The DEPS Project

(DEsign Problem Specification)

GDR GPL

GT Ingénierie des exigences & Génie Logiciel pour les CPS

January 18th, 2019

Laurent ZIMMER
Dassault Aviation

Pierre-Alain YVARS
SupMéca-Quartz

Outlines

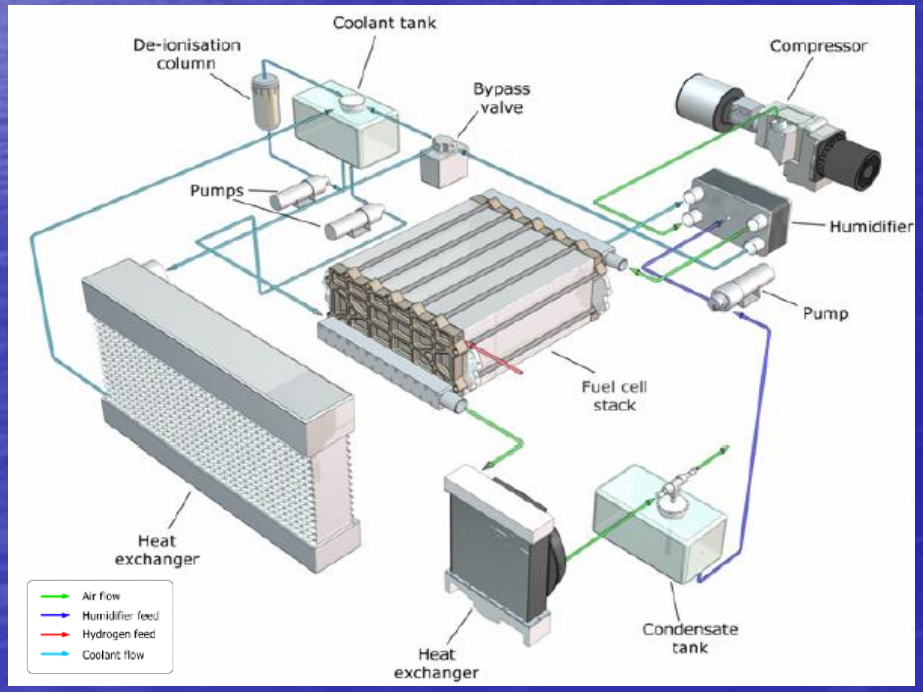
- Context
- The DEPS project
- The DEPS language
- The DEPS solver
- DEPS by example
- Use-case IMA
- Ongoing studies and developments

Context

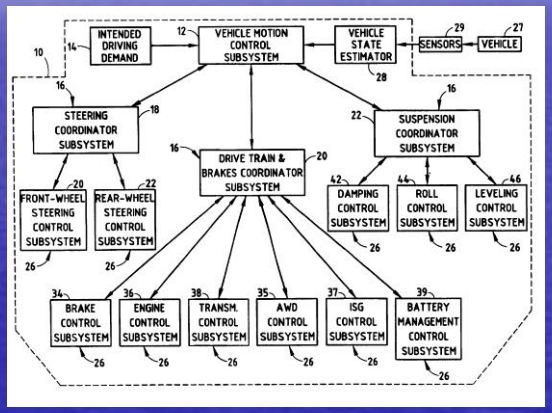
A system is a construct or collection of different elements that together produces results not obtainable by the elements alone (INCOSE Definition).

…

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)



Some system design characteristics

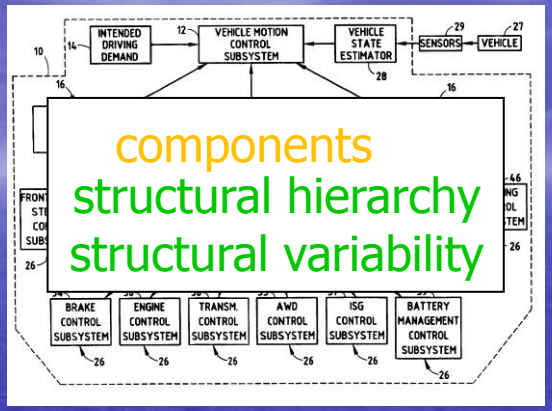


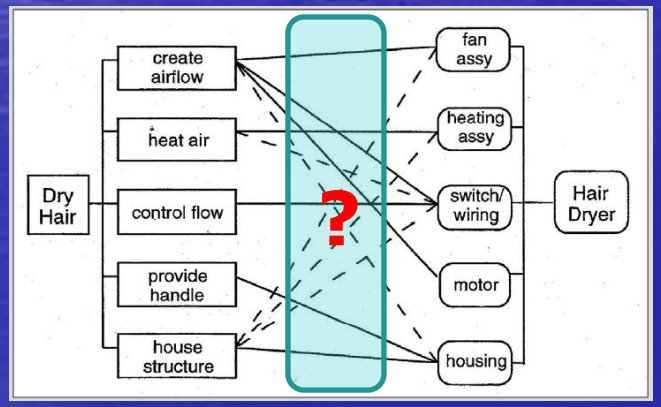
functions
functional hierarchy
+
functional variability

components
structural hierarchy
+
structural variability

A Mixed Calculus world !

Moreover .. There are missing links

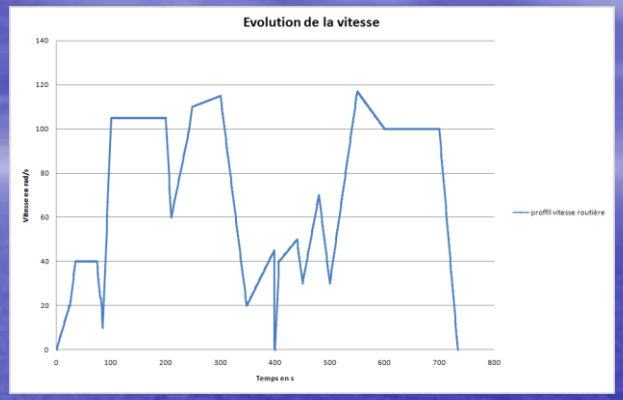


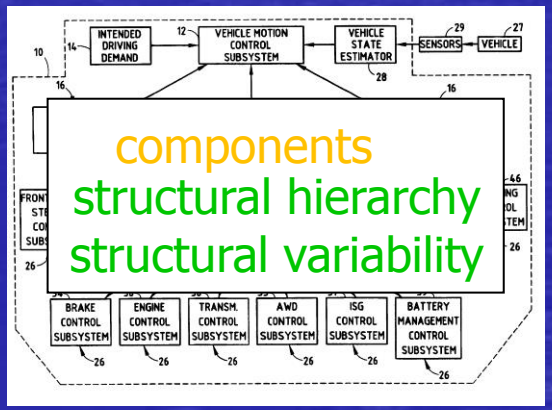


function – structure **allocations**

Moreover .. There are a missing links

use-case – functioning modes **associations**





What is common to Design Problems ?

- Sizing
  - Some Design Parameters are not fixed
- Configuration
  - Number and type of some components are not fixed
- Allocation
  - Resources required by some elements are not fixed
- Architectural design
  - A mixing of everything above

What is common to Design Problems ?

