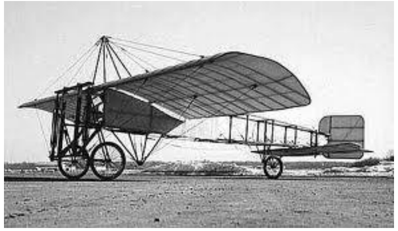
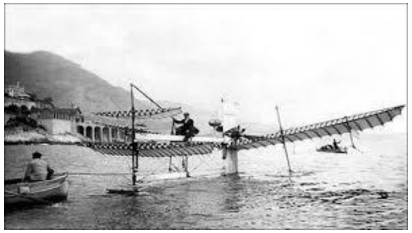
Elements of
Complex System Engineering

Antoine B. Rauzy
Department of Mechanical and Production Engineering (MTP)
Norwegian Science and Technology University (NTNU)
and
Chaire Blériot-Fabre
Centrale-Supélec, SAFRAN


CentraleSupélec

LECTURE 3.
SYSTEM ARCHITECTURE &
DESIGN STRUCTURE MATRICES

Notions:

- Functional and Physical System Architectures
- Design Structure Matrices

LECTURE 3. PART 1. INTRODUCTION

Objective of this lecture*

Mastering the complexity of a system requires to be able to organize and to classify its components as well as to characterize their interactions. In a word, it requires to make an order emerge, to create a structure. Organizing means understanding.

Based on a concrete use case – a climate control system of a vehicle – this lecture introduces the notions of physical and functional architecture of a system as well as the so-called "Design Structure Matrices" modeling technique.

It introduces also some algorithmic considerations and two little digressions on social networks and cognitive sciences.

(*) For the system architecture part, this lecture is deeply inspired by the work of Prof. Daniel Krob (Ecole Polytechnique, France) and the CESAMES method he develops.
For the Design Structure Matrices part, this lecture is partly inspired from the tutorial delivered by Tyson R. Browning during the international conference "DSM" held at Ecole Centrale Paris in July 2014.

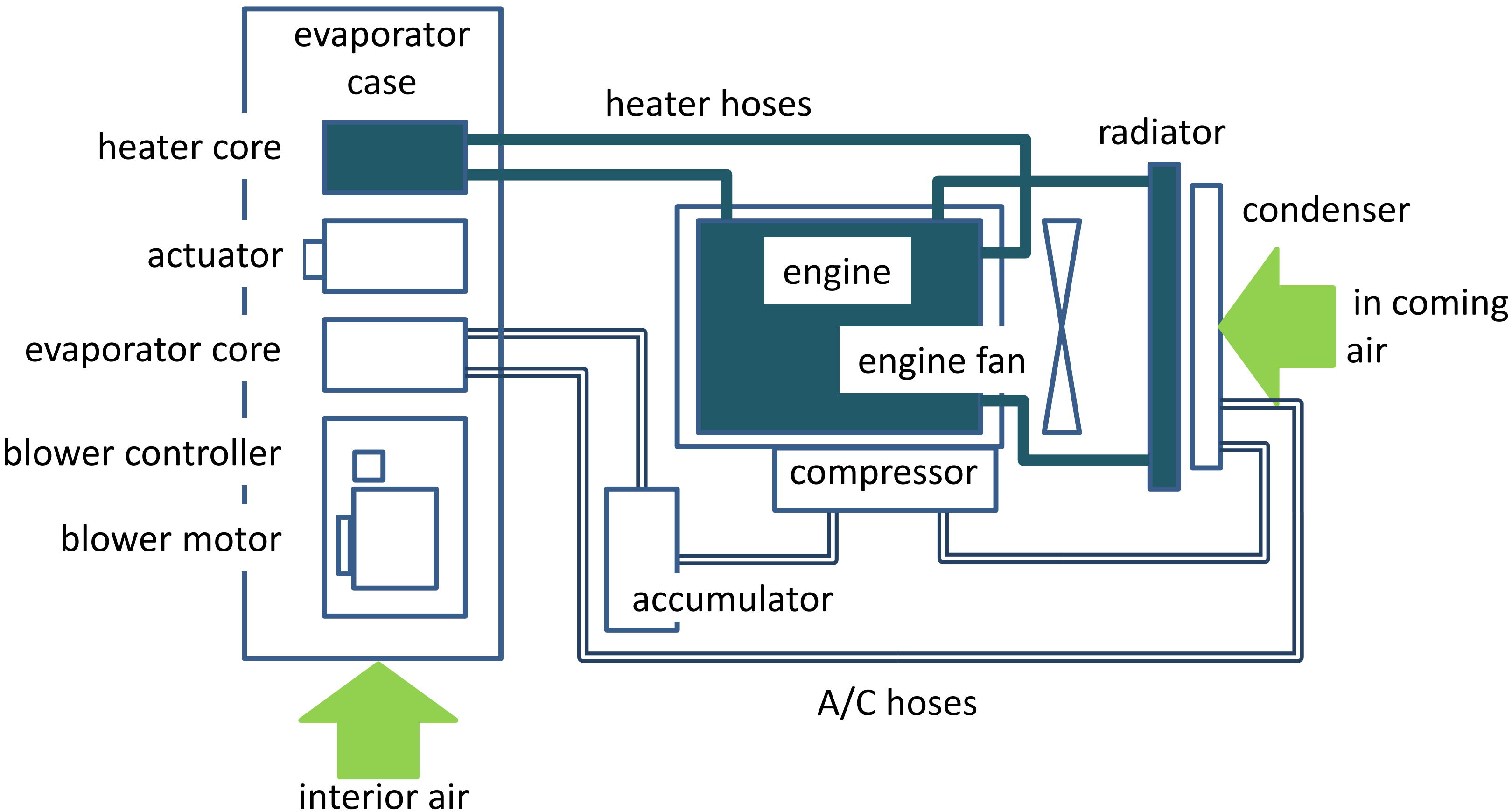
Use Case: Climate Control System

This use case comes from:

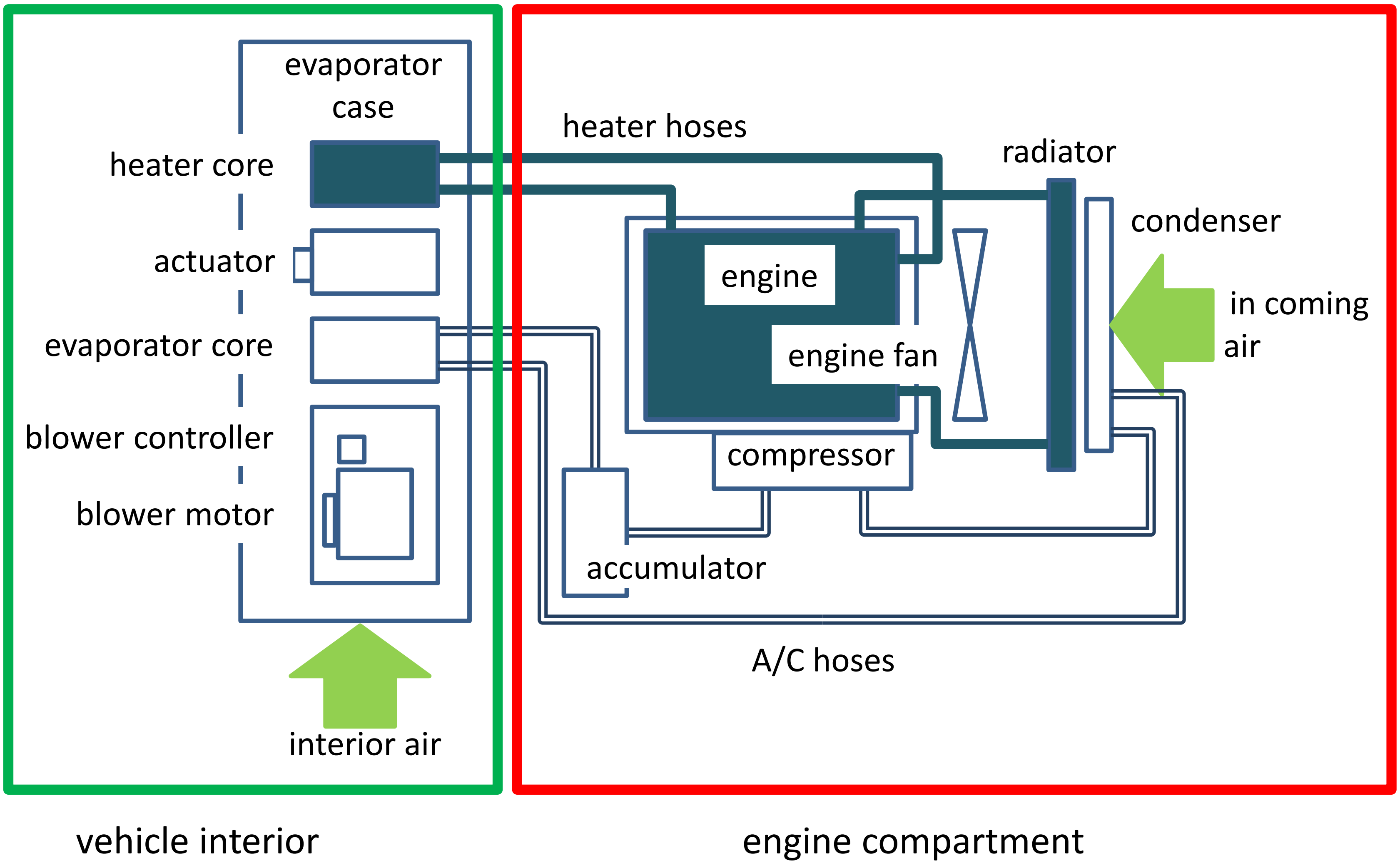
Thomas U. Pimmler and Steven D. Eppinger. Integration Analysis of Product Decompositions. *Proceedings of the ASME International Design Engineering Technical Conferences*. Minneapolis, USA. September 1994.

Summary: The Climate Control Division of the Ford company wanted to better understand the interactions between the many components of the heating/air conditioning systems of its cars and trucks. One of the objectives was to study whether it was possible to have a more **modular design**. A modular design is expected to improve both the **design** of components (possibly by giving the responsibility of this design to well identified teams or suppliers) and their **integration**. Integration is a key issue, because it makes it possible to change of supplier and therefore to some extent of component, when required by economical circumstances.

Climate Control System

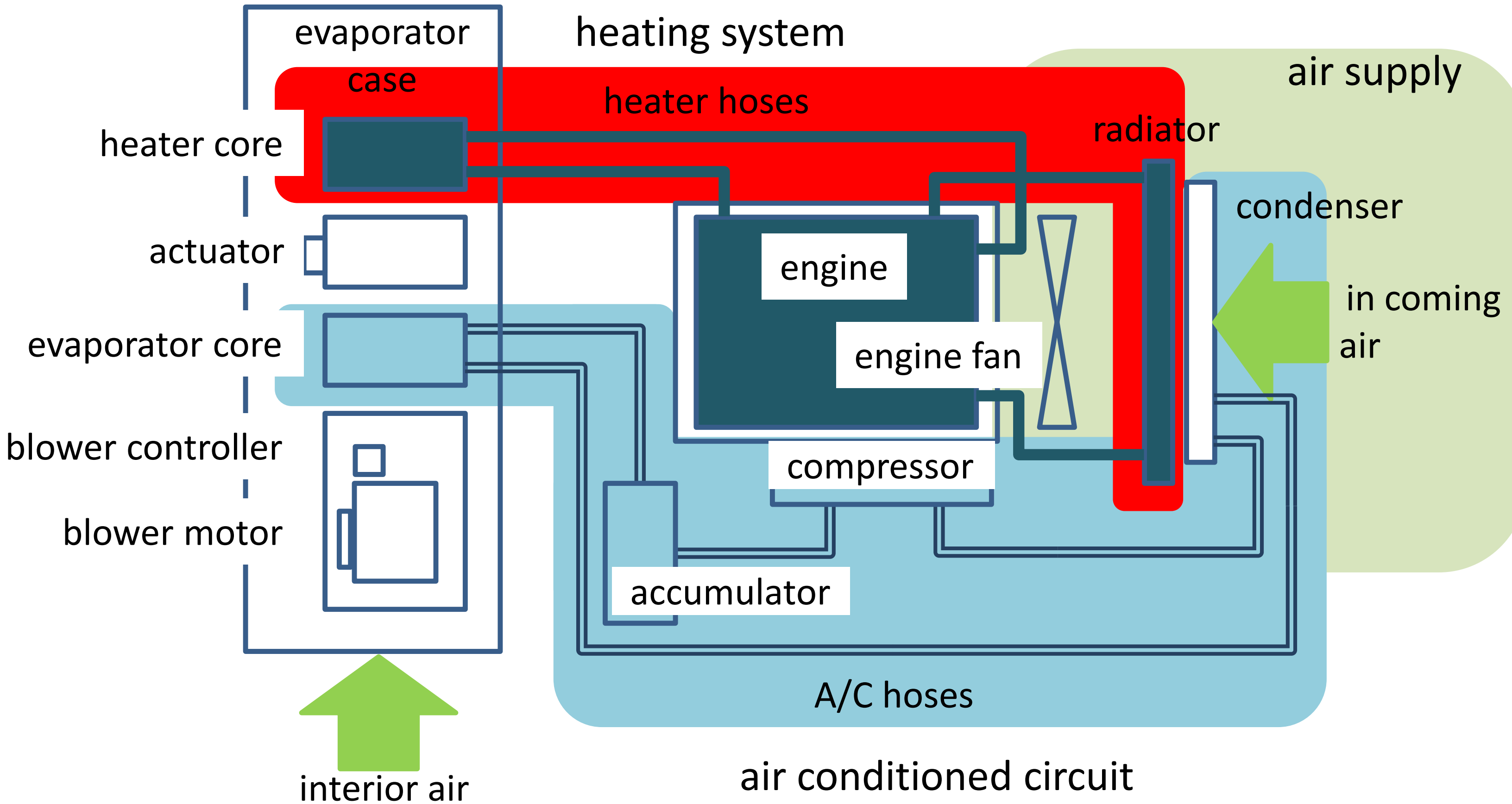


Physical Decomposition*



(*) as given in the cited article

Functional Decomposition*



(*) as given in the cited article

Discussion

The objective here is to understand the structure of a **product**, or more exactly of a **line of products** (the climate systems) in order to improve its design and its productions. To do so, we have at hand a **physical** and a **functional decompositions** (decompositions) of the product, as well as the **interactions** between the components. As we shall see, neither the physical decomposition nor the functional decomposition make are sufficient to achieve a modular design. We need to organize better these decompositions.

But before doing so, we need to define more formally what are physical and functional decompositions and which role they play in system design.

LECTURE 3. PART 2.
FIRST ELEMENTS OF SYSTEM ARCHITECTURE

CentraleSupélec

System Architecture

System Architecture is an emerging discipline that aims primarily at **integrating** other engineering disciplines. Industrial systems are nowadays so complex that the traditional discipline silo-ed, **divide-and-conquer,** approach for system design is no longer sufficient.

System architecture designates both a **process** and the **result of this process**.

System architects apply methodologies that involve the design of models. These methodologies are often called **architecture frameworks**.

The **SysML** modeling notation is often used to design models, although it has essential drawbacks.

In this course, we shall adopt **Krob Architectural Framework** which is, to our knowledge, the most advanced existing one.

Krob Architectural Framework

What are the missions of the system?

