

Produkto reikalavimų inžinerijos procesas

Dr. Asta Slotkienė

New and Changed Practices

- Technical Solution!

Level	Focus	Process Areas	Quality Productivity
5 Optimizing	Continuous Process Improvement	Causal Analysis and Resolution (CAR) Organizational Performance Management (OPM)	
4 Quantitatively Managed	Quantitative Management	Organizational Process Performance (OPP) Quantitative Project Management (QPM)	
3 Defined	Process Standardization	Integrated Project Management (IPM) Risk Management (RSKM) Decision Analysis and Resolution (DAR) Requirements Development (RD) Technical Solution (TS) Product Integration (PI) Verification (VER) Validation (VAL) Organizational Process Focus (OPF) Organizational Process Definition (OPD) Organizational Training (OT)	
2 Managed	Basic Project Management	Configuration Management (CM) Measurement and Analysis (MA) Project Monitoring and Control (PMC) Project Planning (PP) Process and Product Quality Assurance (PPQA) Requirements Management (REQM) Supplier Agreement Management (SAM)	
1 Initial			Risk Rework

CMMI Development V1.3

Level	✓ ← PA also exists at higher level	Practice Areas	Performance Capability
5 Optimizing		CAR, MPM (level 5 practices)	
4 Quantitatively Managed		CAR, PCM, SAM, MPM, PLAN, GOV (level 4 practices)	
3 Defined	✓ ✓	Causal Analysis and Resolution (CAR) Decision Analysis and Resolution (DAR) Risk and Opportunity Management (RSK) Organizational Training (OT) Process Management (PCM) Process Asset Development (PAD) Peer Reviews (PR) Verification and Validation (VV) Technical Solution (TS) Product Integration (PI)	
2 Managed	✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓	Supplier Agreement Management (SAM) Managing Performance and Measurement (MPM) Process Quality Assurance (PQA) Configuration Management (CM) Monitor and Control (MC) Planning (PLAN) Estimation (EST) Requirements Development & Management (RDM) Governance (GOV) Implementation Infrastructure (II)	
1 Initial / 0 Incomplete			Risk Rework

CMMI Development V2.0 – Simple View

Technical Solution (TS) (CMMI-DEV)

- The purpose of Technical Solution (TS) is to **select, design, and implement solutions** to requirements.
- Solutions, designs, and implementations encompass **products, product components**, and product related lifecycle processes either singly or in combination as appropriate.
- Its intention is **to ensure that designed and prepared products are in line with customer requirements**.
- TS value is providing effective design and solution that meets customer requirements while reducing costs and reworks. (ML1, ML2, ML3)

Technical Solution (TS) (CMMI-DEV)

TS.SG 1 Select Product Component Solutions Product or product component solutions are selected from **alternative solutions**.

TS.SG 2 Develop the Design Product or product component designs are developed.

TS.SG 3 Implement the Product Design Product components, and associated support documentation, are implemented from their designs.

Technical Solution (TS) (CMMI-DEV): **TS.SG 1** **Select Product Component Solutions Product**

- Architectural choices
- Architectural patterns
that **support achievement of quality attribute requirements**
- The use commercial off-the-shelf (COTS) product components are considered relative to cost, schedule, performance, and risk.
 - COTS alternatives can be used with or without modification

Alternative solutions and selection criteria (CMMI-DEV)

- Cost of development, manufacturing, procurement, maintenance, and support,
- Achievement of key **quality attribute requirements**, such as product timeliness, safety, reliability, and maintainability
- **Complexity of the product component and product-related lifecycle processes**
- **Robustness to product operating** and use conditions, operating modes, environments, and variations in product-related lifecycle processes
- **Product expansion and growth**
- **Technology limitations**
- Sensitivity to construction methods and materials
- Risk
- Evolution of requirements and technology
- Disposal
- Capabilities and limitations of end users and operators
- *Characteristics of COTS products*

