fundamentals Risk Management

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

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

Systems Engineering offers many methods and techniques to detect and mitigate risks. This presentation touches upon methods like Failure Mode Effect Analysis, Hazard Analysis, etc. addressing related concerns such as reliability, safety, and security.

Distribution

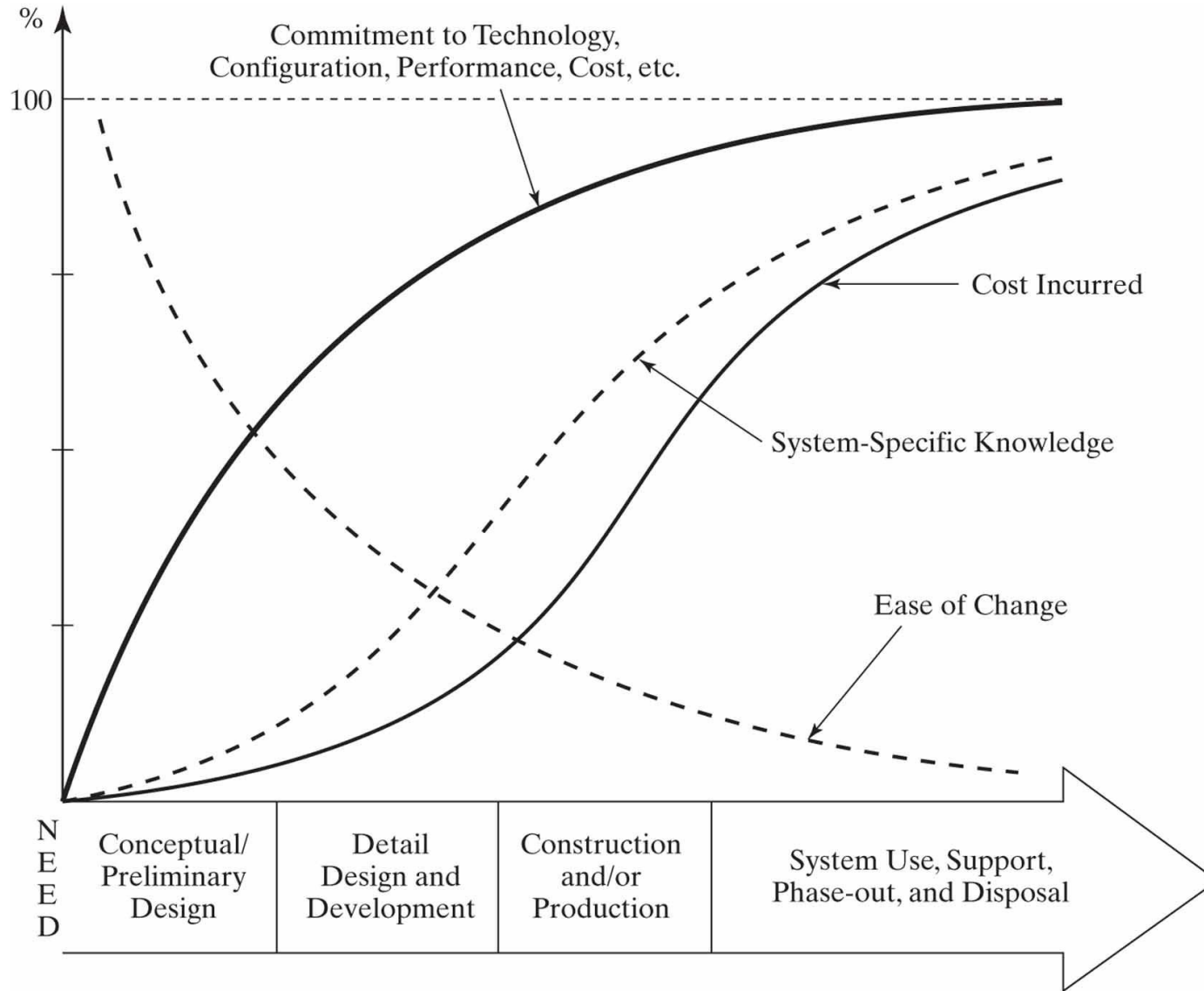
This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026
status: planned
version: 0

logo
TBD

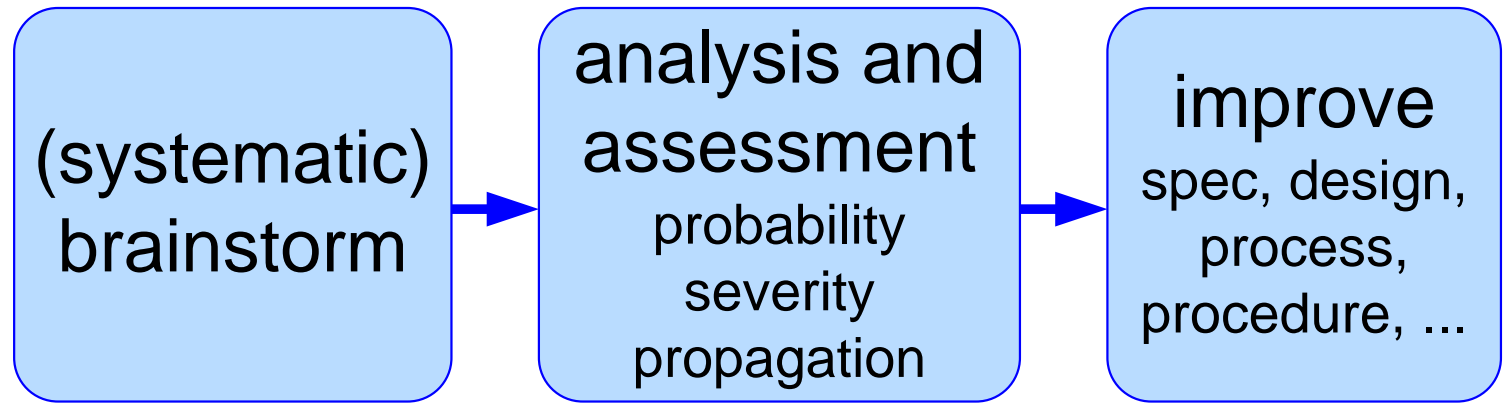
Life-cycle Commitment, Knowledge, and Incurred Cost

Systems Engineering and Analysis, Fifth Edition
 Benjamin S. Blanchard • Wolter J. Fabrycky
 Copyright © 2011, © 2006, © 1998 by Pearson Education, Inc.
 Upper Saddle River, New Jersey 07458



