Models of requirements for avionics architecture synthesis: safety, capacity and security

Laurent ZIMMER, Michaël LAFAYE
Dassault Aviation
Direction de la Prospective
laurent.zimmer@dassault-aviation.com
michael.lafaye@dassault-aviation.com

Pierre-Alain Yvars
Institut Supérieur de Mécanique de Paris
Laboratoire Quartz – EA 7393
pierre-alain.yvars@supmeca.fr

OUTLINE

- CONTEXT
- PROJECT
- THE DEPS LANGUAGE
- APPLICATION
- SUMMARY
- ONGOING AND FUTURE WORK

CONTEXT

SYSTEMS

- "A system is a construct or collection of different elements that together produces results not obtainable by the elements alone" (INCOSE)

- Systems are everywhere, of any kind and in everything: technical, embedded, real-time, software-intensive, cyber-physical, systems-of-systems

❑ Systems are more and more complex

DARPA 2010

- Avionics systems are becoming increasingly complex

➢ Explosion in development costs

➢ Need of new design methods and tools

1. **Abstraction-based design tools**
2. System complexity metrics
3. **Advanced methods of architecture synthesis**
4. Robust uncertainty management

DARPA 2010 : Abstraction Based Complexity Management

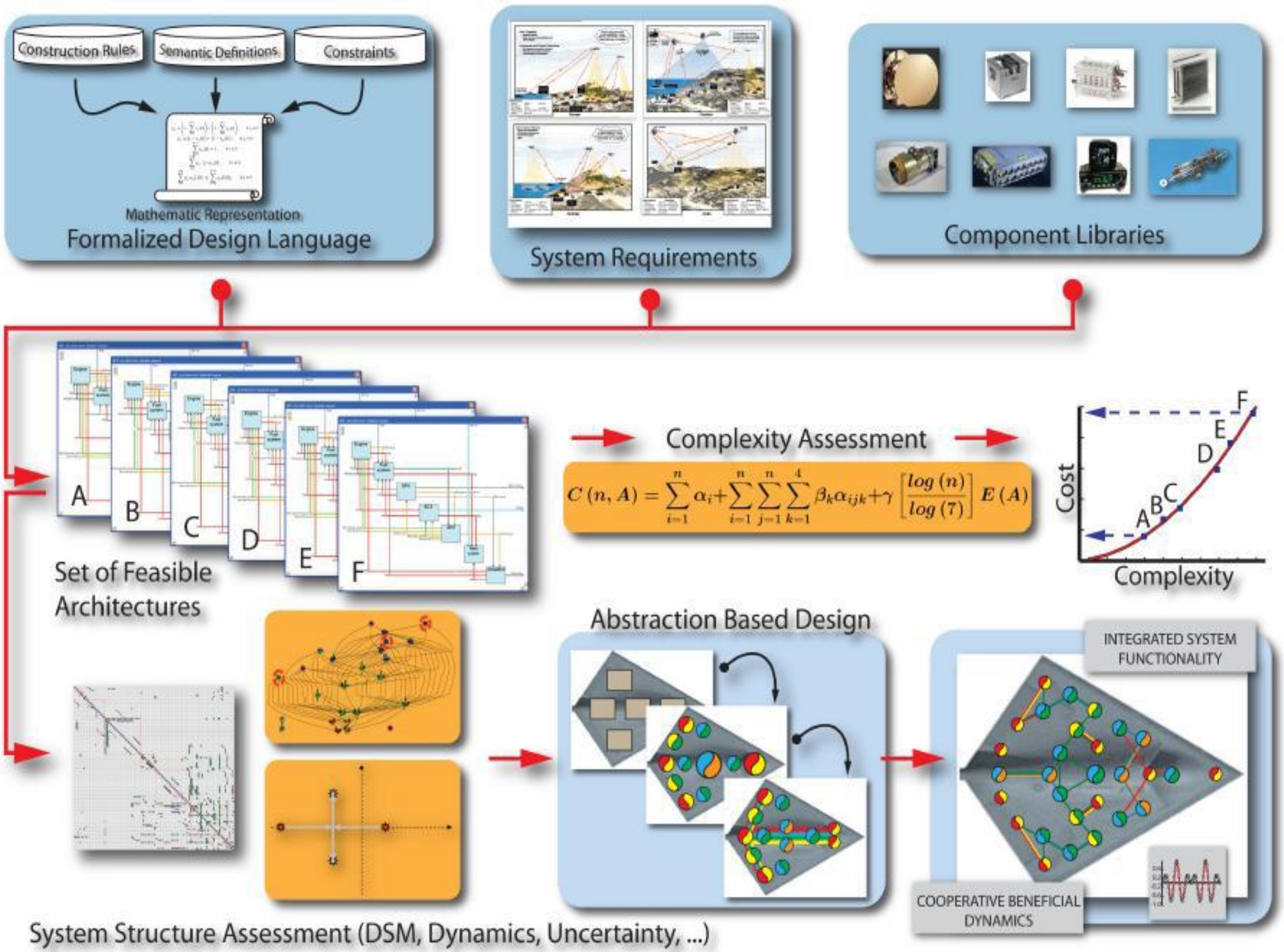


Figure F.2: Design flow to manage complexity in heterogeneous cyber-physical systems