Software Architect vs. Software Design

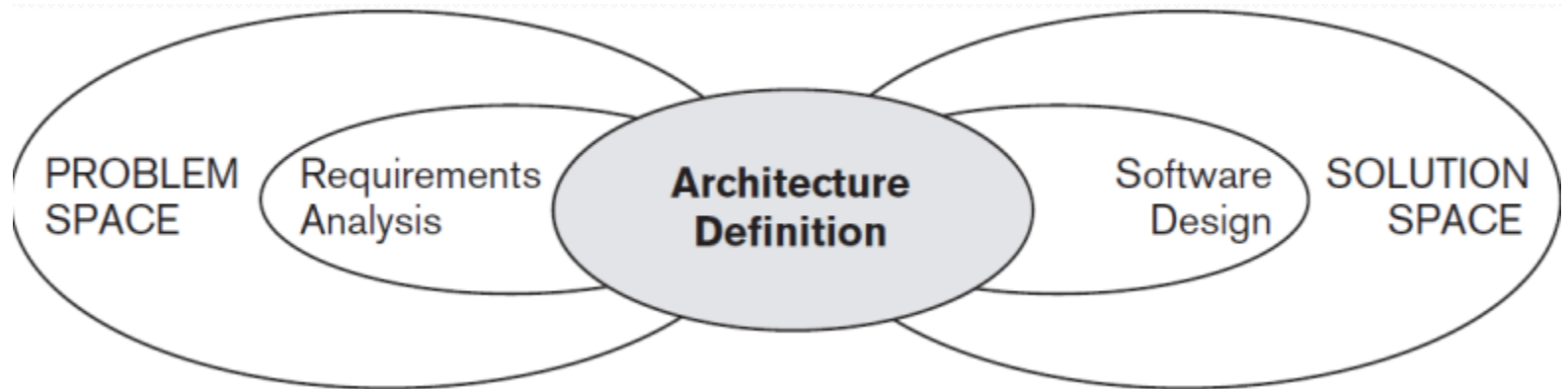
- Subsystems
- Systems
- Integration
- Code
- Classes
- Components
- Packages
- Modules

All architecture is design but not all design is architecture.

Architecture represents the significant design decisions that shape a system, where significant is measured by cost of change.

[Grady Booch]

