UNIVERSITY OF TWENTE.

Architecture starts when you carefully split a system into two subsystems.

There it begins...

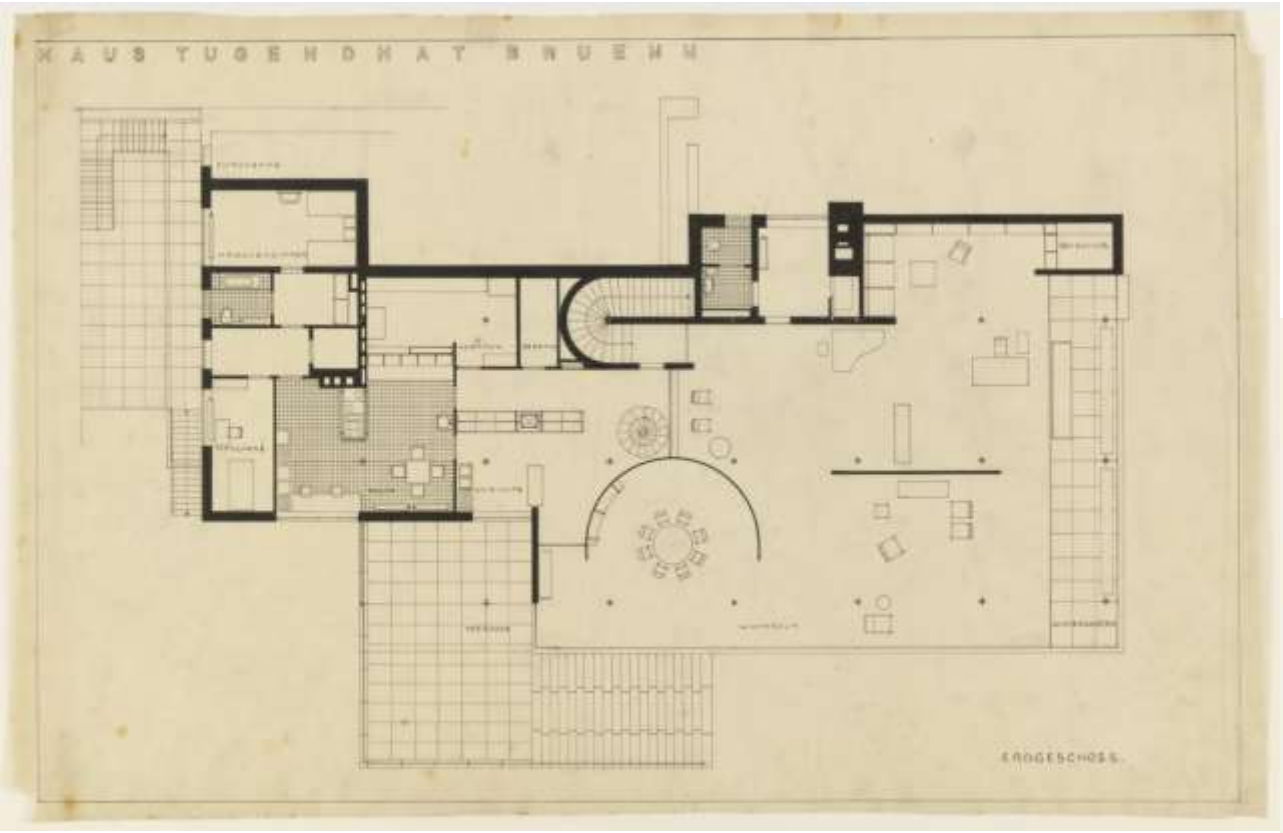
Maarten Bonnema
Full Professor of Systems Engineering & Multidisciplinary Design (SEMD)
University of Twente, the Netherlands

We'll be talking about *systems* architectures, but take inspiration from *building* architecture

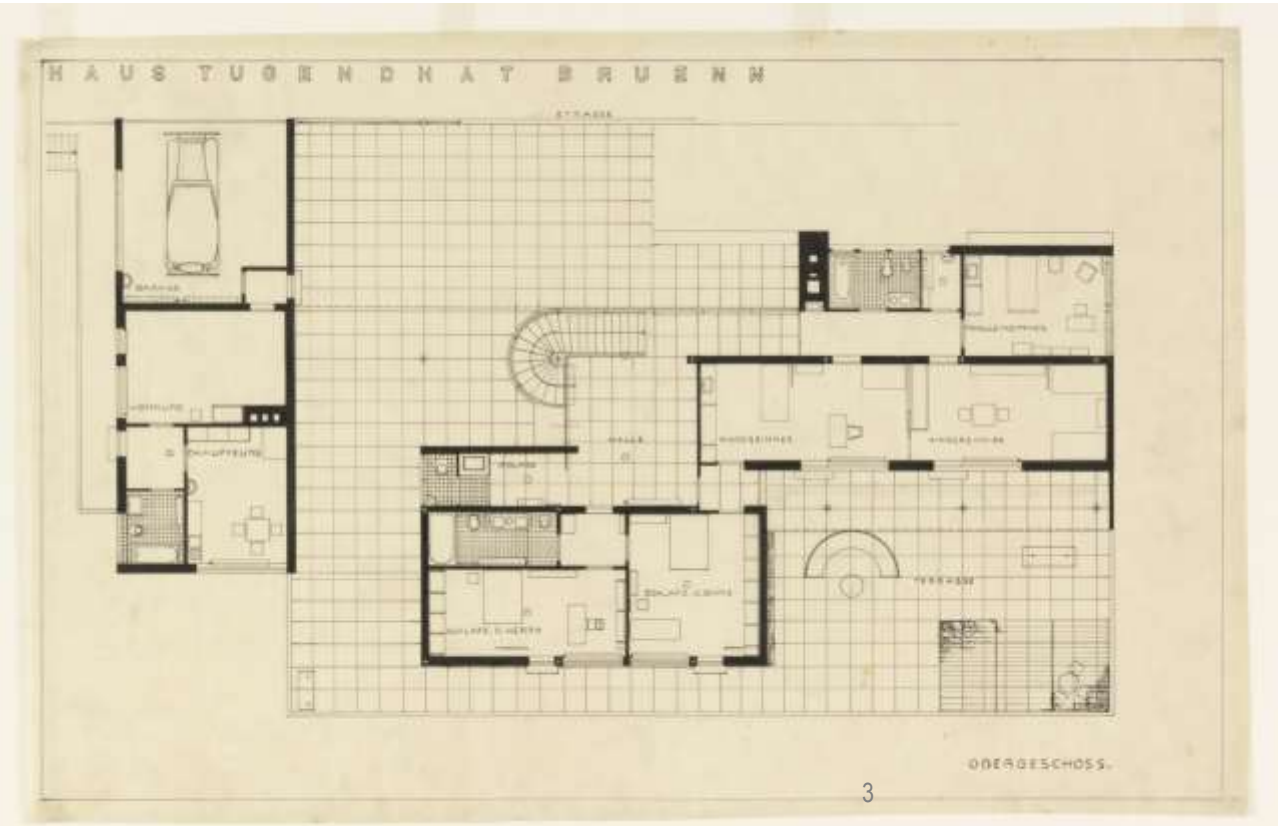
Ludwig Mies van der Rohe
German-American Architect
1886 - 1969

Architecture starts when you carefully split a system into two subsystems. There it begins...