DARPA 2010 : Abstraction Based Complexity Management

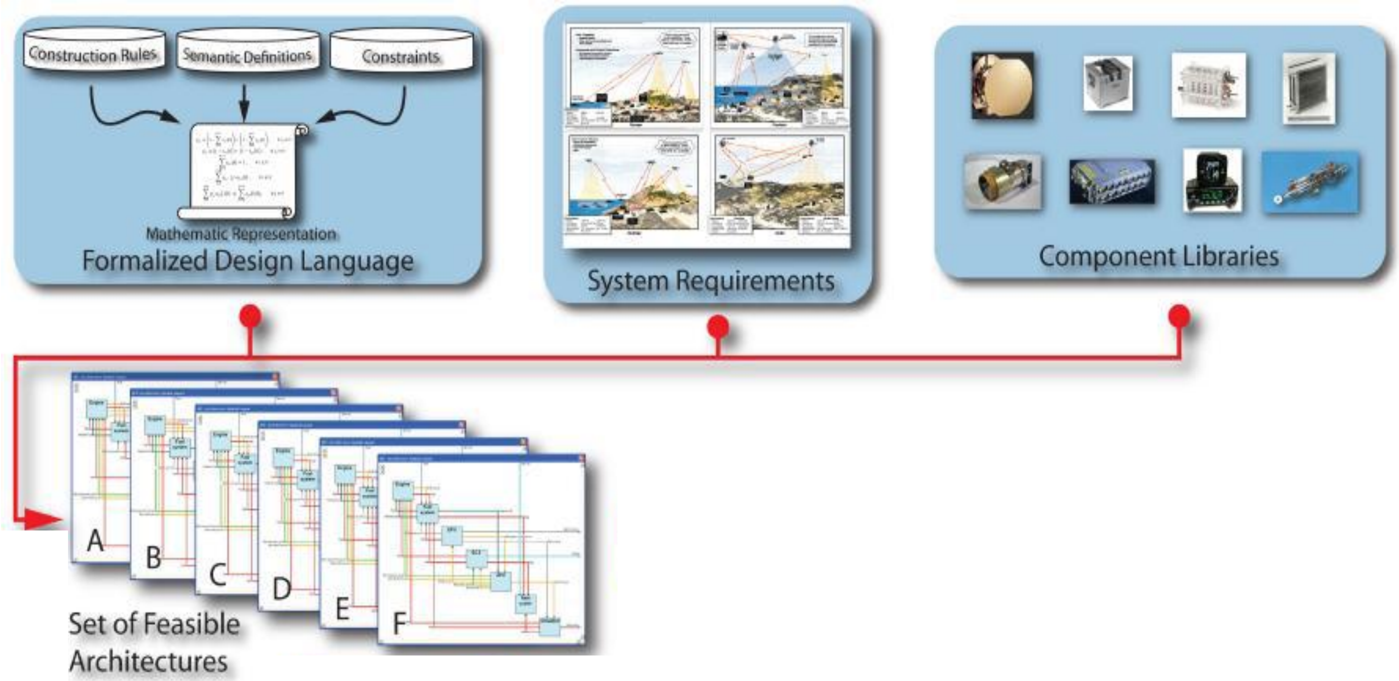
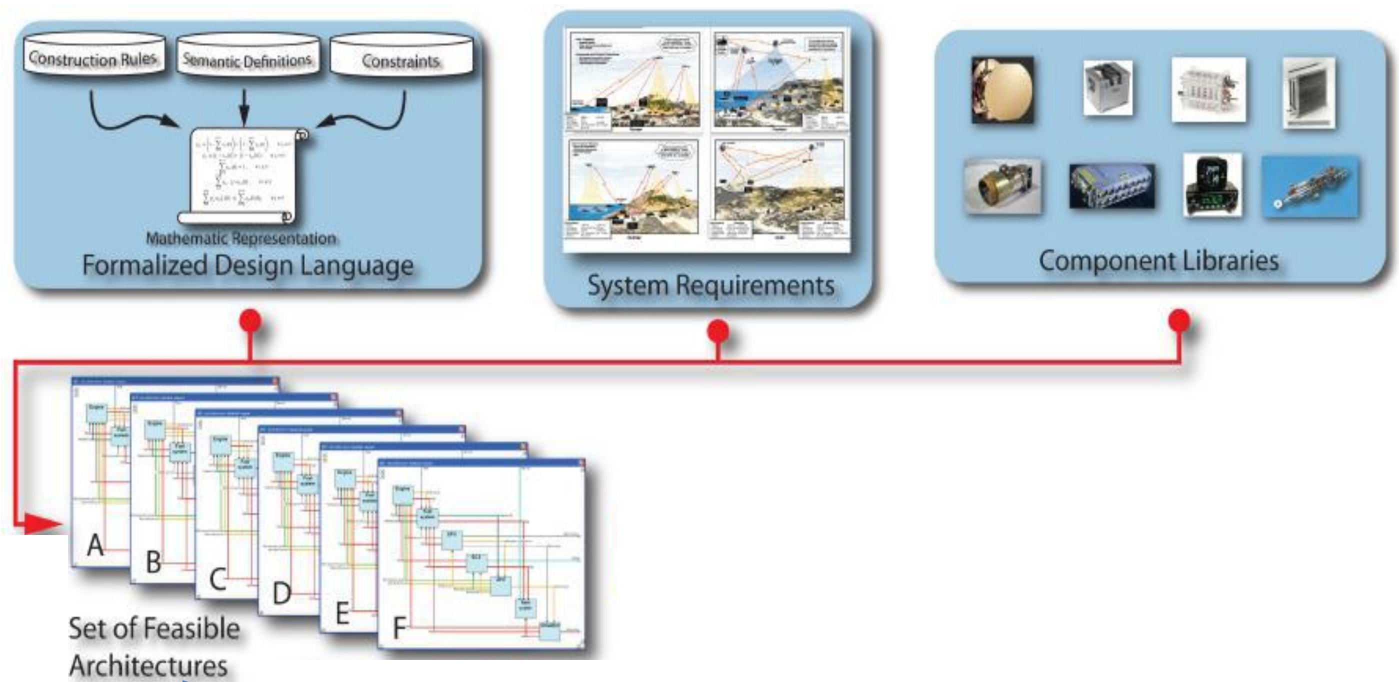


Figure F.2: Design flow to manage complexity in heterogeneous cyber-physical systems

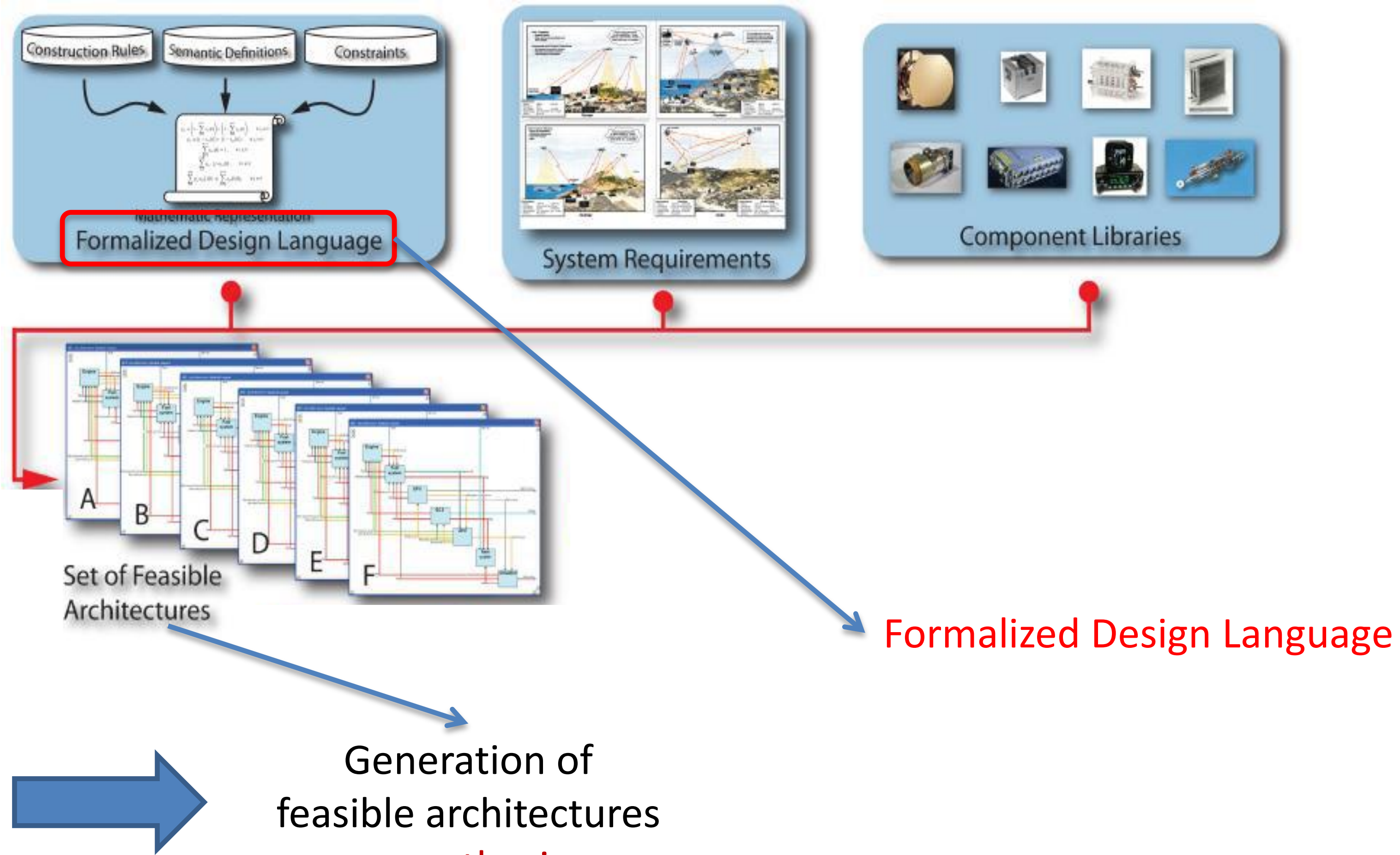
DARPA 2010 : Abstraction Based Complexity Management



Generation of
feasible architectures
= synthesis

Figure F.2: Design flow to manage complexity in heterogeneous cyber-physical systems