Software Architect vs. Software Design

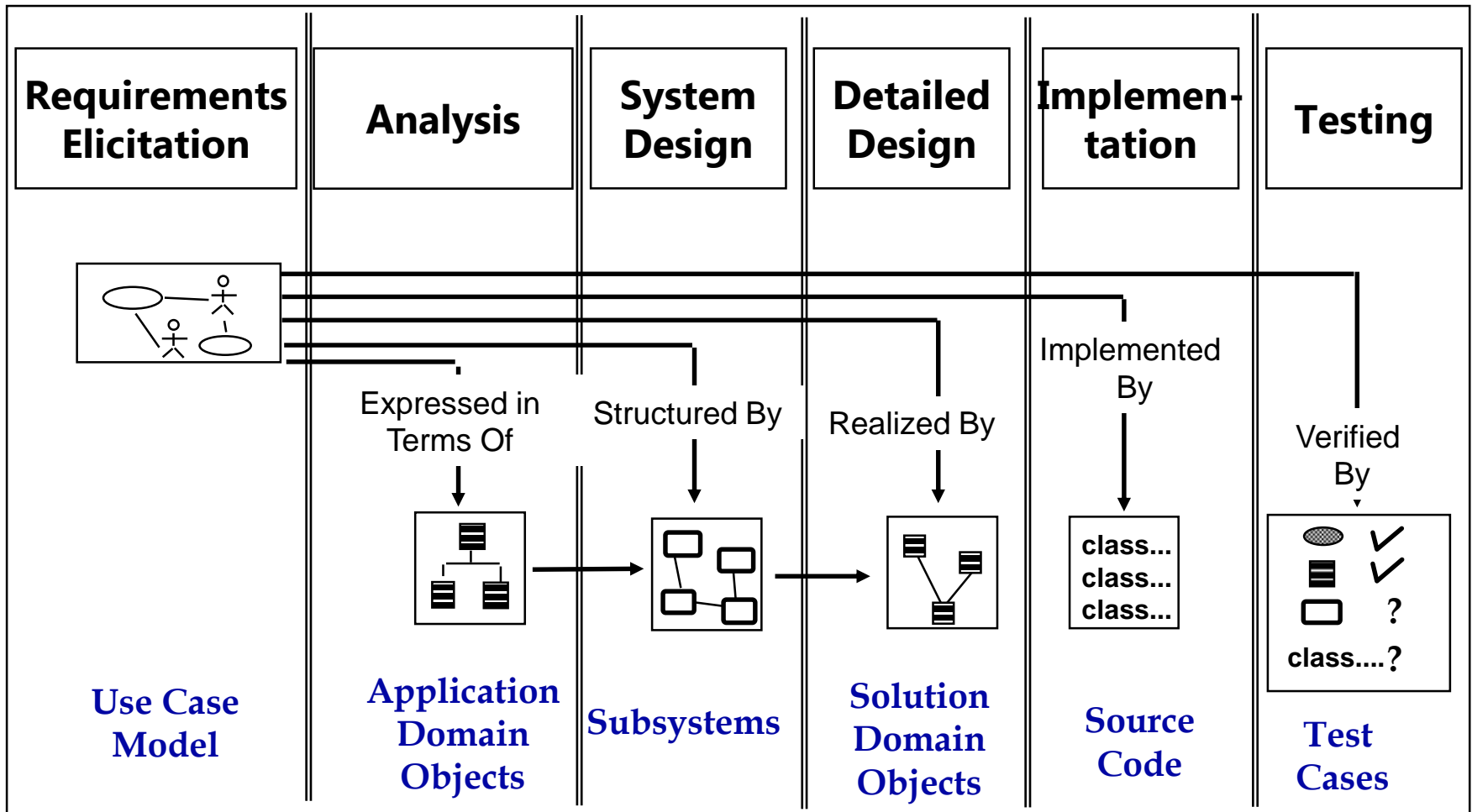


All architecture is design but not all design is architecture.

Architecture represents the significant design decisions that shape a system, where significant is measured by cost of change.

[Grady Booch]

Activities of Software System Lifecycle



From Requirements to Design

The requirements addressed the

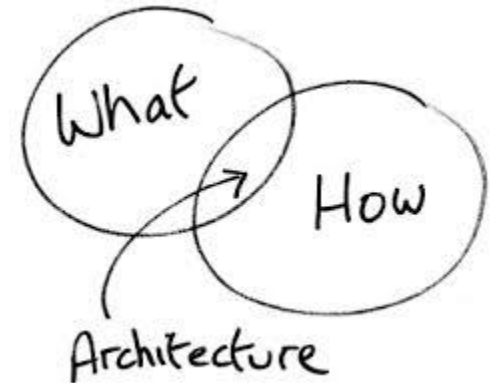
“What?”

what the system is supposed to do, what are the constraints, etc.

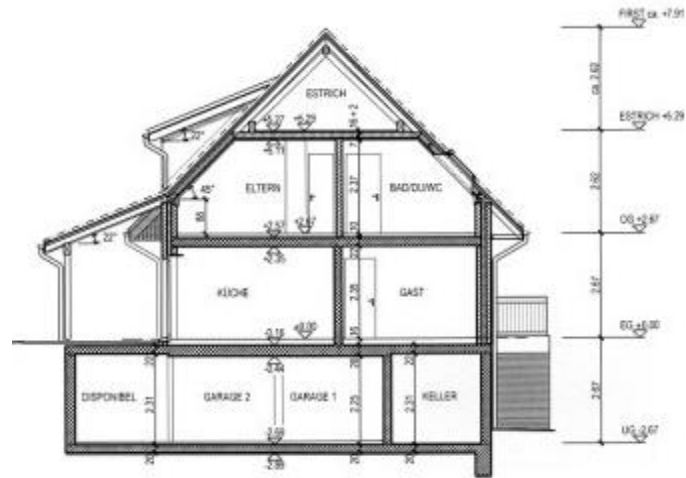
design addresses the

“How?”

how the system is decomposed into components, how these components interface and interact, how each individual component works, etc.