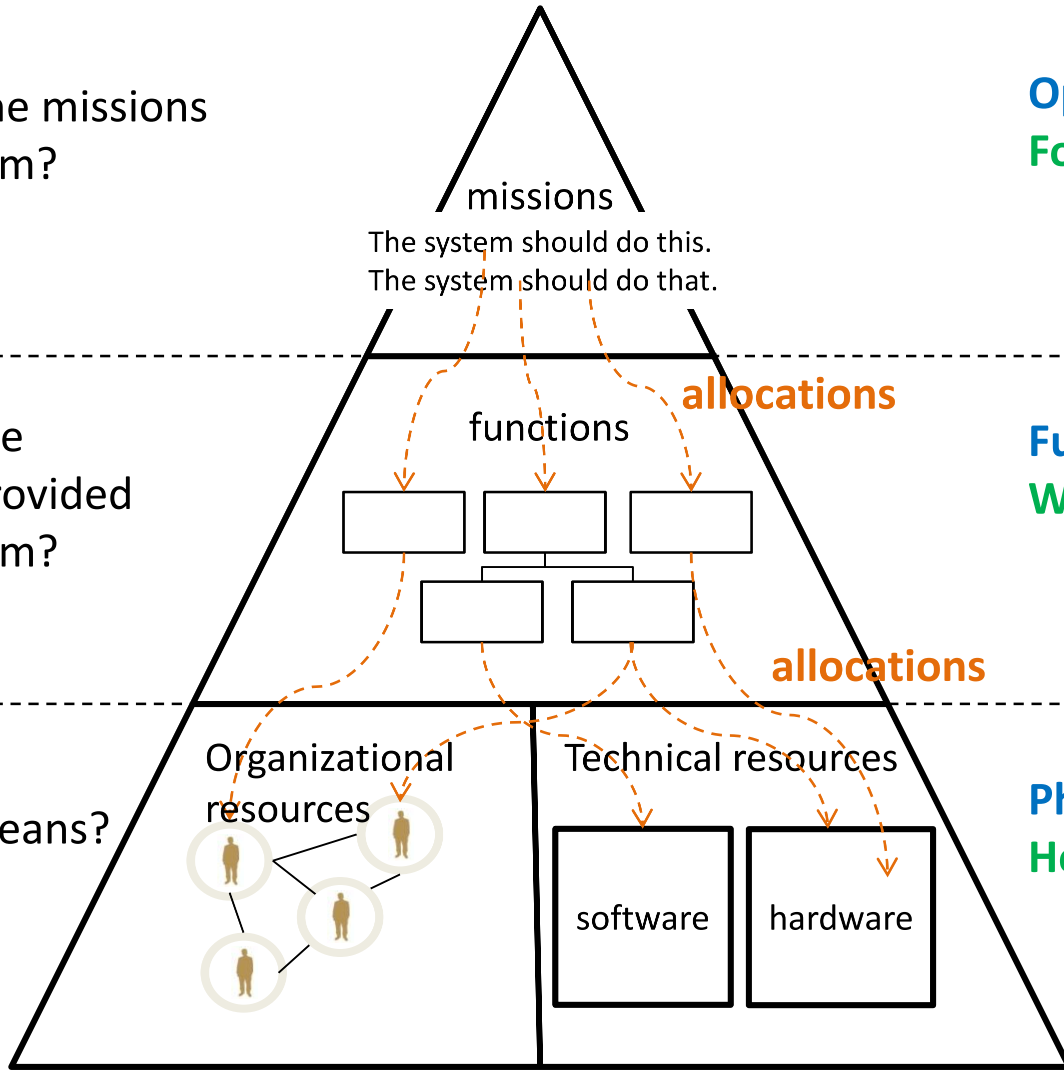
Operational analysis:
For whom and why?

What are the functions provided by the system?

Functional architecture:
What?

By which means?

Physical architecture:
How?

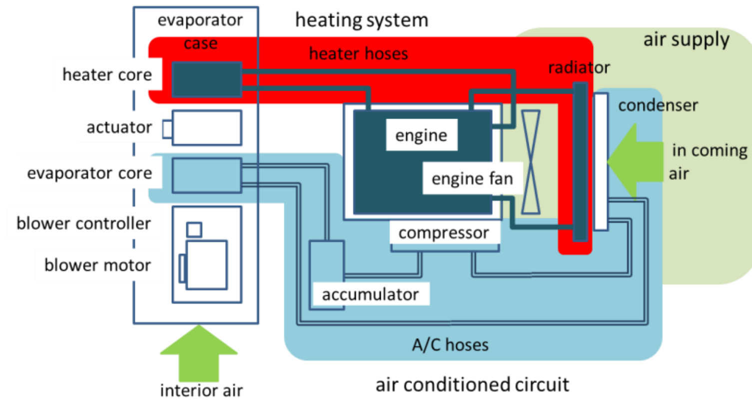


Krob Architectural Framework

Points of view	Questions	Analyses	Keywords	Models
Operational	For whom? Why?	Analysis of the environment of the system	Missions, use case, requirement, operational context, life cycle	Interactions of the system with external systems
Functional	What?	Abstract analysis of the system	Function, task, process, mode	Abstract functions of the system
Physical	How?	Concrete analysis of the system	Component, part, architecture, configuration	Concrete components of the system

Functional Decomposition

(partial view)