Copyright © 2011 Pearson Education, Inc. publishing as Prentice Hall

FMEA-like Analysis Techniques



safety hazard analysis	potential hazards	damage	measures
reliability FMEA	failure modes exceptional cases	effects	measures
security	vulnerability risks	consequences	measures
maintainability	change cases	impact, effort, time	decisions
performance	worst cases	system behavior	decisions

Severity-Probability Classification Matrix

probability ↑	E	moderate	moderate	high	unacceptable	unacceptable	unacceptable
	D	low	moderate	moderate	high	unacceptable	unacceptable
	C	low	low	moderate	moderate	high	unacceptable
	B	low	low	low	moderate	high	unacceptable
	A	low	low	low	low	moderate	high
		I	II	III	IV	V	VI
							severity →

based on https://en.wikipedia.org/wiki/Failure_mode_and_effects_analysis

Mastering Systems Integration; Readiness Levels

by *Gerrit Muller* [TNO-ESI, University of South-Eastern Norway]

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

Readiness level models offer a yardstick to assess the status of specific project aspects. Examples are technology readiness and integration readiness.

Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026
status: planned
version: 0

logo
TBD

Technology Readiness Levels

TRL 9	actual system proven in operational environment
TRL 8	system complete and qualified
TRL 7	system prototype demonstration in operational environment
TRL 6	technology demonstrated in relevant environment
TRL 5	technology validated in relevant environment
TRL 4	technology validated in lab
TRL 3	experimental proof of concept
TRL 2	technology concept formulated
TRL 1	basic principles observed

after: <https://serkanbolat.com/2014/11/03/technology-readiness-level-trl-math-for-innovative-smes/>

Integration Readiness Levels

TRL 7	The integration of technologies has been verified and validated with sufficient detail to be actionable.
TRL 6	The integrating technologies can accept, translate, and structure information for its intended application.
TRL 5	There is sufficient control between technologies necessary to establish, manage, and terminate the integration.
TRL 4	There is sufficient detail in the quality and assurance of the integration between technologies.
TRL 3	There is compatibility (i.e. common language) between technologies to orderly and efficiently integrate and interact.
TRL 2	There is some level of specificity to characterize the interaction (i.e. ability to influence) between technologies through their interface.
TRL 1	An interface (i.e. physical connection) between technologies has been identified with sufficient detail to allow characterization of the relationship.

from: From TRL to SRL: The Concept of Systems Readiness Levels, CSER2006, by Sauser et al.

Mastering Systems Integration; Process and Positioning

by *Gerrit Muller* [TNO-ESI, University of South-Eastern Norway]

e-mail: gaudisite@gmail.com

www.gaudisite.nl

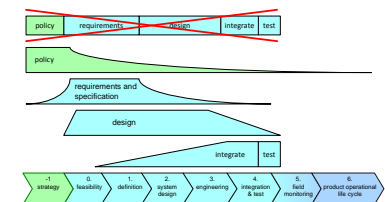
Abstract

This lesson positions systems integration as process in the development processes.

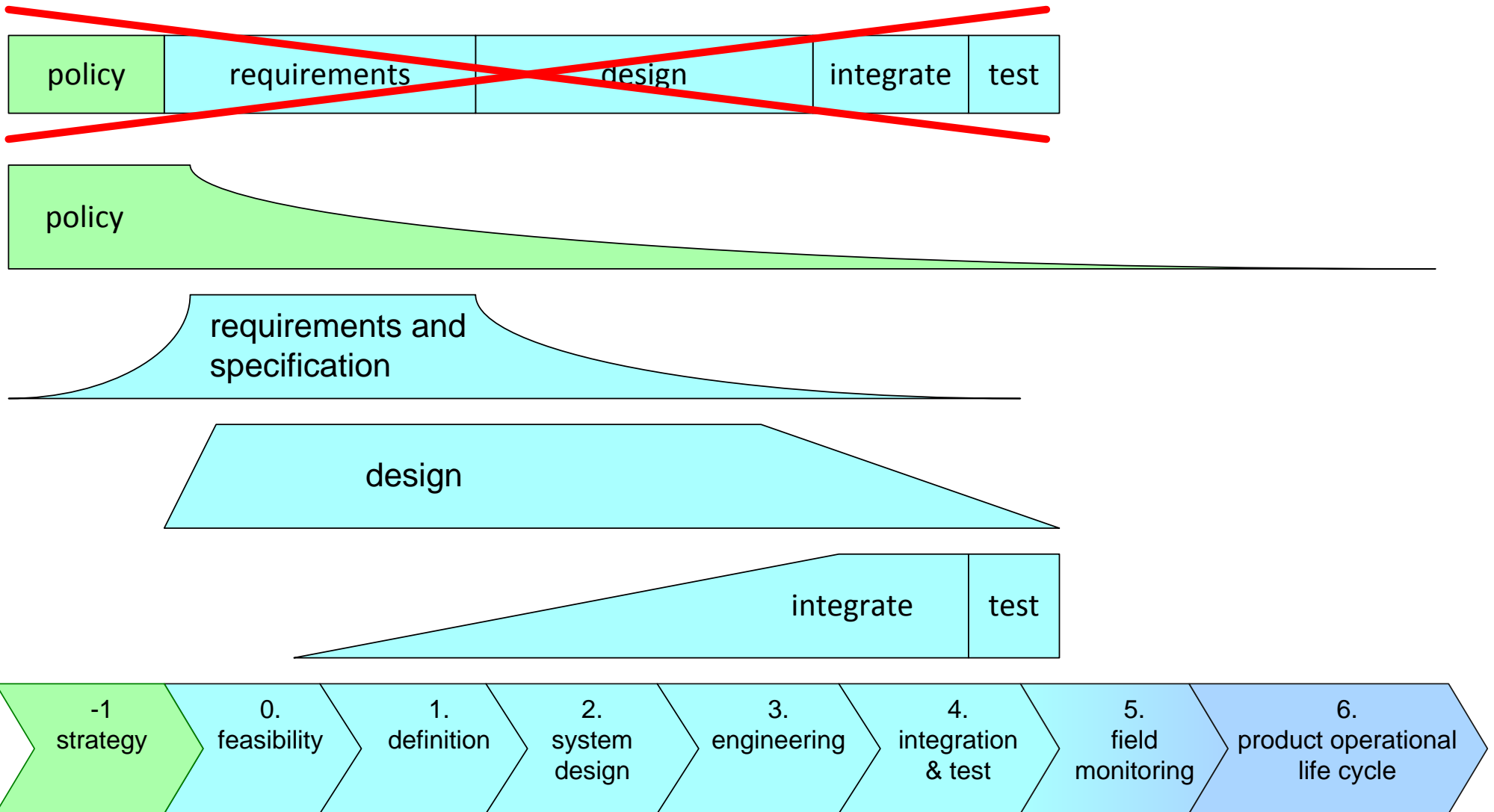
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