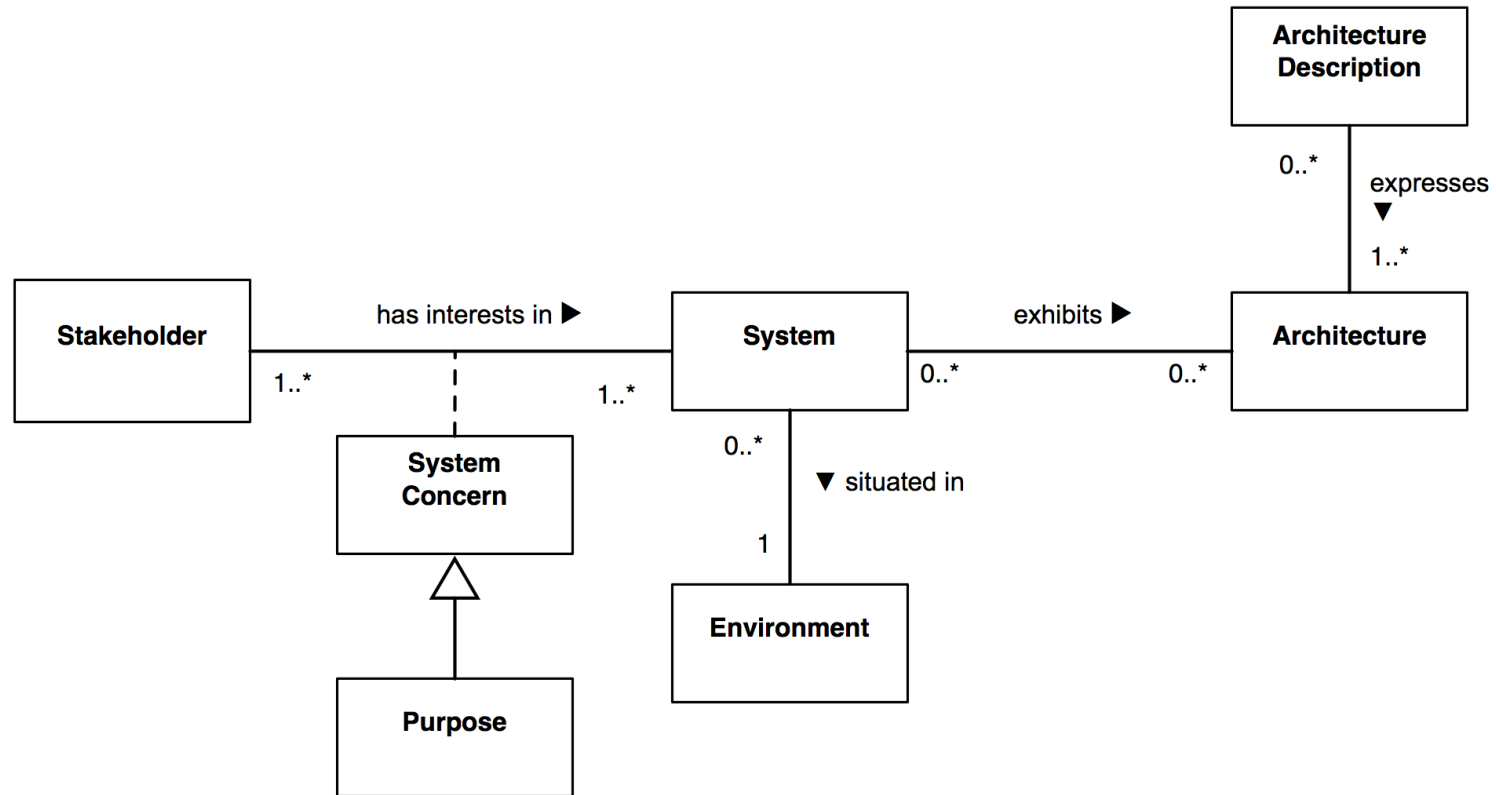
What is architecture?