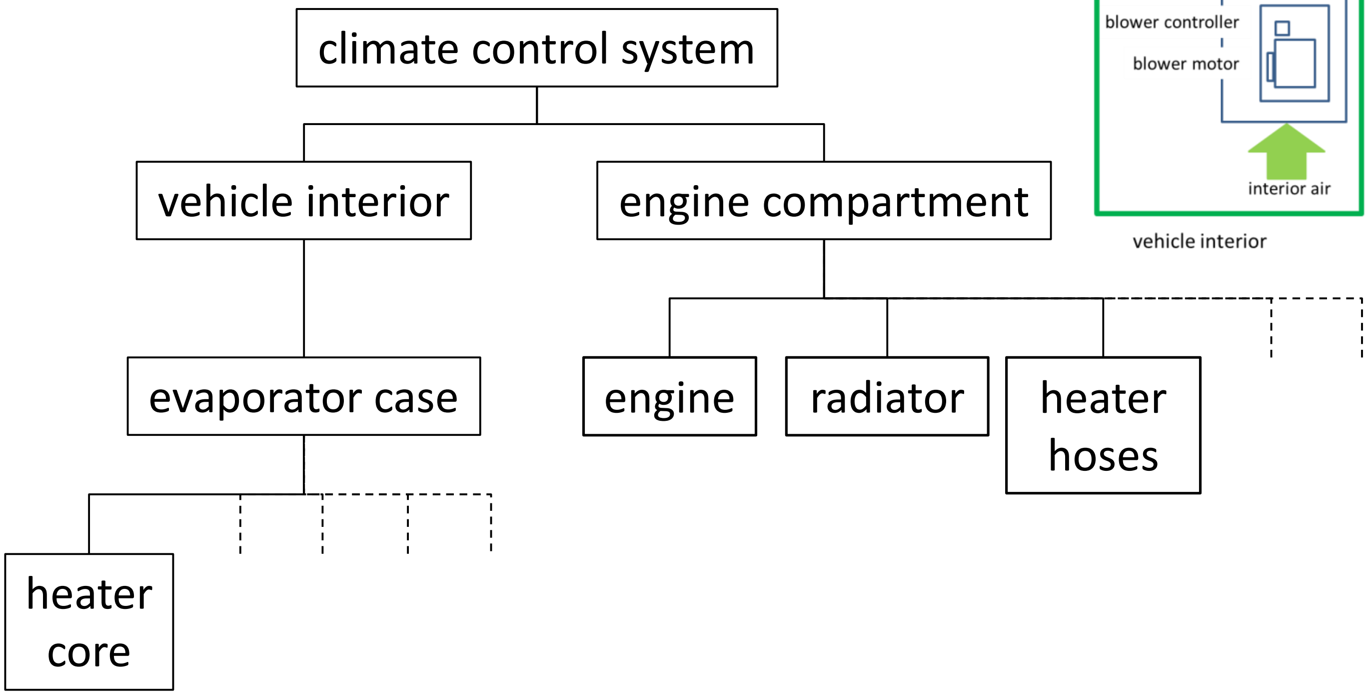


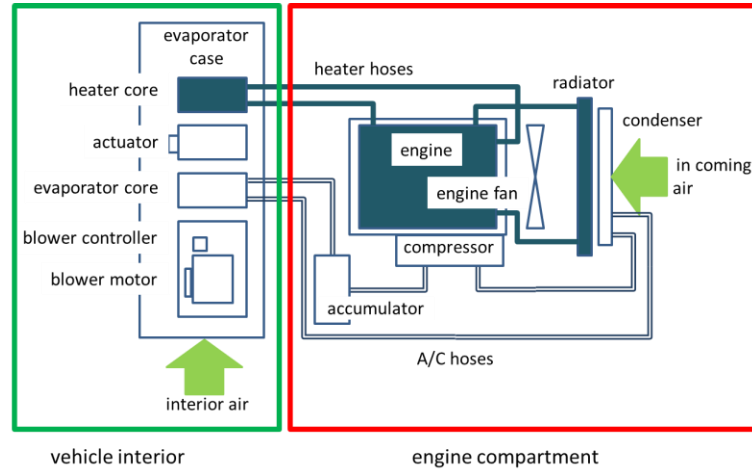
- climate regulation
  - hot air supply
    - fluid heating
    - fluid circulation
    - heat exchange
  - cold air supply

Functions are decomposed into sub-functions. These sub-functions are themselves decomposed into sub-sub-functions and so on until the **suitable granularity** is reached. The result is a **tree-like representation**. However, some nodes may be shared by several branches, e.g. to represent support functions like power supply.

Physical Decomposition

(partial view)

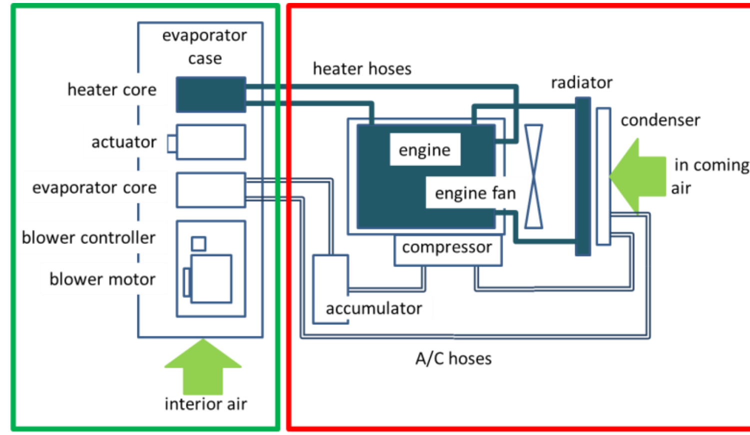




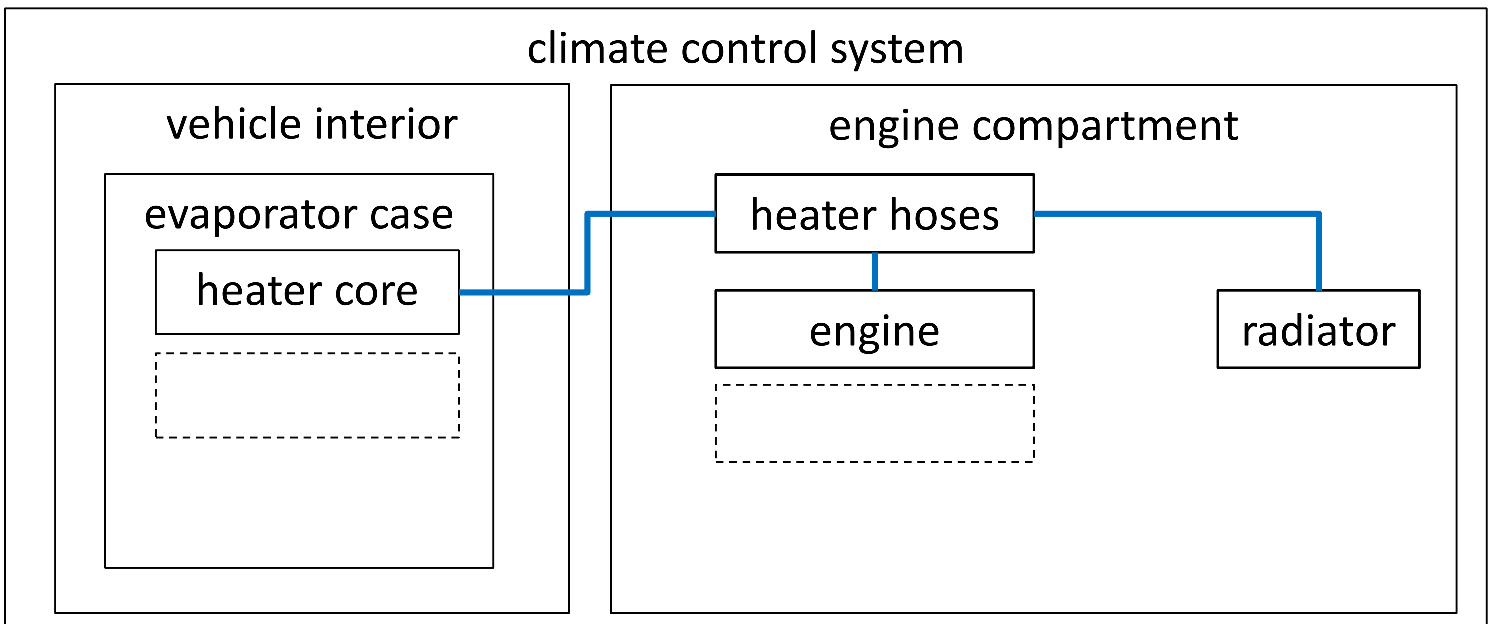
The system is decomposed into sub-systems These sub-systems are themselves decomposed into sub-sub-systems and so on until the **suitable granularity** is reached.