BUILDING ARCHITECTURE



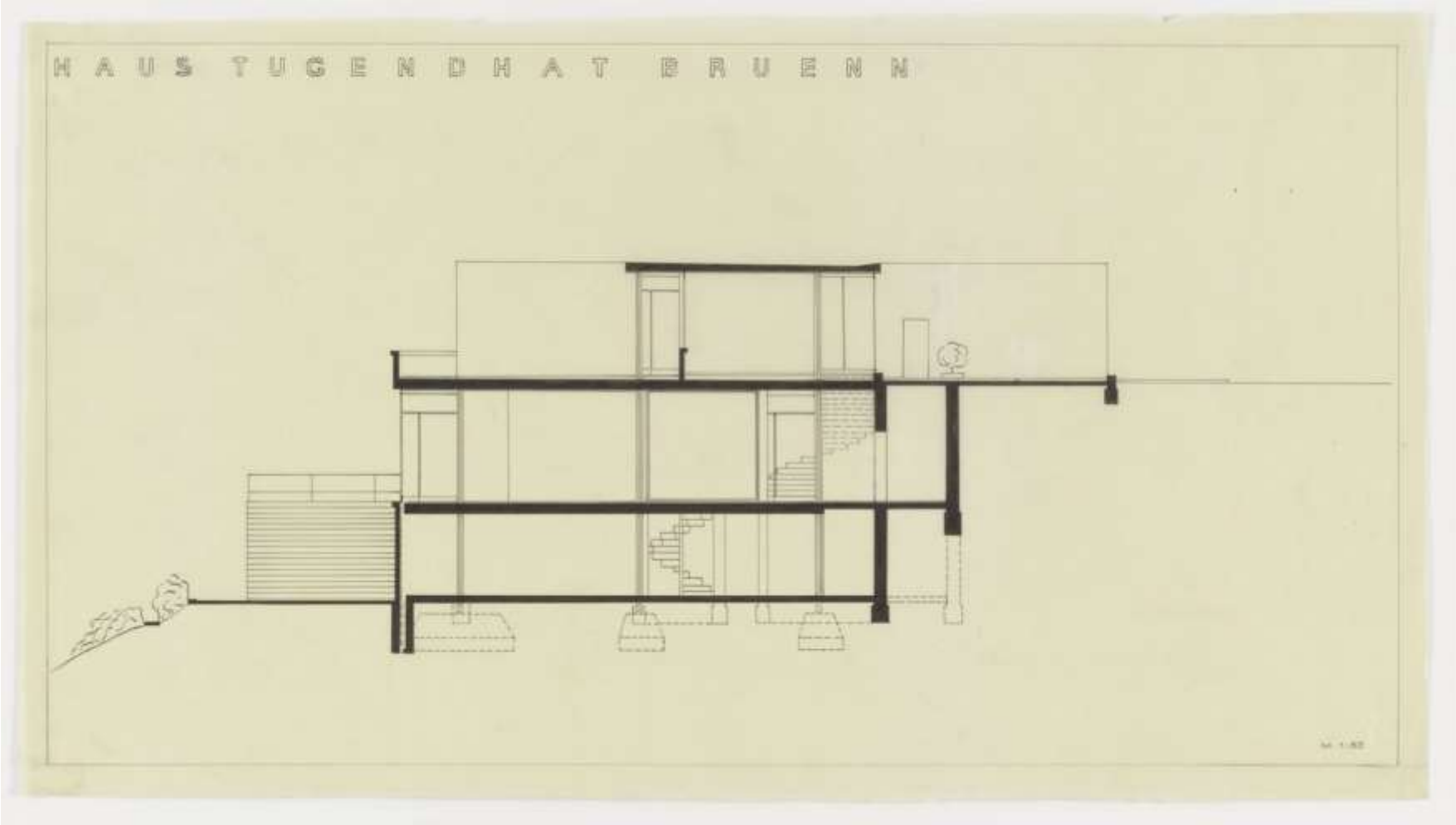
Functions, and their arrangement and allocation to main parts



The Tugendhat Haus in Brno, CZ
by Ludwig Mies van der Rohe
Source: Metalocus | Copyright by MoMA Archives

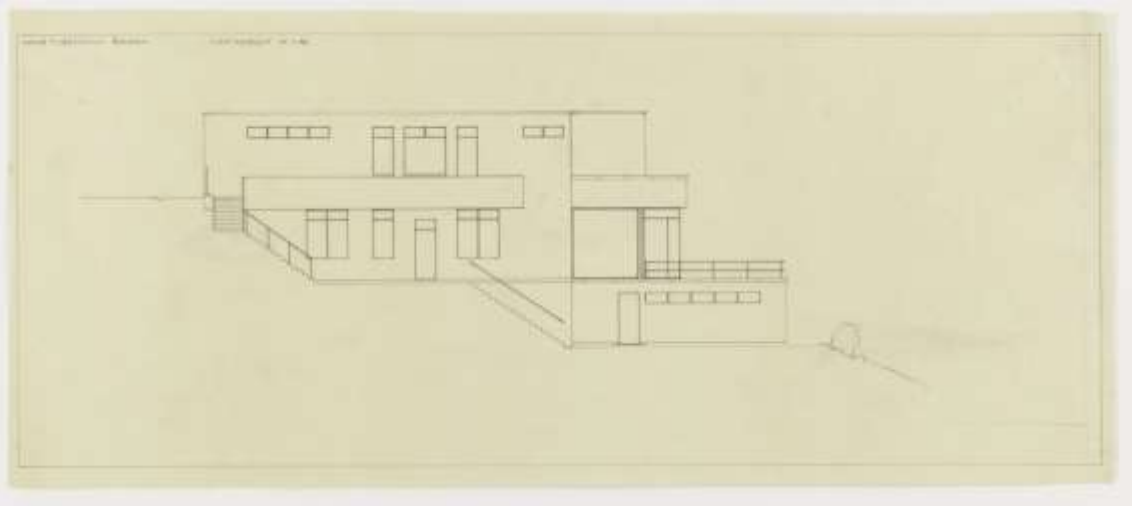
BUILDING ARCHITECTURE

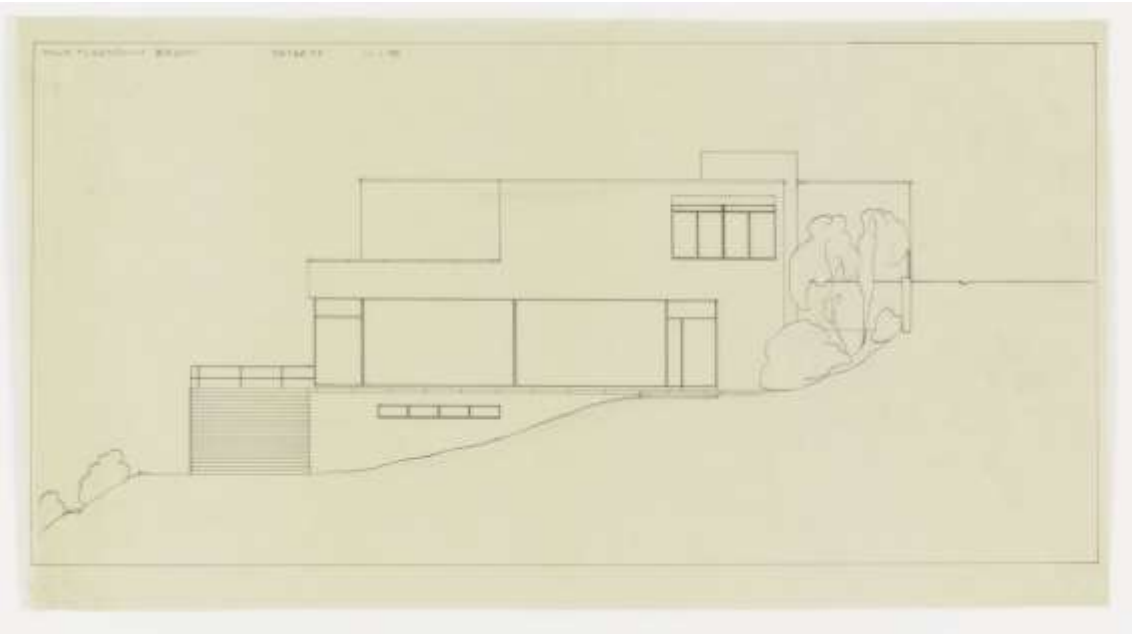
Interface between parts and to context and environment