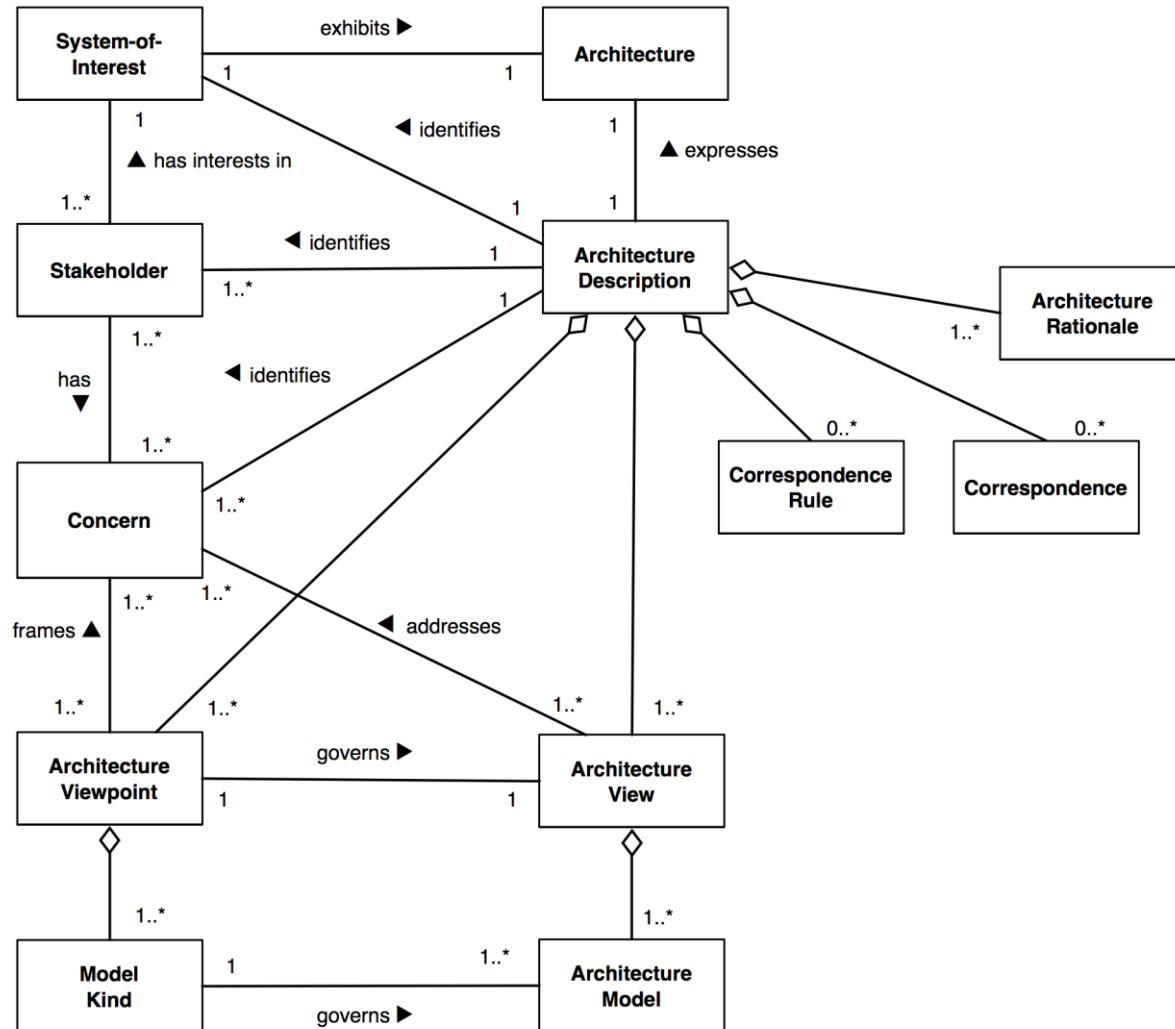
Definition of Software Architecture

The fundamental organization of a system embodied in its **components, their relationships to each other**, and to the environment, and the principles guiding its design and evolution.