Interactions

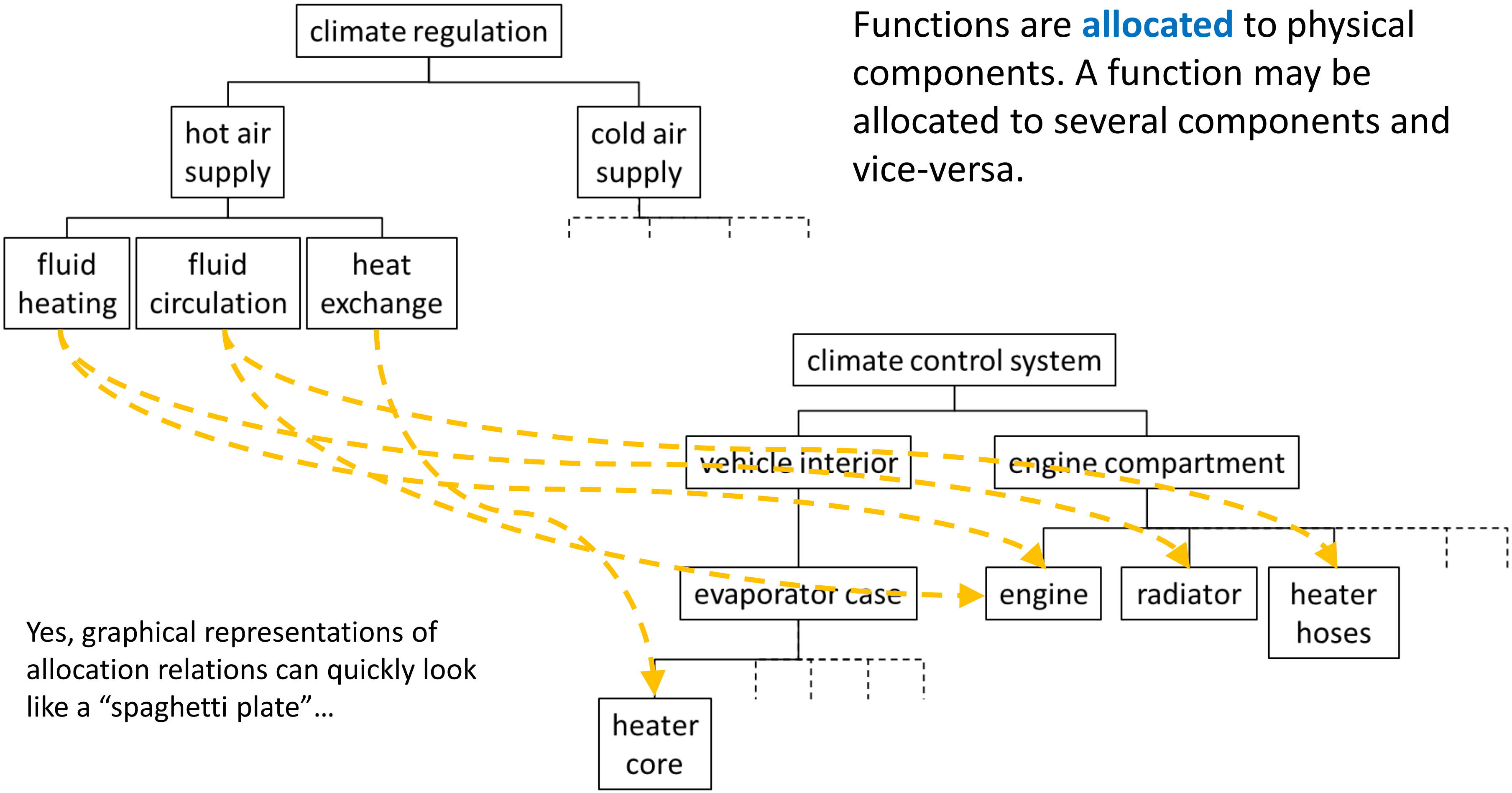
The **tree-like representation** of functional or physical decompositions is well adapted to represent **hierarchical relations**. It is not very suited to represent **interactions**. To represent interactions, **block-diagrams** are in general better suited.
Tree-like diagrams and block diagrams are **two representations** of the **same model**.



vehicle interior engine compartment



Allocations



Functions are **allocated** to physical components. A function may be allocated to several components and vice-versa.

Yes, graphical representations of allocation relations can quickly look like a “spaghetti plate”…

CentraleSupélec

LECTURE 3. PART 3.
SYSTEM ARCHITECTURE LANGUAGE

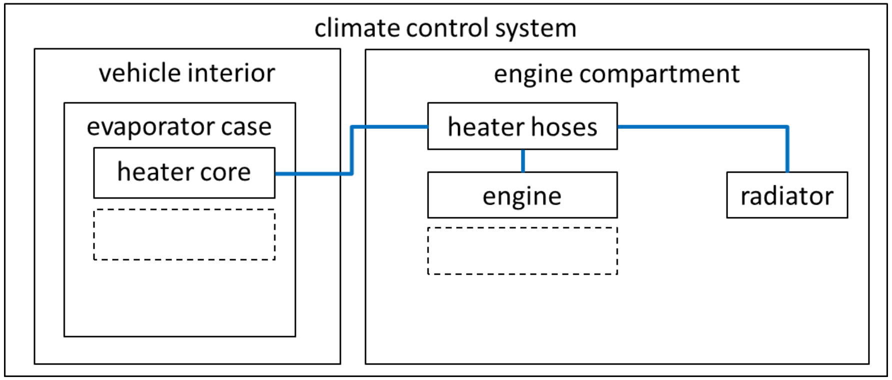
Making Architecture Diagrams a Textual Language

As for BMPN in the previous lecture, there is clearly a need for a textual representation of architecture models.
Here follows the elements of these models (this list may be non exhaustive):

Modeling Elements	Functional Decompositions	Physical Decompositions
(hierarchy of) Blocks	(hierarchy of) Functions	(hierarchy of) Components
Ports	Inputs and outputs of functions	Inputs and outputs of components
Connections	Interactions of functions (trough their ports)	Interactions of components (trough their ports)
Allocations	Allocation of functions onto components	

Functional and Physical Decompositions (sketch)

```
block EngineCompartment
    …
    block Radiator
        Port toHeaterHoses;
            …
    end
    block HeaterHoses
        Port toRadiator;
        …
    end
    …
    connections
        connect Radiator.toHeaterHoses HeaterHoses.toRadiator;
        …
end
```



Model (sketch)

```
model ClimateControlModel
   block ClimateRegulation // Functional decomposition
      block HotAirSupply
         …
      end
      …
   end
   block ClimateControlSystem // Physical decomposition
      block VehicleInterior
         …
      end
      …
   end
   allocations
      allocate ClimateRegulation.HotAirSupply.HeatExchange
               ClimateControlSystem.VehicleInterior.
               EvaporatorCase.HeaterCore;
      …
end
```

EBNF of the Textual Language

See assignment

CentraleSupélec

LECTURE 3. PART 4.
DESIGN STRUCTURE MATRICES

Design Structure Matrices

Design Structure Matrices (**DSM**) are a simple but efficient tool to understand and to organize **interactions** between components of a system. In a word, DSM are a good support tool to **structure** the vision (and the models) of a system.

A **Design Structure Matrix** is:

- a **square matrix** showing **dependencies** amongst the components (the component 1 uses the component 5 and is used by the component 2),
- that allows to represent asymmetric relationships.

The analysis of a DSM consists in permuting components, i.e. lines (and thus columns) so to:

- visualize **architectural patterns** (**modules**, **loops**), and
- find out **modules** (**clusters**) and **sequences**.