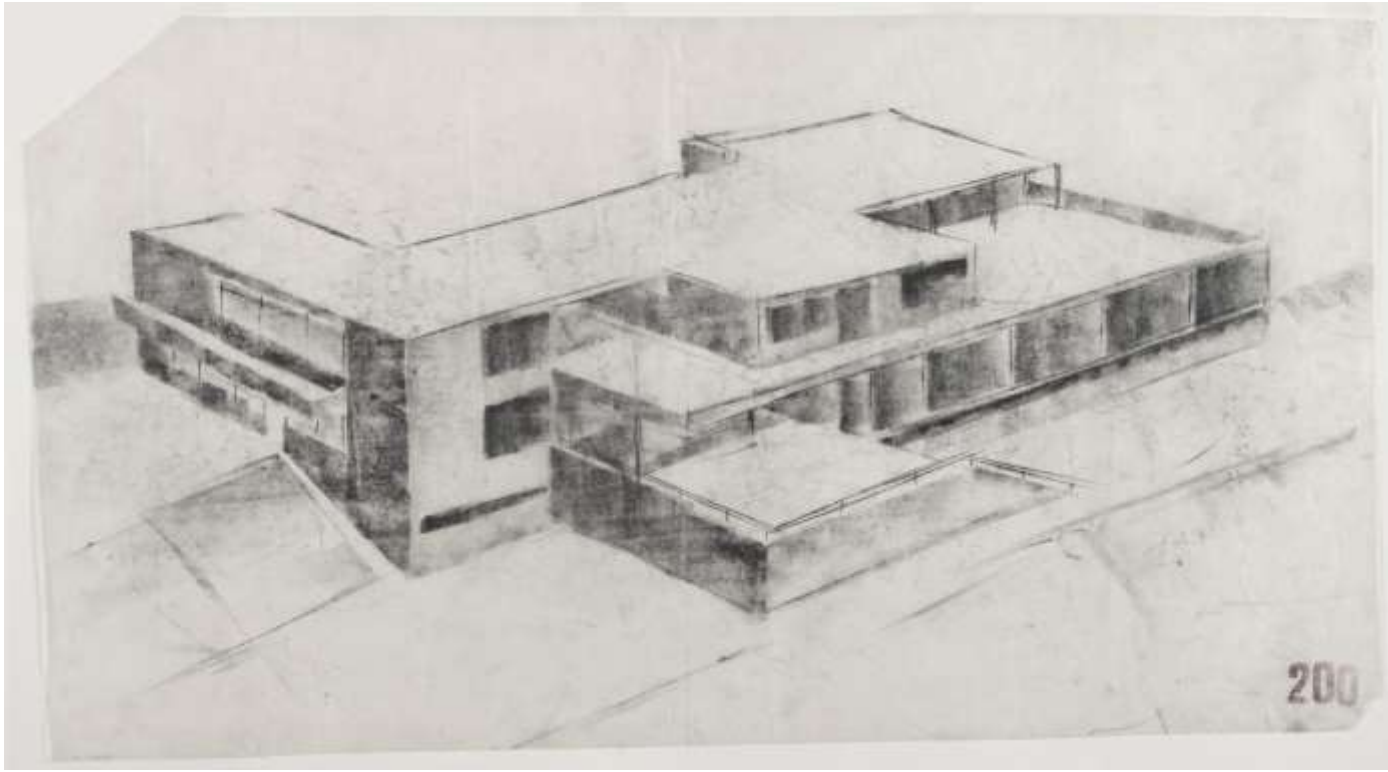
The Tugendhat Haus by Ludwig Mies van der Rohe Source: Metalocus | Copyright by MoMA Archives

BUILDING ARCHITECTURE





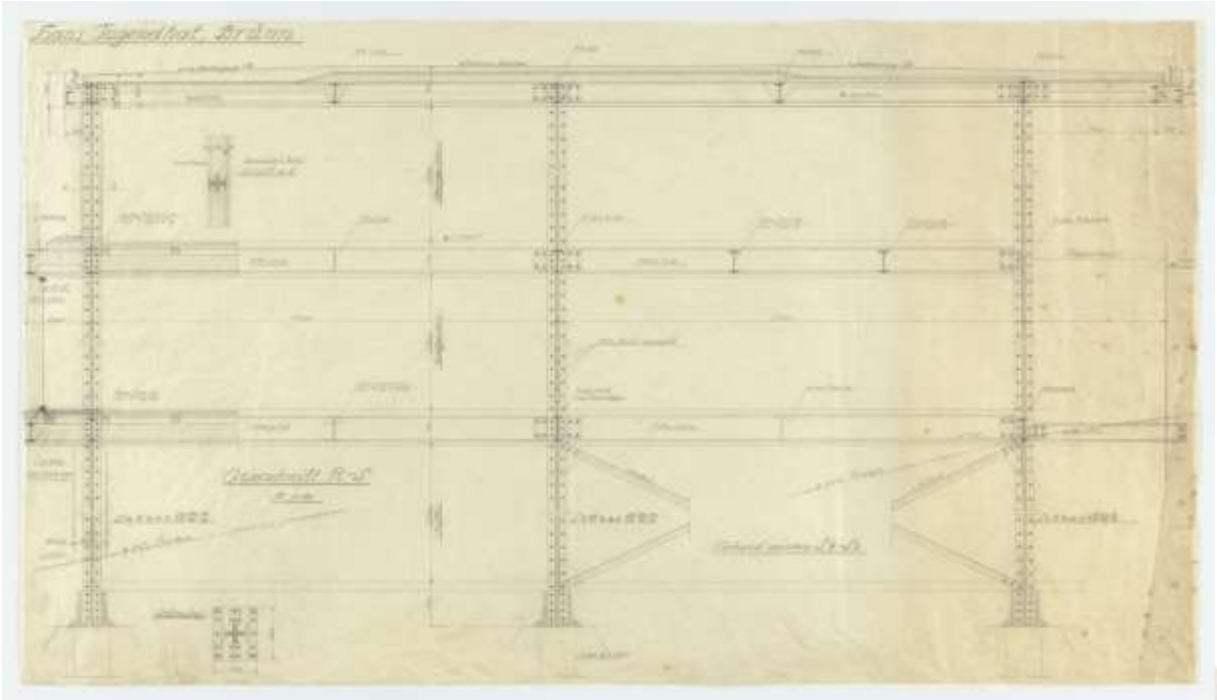
Big Picture and appearance to the (end) users



The Tugendhat Haus by Ludwig Mies van der Rohe

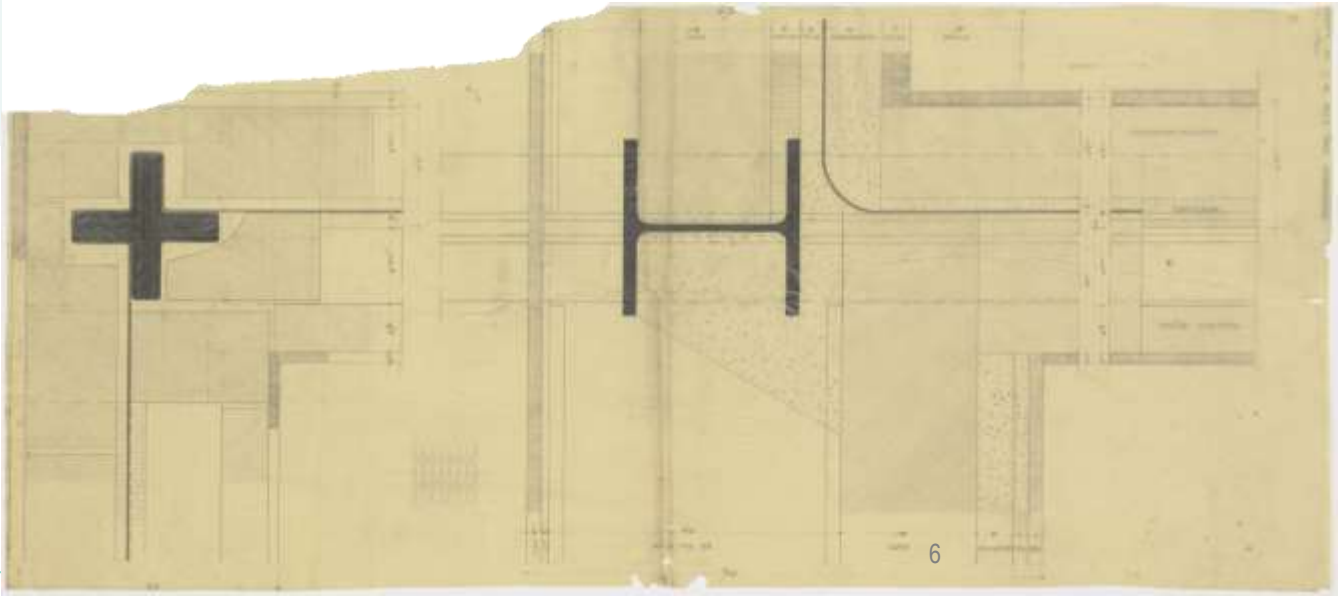
Source: Metalocus | Copyright by MoMA Archives

BUILDING ARCHITECTURE



The Tugendhat Haus by Ludwig Mies van der Rohe
Source: Metalocus | Copyright by MoMA Archives

Crucial details



DEFINITION OF SYSTEM ARCHITECTURE

So more than the thing we build!