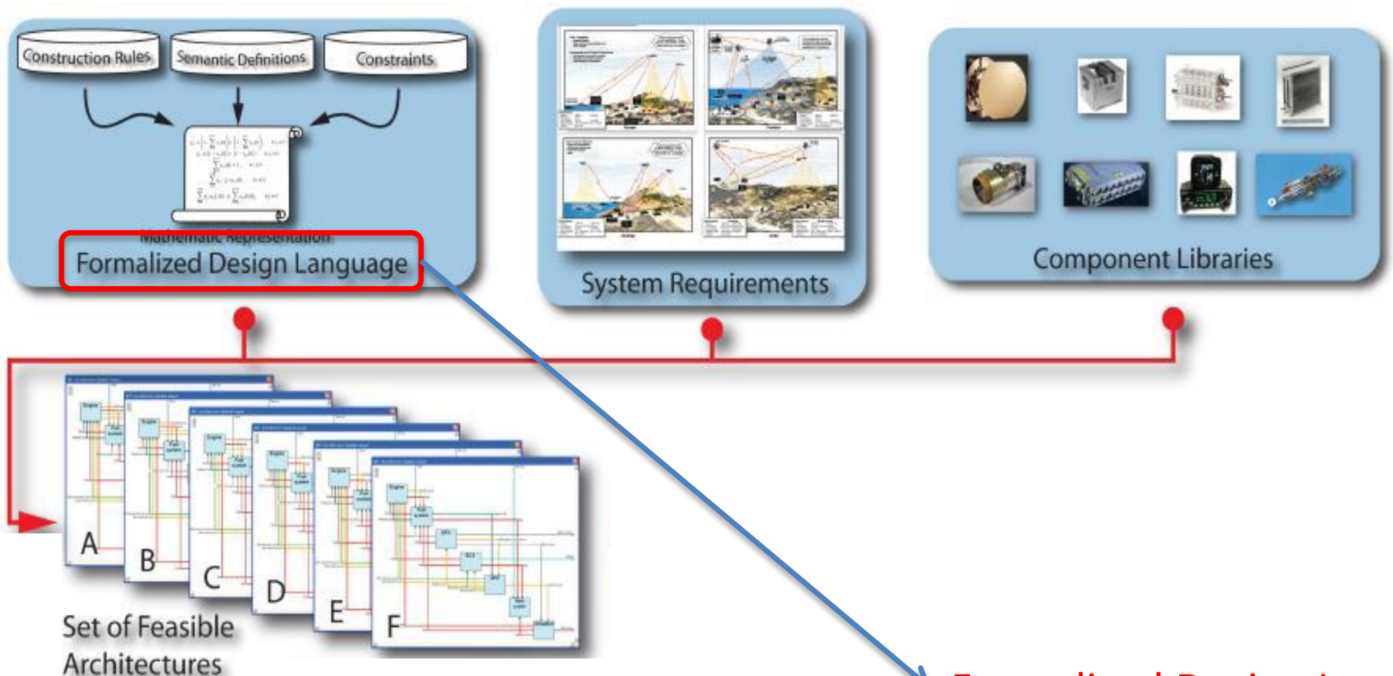
DARPA 2010 : Abstraction Based Complexity Management



Generation of feasible architectures = synthesis

Figure F.2: Design flow to manage complexity in heterogeneous cyber-physical systems

DARPA 2010 : Abstraction Based Complexity Management



Formalized Design Language

Generation of feasible architectures = synthesis

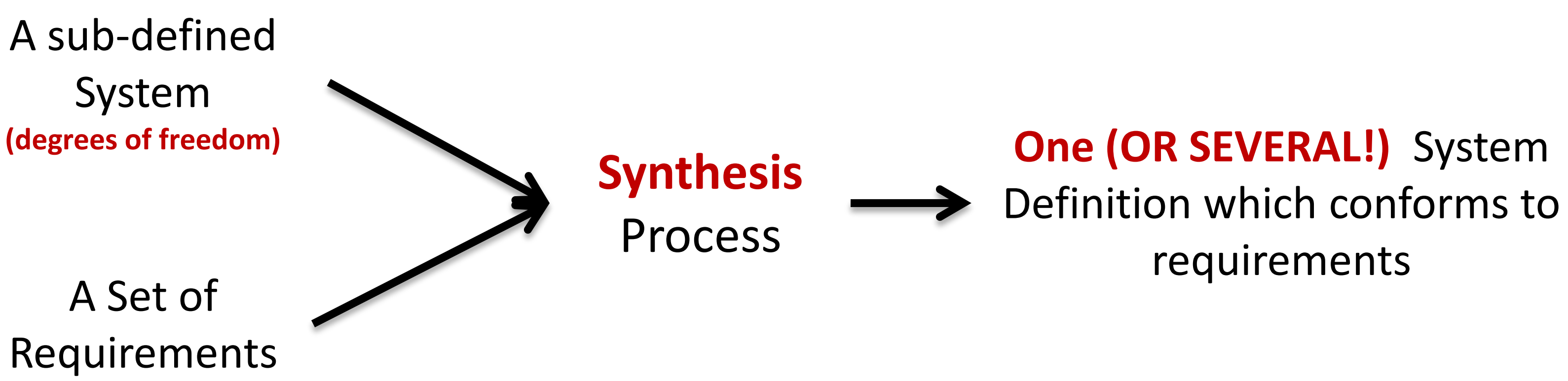
Our approach: using the DEPS language

Figure F.2: Design flow to manage complexity in heterogeneous cyber-physical systems

CONTEXT

MBSE HAS TO BE REVISED TO ADRESS SYNTHESIS PROBLEMS:

- Modeling languages to capture definitions of problem
- Synthesis software tools to fix, select, allocate …



The DEPS language
A Domain Specific Language

- **Declarative** MBSE language for problem specification (EBNF)
- **Object-oriented** Knowledge Representation (*Models* are classes, *elements* are instances)
  - class-instance model
  - inheritance, composition, association, polymorphism
  - some attributes can be sub-defined (*variables*)
- **Formal properties** encapsulated inside or between Models
  - equations, inequalities between algebraic expressions (IEEE 754)
  - data catalogs
- **Ontology for engineers**
  - quantities, dimensions, units

EXAMPLE

Sub-Defined Model

Model Partition ()
Constants
Variables
icpu: CpuIndex ;
Elements
Properties
End

Quantity

Quantity CpuIndex
Kind : Integer ;
Min : 1 ;
Max : 4 ;
Unit : u ;
End

Domain of possible values

Formal Property

Model Co-location (P1, P2)
Constants
Variables
Elements
P1 : Partition ;
P2 : Partition ;
Properties
P1.icpu = P2.icpu;
End

Constraint Declaration

The DEPS Studio IDE

An Integrated Modelling and Solving Environment

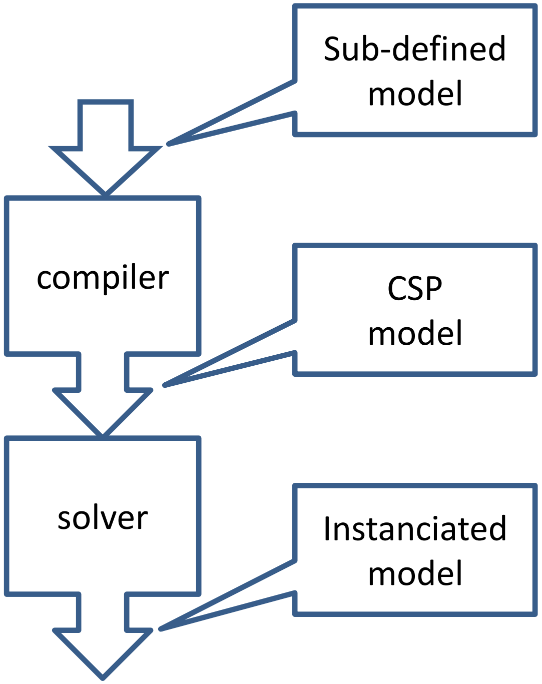
A SYNTHESIS TOOL CHAIN:

DEPS COMPILER

- Ahead-of-time with static type checking
- generation of sub-defined model instances with constraints

DEPS SOLVER

- constraint programming paradigm
- Purpose-built
- Mixed (integer/real) solving capabilities



IMA APPLICATION

DEPLOYMENT OF AIRCRAFT SYSTEMS ON AN AVIONICS PLATFORM

- The aim is to size the processing capacity of the platform and to generate a correct by construction multi-system deployment

- Build all DEPS models of this deployment problem (including the *problem* itself)
  - Systems (functions) and a sub-defined platform (structure)
    - computational resources allocation is unknown
  - design requirements and design constraints (properties)

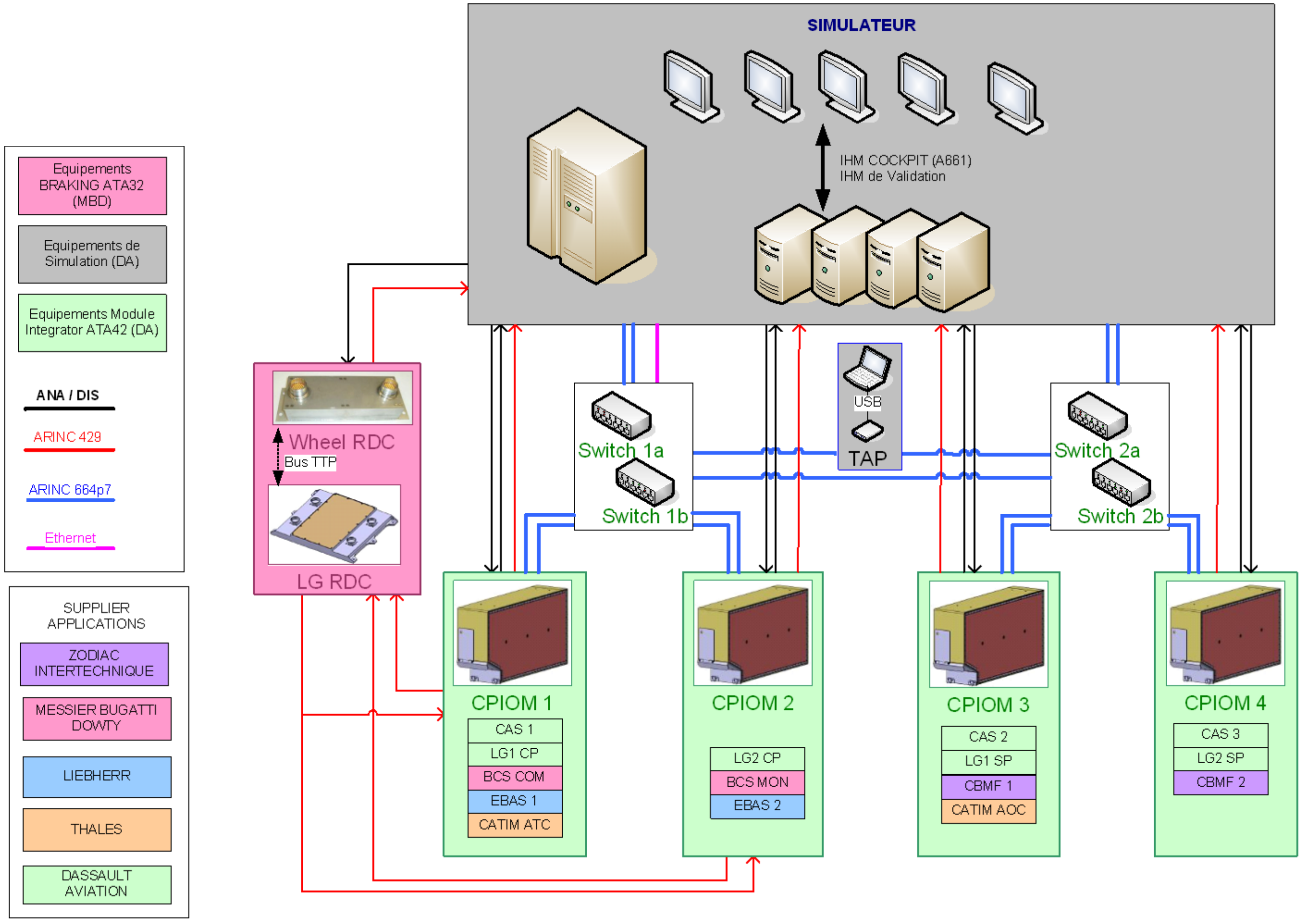
Vocabulary

- Avionics functions are composed of channels
- (Processing) channels represent different ways to realize the same function
- Channels are composed of (software) IMA applications
- IMA application are composed of partitions (≈ threads)
- CPIOM (computing process and I/O modules) are IMA calculators

Requirements and constraints

- **The safety requirements** are issued from a preliminary Safety Analysis of systems leading to :
  - Duplication, triplication … of processing channels, redundancy of applications
  - Material segregation of resources used by duplicated paths or applications
  - ➢ formal DEPS models of architectural patterns
- **The capacity constraints** express the memory limit of each CPIOM for the partitions deployed on it
- **The security requirements** express a segregation between functions (and not channels)

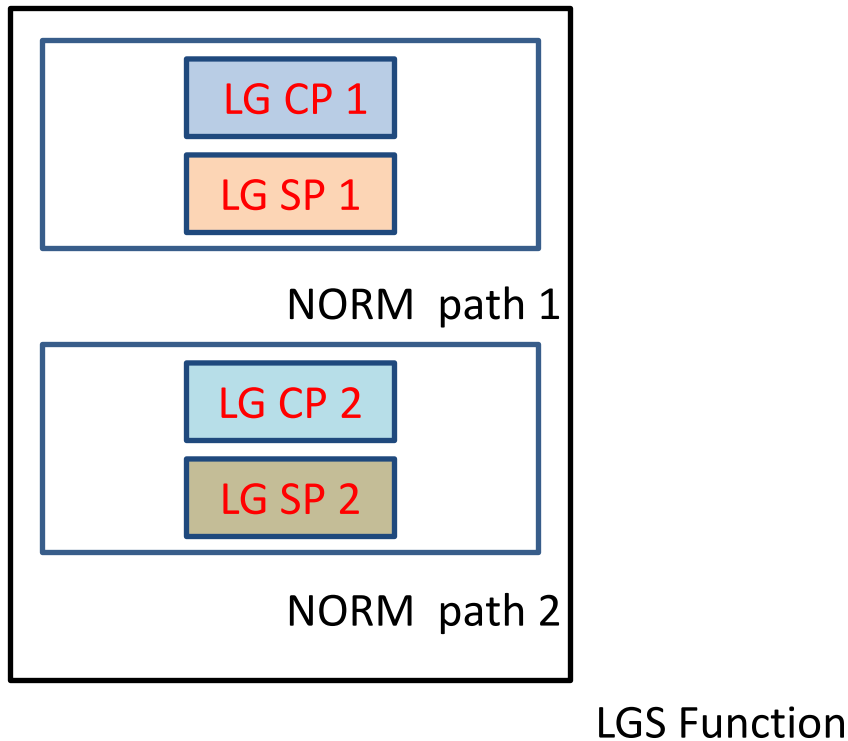
One (amongst other) Deployment Solution



Avionics functions :