A Conceptual Model of Architecture Description



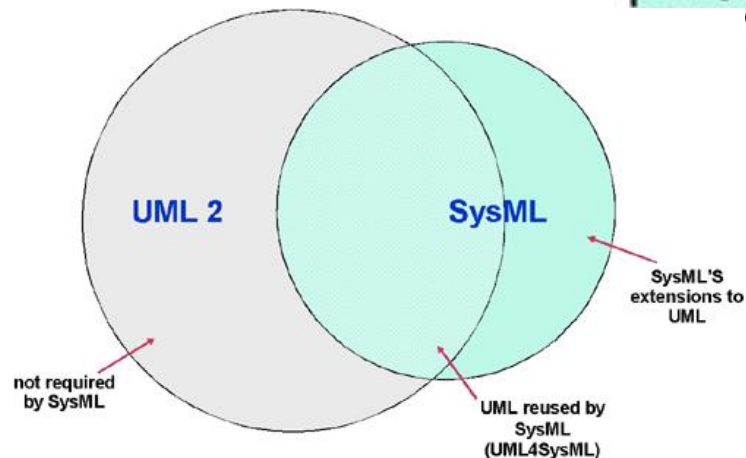
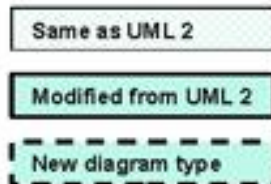
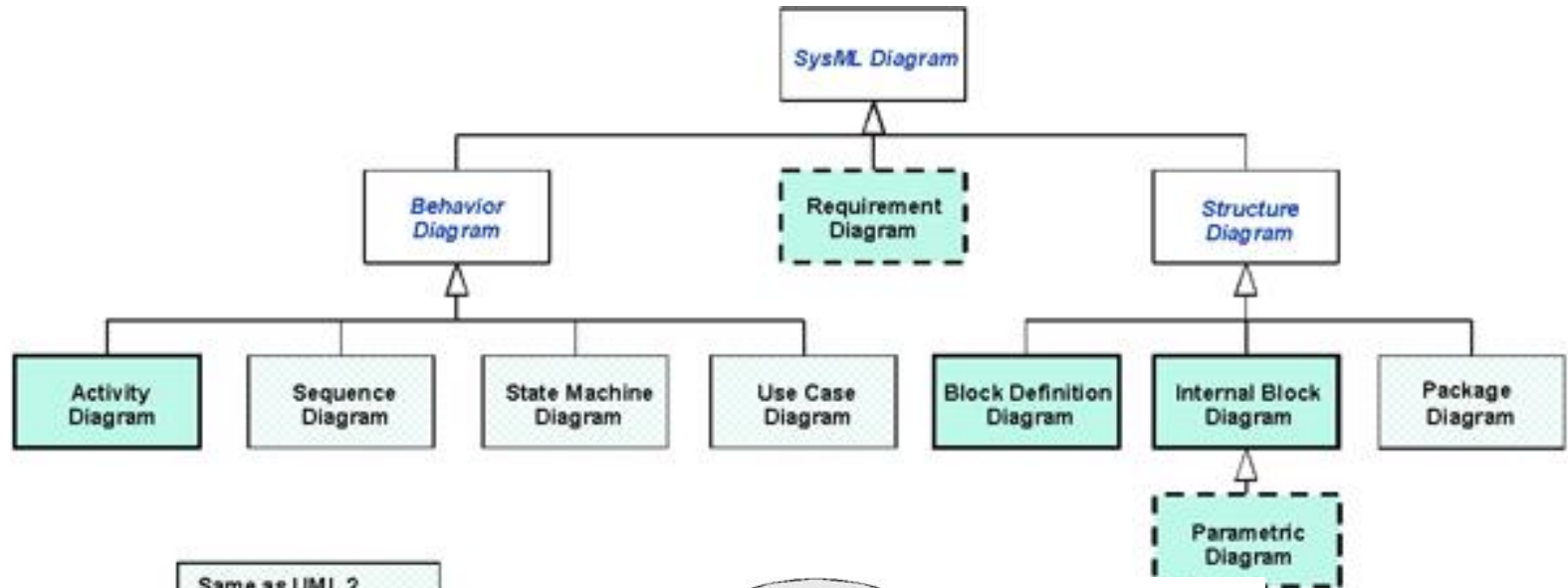
Software Architecture Description



