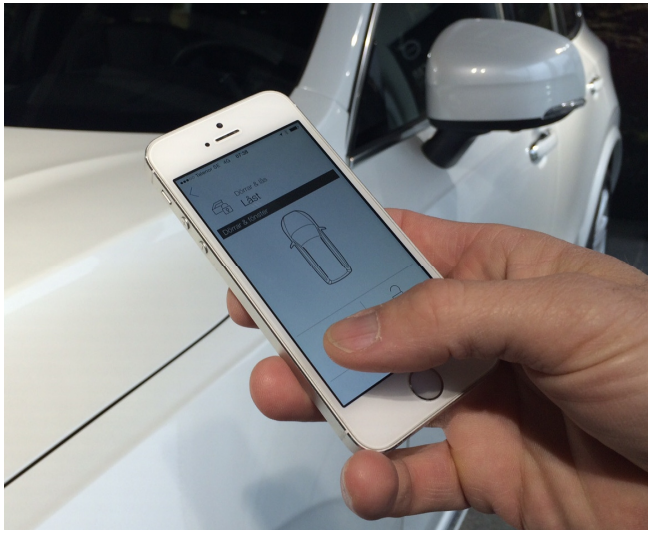

CONNECTING SYSTEMS ENGINEERING AND HUMAN FACTORS THROUGH MBSE
VOLVO
ROBERT NILSSON
VOLVO CAR CORPORATION
LARS-OLA BLIGÅRD
CHALMERS UNIVERSITY OF TECHNOLOGY

A STORY OF CONNECTION

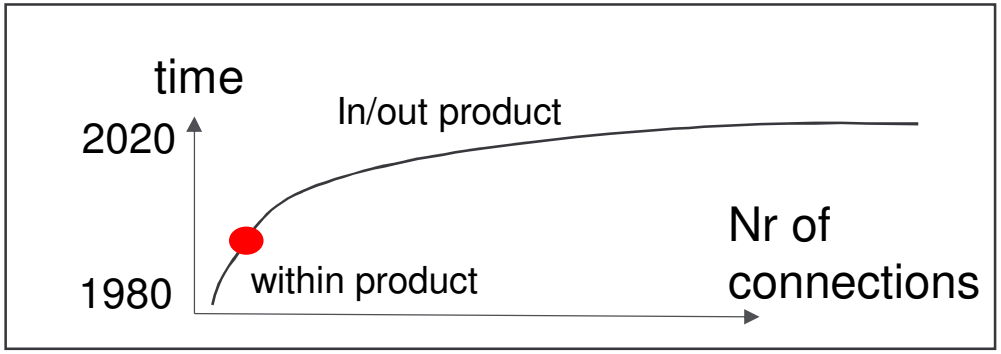
"reflecting on security & comfort"

Traditional mechanical

1st connected version of key



2nd connected version of key



HUMAN FACTORS

"Ergonomics (or human factors) is the scientific discipline concerned with the understanding of interactions among humans and other elements of a system, and the profession that applies theory, principles, data and methods to design in order to optimize human well-being and overall system performance."

International Ergonomic Society

http://www.iea.cc/whats/index.html [150413]