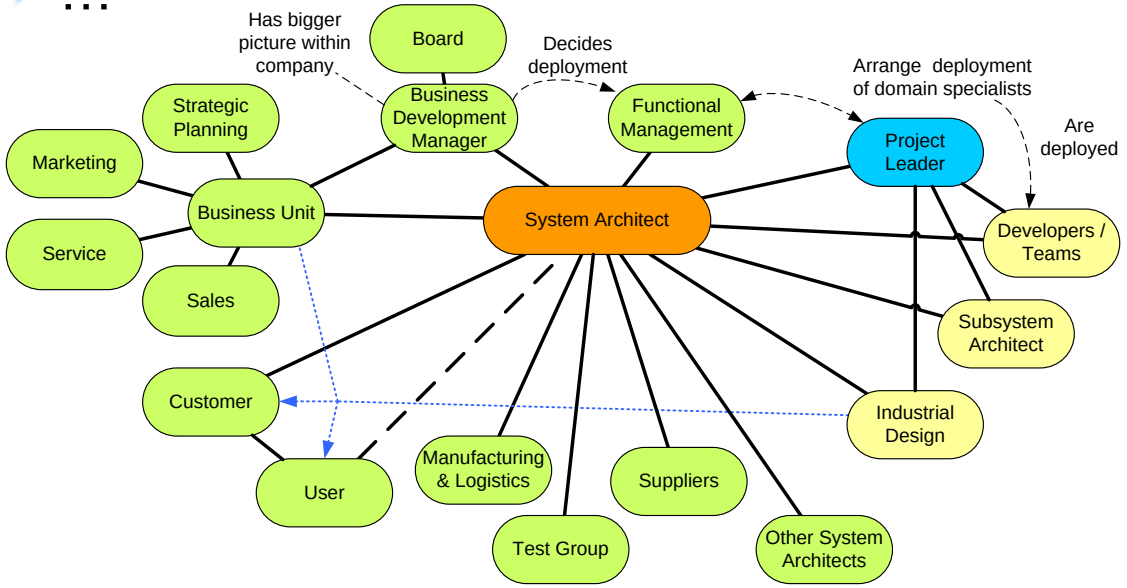
A **System architecture** defines the **parts** constituting a system and allocates the system's **functions** and **performance** over its parts, its user, its super system and the environment in order to **meet system requirements,** and so that **the parts, user(s) and super system** can be **integrated** to **cooperate constructively.**

Adapted from: Bonnema, G. M. (2008). FunKey Architecting - An Integrated Approach to System Architecting Using Functions, Key Drivers and System Budgets. PhD-thesis, University of Twente.

THE NEED FOR AN ARCHITECTURE

- When it becomes too large for one person/small team
- Division of work
- Identification of *Interfaces* (next talk)
- Re-use of solutions
- Create and maintain the Big Picture
- Support reasoning about (the impact of) changes

- Facilitating Discussions with:
  - Marketing
  - Engineering
  - (End-) users
  - …



Haveman, S. (2009). Project Buzz Tracker - supporting system architects in the future workspace. Master of Science thesis, University of Twente.

ARCHITECTURE AND BUSINESS

Bonnema, G. M. (2008). *FunKey Architecting: an integrated approach to system architecting using functions, key drivers and system budgets*. PhD Thesis. University of Twente.
https://research.utwente.nl/en/publications/funkey-architecting-an-integrated-approach-to-system-architecting

Development in architecture (function allocation) brings possibilities for earning money:

- Gillette
- Cut salad mix
- Coffee

- Different function allocations result in different business opportunities/models
- QFD or FunKey architecting helps here
- Mapping the contribution of functions to key-drives

1 kg beans €10 - €20

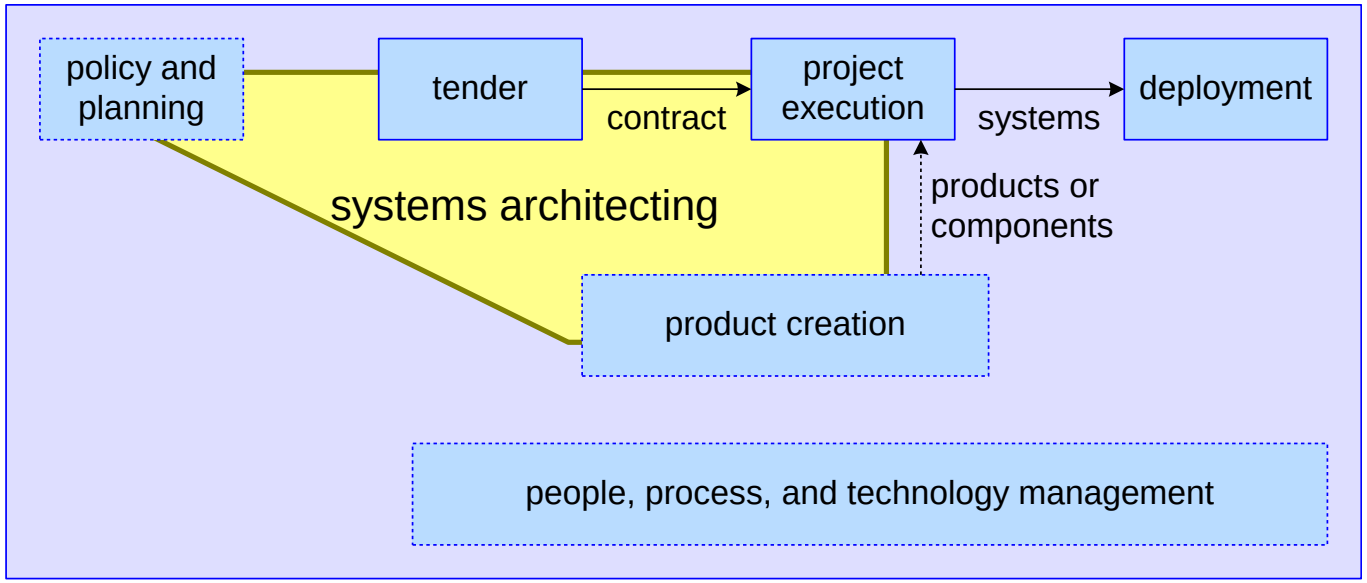
1 kg ground coffee €13 - €20

1 kg Nespresso coffee € 50 - €80

1 kg Senseo coffee € 20 - €25

ARCHITECTURE: THE BEST WAY TO PREDICT THE FUTURE IS TO CREATE IT (Peter Drucker)

- The System Architect should understand
  - the customer
  - your own company
  - the market
- And see opportunities and limitations



Source: Gerrit Muller, http://www.gaudisite.nl/SABP.html

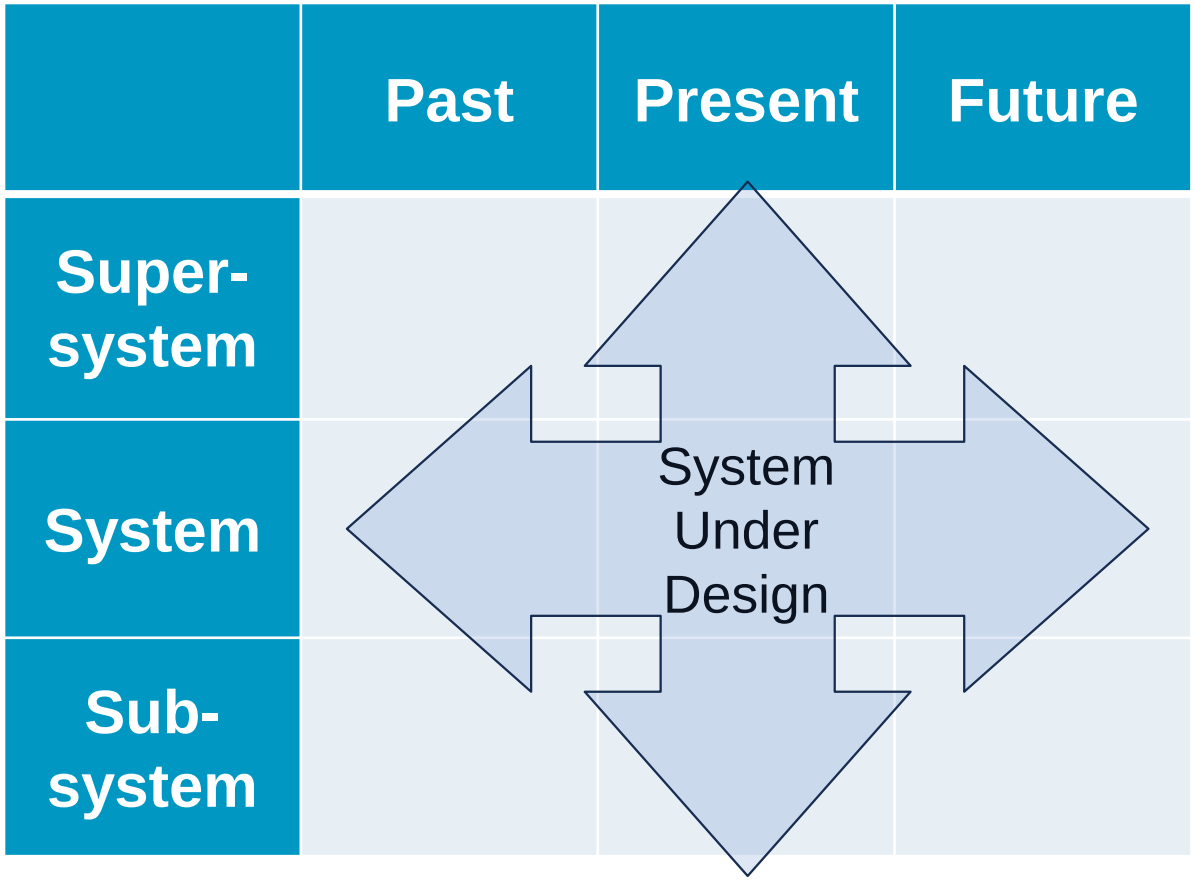
EXPLORING THE FUTURE: 9-WINDOW DIAGRAM

Put system central, instead of focussing on details

- Zoom out to see the whole
- Zoom in on crucial issues only

- Holistic thinking / Big picture
- Also time-wise

Pitfall: stay too general or superficial.