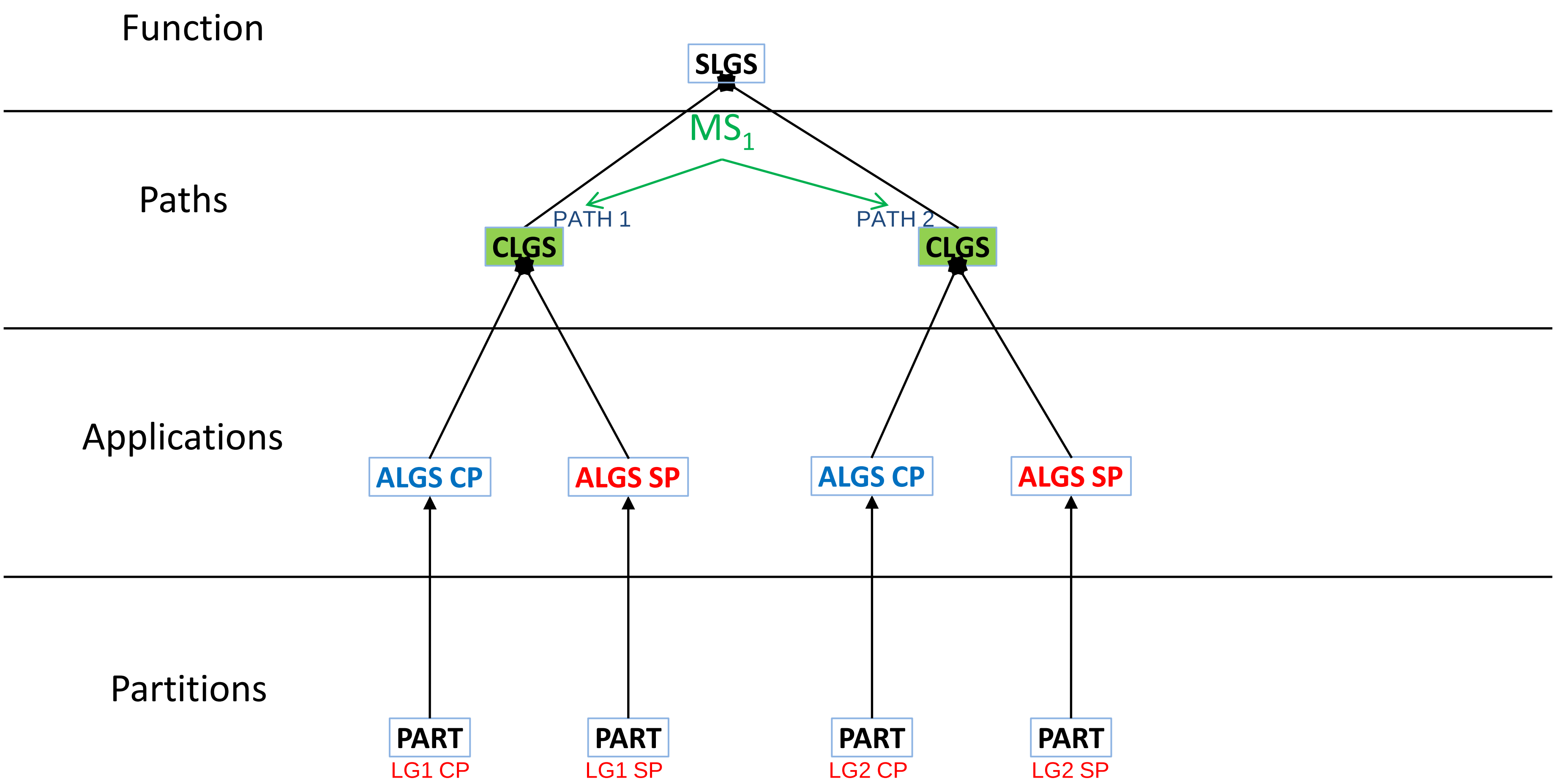
- CATIM (communication)
- LGS (landing gear system)
- BCS (breaking control system)
- EBAS (air sampling)
- CBMF (breaking monitoring)
- CAS (crew alerting system)
- AFCS (flight control system)