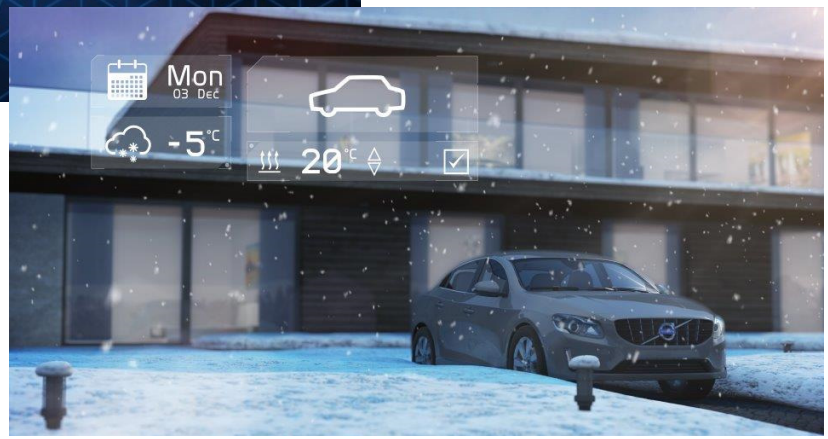
SYSTEMS ENGINEERING

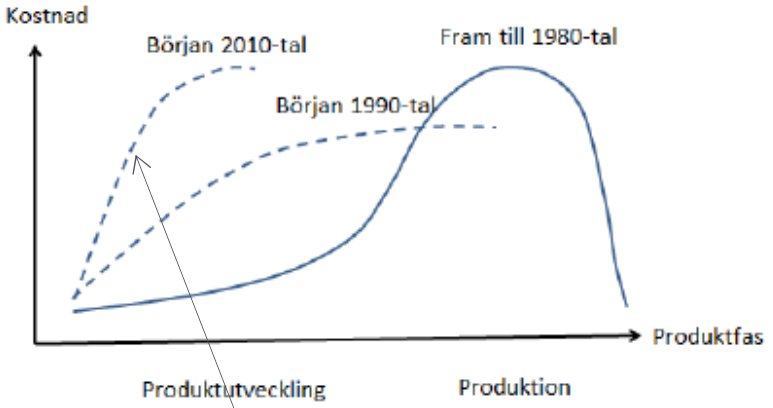
”Systems Engineering is an engineering discipline whose responsibility is creating and executing an interdisciplinary process to ensure that the customer and stakeholder's needs are satisfied in a high quality, trustworthy, cost efficient and schedule compliant manner throughout a system's entire life cycle.”

http://www.incose.org/AboutSE/WhatIsSE

WHY IS SYSTEMS ENGINEERING OF INTEREST

Increased complexity





Bligård, L-O., Nilsson, R., (2015), PU²B-modellen - En introduktion till Model Based Systems Engineering (MBSE) utifrån operatörscentrerad systemdesign, Chalmers – based on: Wickelgren, M. (2005).

“Demands structure”

Goal:
Conscious decisions during product development

THE ROLE OF HUMAN FACTORS IN SHORT

INCOSE definition of a system:

" A system is a construct or collection of different elements that together produce results not obtainable by the elements alone. The elements, or parts, can include people, hardware, software, facilities, policies, and documents; that is, all things required to produce systems-level results. The results include system level qualities, properties, characteristics, functions, behavior and performance. The value added by the system as a whole, beyond that contributed independently by the parts, is primarily created by the relationship among the parts; that is, how they are interconnected (Rechtin, 2000)."

Explicit HF

Implicit HF

"Competitive advantages accrues to firms that can bring a technology into the market place in a product that meets the customer needs efficiently and in timely manner"
(Wickelgren, M., 2005, 2nd source ref: Clark and Fujimoto, 1991, p.96)

1. If we are to build future systems smart we need to understand the users.

2. Human Factors = Lean

TRADITIONAL VS FUTURE PRODUCT DEVELOPMENT (PD)

Life cycle stages:

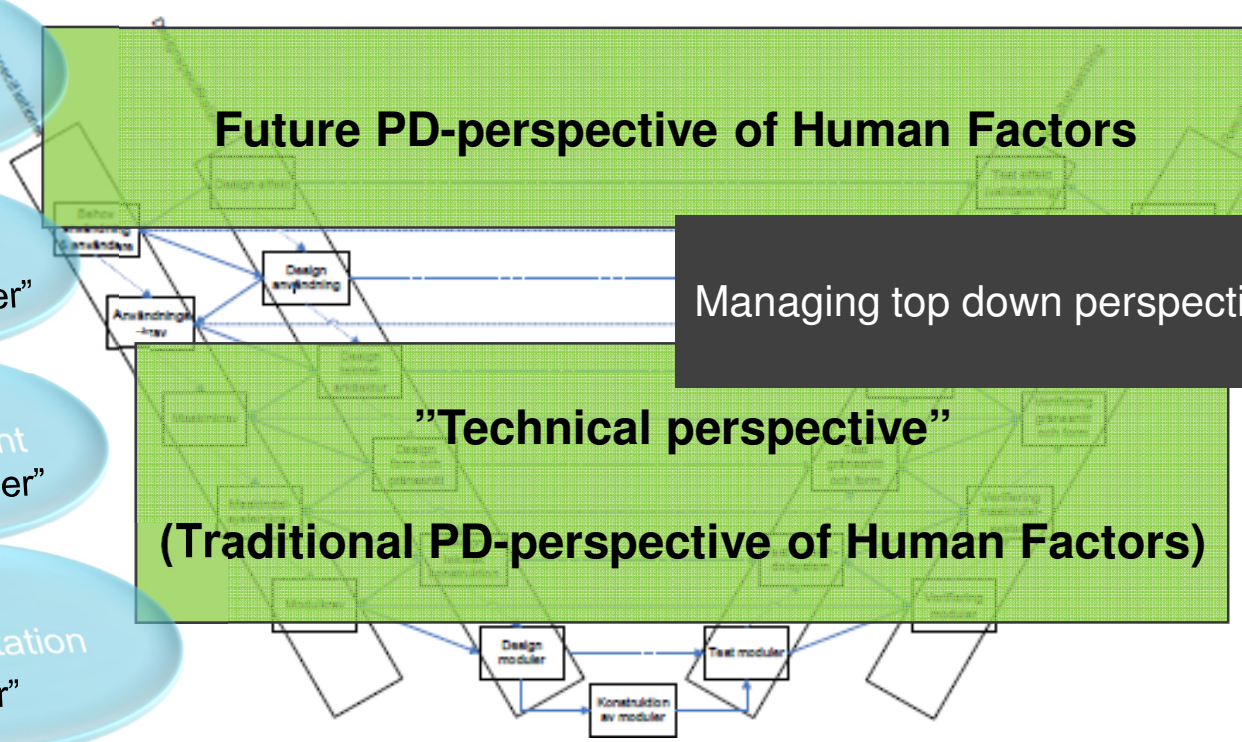
- Exploratory research "What to do"
- Concept "What to deliver"
- Development "How to deliver"
- Implementation "Deliver"

Future PD-perspective of Human Factors

Managing top down perspective enables LEAN.

"Technical perspective"

(Traditional PD-perspective of Human Factors)



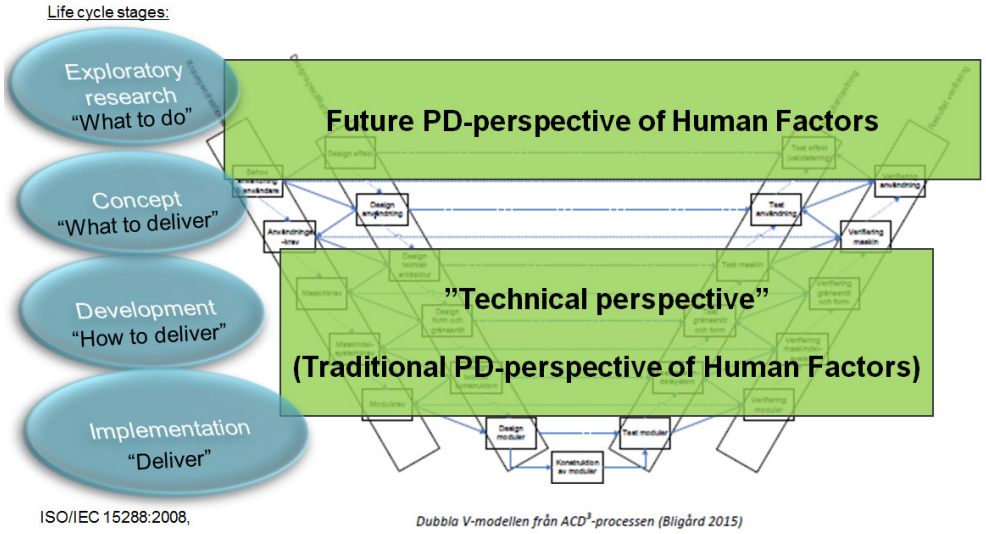
Dubbla V-modellen från ACD³-processen (Bligård 2015)

ISO/IEC 15288:2008,
Systems engineering handbook, V 3.2.2, p25

CO-EXISTENCE...