Components	1	2	3	4	5
1		x			
2			x	x	
3				x	
4					x
5	x				

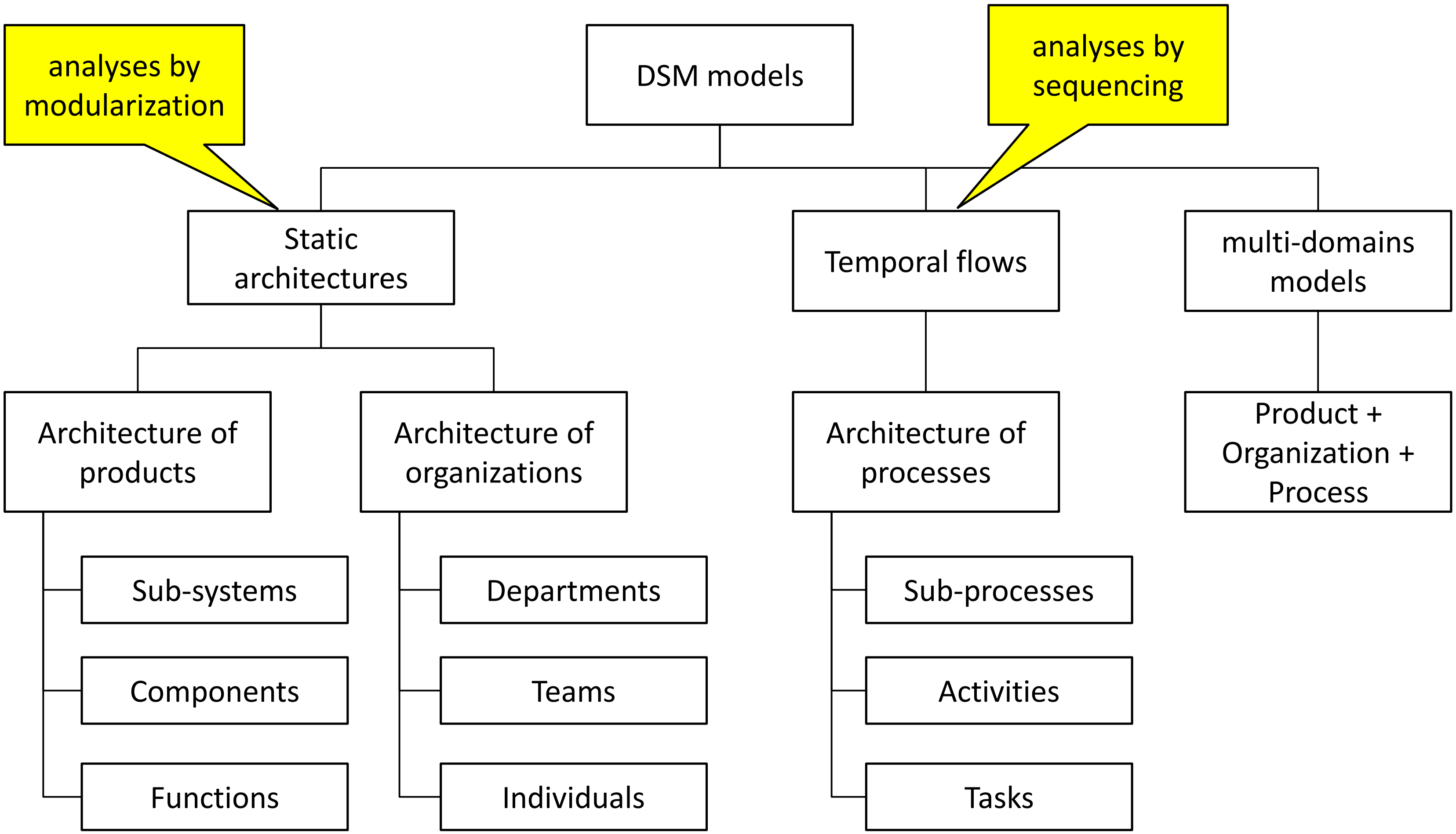
When designing a DSM, it is of primary importance to well document interactions. It is sometimes of interest to design one matrix per type of interaction.

Climate Control System

DSM	16	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Radiator	1		**2**			**2**											
Engine Fan	2	**2**				**2**								**1**			
Heater Core	3							**2**	**1**								**2**
Heater Hoses	4					**1**				**1**							
Condenser	5	**2**	**2**				**2**		**2**								
Compressor	6					**2**			**2**	**2**	**2**	**2**		**1**			
Evaporator Case	7			**2**					**2**						**2**	**2**	**2**
Evaporator Core	8			**1**		**2**	**2**	**2**		**2**							**2**
Accumulator	9				**1**		**2**		**2**		**1**						
Refrigeration Controls	10						**2**			**1**		**2**		**1**			
EATC Controls	11						**2**				**2**		**2**	**1**	**2**	**2**	
Sensors	12											**2**		**1**			
Command Distribution	13		**1**				**1**				**1**	**1**	**1**		**1**	**1**	**1**
Actuators	14							**2**				**2**		**1**			
Blower Controller	15							**2**				**2**		**1**			**2**
Blower Motor	16			**2**				**2**	**2**					**1**		**2**	

1: weak interaction, 2: strong interaction

Taxonomy of Design Structures Matrices

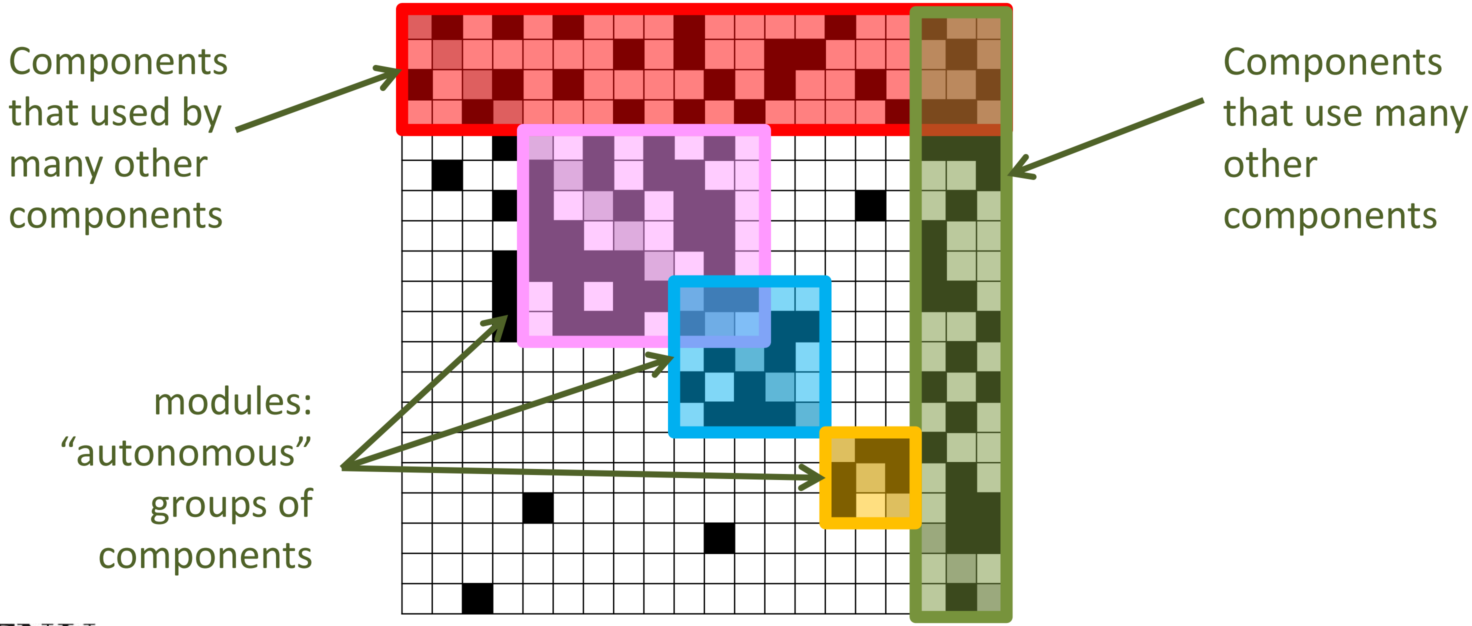


Modularization

The **analysis** of a design structure matrix that represents interactions between components of a product aims at making emerge:

- Components that are involved in the **control** of the product, i.e. components that either use or are used, or both by many other components.
- **Modules**, also called “**clusters**”, i.e. groups of components that have strong interactions among them but few interactions with other components (except possibly the control components).

To achieve this goal, the idea is to switch lines (and thus columns) of the matrix:



Climate Control System

Here is the result obtained with a “good” modularization algorithm

DSM	16	4	10	11	12	13	14	1	2	5	6	9	8	3	16	15	7
Heater Hoses	4									1		1					
Refrigeration Controls	10			2		1					2	1					
EATC Controls	11		2		2	1	2				2					2	
Sensors	12			2		1											
Command Distribution	13		1	1	1		1		1		1				1	1	
Actuators	14			2		1											2
Radiator	1								2	2							
Engine Fan	2					1		2		2							
Condenser	5							2	2		2		2				
Compressor	6		2	2		1				2		2	2				
Accumulator	9	1	1								2		2				
Evaporator Core	8									2	2	2		1	2		2
Heater Core	3												1		2		2
Blower Motor	16					1							2	2		2	2
Blower Controller	15			2		1									2		2
Evaporator Case	7						2						2	2	2	2	

Climate Control System

This clustering "makes sense"