A System to be designed is sub-defined

Designing a System means completing a sub-defined system which conforms to Requirements

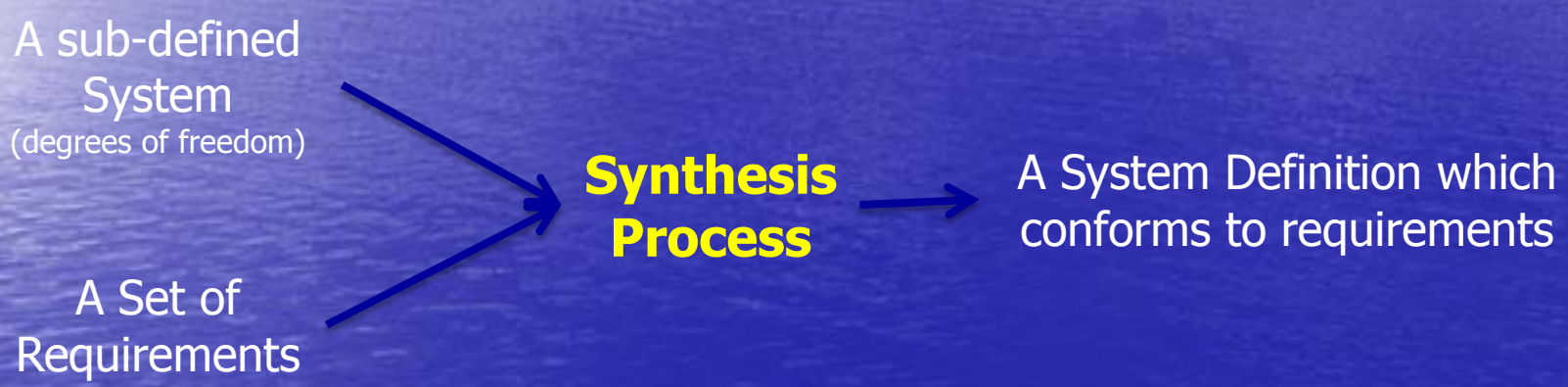
Our Manifesto

Solving a Design Problem

Is

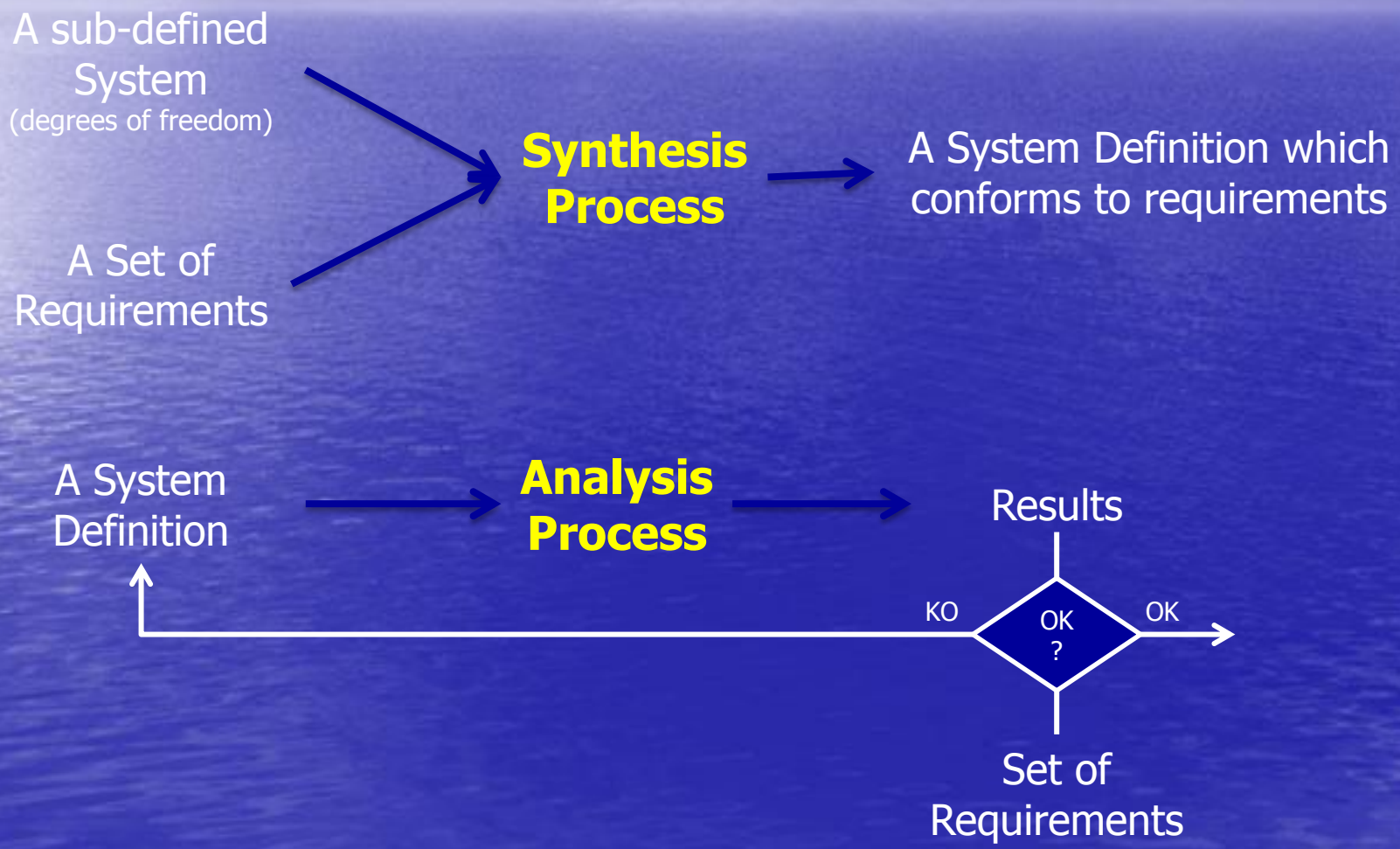
Completing a Sub-defined Model

A Design Problem is a matter of Synthesis

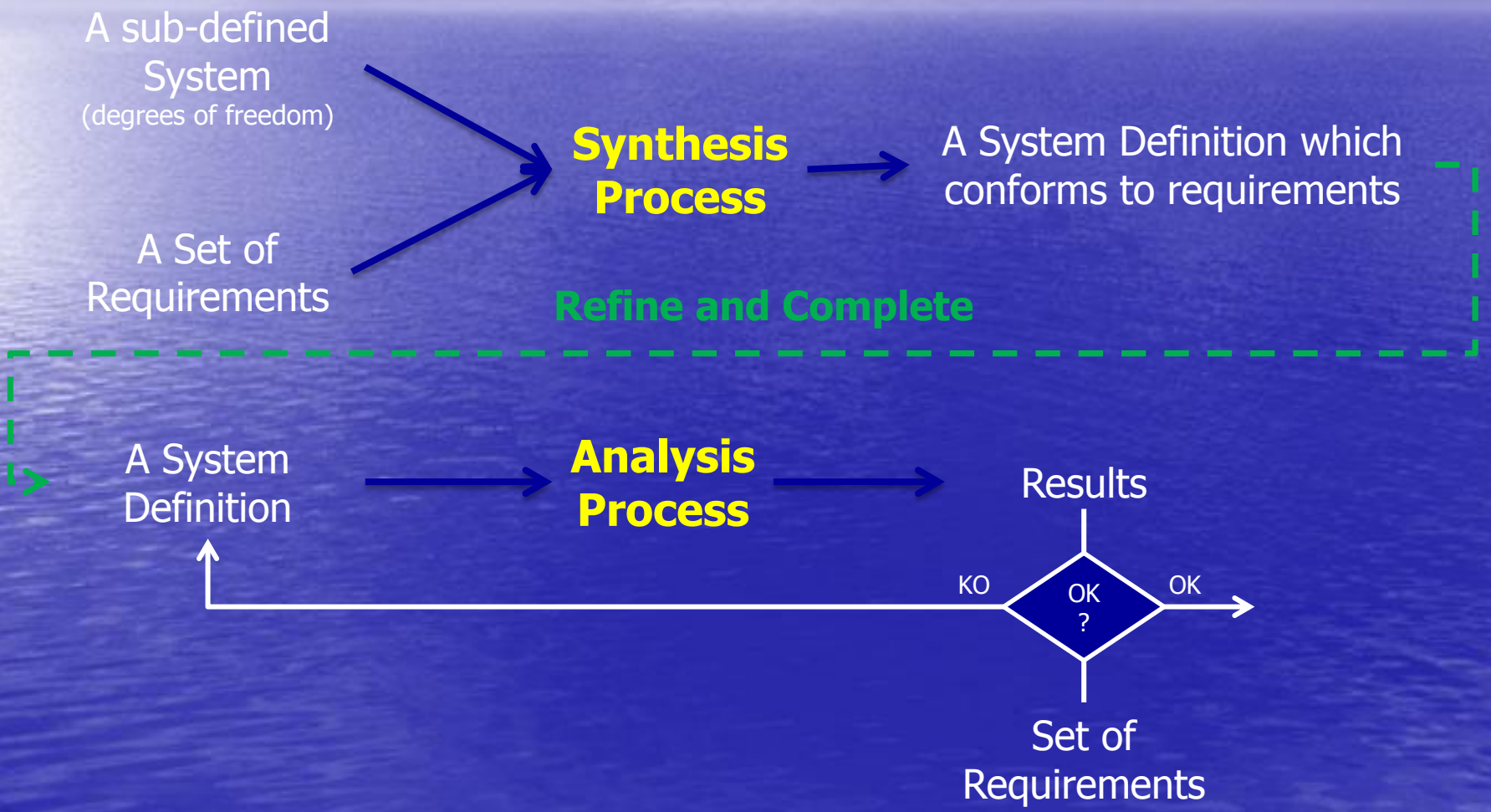


Synthesis isn't Analysis

Synthesis and Analysis are complementarity



Synthesis and Analysis are complementary



Outlines

- Context
- The DEPS project
- The DEPS language
- The DEPS solver
- DEPS by example
- Use-case IMA
- Ongoing studies and developments

The DEPS Project

- Develop a formal modeling language for specifying System Design Problems
- Develop Problem Solving methods and tools

The DEPS Project

- Target: engineering, embedded, real-time, cyber-physical, software-intensive … Systems
- Design steps: preliminary design, architecture generation, system integration, system verification…

Outlines

- Context
- The DEPS project
- The DEPS language
- The DEPS solver
- DEPS by example
- Use-case IMA
- Ongoing studies and developments

The DEPS language declarative features

- Model-based Knowledge Representation
  - inheritance, composition, aggregation, polymorphism
  - Attributes
    - constants, variables
    - integer or real values
  - Properties
    - algebraic equations or inequalities
- Ontology for engineers
  - quantities, dimensions, units

The DEPS language the compiler

- Ahead-of-time
- Package management
- Parsing/Error handling
- Generation of sub-defined model instances

The DEPS Language

Object-Oriented Language + Constraint Solving Language = DEPS Language