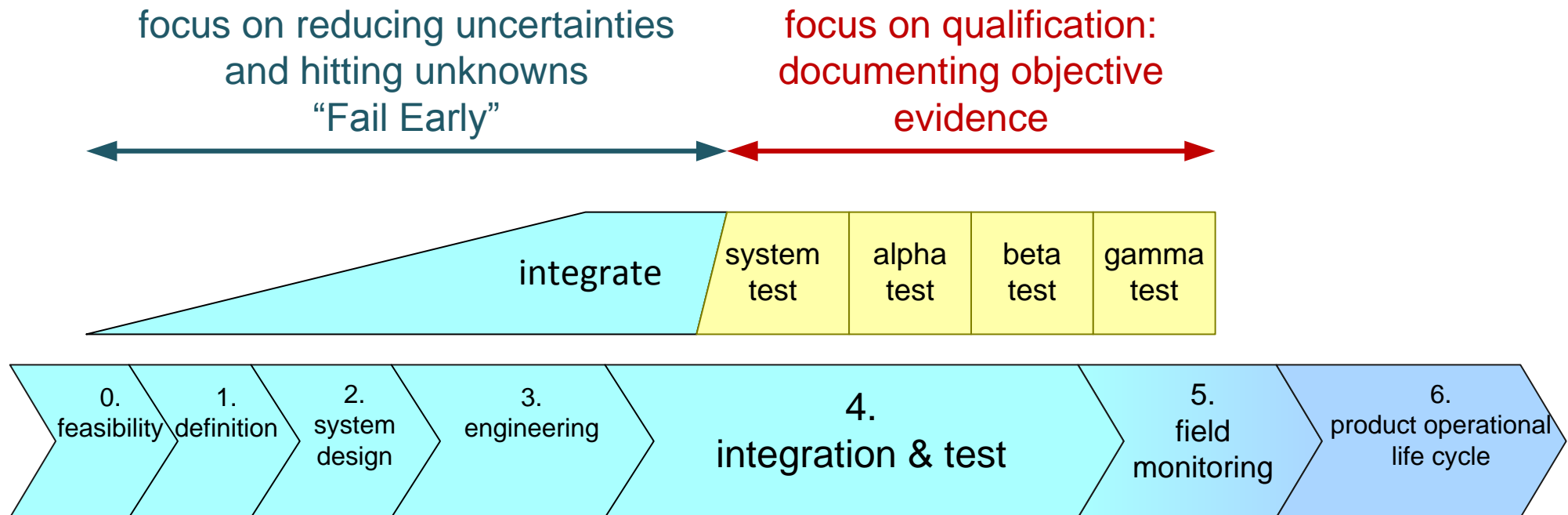
February 8, 2026
status: planned
version: 0.3



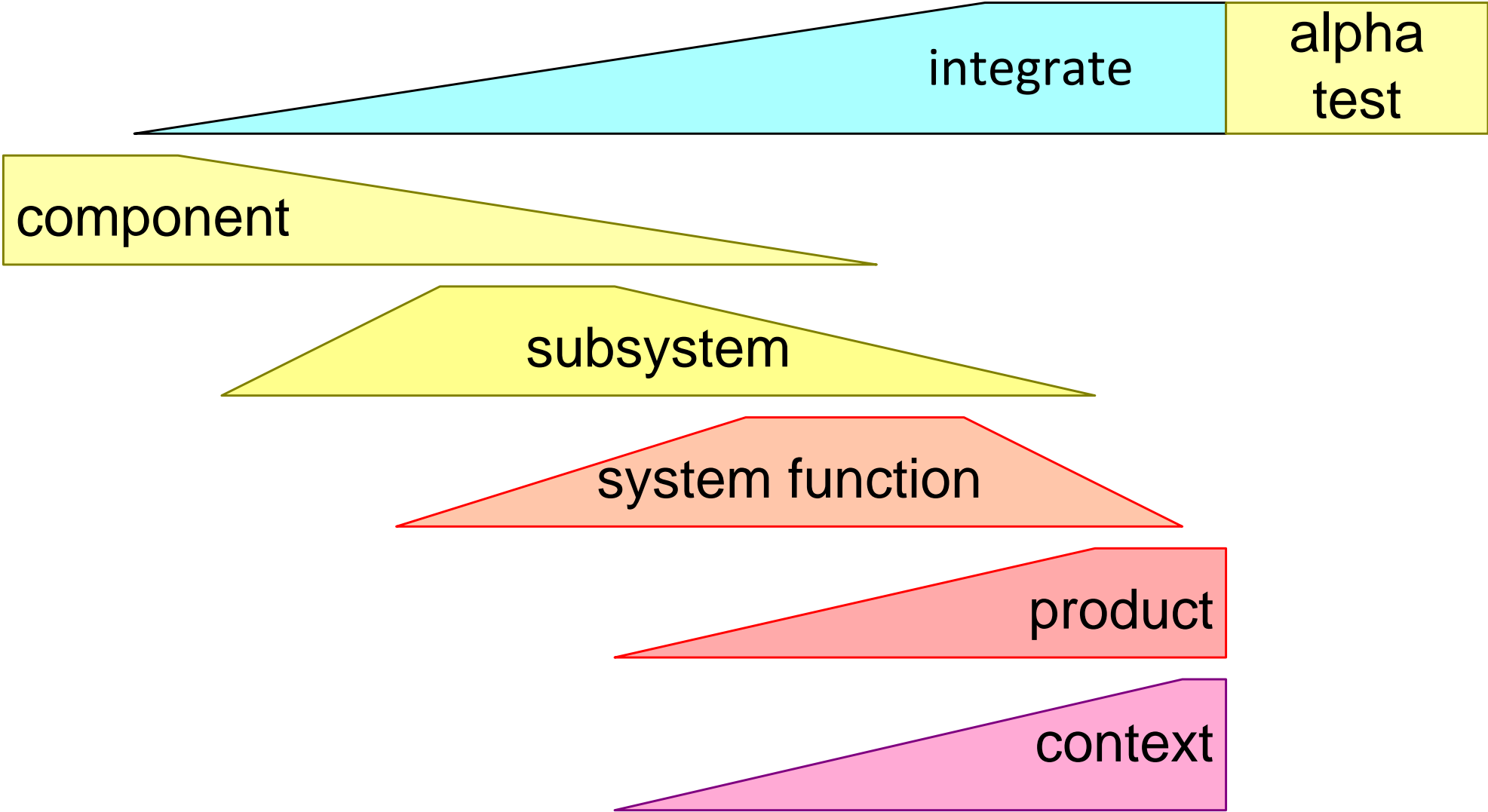
Typical Concurrent Product Creation Process