DSM	16	4	10	11	12	13	14	1	2	5	6	9	8	3	16	15	7
Heater Hoses	4									1		1					
Refrigeration Controls	10			2		1					2	1					
EATC Controls	11		2		2	1										2	
Sensors	12			2		1											
Command Distribution	13		1	1	1		1		1		1				1	1	
Actuators	14			2		1											2
Radiator	1								2	2							
Engine Fan	2							2		2							
Condenser	5							2	2		2		2				
Compressor	6		2	2		1				2		2	2				
Accumulator	9	1	1								2		2				
Evaporator Core	8									2	2	2		1	2		2
Heater Core	3												1		2		2
Blower Motor	16					1							2	2		2	2
Blower Controller	15			2		1									2		2
Evaporator Case	7						2						2	2	2	2	

Control and connections

Front-End Air

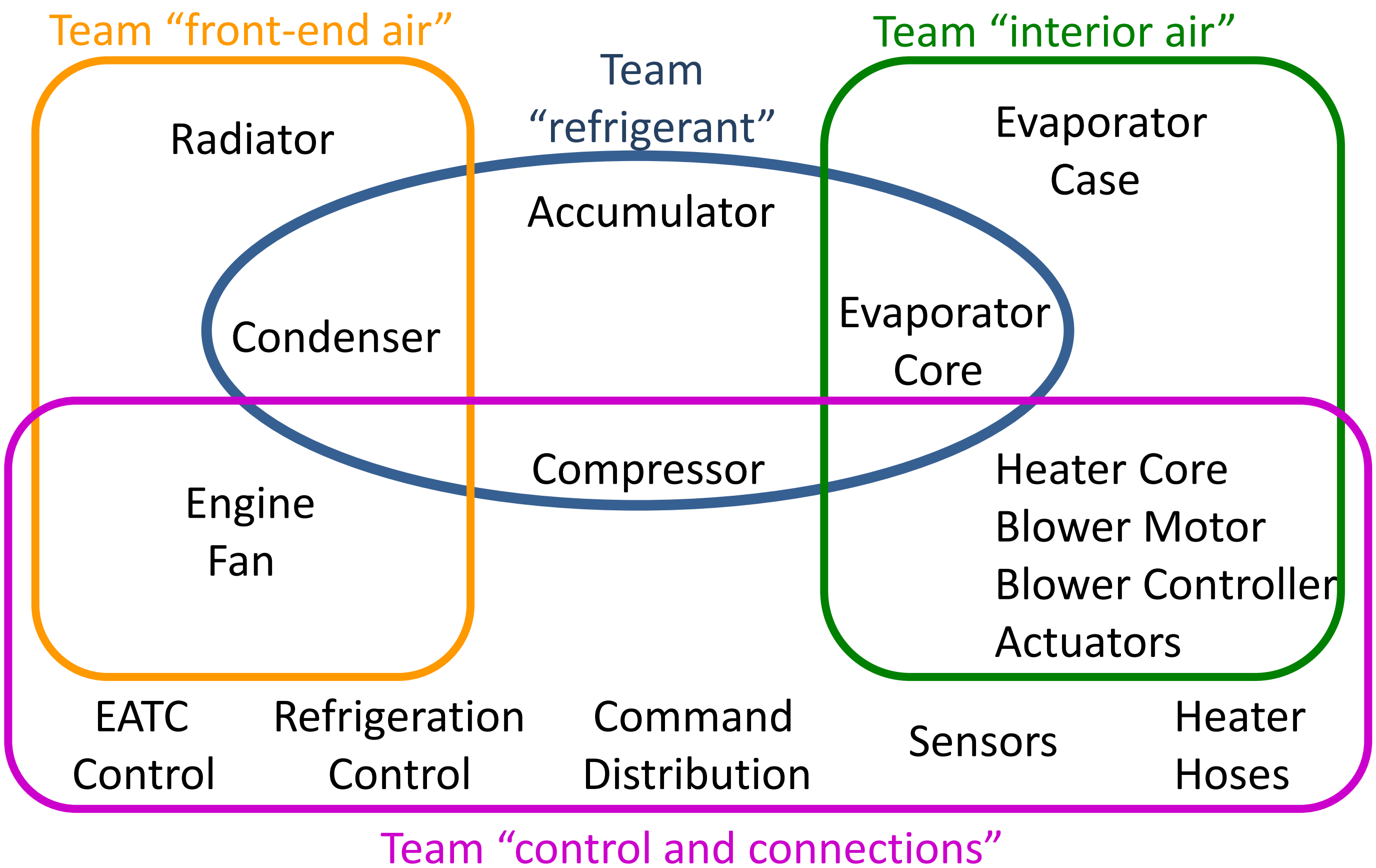
Refrigerant

Interior Air

Note: modules are not necessarily disjoint.

Assignment to Teams

The clustering obtained thanks to the analysis of the DSM makes it possible to split the climate control system into **well identified and relatively independent modules**. These modules can be assigned to **separate teams** in charge of the design and the production. The analysis makes it also possible to master **interactions** between the teams.



Assessment of a Clustering

As there is **no unique way** to modularize a DSM, a **heuristic** should be designed to choose amongst candidate clustering's. Such a heuristic has two objectives:

- Minimize the size of the modules;
- Minimize the number of inter-modules connections.

Moreover, it can choose:

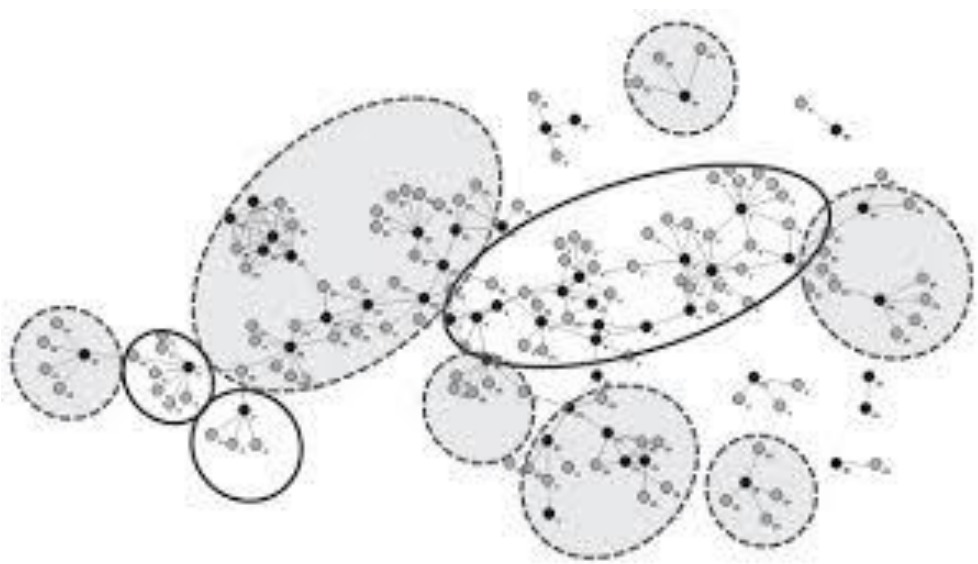
- To consider of not overlapping modules;
- To put aside or not elements with many interactions (which requires to define a threshold).

A commonly used reward function to assess the benefit of a clustering is as follows.

$$\alpha\left(\sum_{i=1}^{m} {M_i}^2\right) + \beta X \quad \text{with} \begin{cases} m\text{: number of modules} \\ M_i\text{: number of elements of the ith module} \\ X\text{: number of interactions bewteen modules} \\ \alpha \text{ and } \beta\text{: two constants to be defined} \end{cases}$$

Digression: a small world

In 1929, Frigyes Karinthy stated his theory of the **six degrees of separation** that asserts that anybody, dead or alive, can be linked to anybody else, dead or alive, by a chain comprising at most five other individuals. This theory has been re-investigated in 1967 by Stanley Milgram through his **small world** study. It is actually empirically verified and tested on a large scale on **social networks** such as Facebook.



Social networks are actually organized into **clusters**, which explains why they are "small world". **Clustering algorithms** are used to understand and to predict behaviors… just by using **metadata** (connections) and without entering in (the detail of) interactions between the members of the network.