Pattern example

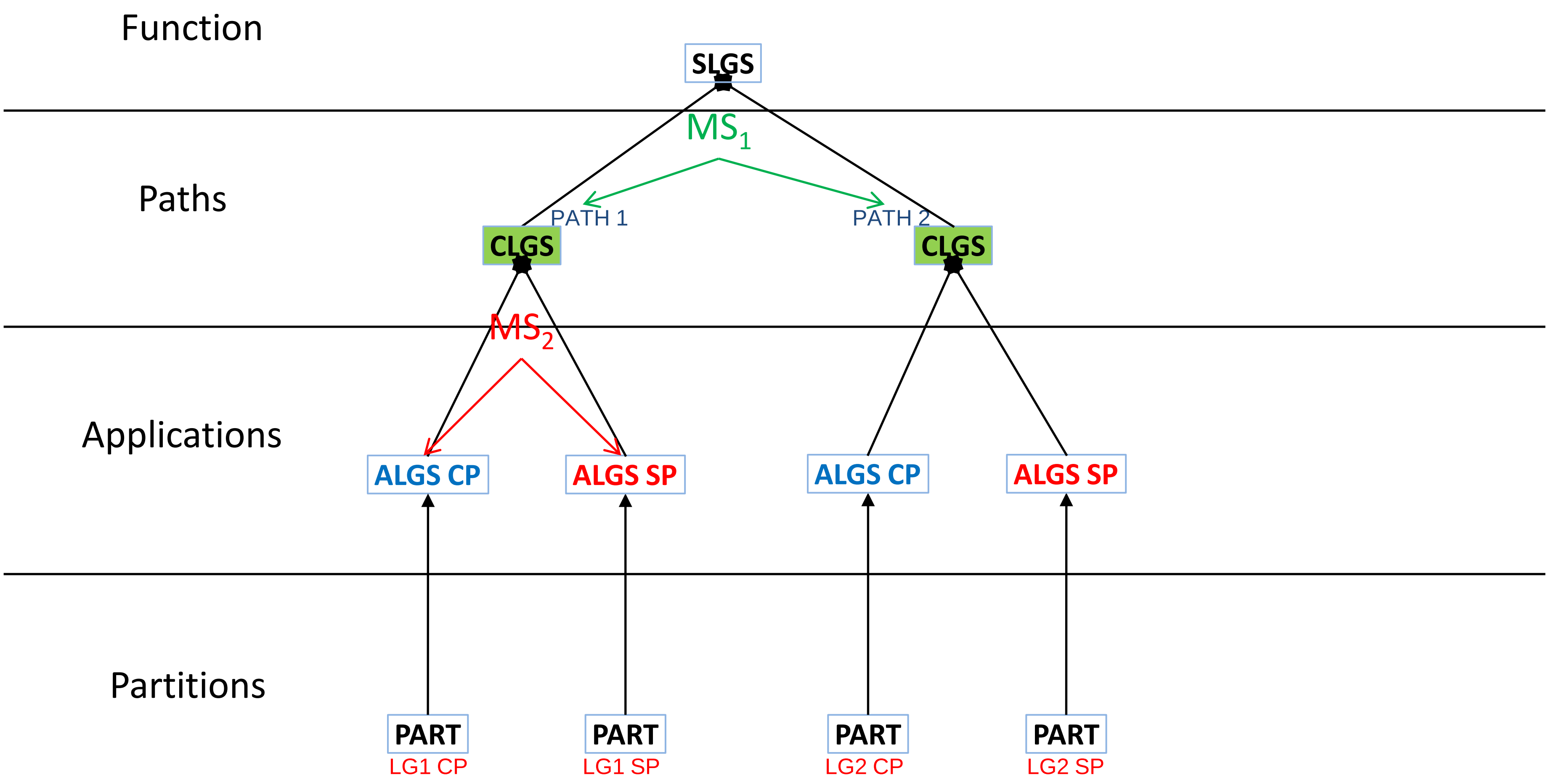


CP = control path
CS = safety path

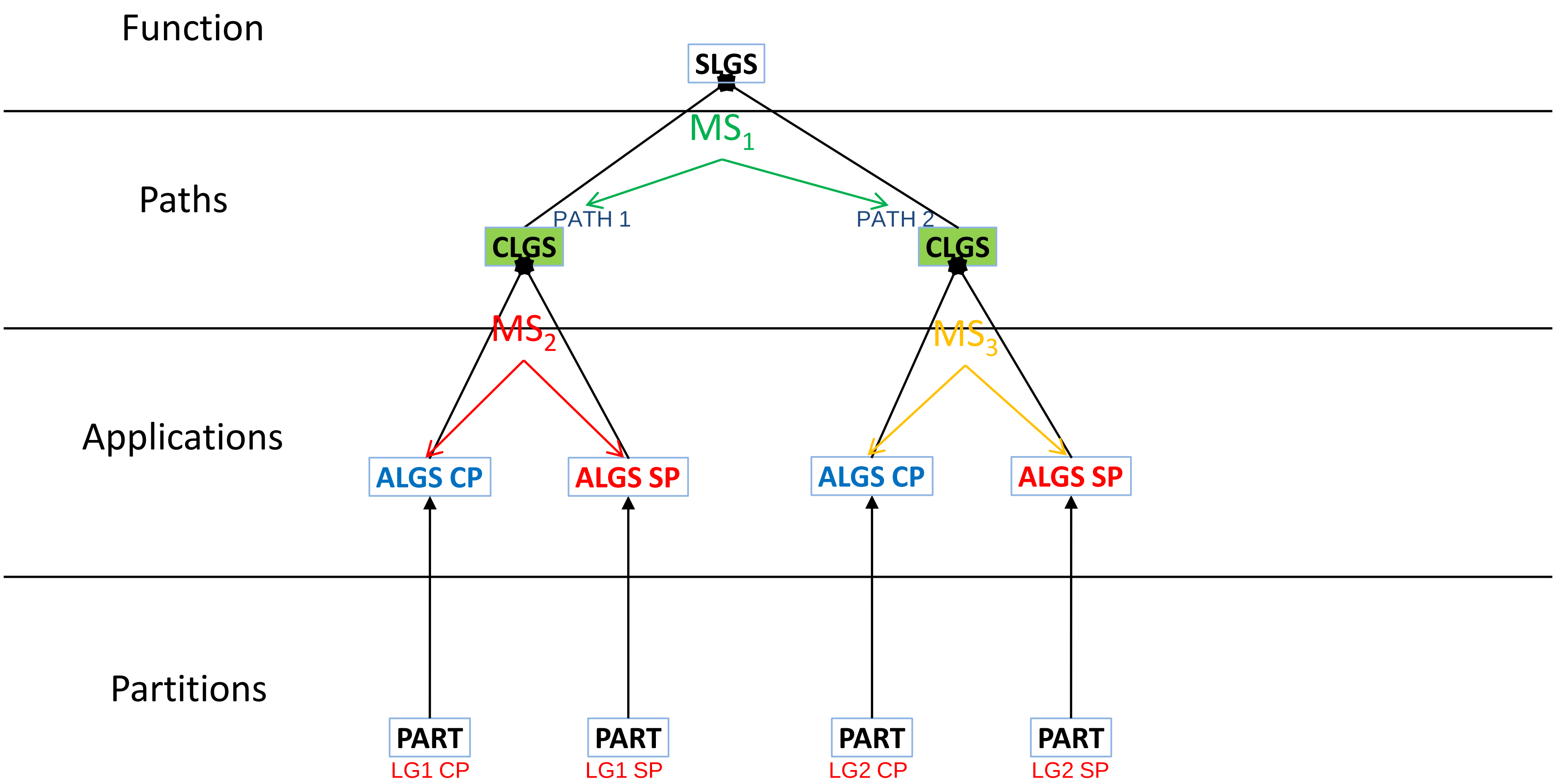
LGS: model of pattern



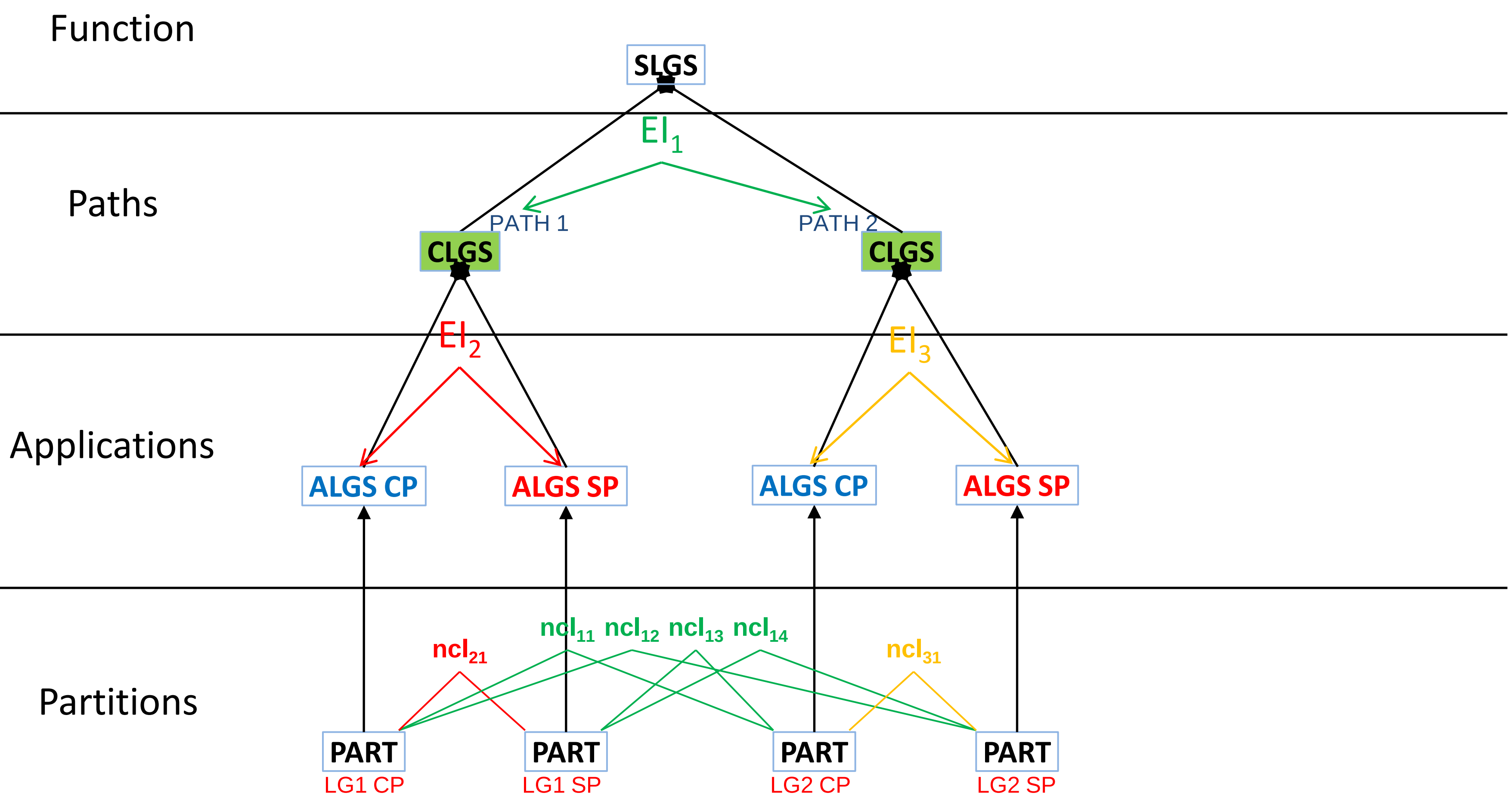
LGS: model of pattern



LGS: model of pattern

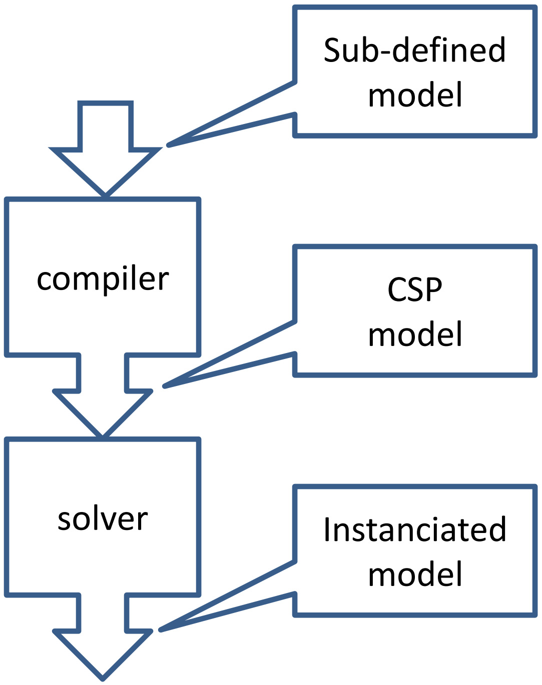


LGS: deployment constraints

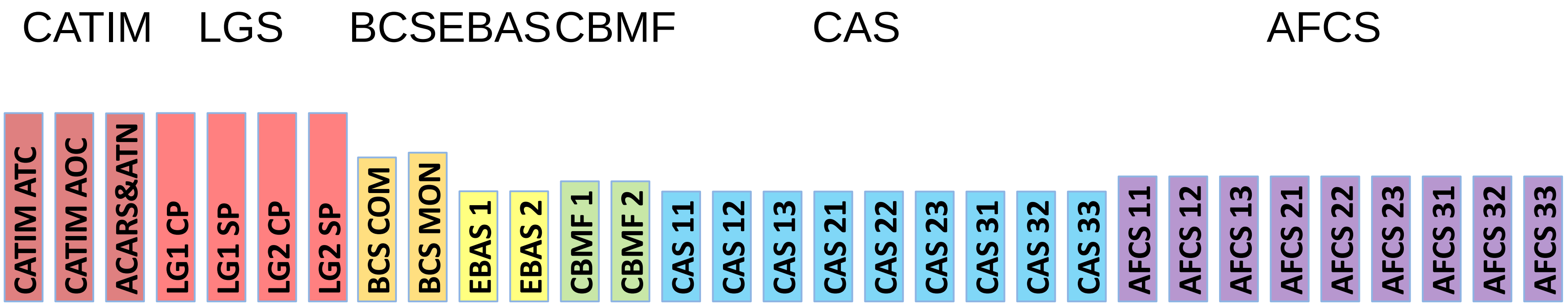


Modelling and solving process with DEPS Studio