Sub-defined Models + Constraints = Models of Properties

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

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

```
P1.icpu = P2.icpu;
```

```
Model colocalisation (P1, P2)
Constants
Variables
Elements
 P1 : Partition ;
 P2 : Partition ;
Properties
  P1.icpu = P2.icpu;
End
```

Outlines

- Context
- The DEPS project
- The DEPS language
- The DEPS solver
- DEPS by example
- Use-case: Synthesis of avionics embedded systems
- Ongoing studies and developments

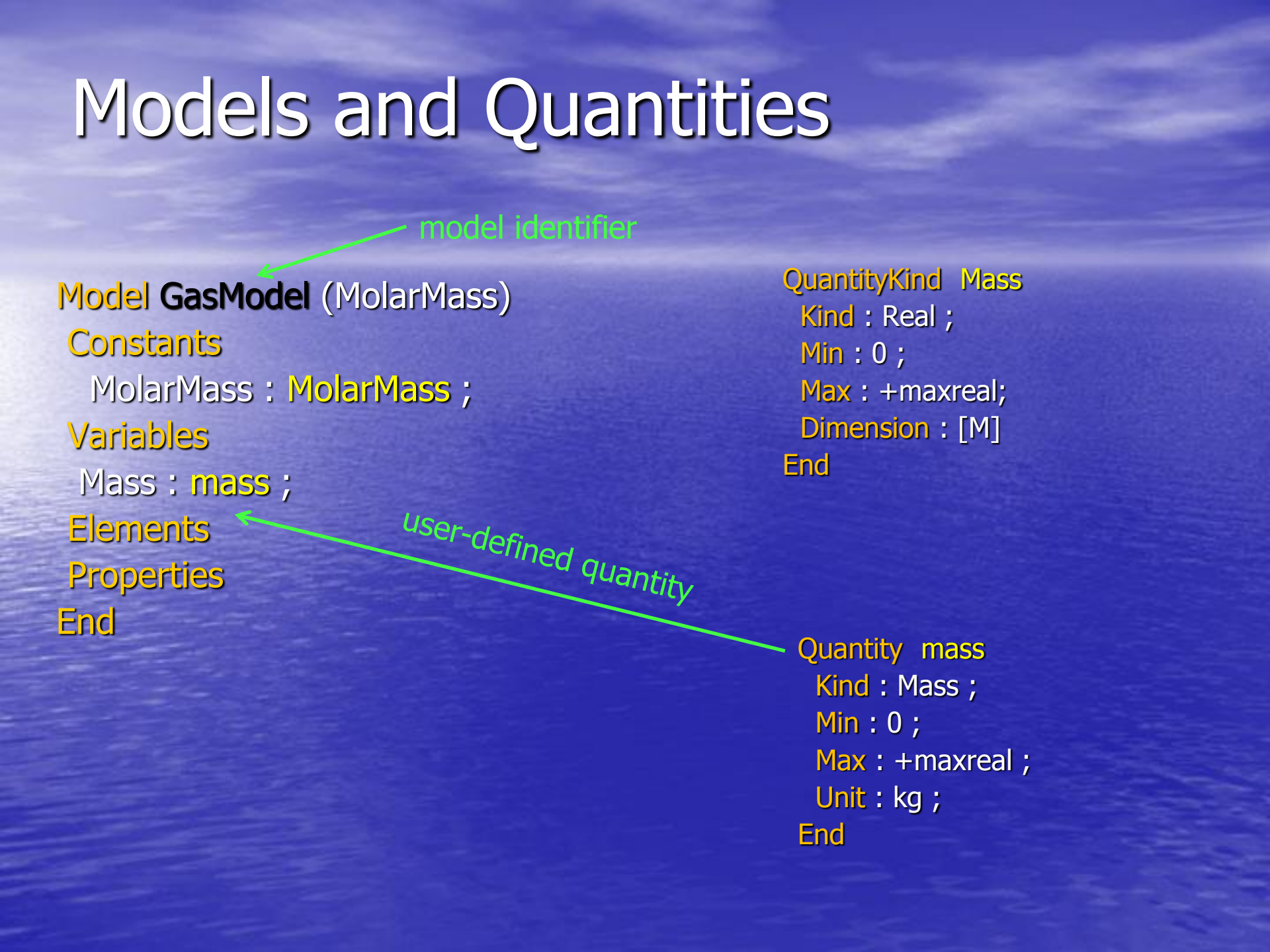
DEPS Solver

- A Need of solving capabilities addressing :
  - under constrained problems,
  - mixed non linear mathematical problems,
  - both equations and inequalities,
  - other relations.
- A Constraint based oriented solver
  - constraint programming on mix domains
- Built for dealing with DEPS models
- An object oriented architecture
  - extensible

Outlines

- Context
- The DEPS project
- The DEPS language
- The DEPS solver
- DEPS by example
- Use-case IMA
- Ongoing studies and developments

Models and Quantities



Part and shared models

reference binding

```
Model Tank(p, t, Gas)
 Constants
  R : Real = 8.314 ;
  p : Pressure ; instance declaration
  t :  Temperature ;
 Variables
  V : Volume ;
 Elements
 Gas : GasModel ;
 Properties
  p*V= (Gas.Mass/Gas.MolarMass)*R*t;
End
```

Instance construction
O2 part of Problem

reference binding
O2 shared by T1 and T2

```
Model Problem()
 Constants
 Variables
 Elements
  O2:GasModel(0.032);
  H2:GasModel(0.002);
  T1,T2:Tank(2.56e+7, 300, O2);
  T3:Tank(7.00e+7, 300, H2);
 Properties
  O2.Mass = 10 ;  (* or T1.Gas *)
  T1.V+T2.V< 500 ;
End
```