The 7 ± 2 rule

Design Structure Matrices help to **group** the components of a system in a suitable way. This grouping aims first at easing the **understanding** of the system by the **stakeholders**.

When we visualize graphical representations such as functional and physical decompositions or design structure matrices, we use our **short term memory**. The short term memory can be seen as the **working space** of the brain. It is opposed to the **long term memory** which is the place where **stabilized knowledge** is stored.

A number of cognitive science experiments have been performed that show that the short term memory is able to distinguish **7 ± 2 items** (so called chunks) [Miller 1975]*.

Mutatis mutandis, the Miller's rule is a good **heuristic** to group items so to obtain easy to read and easy to understand **hierarchical representations**: each node should contain 7 ± 2 sub-nodes.

(*) G.A. Miller, The Psychology of Communication, Basic Books, New York, second edition, (1975)

LECTURE 3. PART 5.
WRAP-UP & ASSIGNMENT

Wrap-Up

- **System Architecture** is an emerging discipline that aims primarily at **integrating** other engineering disciplines.
- System architects apply methodologies that involve the design of models. These methodologies are often called **architecture frameworks**.
- **Functional and physical decompositions** (as well as **allocation relations**) play a central role in system architecture frameworks.
- **Design Structure Matrices** are a simple but efficient tool to study and to organize the **dependencies amongst components** of a system. This technique applies to products, to organizations and to processes.
- The analysis of a design structure matrix consists in reorganizing the matrix so to make emerge **architectural patterns** such as **modules** or **sequences**.
- Design Structures Matrices have their website: **http://www.dsmweb.org/**

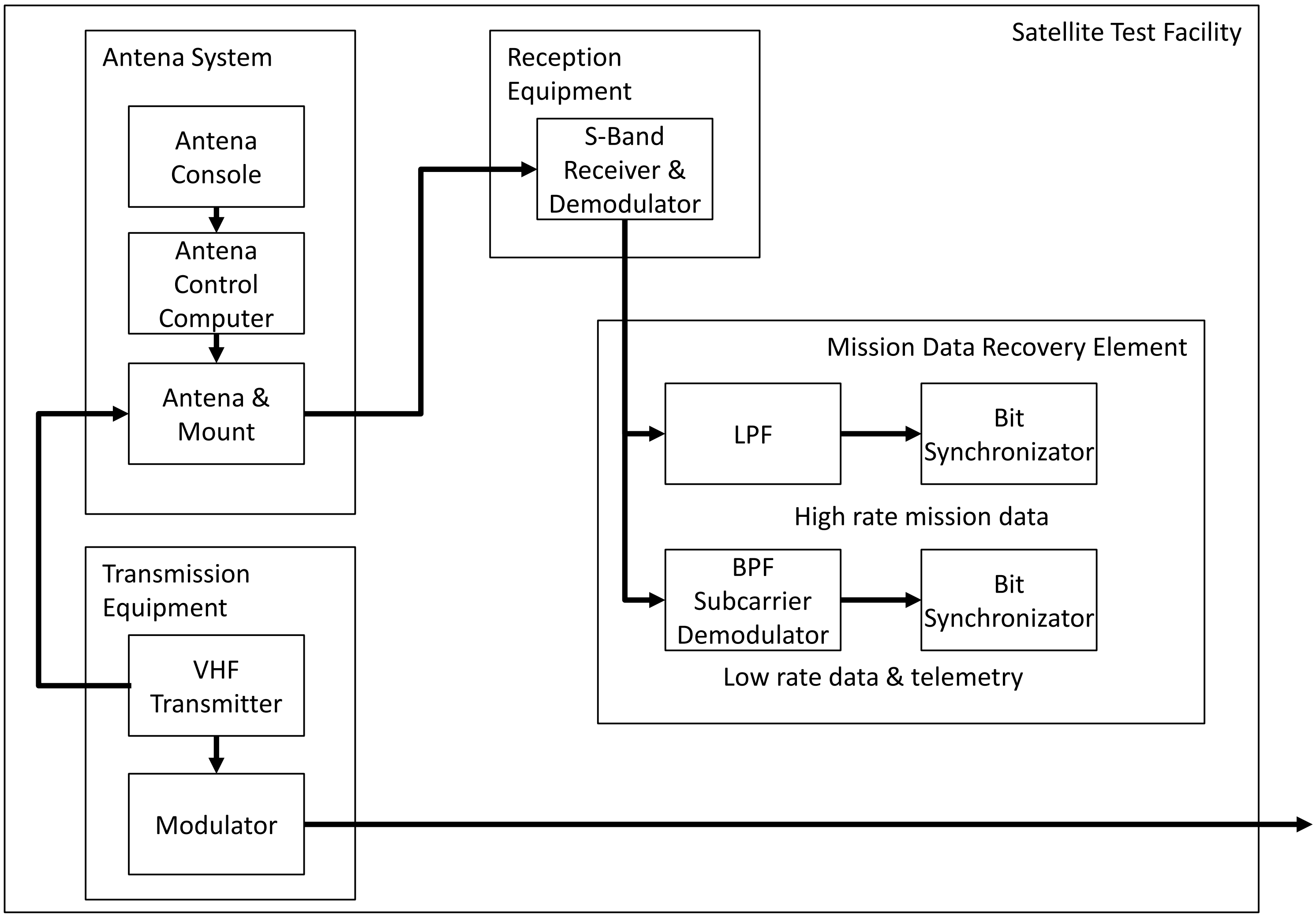
Assignment

1. Write the Extended Backus-Naur Form of the architecture language sketched slides 19, 20 and 21 of this presentation.

Consider the physical decomposition of the GEOSAT ground system test facility given next slide.

2. Design a tree-like representation for this physical decomposition.
3. Design a textual model for this physical decomposition.
4. Design a design structure matrix for this system.
5. Modularize this design structure matrix and comment.

GEOSAT Ground System Test Facility



Recommend Readings

Books or articles about System Architecture:

Benjamin S. Blanchard and Wolter J. Fabrycky. Systems Engineering and Analysis. Pearson. Upper Saddle River, NJ 07456, USA. ISBN 978-0137148431. 2008.

Sanford Friedenthal, Alan Moore and Rick Steiner. A Practical Guide to SysML: The Systems Modeling Language. Morgan Kaufmann. The MK/OMG Press. San Francisco, CA 94104, USA. ISBN 978-0123852069. 2011.

INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, fourth edition. David D. Walden, Garry J. Roedler, Kevin J. Forsberg, R. Douglas Hamelin and Thomas M. Shortell Ed.. Wiley-Blackwell. Hoboken, NJ, USA. ISBN 978-1118999400. August, 2015.

Daniel Krob website: http://krob.cesames.net/

Books or articles about Design Structure Matrices:

Thomas U. Pimmler and Steven D. Eppinger. *Integration Analysis of Product Decompositions*. Proceedings of the ASME International Design Engineering Technical Conferences. Minneapolis, USA. September, 1994.

Steven D. Eppinger & Tyson R. Browning. *Design Structure Matrix Methods and Applications*. MIT Press (19 juin 2012). Engineering Systems. ISBN-10: 0262017520. ISBN-13: 978-0262017527

Non-scientific books about social networks:

Nicholas A. Christakis and James H. Fowler. *Connected*. Back Bay Books (12 janvier 2011). ISBN-10: 0316036137, ISBN-13: 978-0316036139.

Duncan Watts. *Six Degrees: The New Science of Networks*. Vintage (6 mai 2004). ISBN-10: 0099444968. ISBN-13: 978-0099444961.

Louis Charles Joseph Blériot (1872 -1936) is an airplane designer and one of the pioneer pilot of French aviation. He has been the first to cross the channel on July the 25th onboard of the Blériot XI.
He graduated from Ecole Centrale de Paris

Henri Marie Léonce Fabre (1882 -1984) is a French engineer and pilot. He invented the seaplane in 1910.
He graduated from Supélec.