- **Modelling the problem with DEPS language**
  - Aircraft functions, processing channels, paths, applications, partitions
  - CPIOM
  - Safety requirements
  - Capacity constraints
  - Security requirements
- **Compiling the problem**
- **Solving the problem**
  - Generation of one or several solutions (zero is possible too)

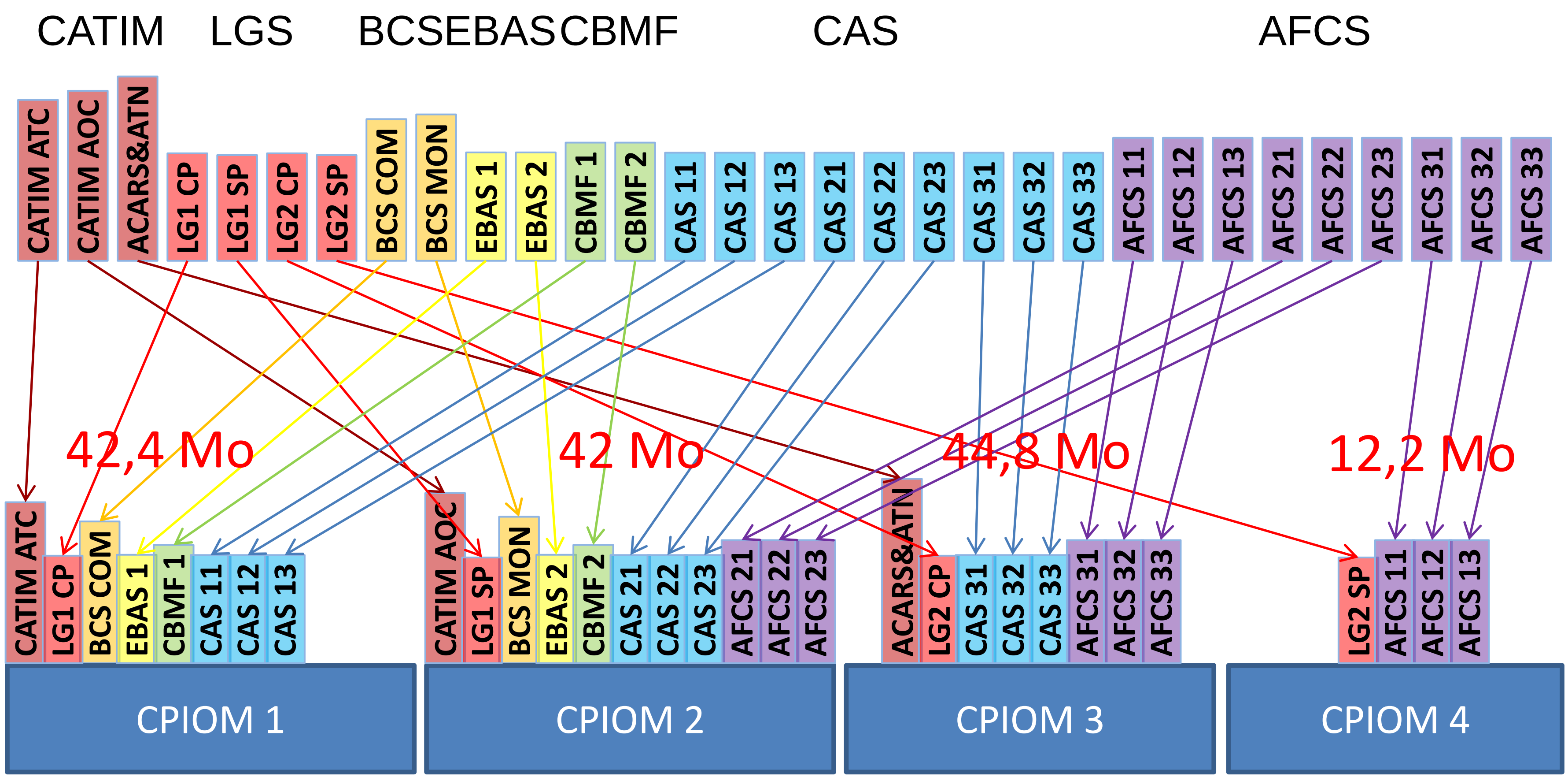


Demo.