Zooming in on Integration and Tests



Integration Takes Place in a Bottom-up Fashion



Mastering Systems Integration; Integration Strategy

by *Gerrit Muller* [TNO-ESI, University of South-Eastern Norway]

e-mail: gaudisite@gmail.com

www.gaudisite.nl

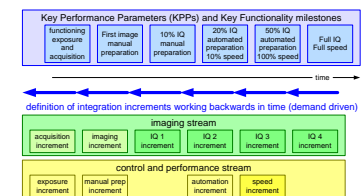
Abstract

This presentations discusses the strategy for integration. The strategy is transformed into an approach to determine an integration sequence based on Key Performance Parameters and potential risks to achieve them.

Distribution

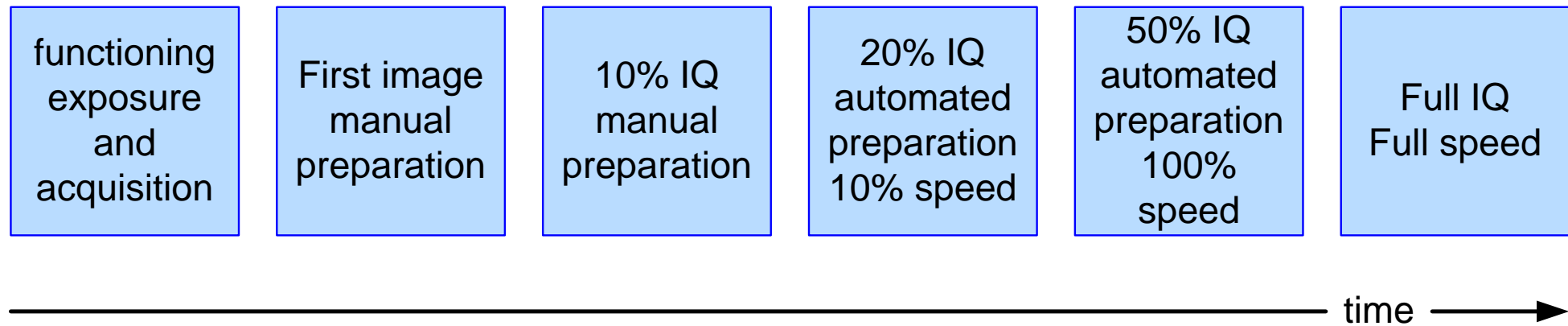
This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026
status: preliminary
draft
version: 0.5



- Get Key Performance Parameters functioning ASAP
- Work on highest risks ASAP
- Use a pacing process (regular visible results)
 - with regular milestones
 - and increments in functionality and performance
- Merge constraints from test configurations, suppliers, resources, etc.

Pacing Milestones



pacing:

maximum 6 month between milestones
depending on technology and domain

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.

Mastering Systems Integration; Project Management

by *Gerrit Muller* [TNO-ESI, University of South-Eastern Norway]

e-mail: gaudisite@gmail.com

www.gaudisite.nl

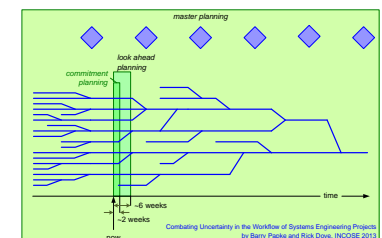
Abstract

Systems Integration requires specific project management. The challenge for project managers is to plan ahead, knowing that the integration plan will need continuous adaptations.

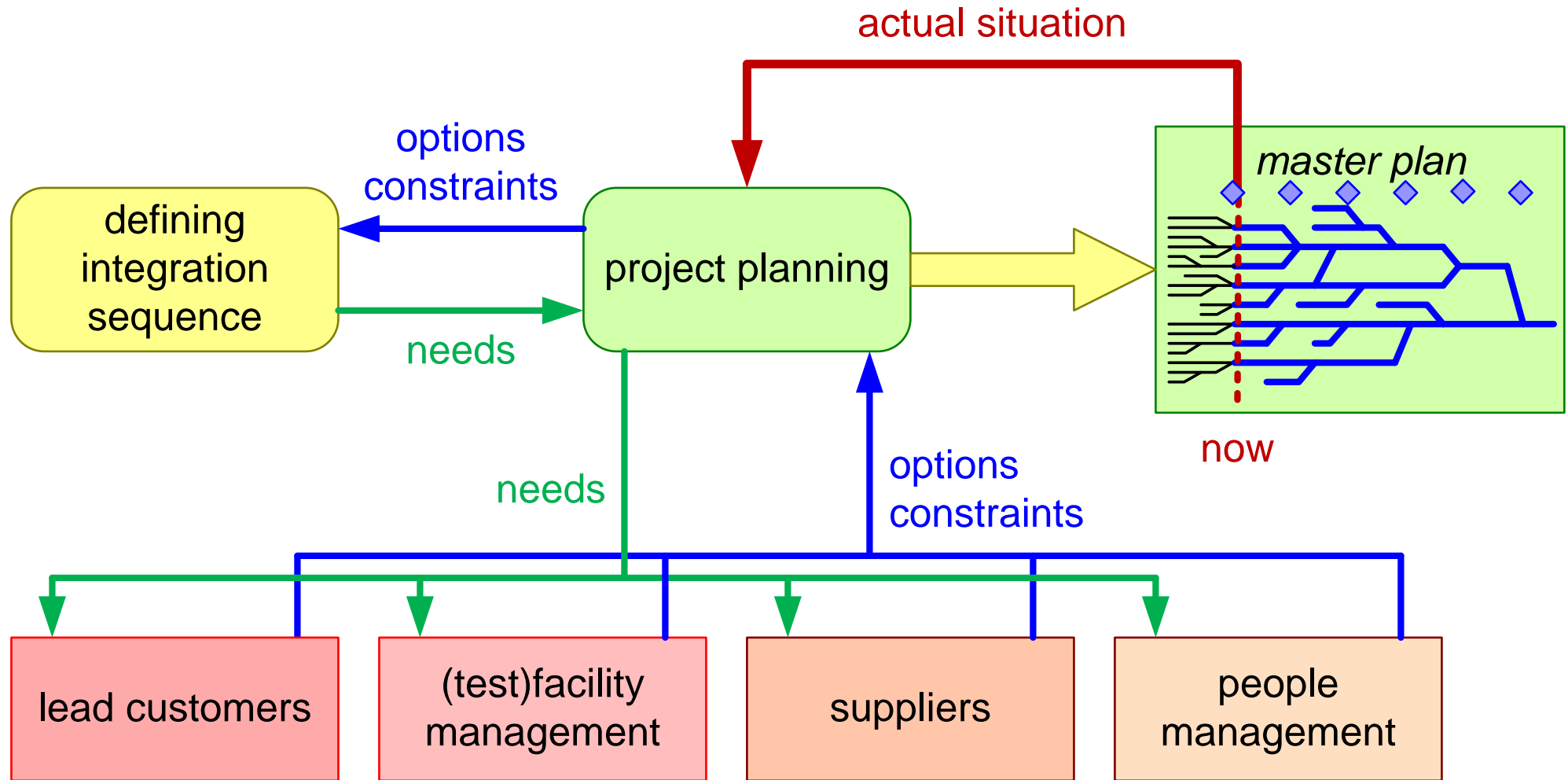
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