EXPLORING THE FUTURE: 9-WINDOW DIAGRAM

Put system central, instead of focussing on details

- Zoom out to see the whole
- Zoom in on crucial issues only

- Holistic thinking / Big picture
- Also time-wise

Pitfall: stay too general or superficial.

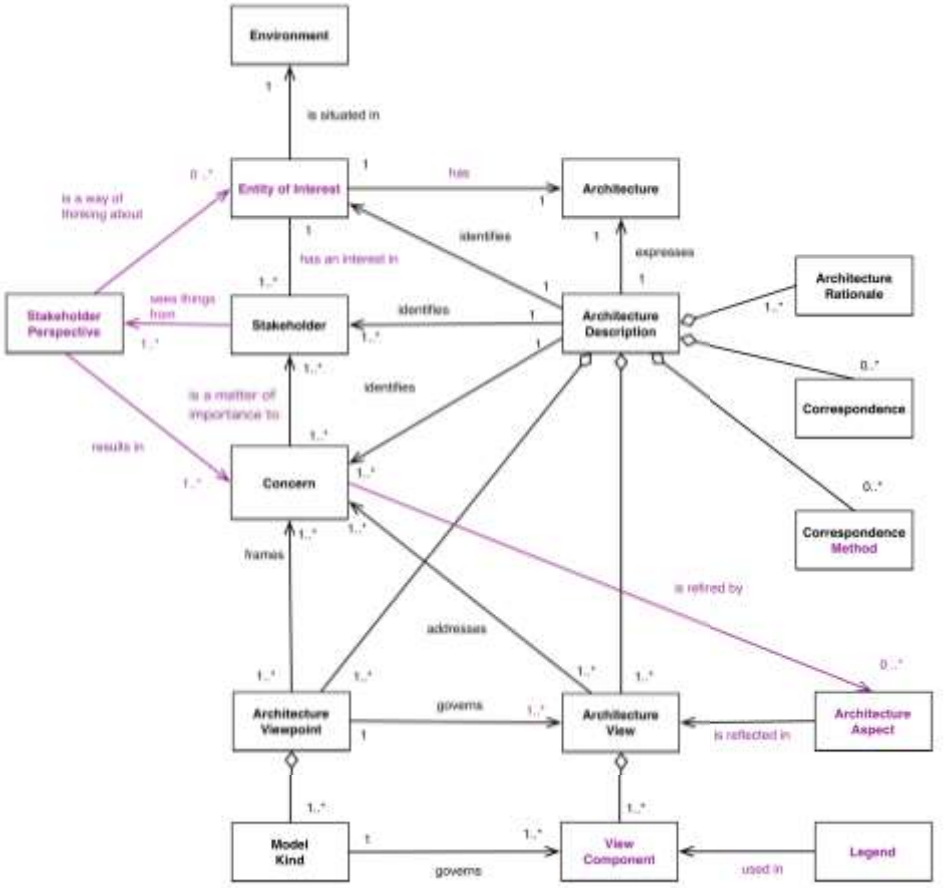
	Past	Present	Future
Super-system	Medical treatment in hospital	Medical Treatment by medication	Continuous diagnosing and treatment
System	Nurse	Smart Pill Box	Personalized delivery
Sub-system	Recipes, Procedures	Dose boxes, Reminder, Water	Ampuls, Doser, IV

INTERMEZZO: ARCHITECTURE AND REPRESENTATION

- Every system of some size *has* an architecture
- Not all architectures are made *explicit* or *tangible*
- An architecture *description* contains one or more *views*.
- Views are created from *viewpoints*

PRINCIPLE It is not possible to capture the functional features and quality properties of a complex system in a single comprehensible model that is understandable by, and of value to, its stakeholders.

Rozanski, N. and E. Woods (2012). Software Systems Architecture. Upper Saddle River, Addison Wesley.



ISO/IEC/IEEE 42010
http://www.iso-architecture.org/42010/cm/

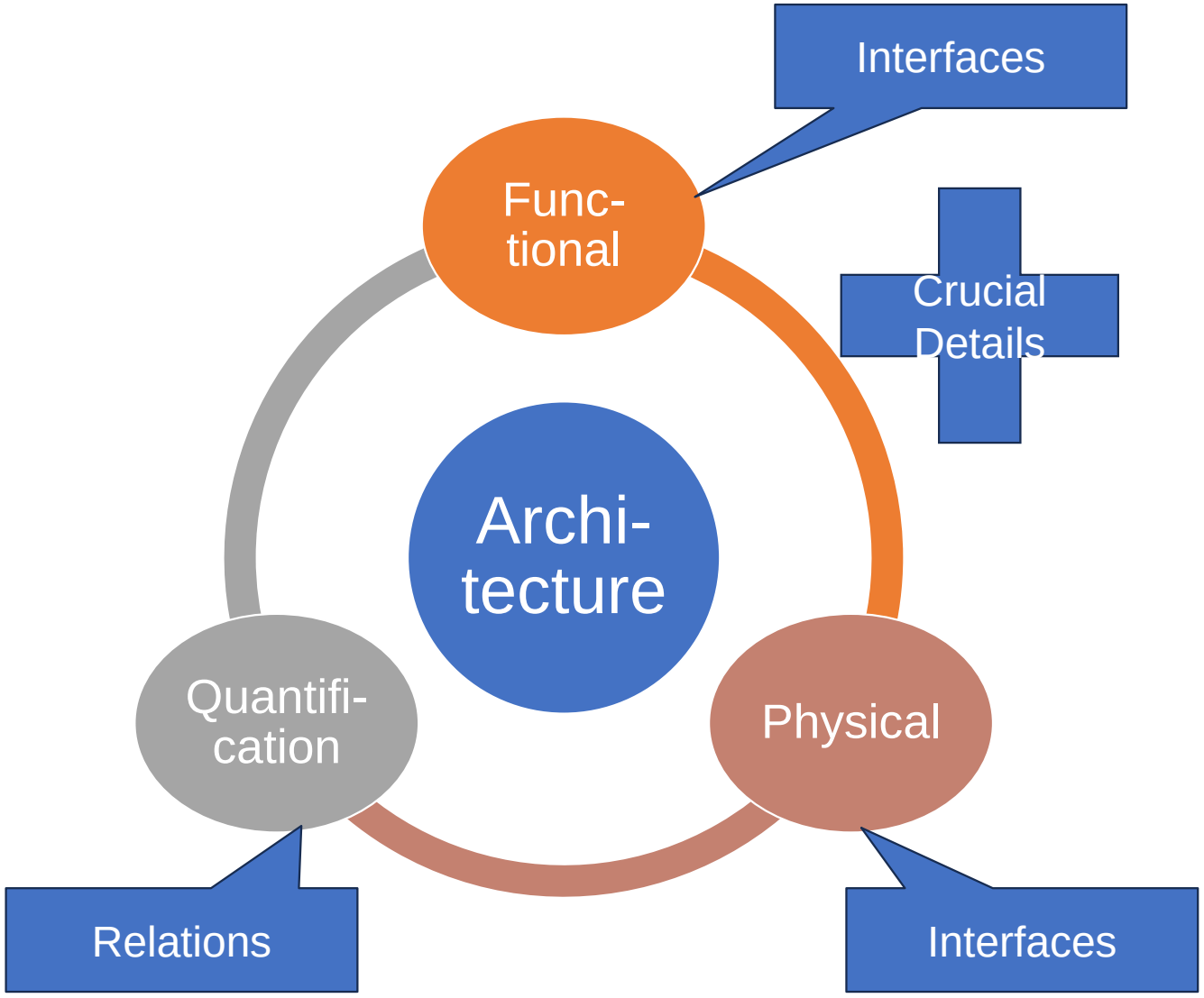
MULTIPLE VIEWS: WHICH ONES DO WE NEED?