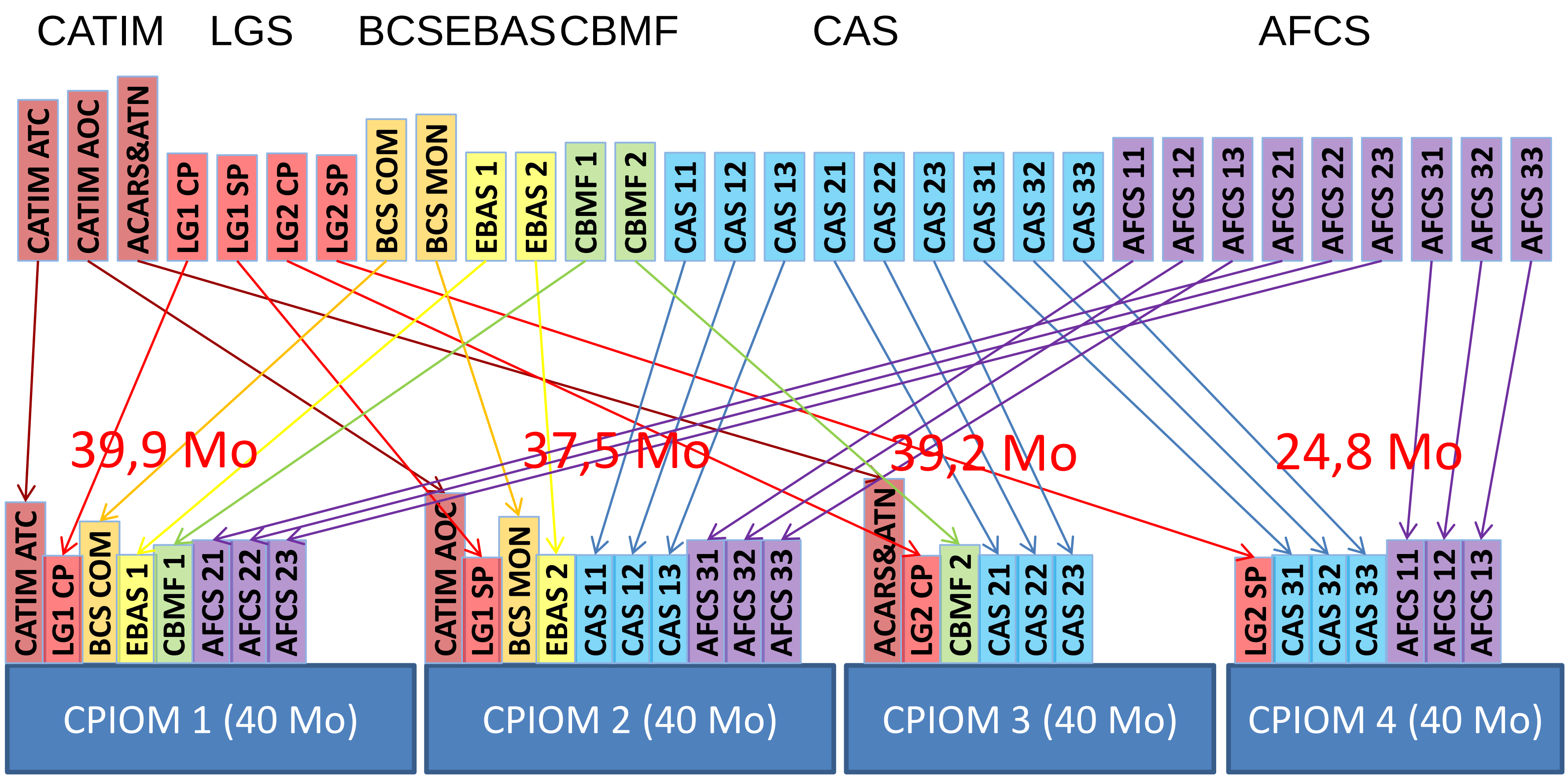


Deployment of several aircraft functions

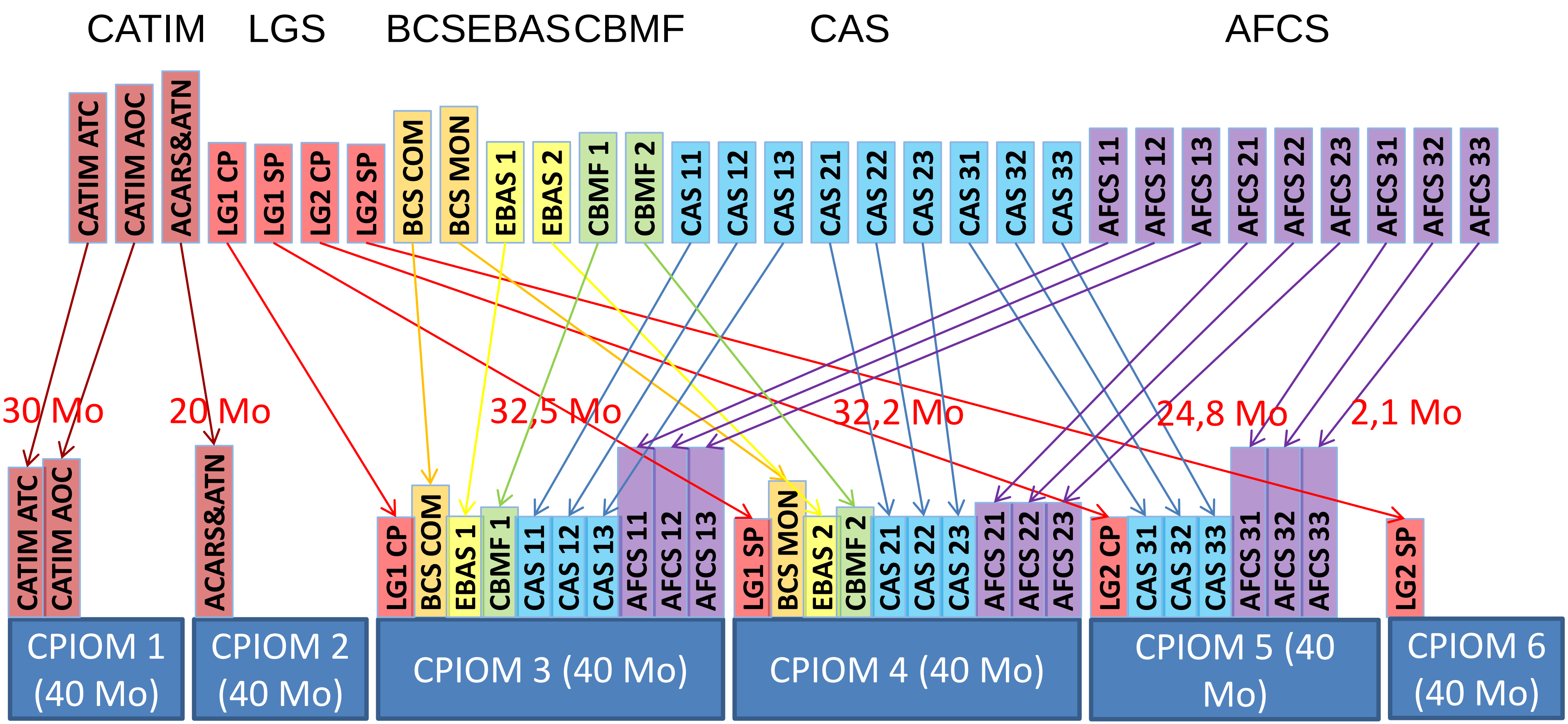
One solution respecting the safety patterns



One solution respecting the capacity constraints too



One solution respecting the security levels of the functions too



SUMMARY

The DEPS language

A high level problem modeling language:

- to represent a sub-defined system, its objects and their organization
- to express requirements as properties between models

DEPS Studio

- an integrated problem solving tool chain to address design problems:
  - sizing, deployment, configuration, architecture synthesis

IMA APPLICATION

- a deployment problem set in an elegant way and solved efficiently

ONGOING AND FUTURE WORK

Develop DEPS language

- Quantities extension and dimensional analysis
- Collectors of elements (model instances)
- Constraints expressed from experimental data or from simulation data

- DEPS is supported by the DEPS Link non profit organization
  - www.depslink.com

Test on system problems

- IRT SystemX: I(SC)^2 project
  - Verification of the fail-safe nature of an on-board electrical generation and distribution system architecture (ATA 24) (Dassault Aviation DPR)
  - Sizing of a satellite optical instrument (Thales TRT)
  - Allocation of heterogeneous computing resources to system functions (Thales TRT)
- Electrical Engineering
  - Configuration and sizing of Battery architecture

    (PhD thesis Supmeca/UTC)
  - Synthesis of Offshore wind turbine network architecture (SupMeca/IREENA)

THANKS FOR YOUR ATTENTION

QUESTIONS ?