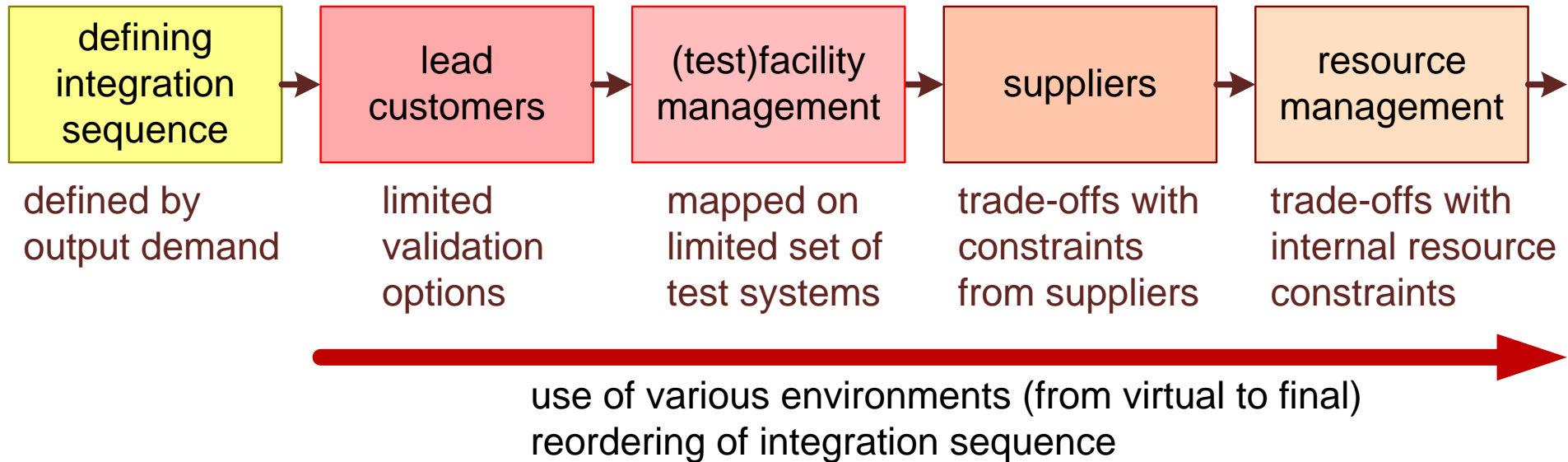
February 8, 2026
status: preliminary
draft
version: 0.6