Methods for context analysis

Methods for user experience



ISO/IEC 15288:2008,
Systems engineering handbook, V 3.2.2, p25

Dubbla V-modellen från ACD³-processen (Bligård 2015)

Life cycle perspective

Complete system understanding

Example of Human Factors tool:

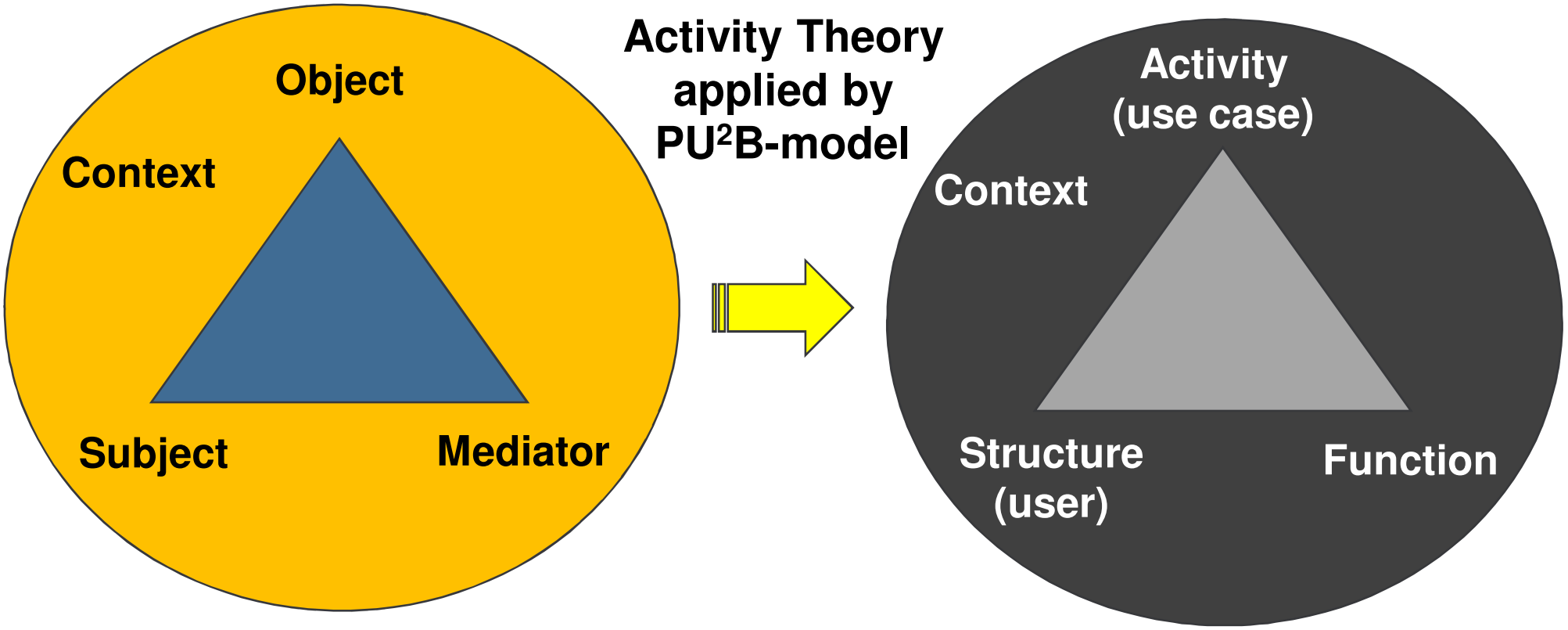
The method of Hierarchical Task Analysis (HTA) was developed in the 1960s. Initially, the method was developed for training in process control tasks performed in steel and petrochemical industries.

Today the HTA is used in many different contexts like for example interface design and error analysis.