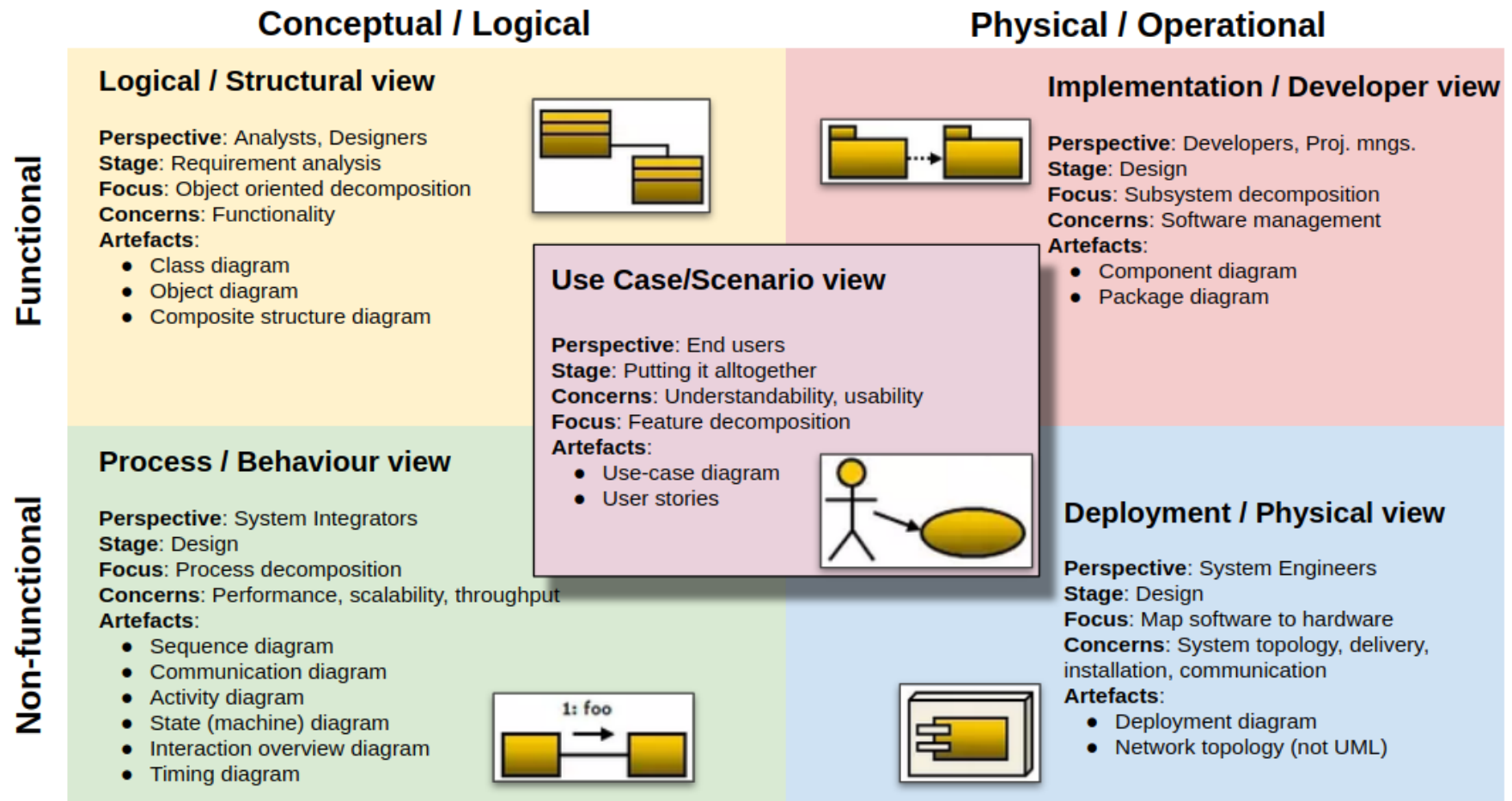
Software architecture is driven by

- Functional needs
 - Your requirements.
- Quality needs
 - Non-functional requirements, e.g., performance, security.
- Constraints
 - Technical, legal, economic.

OMG SysML/UML diagram



4+1 Architectural view model: UML diagrams



CMMI-Dev: Technical Solutions

- Purpose:
 - The focus of the technical solution **is to design and implement solutions according to the requirements.**
 - Develop solutions for design alternatives that:
 - feature current vs. new technologies,
 - COTS products, tools and so on

Always are more than one possible solution how to implement a requirement

https://en.wikipedia.org/wiki/Commercial_off-the-shelf

<https://searchdatacenter.techtarget.com/definition/COTS-MOTS-GOTS-and-NOTS>

CMMI-Dev: Technical Solutions

A couple things to keep in mind when developing the design:

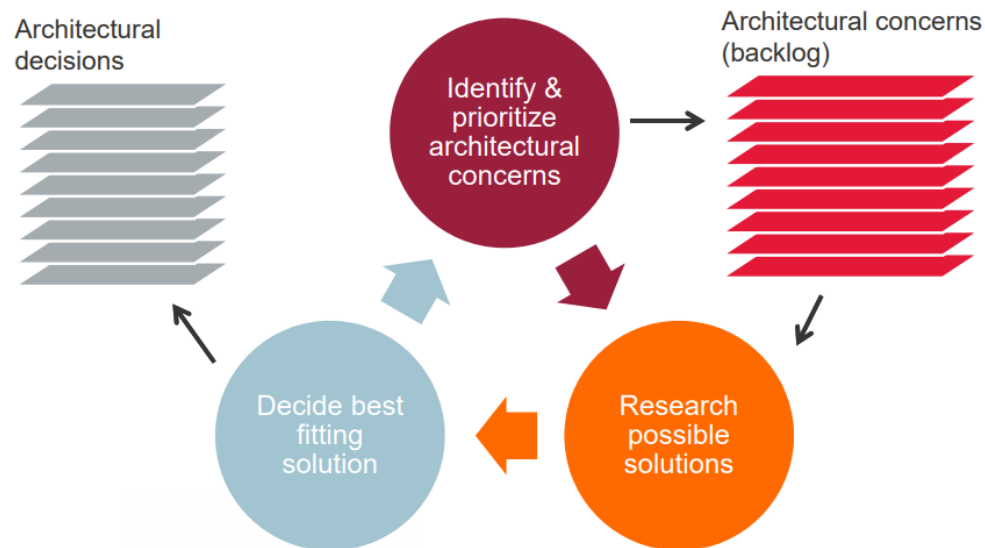
- Employ techniques and methods known for producing effective design (
 - prototypes,
 - structural models,
 - object oriented design,
 - entity relationship models
- Ensure that design standards are adhered to while preparing the design:
 - ISO 42010, TOGAF, Archimate, 4 + 1, etc