Aliases

```
Model Gas (ckilo, molarMass)
 Constants
  ckilo : DollarCostPerKilo;
  molarMass : MolarMass;
 Variables
  M : mass ;
  expr CoutStockage : DollarCost  ;
 Elements
 Properties
  CoutStockage := ckilo*M ;
End
```

Alias declaration

Alias definition

Universal constants

```
Model circle ()
 Constants
 Variables
  Diameter, Circumference: length ;
 Elements
 Properties
  Circumference = uPi*D ;
End
```

Derived models (inheritance)

```
Model Component (index)
 Constants
  index : ComponentIndex ;
 Variables
  I : intensity ;
 Elements
   P1, P2 : Port() ;
 Properties
   P1.I :=  I;
   P2.I := -I;
End
```

```
Model Port
 Constants
 Variables
  V : voltage ;
  expr I : intensity ;
 Elements
 Properties
End
```

```
Model Resistor(R) extends Component
 Constants
  R : Resistor;
 Variables
 Elements
 Properties
  P1.V-P2.V = R*I;
End
```

1 resistor model:

- 3 variables (unknowns)
- 1 equation
- 2 expressions

Model Signature

```
Model Component (index)
Constants
  index : ComponentIndex ;
 Variables
  I : intensity ;
 Elements
   P1, P2 : Port() ;
 Properties
   P1.I :=  I;
   P2.I := -I;
End
```

```
Model Resistor(R) extends Component [ComponentIndex]
 Constants
  R : Resistor;
 Variables
 Elements
 Properties
  P1.V-P2.V = R*I;
End
```

```
Model Resistor() extends Component [ComponentIndex]
 Constants
 Variables
 R : Resistor;
 Elements
 Properties
  P1.V-P2.V = R*I;
End
```

Outlines

- Context
- The DEPS project
- The DEPS language
- The DEPS solver
- DEPS by example
- Use-case IMA
- Ongoing studies and developments

Use-case:
Synthesis of avionics embedded systems

Modélisation d'exigences et synthèse d'architecture de plateforme informatique embarquée

Laurent Zimmer (Direction de la Prospective)

DASSAULT AVIATION

CORAC AME

Processus Générique de Définition d'Architecture

- Objectifs :
  - Définir un processus de conception d'architecture de plateforme informatique embarquée, pour un périmètre fonctionnel étendu à l'ensemble des domaines.
  - Via un ensemble d'étapes successives, le processus doit générer de façon assistée une architecture, répondant aux besoins opérationnels et aux contraintes de ses fonctions embarquées (de sûreté de fonctionnement, de sécurité des données, etc..).
  - Ce processus de génération doit, autant que possible, être prouvé correct par construction. Pour ce faire, il s'appui sur un ensemble de modèles formels (i.e., reposant sur des notations mathématiques), et sur des techniques d'optimisation et de recherche de solutions (de type résolution de contraintes).

CORAC AME

Processus Générique de Définition d'Architecture

- Objectifs :
  - Définir un processus de conception d'architecture de plateforme informatique embarquée, pour un périmètre fonctionnel étendu à l'ensemble des domaines.
  - Via un ensemble d'étapes successives, le processus doit générer de façon assistée une architecture, répondant aux besoins opérationnels et aux contraintes de ses fonctions embarquées (de sûreté de fonctionnement, de sécurité des données, etc..).
  - Ce processus de génération doit, autant que possible, être prouvé correct par construction. Pour ce faire, il s'appui sur un ensemble de modèles formels (i.e., reposant sur des notations mathématiques), et sur des techniques d'optimisation et de recherche de solutions (de type résolution de contraintes).

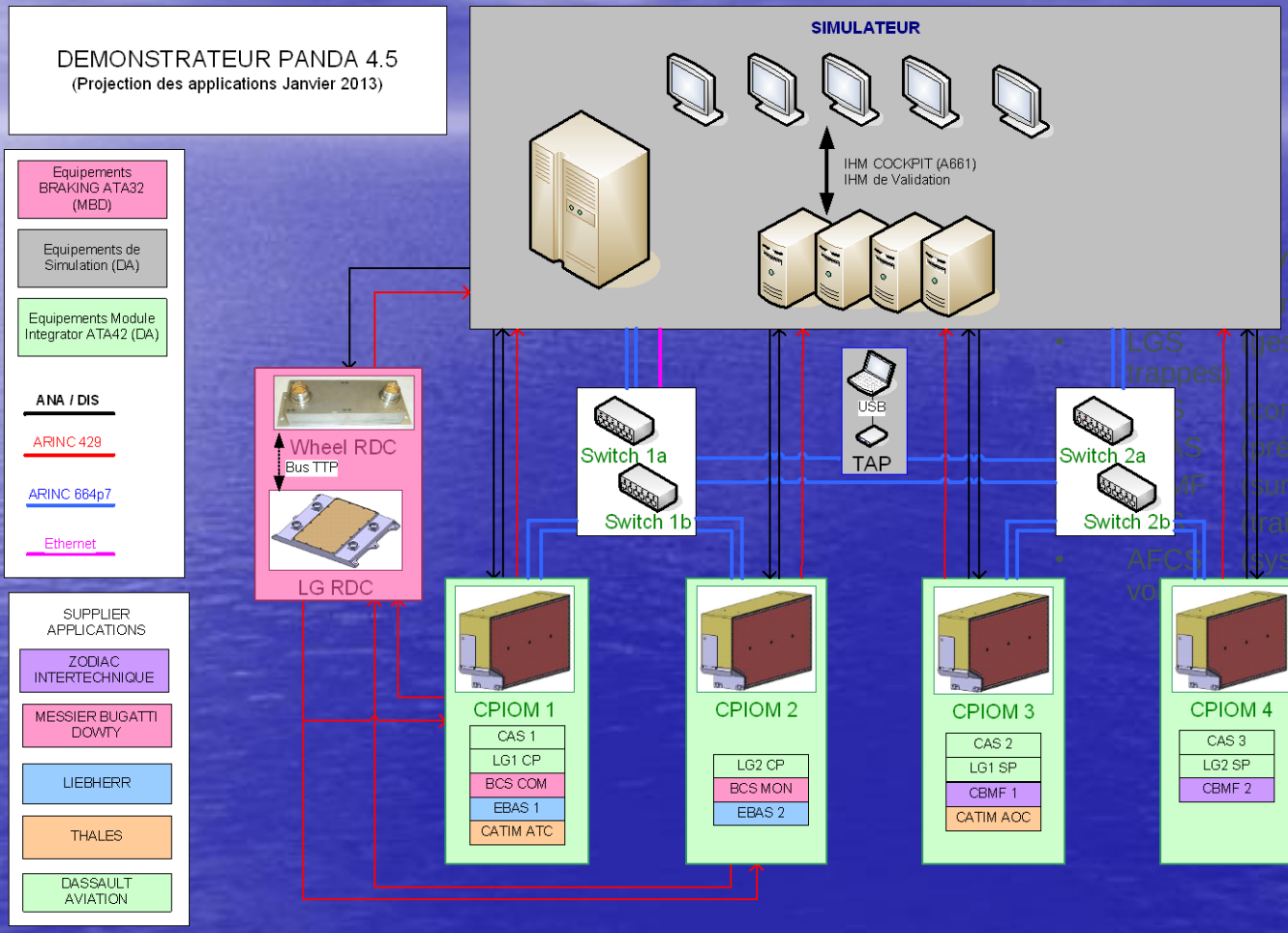
Notre Propos

- Modéliser formellement (en DEPS) le maximum d'exigences et de contraintes de conception qui portent sur les fonctions avion

- Générer (ou vérifier) un déploiement correct par construction sur la plateforme cible en dimensionnant (si besoin) cette dernière

Modélisation en DEPS

- Modélisation système :
  - Modèles des fonctions avion, des canaux, des voies, des applications, des partitions
- Modélisation des éléments de plateforme
  - Modèle des calculateurs
- Modélisation d'exigences :
  1) Modèles de patrons de sûreté de fonctionnement des systèmes
  2) Modèles de contraintes de capacité
  3) Modèles de sécurité inter-systèmes