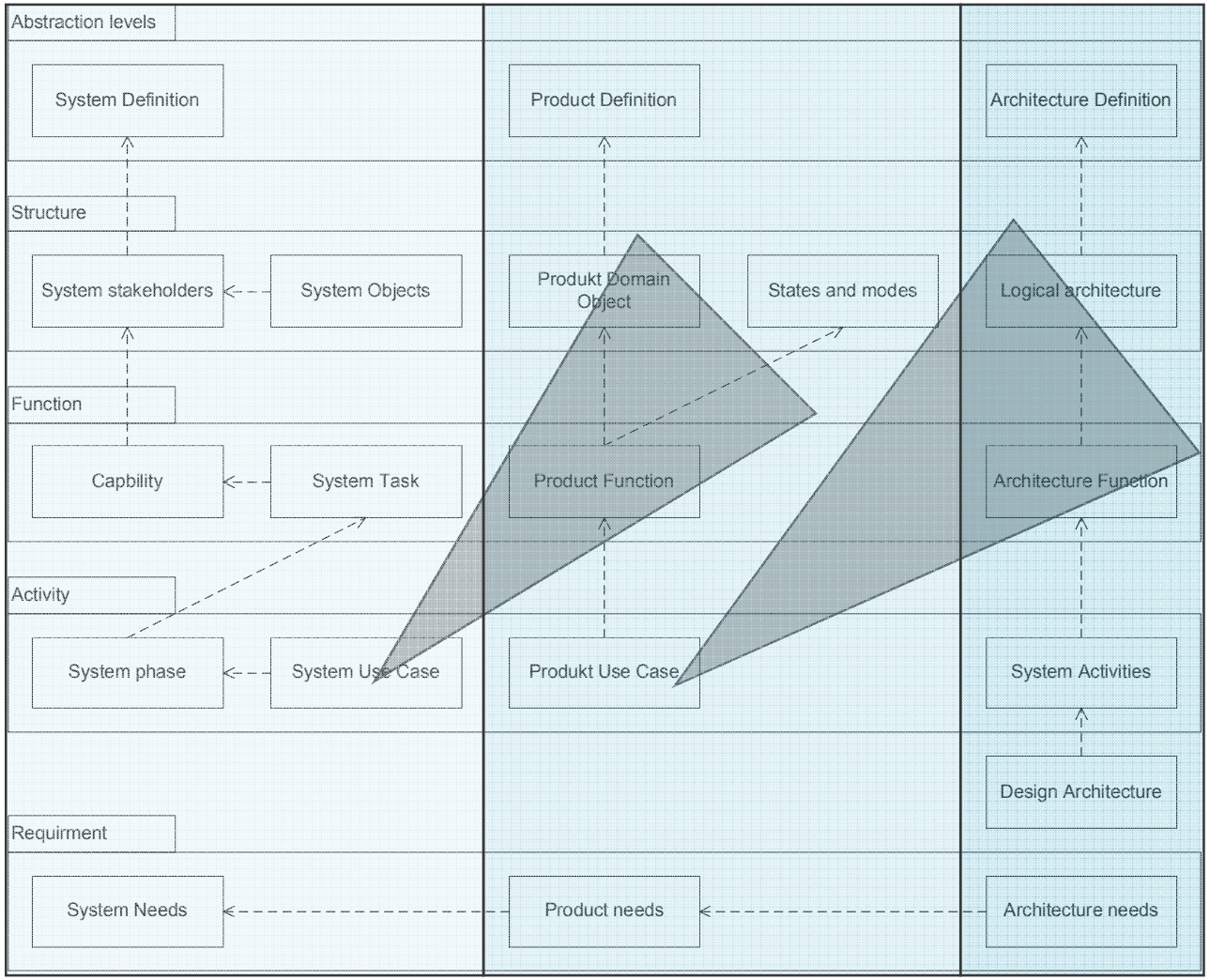
Example of Systems Engineering tool:

- Structured language of development process ISO-15288
- Modelling tools
- PLM-tools

ACTIVITY THEORY KARLSSON, I. C. M. (1996)



PU²B-MODEL


Abstraction levels
System Definition
Product Definition
Architecture Definition
Structure
System stakeholders
System Objects
Produkt Domain Object
States and modes
Logical architecture
Function
Capbility
System Task
Product Function
Architecture Function
Activity
System phase
System Use Case
Produkt Use Case
System Activities
Design Architecture
Requirment
System Needs
Product needs
Architecture needs

"NAPKIN EXAMPLE"

Food Turner

Stake holder: OEM, transport company, Vendor, buyer, active user, passive user...