Architectural decisions

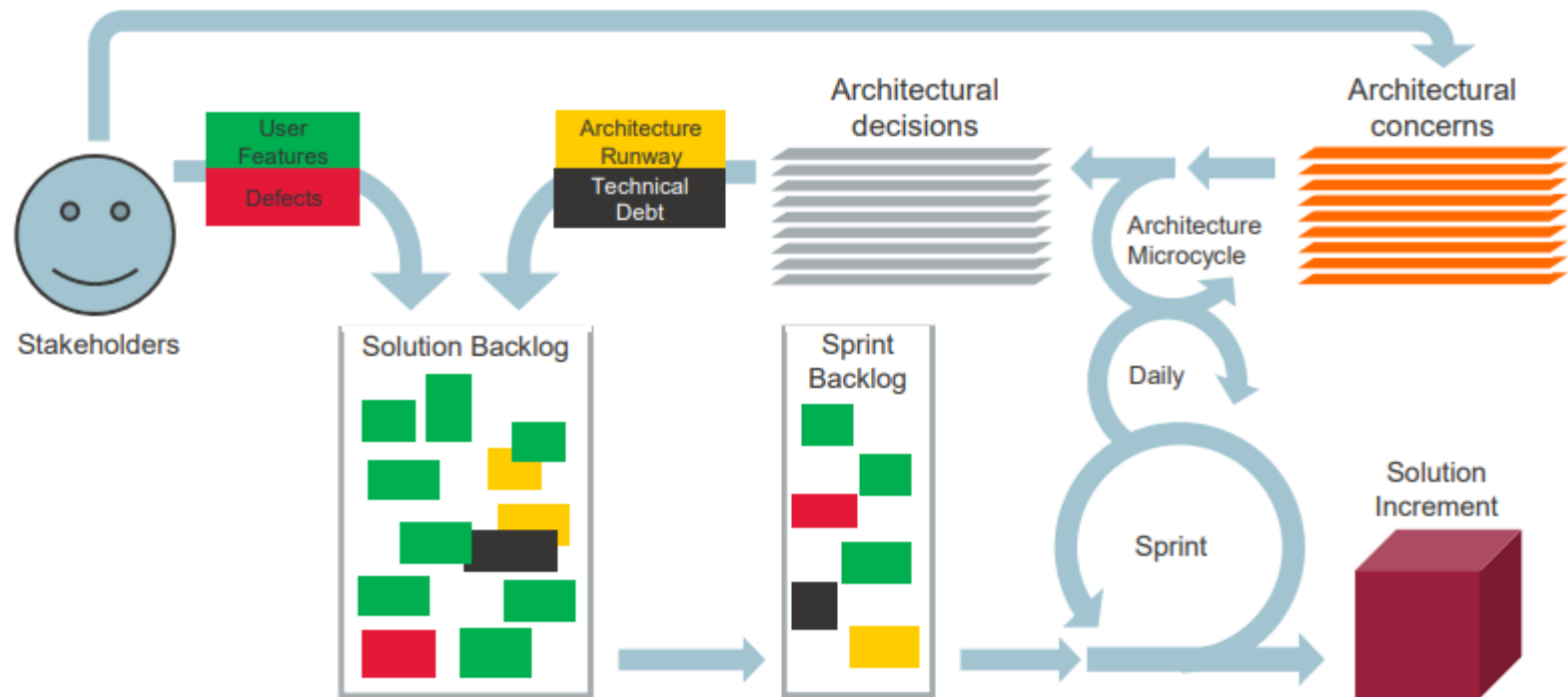
	Traditional	Agile
Development	 <p>Big Bang</p>	 <p>Continuous stream of improvements</p>
Architecture	 <p>Big Up-Front Design</p>	 <p>Continuous stream of Architectural Decisions</p>

What is Continuous Architecture?

Continuous Architecture is about using the appropriate tools to make the **right decisions** and support **Continuous Delivery, Continuous Integration and Continuous Testing**



SCRUM and the Architecture Microcycle



Continuous Architecture Principles

1. Architect Products - Not Just Solutions for Projects.
2. Focus on Quality Attributes - not on Functional Requirements.
3. Delay design decisions until they are absolutely necessary.
4. Architect for Change – Leverage “The Power of Small”.
5. Architect for Build, Test and Deploy.
6. Model the organization of your teams after the design of the system you are working on.

Software architecture evolution

Evolution of Software Architectures