DEMONSTRATEUR PANDA 4.5
(Projection des applications Janvier 2013)
Equipements BRAKING ATA32 (MBD)
Equipements de Simulation (DA)
Equipements Module Integrator ATA42 (DA)
ANA / DIS
ARINC 429
ARINC 664p7
Ethernet
SUPPLIER APPLICATIONS
ZODIAC INTERTECHNIQUE
MESSIER BUGATTI DOWTY
LIEBHERR
THALES
DASSAULT AVIATION
SIMULATEUR
IHM COCKPIT (A661)
IHM de Validation
Wheel RDC
Bus TTP
LG RDC
Switch 1a
Switch 1b
USB
TAP
Switch 2a
Switch 2b
CPIOM 1
CAS 1
LG1 CP
BCS COM
EBAS 1
CATIM ATC
CPIOM 2
LG2 CP
BCS MON
EBAS 2
CPIOM 3
CAS 2
LG1 SP
CBMF 1
CATIM AOC
CPIOM 4
CAS 3
LG2 SP
CBMF 2

Formalisation des exigences de sûreté de fonctionnement

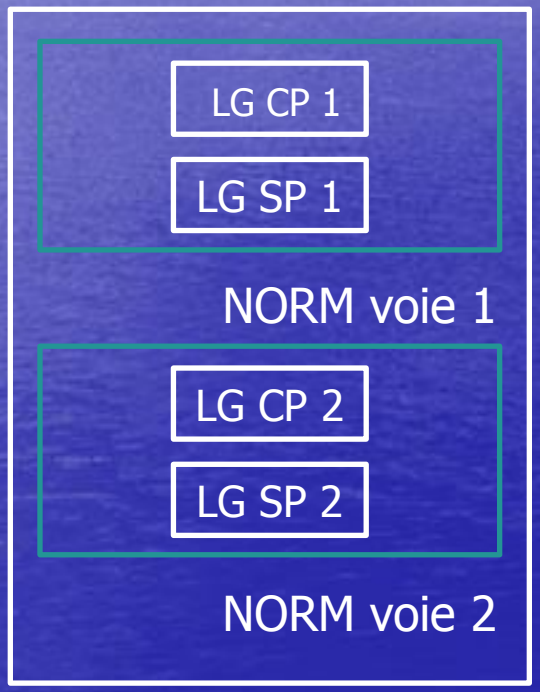
- Les experts en sûreté de fonctionnement étudient les cas de panne de chaque fonction avion et en fonction de la criticité de celles-ci préconisent :
  - des duplications, des triplications de chaînes de traitement, des redondances d'applications …
  - des ségrégations des ressources utilisées par les chaines ou les applications …

➢ Production de patrons d'architecture (schémas)

➢ Définition d'exigences de sûreté de fonctionnement associées (texte)