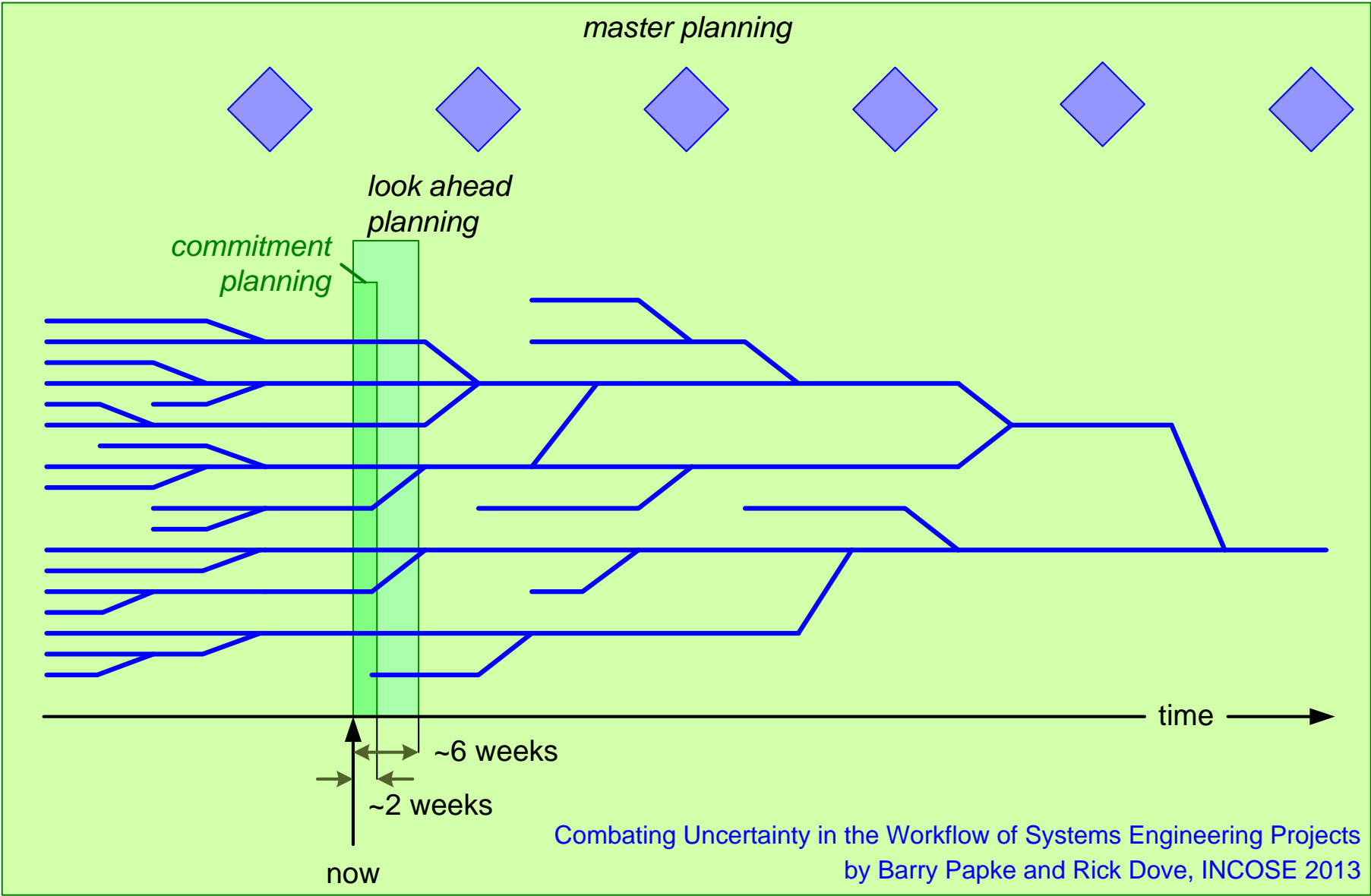
Integration Planning



Demand Driven, Fitting Constraints



Last Planner; Look Ahead!



Mastering Systems Integration; Terminology

by *Gerrit Muller* [TNO-ESI, University of South-Eastern Norway]

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

This presentation defines terms, which are used in relation to systems integration, such as validation, verification, qualification, evidence, approval process, certification, and acceptance.

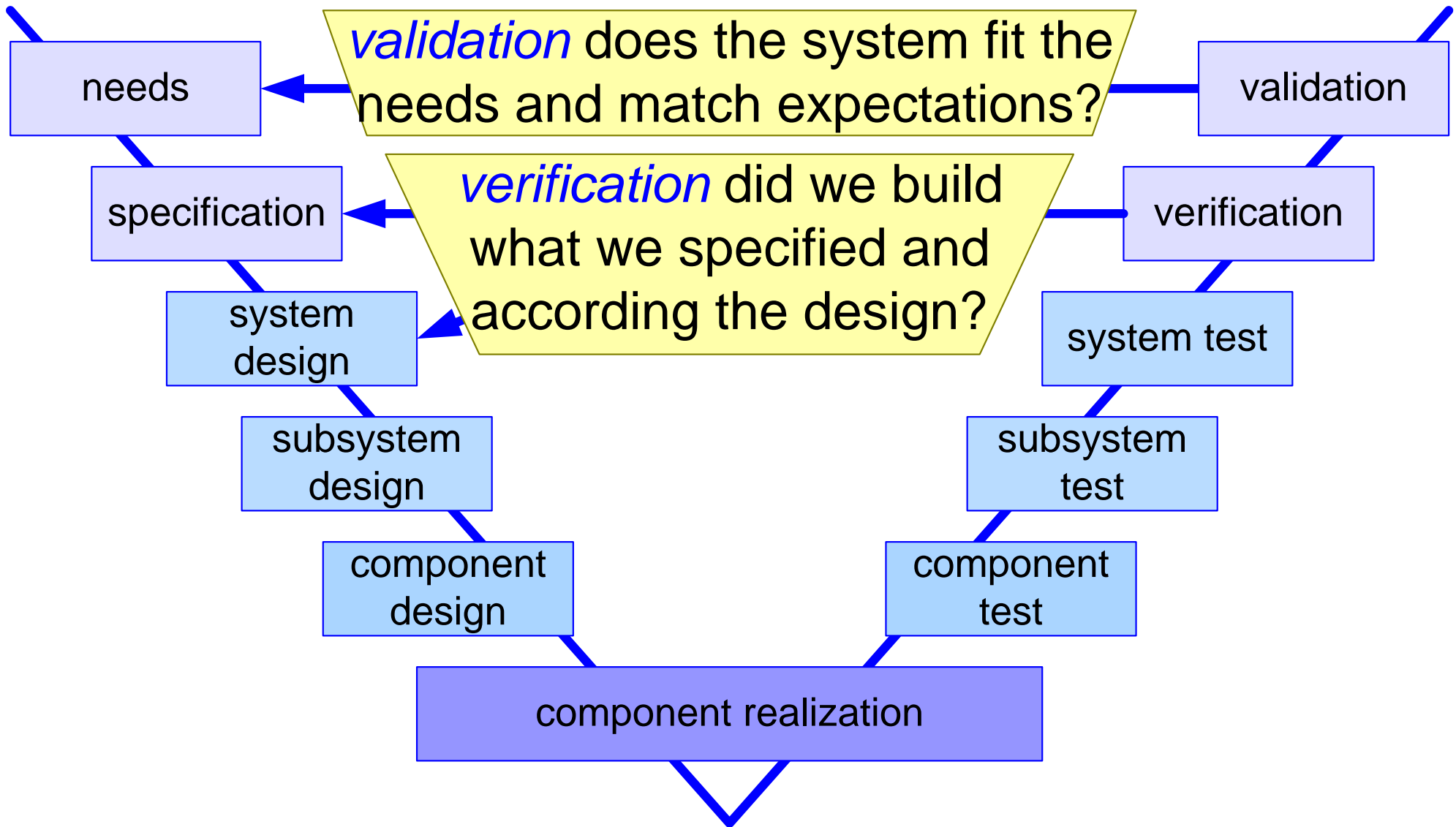
Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