We already saw:

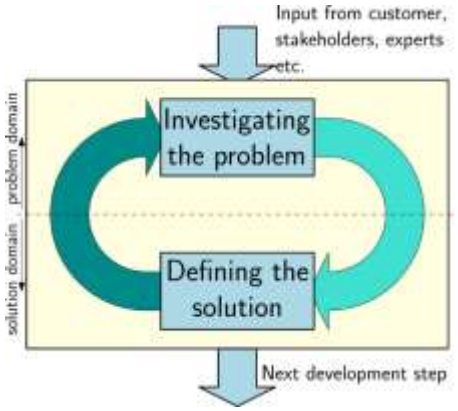
- Functions & function allocation
- Interfaces
- Big picture
- Crucial Details

Also needed:

- Physical
- Quantification



FUNCTIONAL VIEW

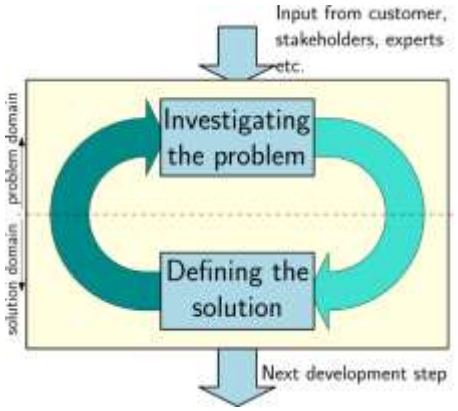


- What is it that the system does?
- How is the main functionality broken down into manageable chunks
- Functional interfaces
  - Energy
  - Matter
  - Information

→ N^2 diagram

- Use Verb + Substantive:
  - (to) Make coffee
  - (to) Transport persons
  - (to) Expose wafer
  - …
- Do not mix the function with the solution
  - Write “Provide mechanical power” instead of “Engine”

QUANTIFICATION VIEW



- How much?
  - Throughput
  - Accuracy
  - Power
  - Customer satisfaction
  - Business contribution
  - …

Competitive Engineering by Tom Gilb

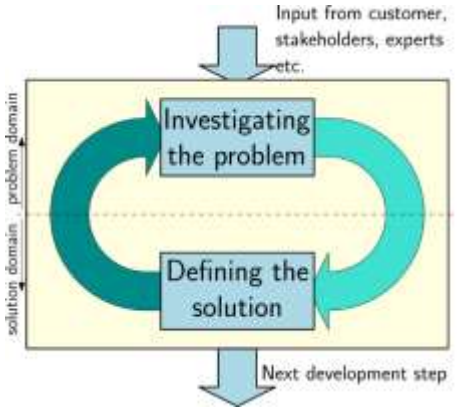
A Handbook For Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage
https://www.gilb.com/competitive-engineering

- At (complex) system level:
  Use **Key Drivers**

  A key driver is a generalized requirement representing stakeholders' main interest.

- As quantifiable as possible;
- 5 to 10 key drivers per system

QUANTIFICATION VIEW: Key Drivers

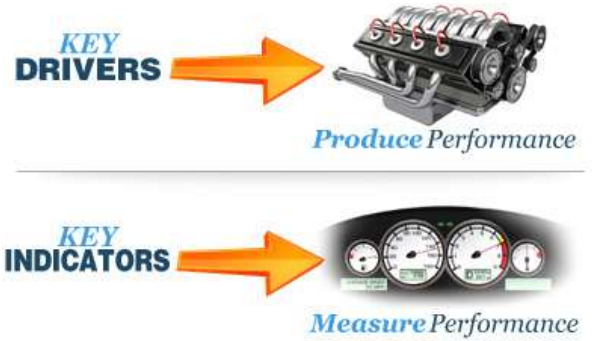


- Telephone system:
  - Number of simultaneous connections
- Commercial planes:
  - Cost per mile per passenger
  - Turn-around time (etc.)

  http://www.wired.com/science/discoveries/news/2006/05/70689
- EV:
  - Range;
  - Charging speed (km/h);
  - Total cost of ownership (TCO)

Defining good key drivers takes time, effort and requires insight

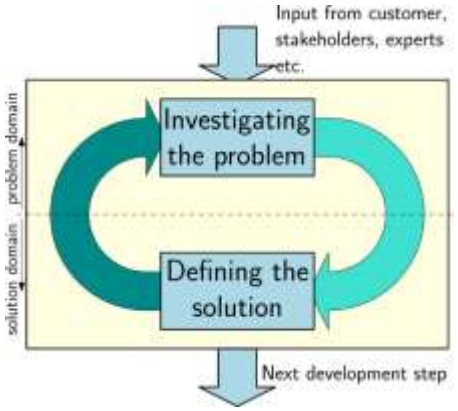
Key drivers *produce* performance and **key (performance) indicators** *measure* performance.
Well-designed indicators provide feedback about your business, but they don't deliver results – drivers do.
(http://www.focus3organizationalculture.com/the-performance-driver-model/)

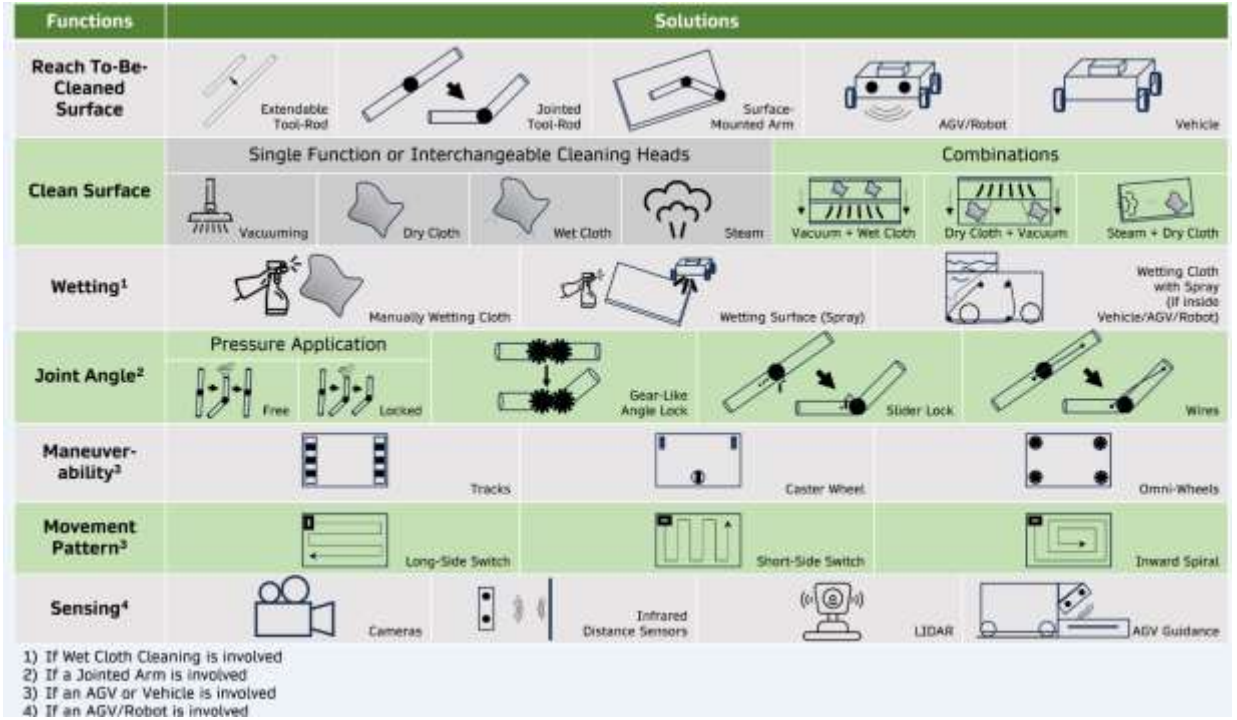


https://centralohiobwpc.com/performance-driver-model/

PHYSICAL VIEW

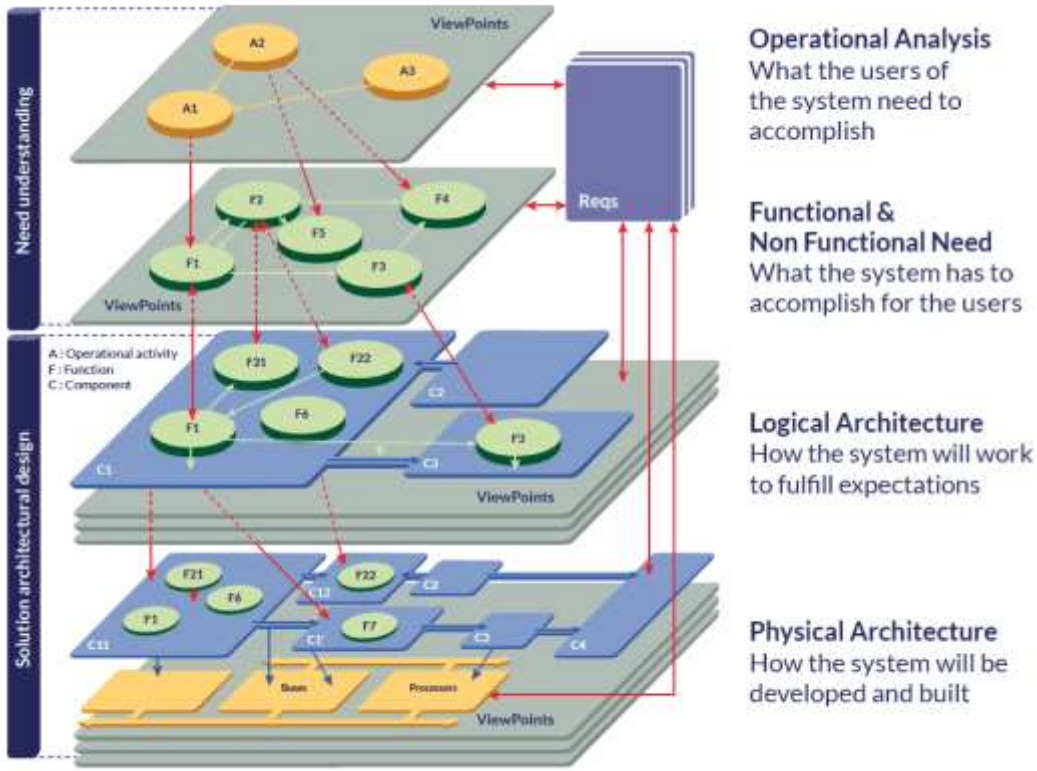
- Allocation of functions to subsystems
- Answer to **HOW**?
- For the crucial ones: operating principle and potential embodiment