➢ *FORMALISATION des patrons et des exigences en DEPS*

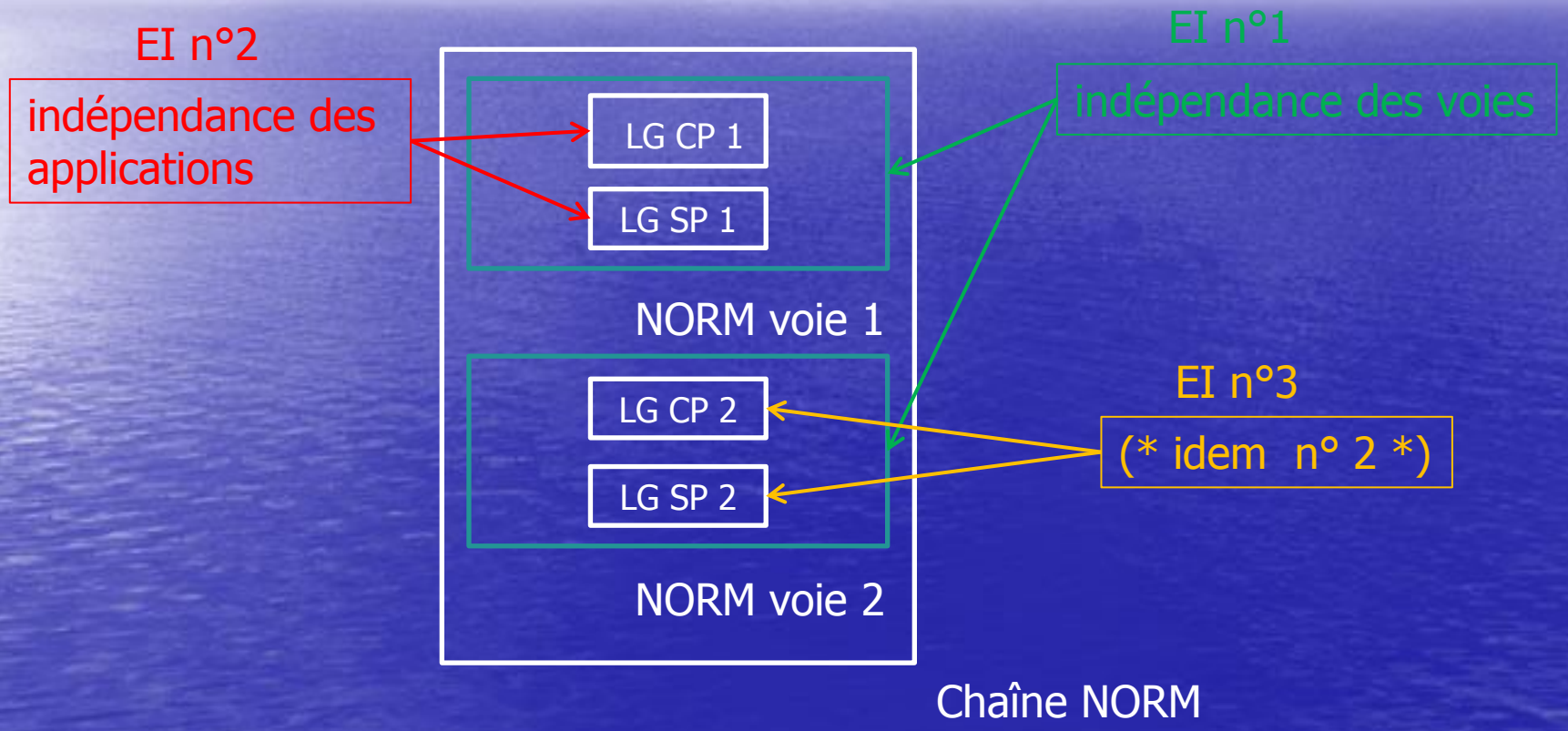
Patron d'architecture fonction LGS



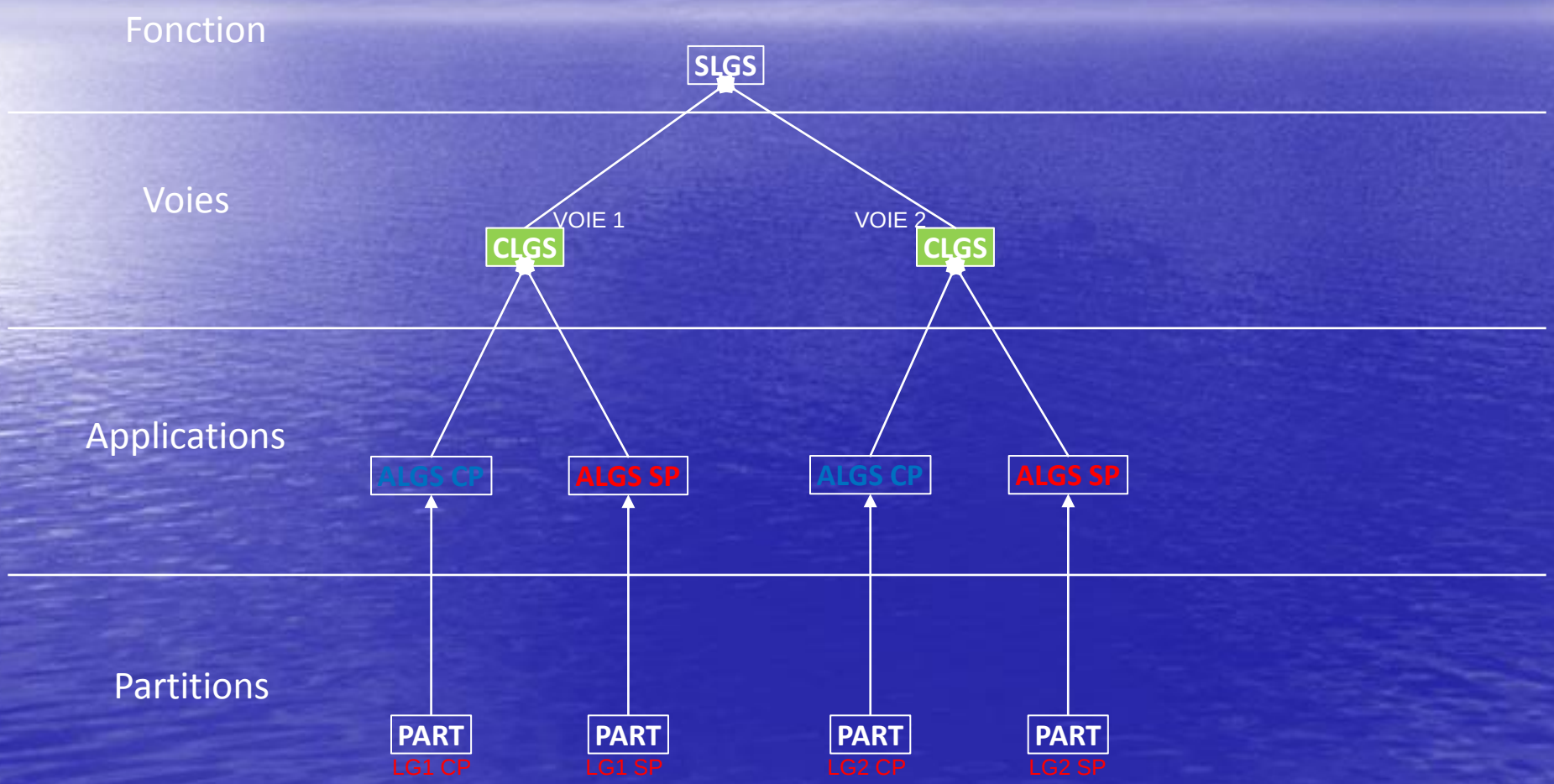
CP = control path
SP = safety path

Chaîne NORM

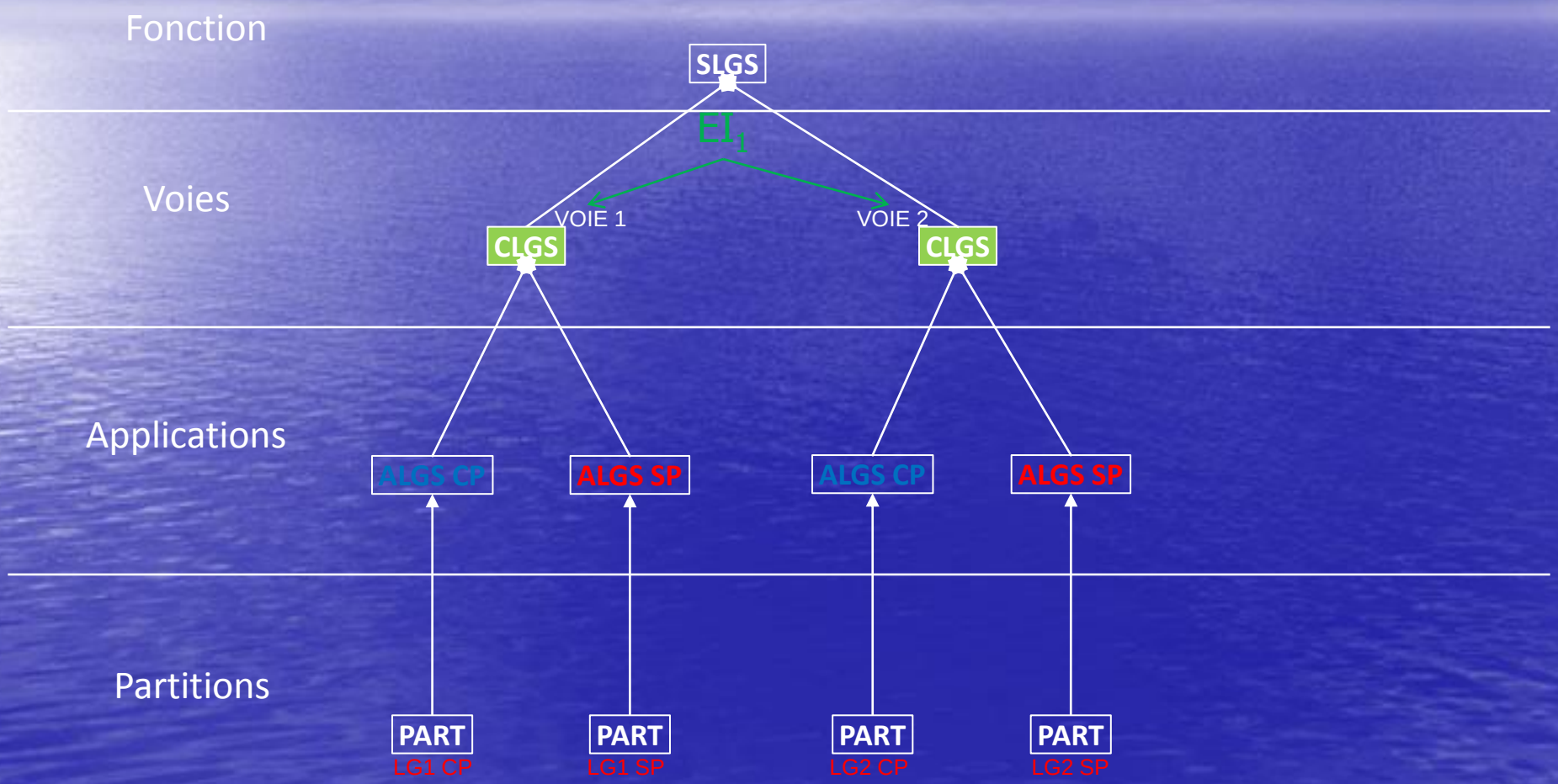
Exigences de Sûreté de Fonctionnement fonction LGS