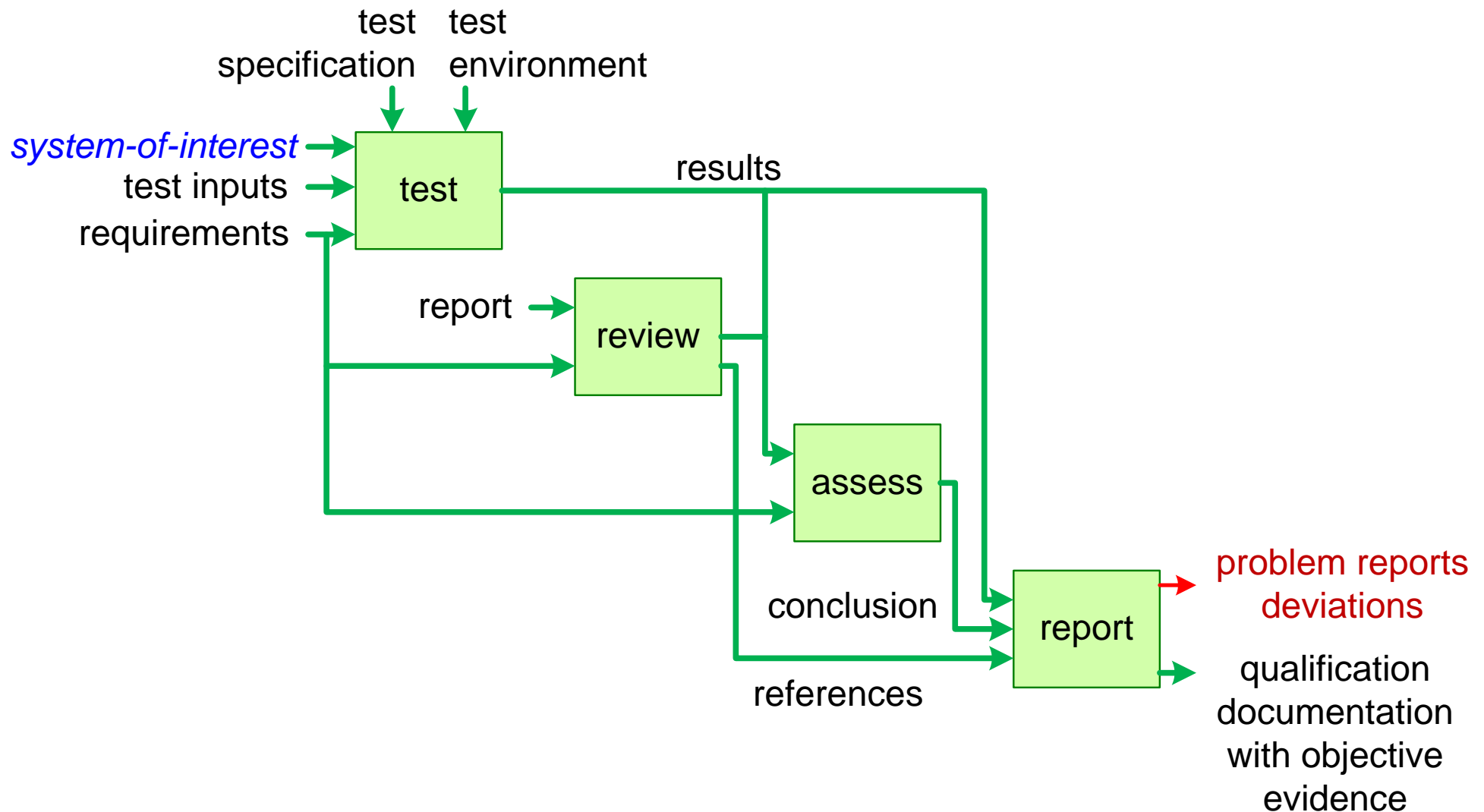
February 8, 2026
status: planned
version: 0

logo
TBD

Validation and Verification in the V-model



Functional Model of Verification



Certification

Certification: an independent agency (e.g. DNV-GL) certifies the quality of the system-of-interest, technology, or process

Self-certification: the company has been accredited by the agency to do the certification themselves.

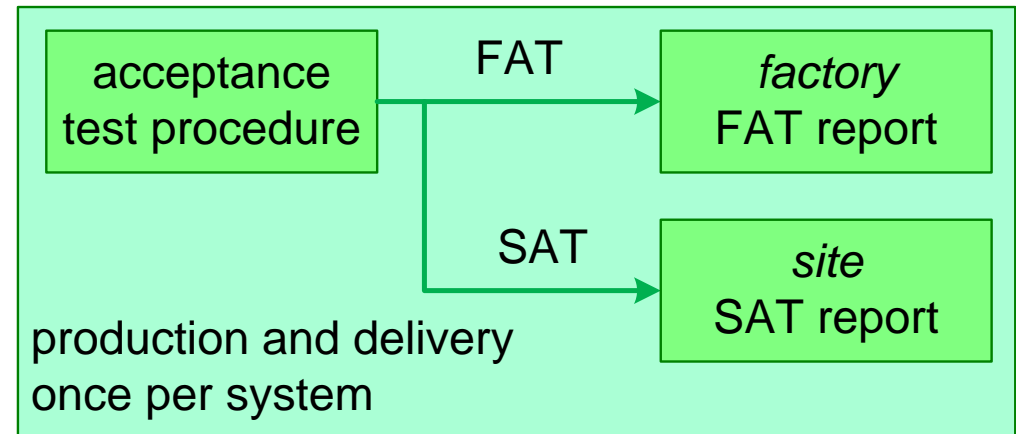
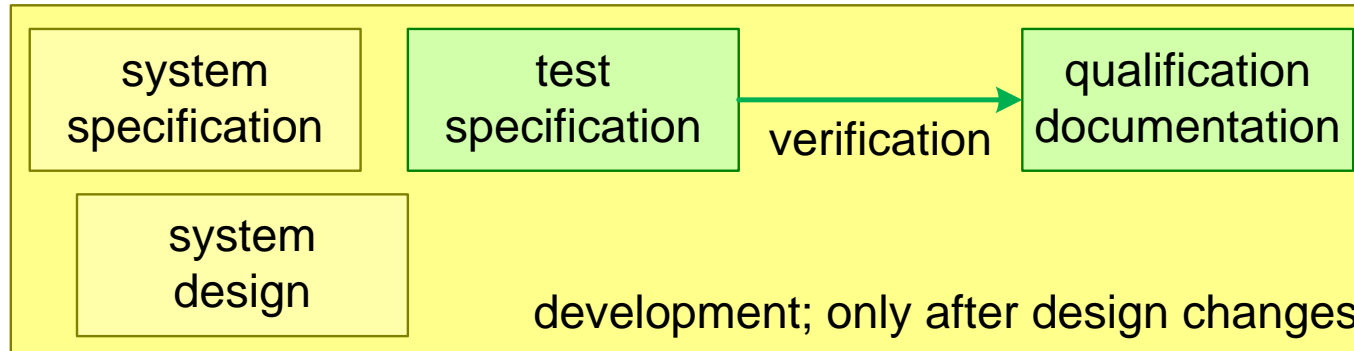
check
qualification data

check process
and organization

optional audit

certify

Development and (repeated) Production



Objective Evidence

From a business perspective: Objective evidence is “information based on facts that can be proved through analysis, measurement, observation, and other such means of research.”