van Zijl, R. (2024). *Adopting CATs: Development of Cleaner Assisting Technologies for a Gradual Technology Adoption and Implementation Process in the Cleaning Sector*. [EngD Thesis, University of Twente]. University of Twente. https://research.utwente.nl/en/publications/adopting-cats-development-of-cleaner-assisting-technologies-for-a

LOGICAL VIEW – ???



Arcadia Architecture Layers

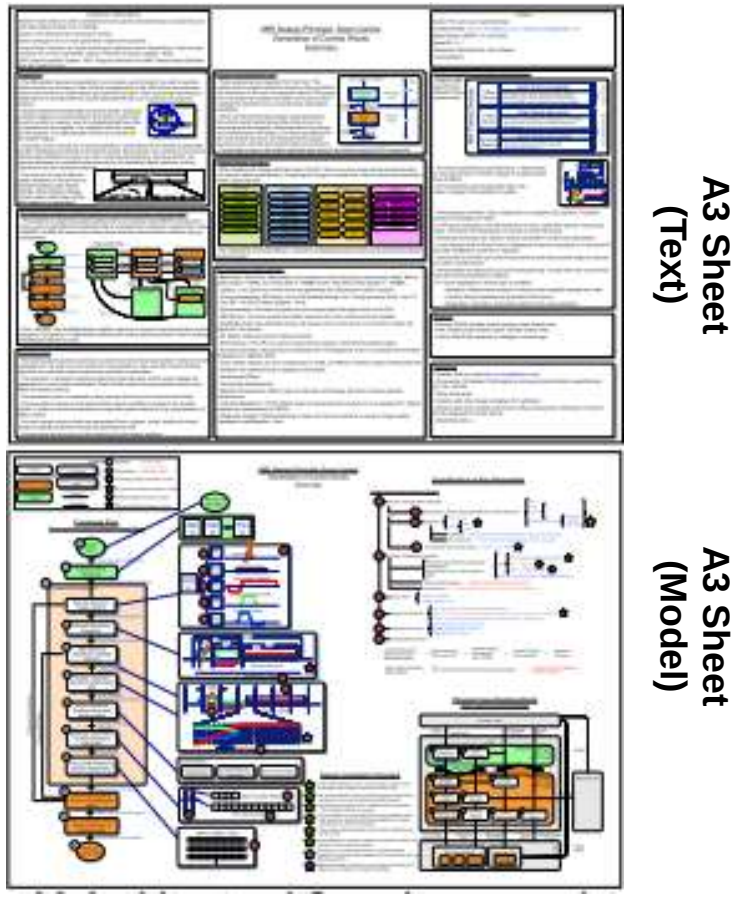
source: https://www.ppi-int.com/systems-engineering-newsjournal/ppi-syen-70/

- "Layer" "between" Functional and Physical
- Logical architecture may make sense in some cases (SW)
- Good architecture should be sound from all viewpoints: functional, physical, quantification

I often see discussion whether a view or an item is "functional", "logical" or "physical"; distracting from the real architecting work…

HOW IT ALL FITS TOGETHER: A3 ARCHITECTURE OVERVIEWS

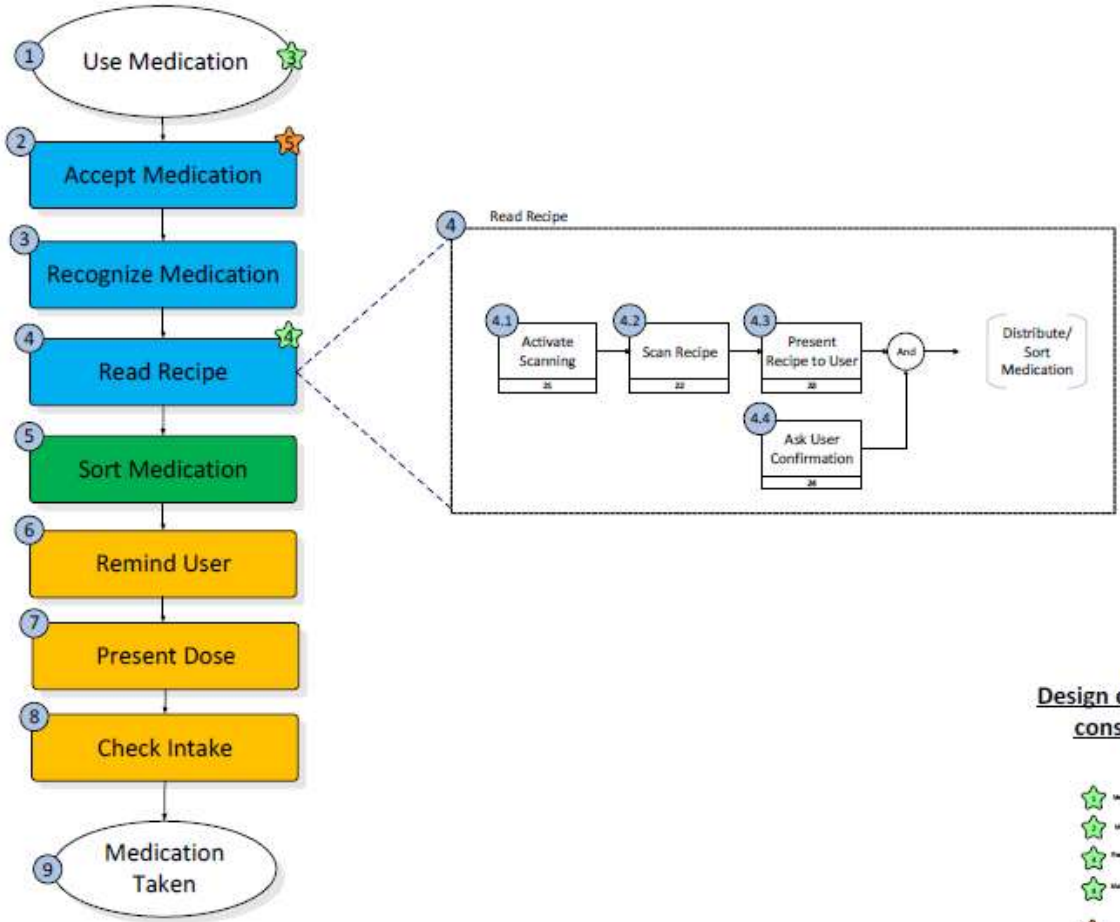
- 2 Sides of an A3 sheet
- Model and Text
- Structured and consistent to improve readability
- Interlinked visual representations to support understanding and discussion
  - Functional
  - Physical
  - Quantification



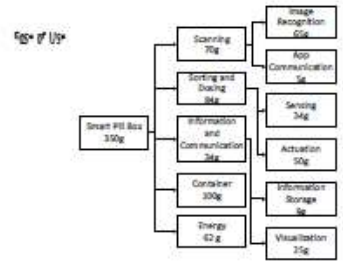
A3 Architectural Overview example

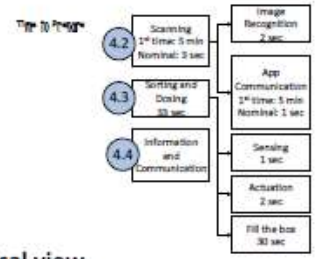
- More information: [A3AO.eu](A3AO.eu)
  (the A3 Architecture Overviews resource site)