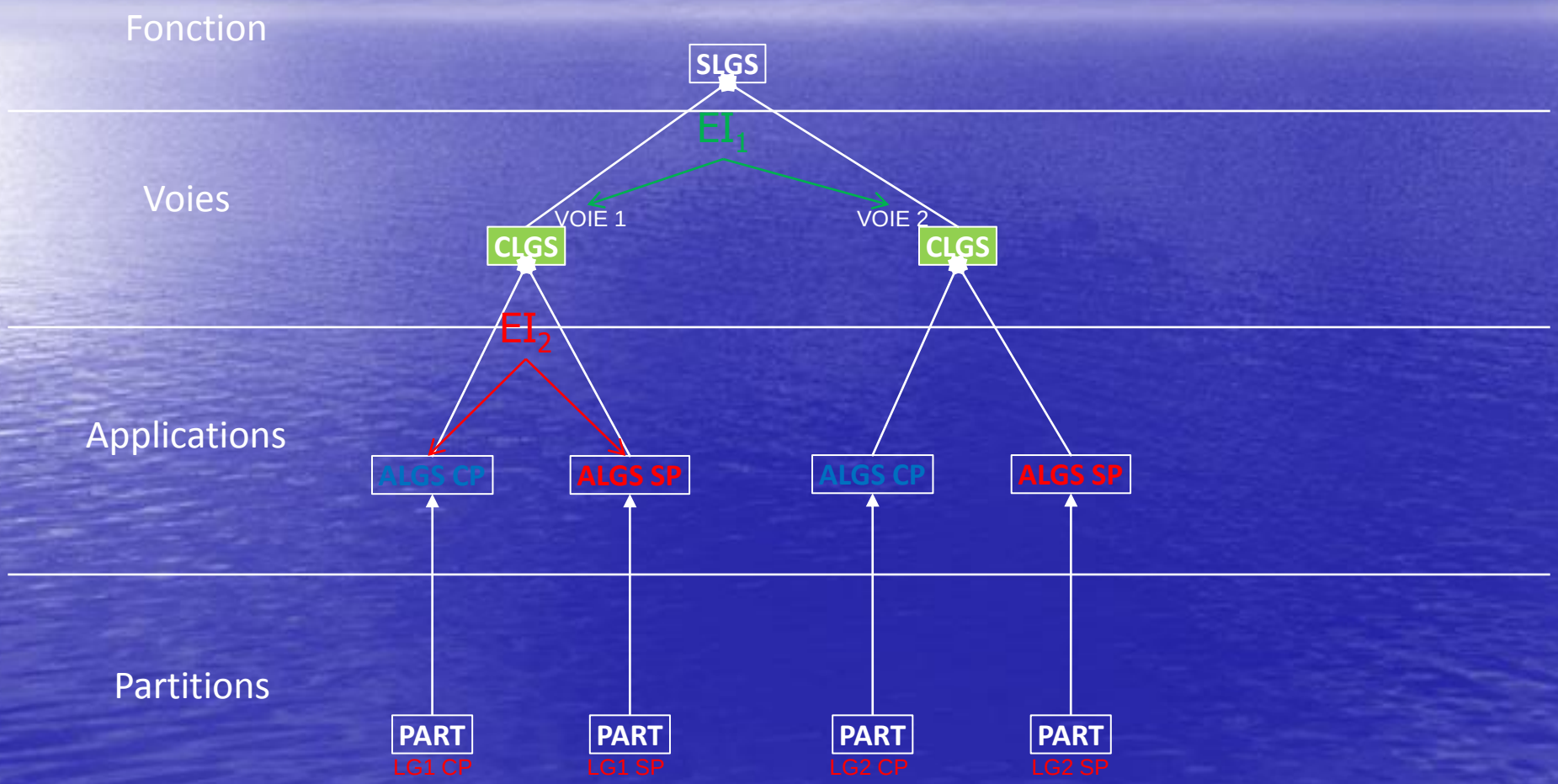
LGS : modélisation du patron



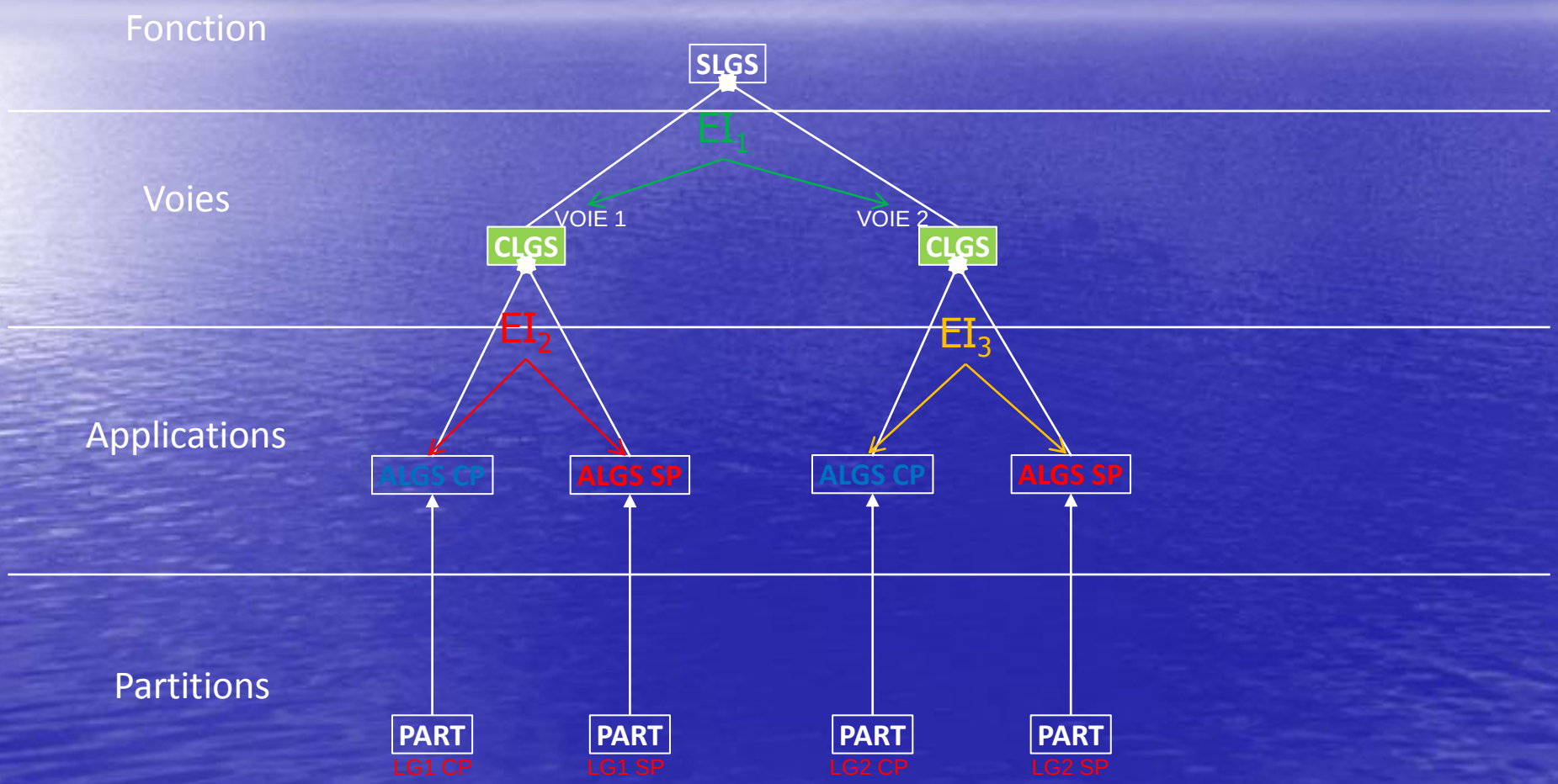
LGS : modélisation des exigences



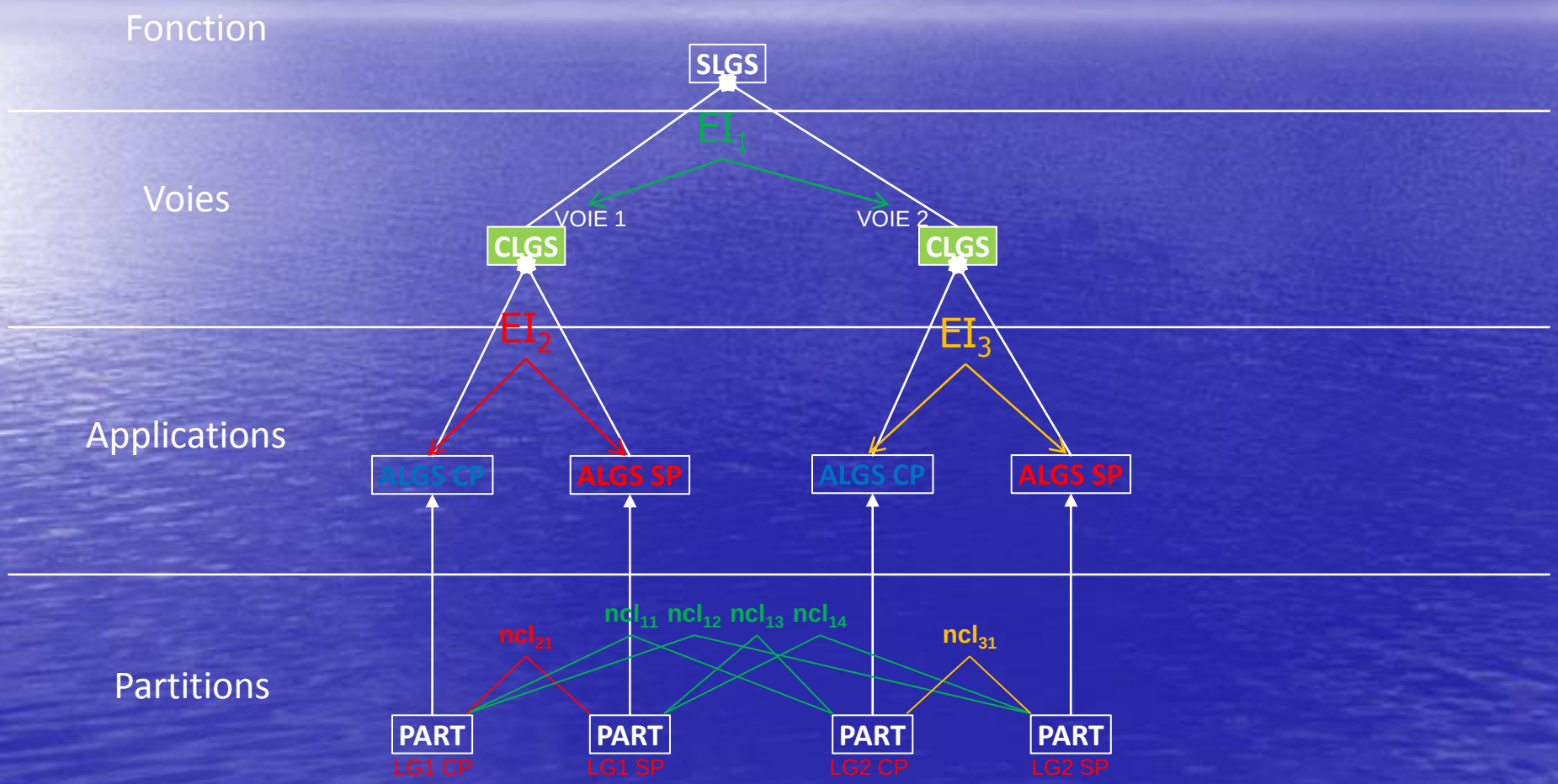