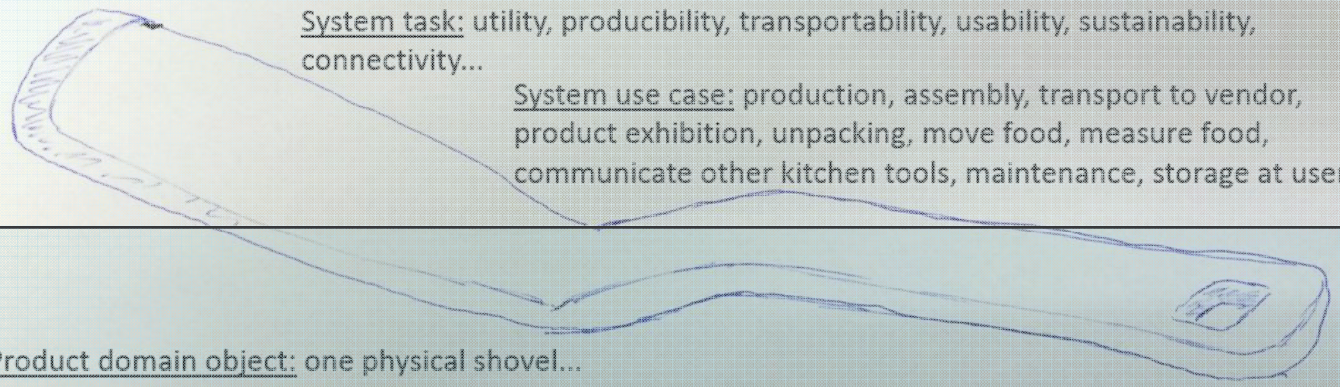
System objects: The turner, the turner package, the user, food, other food utensils...

Capability: Utility, kitchen identity...

System phases: manufacturer, vendor, user, recycling...

System task: utility, producibility, transportability, usability, sustainability, connectivity...

System use case: production, assembly, transport to vendor, product exhibition, unpacking, move food, measure food, communicate other kitchen tools, maintenance, storage at user...



Product domain object: one physical shovel...

States & modes: packaged, unpackaged, used, stored...

Product function: Stand up, move food, separate food, measure food, communicate stove...

Product use case: turn food, move food horizontally, lift food, cut food...

Logical architecture: food contact part, manage part, interface food and manage part...

Architecture function: push food, lift food, turn soft food, turn hard food, cut soft food, cut hard food...

System activities: move food with top part, move food by side part, cut food by push, move food sawing...

Design architecture: Spade, knife egg, handle, slip protection, sensor...

"REFLECTIONS... QUESTIONS AND ANSWERS"

1. Where do a function start? ...when you need to express a "what"

2. Where do a function end? ...when you start to create functions to support other functions and not only describe priorities... Then you are probably creating a solution for that abstraction level