The A3 Architecture Overview of a Smart Pillbox



Quantification of Key Parameters

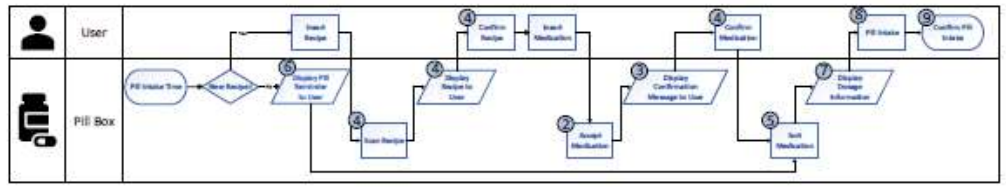




Physical view

Design decisions / constraints

User	Button	Pills	Button Recipe		Button		Button
	Accept Medication	Data: Is pill insertion finished?					
Data: Visual Information		**Recognize Medication**		Data: Medication			
Data: Visual Information (Confirmation Request)			**Read Recipe**	Data: Recipe			
Data: Visual Information (Confirmation Request)				**Sort Medication**	Data: Dosage	Data: Dosage	
Sound/ Vibration					**Remind User**		
Pills						**Present Dose**	
Data: Visual Information (Confirmation Request)				Data: Remaining Pills	Data: Were the pills taken?		**Check Intake**



ARCHITECTURE vs. TECHNICAL CONCEPT

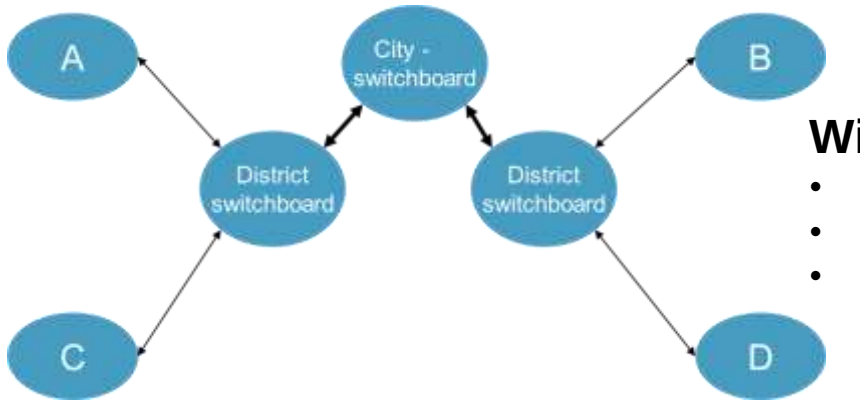
What and where

- Architecture **divides** the functions and performance requirements, yet leaves the implementation (largely) open
- (Technical) Concept determines the **implementation** of the functions (operation and embodiment)

How

Ref. building architecture:

- The architect designs the overall idea and appearance
- The structural engineer develops the technical details and solutions



Wired telephony architecture

- Developed end of the 1800's
- Still relevant today
- Technical implementations have evolved drastically

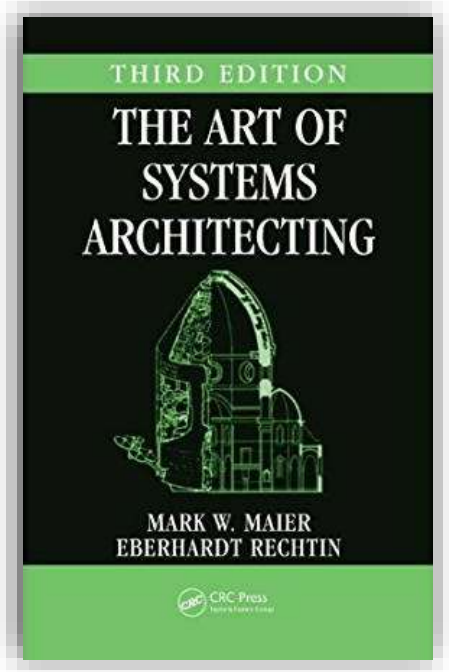
HOW TO CREATE AN ARCHITECTURE – 7 STEPS

1. Context diagram and 9-window diagram to understand the system under design.
2. Stakeholder analysis
3. Define Key-drivers (on system level) and requirements (on lower levels)
4. Inventory functions and interfaces (functional N^2)
5. FunKey (or QFD or similar) to understand the value of the system for the stakeholders, draft Budgets for main key drivers
6. Compare architectures using modular N^2 diagrams, select the preferred one.
7. Put chosen architecture in an A3AO

In reality way less clear-cut

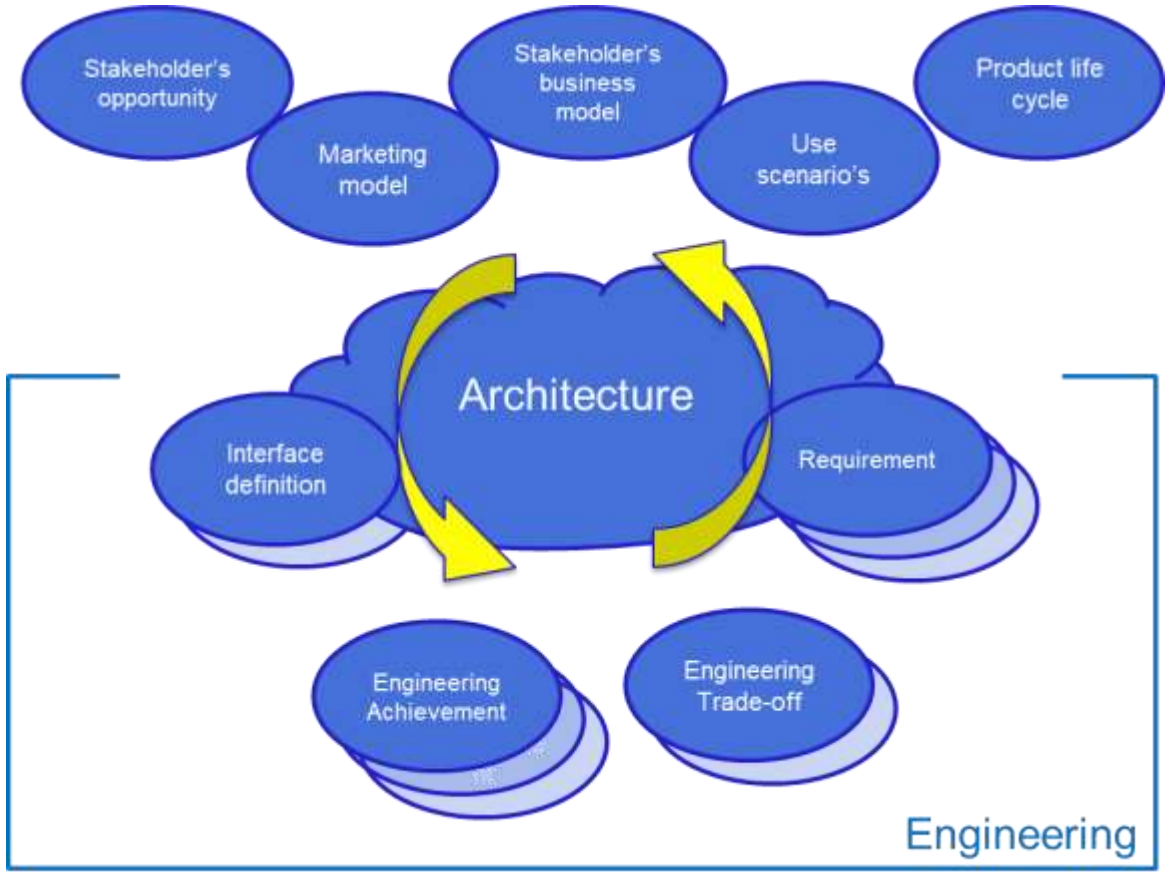
HEURISTICS FROM: MAIER AND RECHTIN "THE ART OF SYSTEMS ARCHITECTING

- In architecting a new [aerospace] system, by the time of the first design review, performance, cost and schedule have been predetermined.
- Group elements that are strongly related to each other, separate elements that are unrelated.
- Subsystem interfaces should be drawn so that each subsystem can be implemented independently of the specific implementation of the subsystems to which it interfaces.
- Do not slice through regions where high rates of information exchange are required.



WRAP-UP

- An Architecture connects worlds:
  - Customer
  - Business
  - Management
  - Engineering
- ***Balance*** is the key-word



TRAINTRACK STRAIGHT ACROSS USA

Plus running material

Architecture 1:

- SS1: Banking & Ballast
- SS2: Rails
- SS3: Overhead wire
- SS4: Safety systems

Architecture 2:

- SS1: Track West section
- SS2: Track Mid-West section
- SS3: Track Mid-East
- SS4: Track East
- SS5: Safety systems

Thank You!

Contact: Maarten Bonnema, g.m.bonnema@utwente.nl
SEMD-chair: www.utwente.nl/SEMD