From a legal perspective: Objective evidence is “real evidence, also known as demonstrative or objective evidence; this is naturally the most direct evidence.”

From a scientific perspective: “To be termed scientific, a method of inquiry must be based on gathering observable, empirical, and measurable evidence subject to specific principles of reasoning. A scientific method consists of the collection of data through observation and experimentation, and the formulation and testing of hypotheses.”

From a list of Plain English definitions related to the ISO 9000, 9001 and 9004: Objective evidence is “data that show or prove that something exists or is true. Objective evidence can be collected by performing observations, measurements, tests, or by using any other suitable method.”

from: Understanding Objective Evidence: (What It Is and What It Definitely Is Not),
by Denise Dion http://www.eduquest.net/Advisories/EduQuest%20Advisory_ObjectiveEvidence.pdf

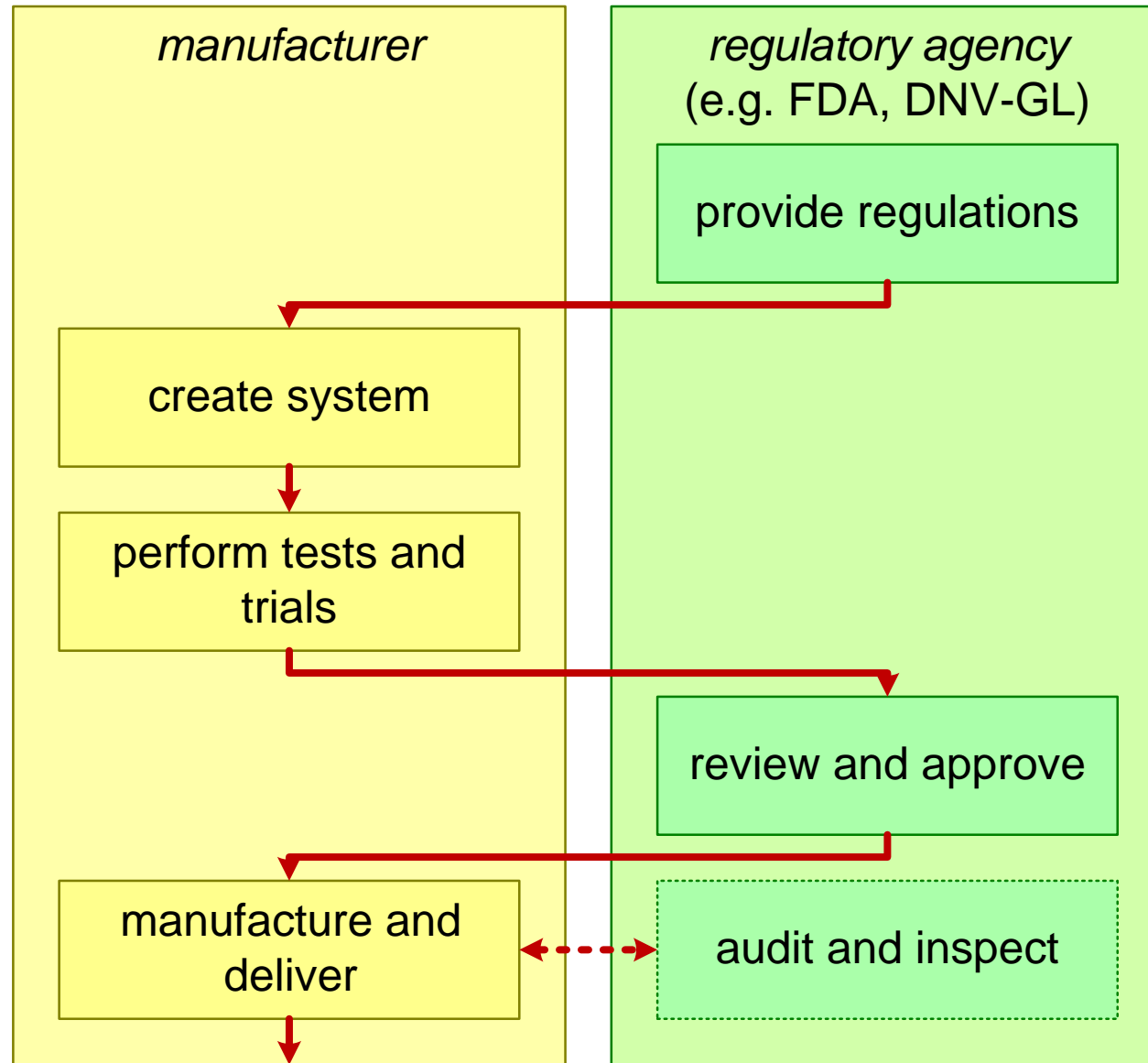
FDA Requirements for Objective Evidence

FDA is a science-based law enforcement agency and, therefore, requires answers that are scientifically and legally supported. FDA expects your objective data to answer the following questions:

- **Scientific** – Can the data be *evaluated by independent observers* to reach the same conclusions?
- **Scientific** – Are the data documented in a manner that *allows re-creation of the data* or the events described?
- **Scientific** – Does the documented evidence provide *sufficient data* to prove what happened, when, by whom, how, and why?
- **Legal** – Was the documentation *completed concurrently* with the tasks?
- **Legal** – Is the documentation *attributable* (directly traceable to a person)?
- **Legal** – Have the data and associated documentation been maintained in a manner that *provides traceable evidence* of changes, deletions, additions, substitutions, or alterations?
- **Legal** – Are the data and associated documentation maintained in a manner that *protects and secures* them from changes, deletions, additions, substitutions, or alterations?

from: Understanding Objective Evidence: (What It Is and What It Definitely Is Not),
by Denise Dion http://www.eduquest.net/Advisories/EduQuest%20Advisory_ObjectiveEvidence.pdf

Regulatory Approval Process



Systems Engineering Deployment and Commissioning

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

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

Deployment covers all steps from receiving shipped components to getting the system live and operating the system in its operational context.

Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

February 8, 2026
status: planned
version: 0

logo
TBD

Deployment Workflow