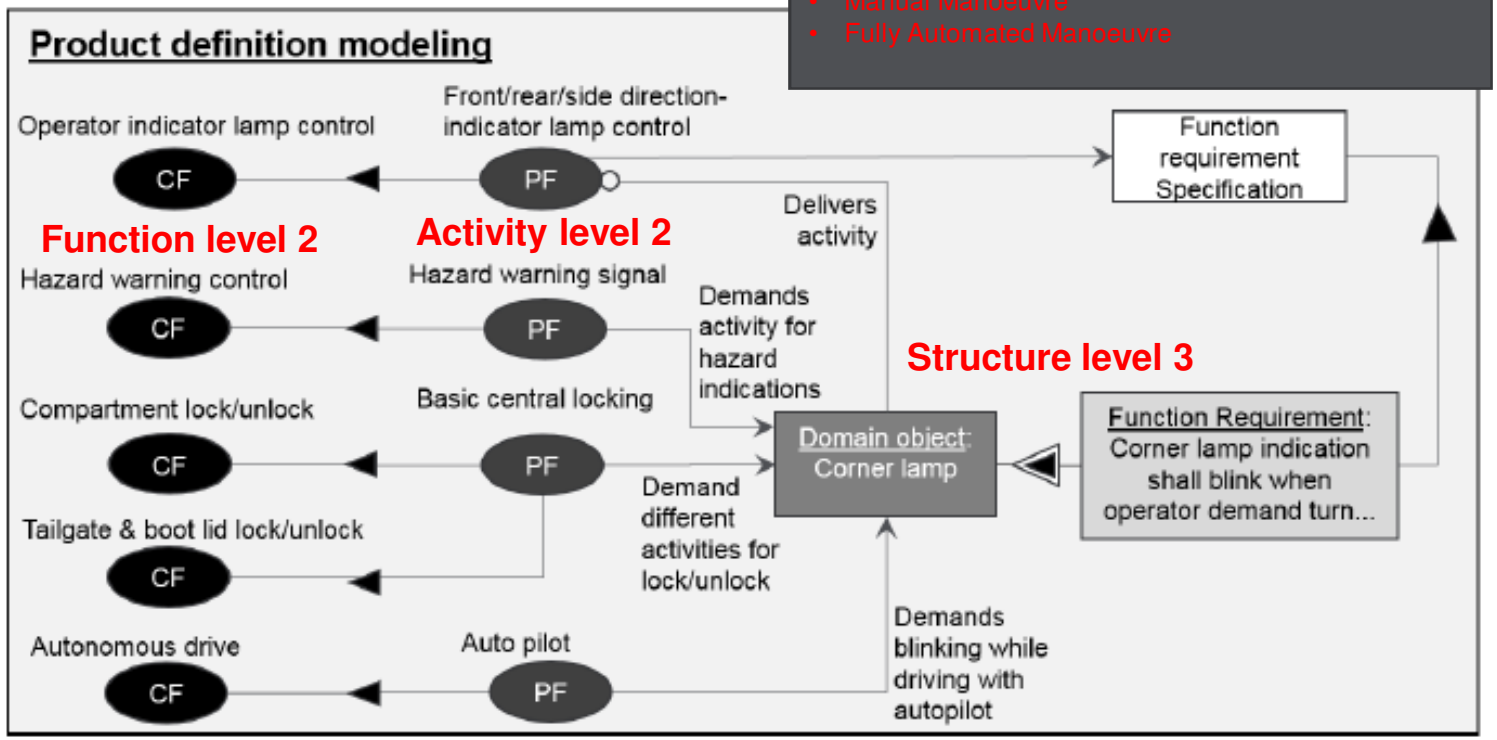
3. What is the answer to the "what" of the function? ...the structure is showing which parts that provides the solution and describing the "how"

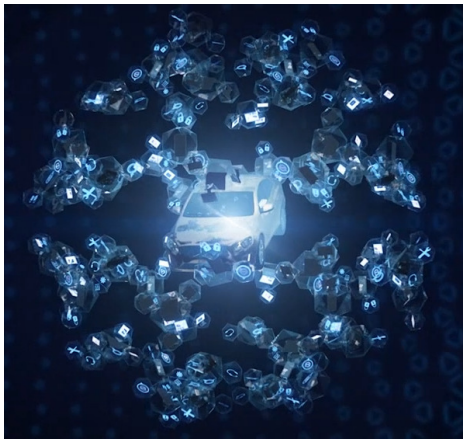
4. Why do we want to state function and structure unambiguously? ...then we can identify activities that can be performed AND this is the prerequisite to truly understand users!

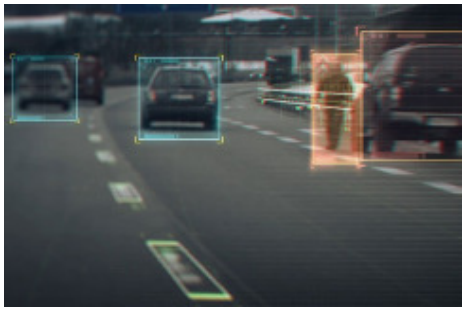
Mind the gap... of abstraction levels !!!

"VOLVO CARS EXAMPLE"

Activity level 1 – Product definition contexts:
- Open Close Vehicle
- Manual Manoeuvre
- Fully Automated Manoeuvre







Structure level 2 – Domain object:
- Vehicle
  - Vehicle HMI
  - Vehicle doors
- Remote Control


VOLVO
THANK YOU!
V40