LGS : modélisation des exigences



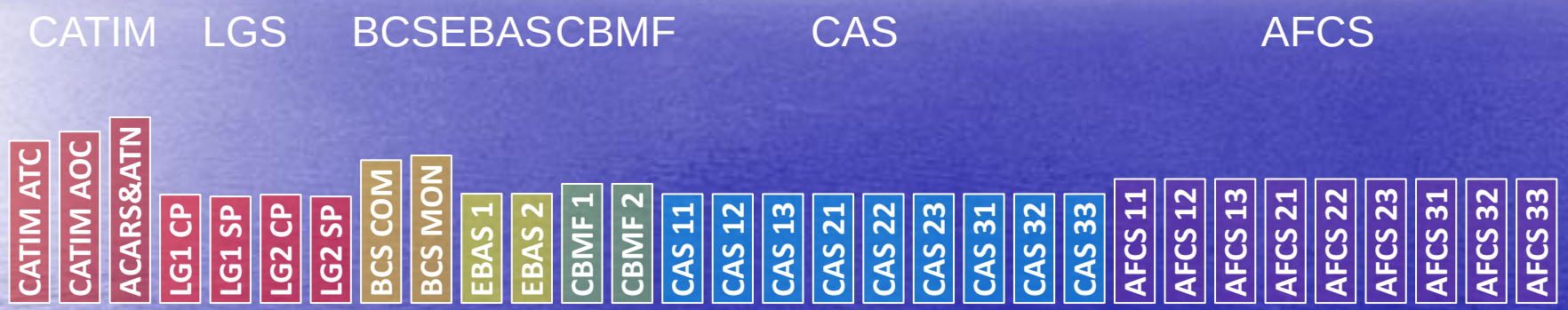
LGS : modélisation des exigences



LGS : contraintes de déploiement

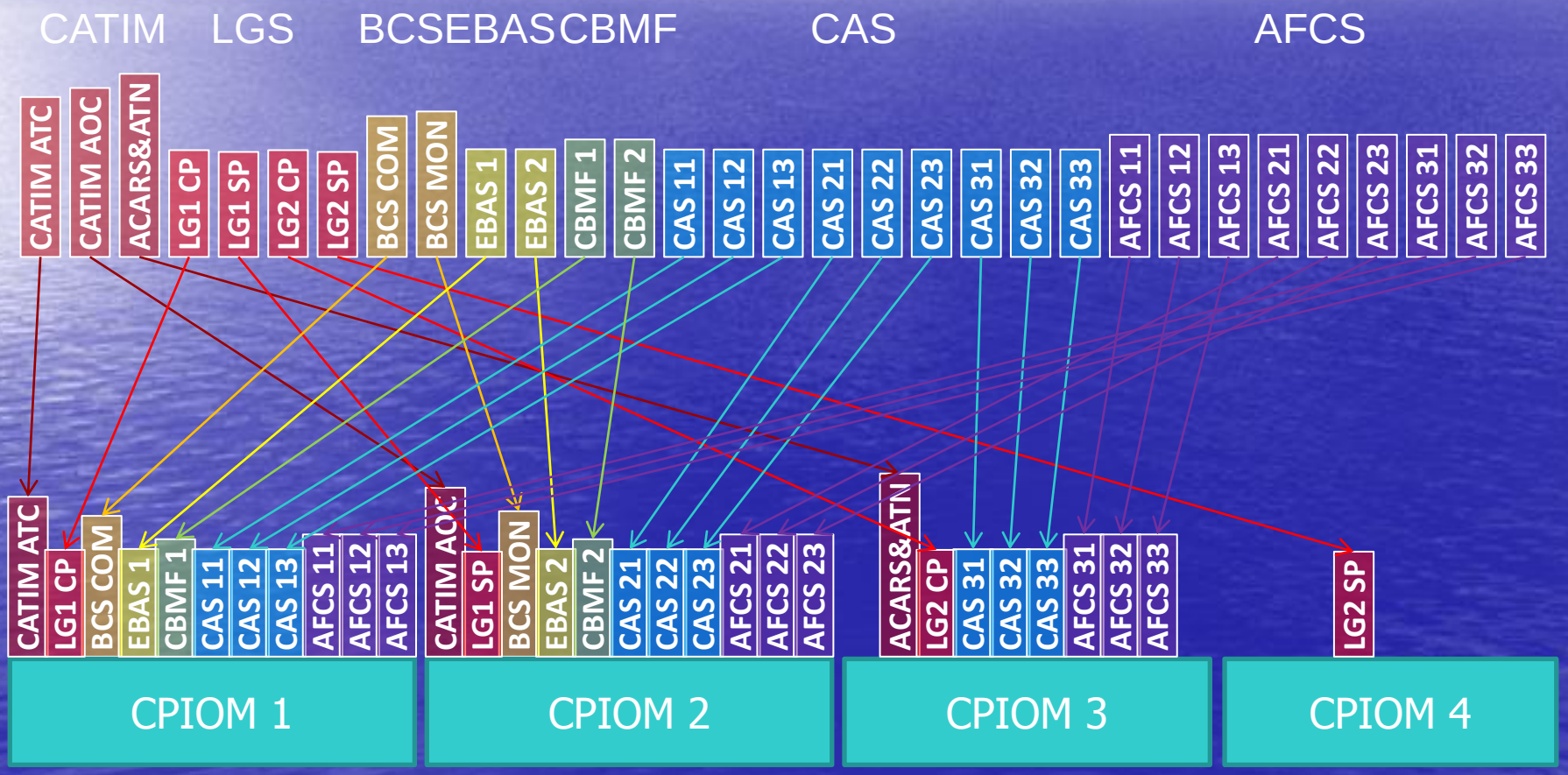


Démonstration

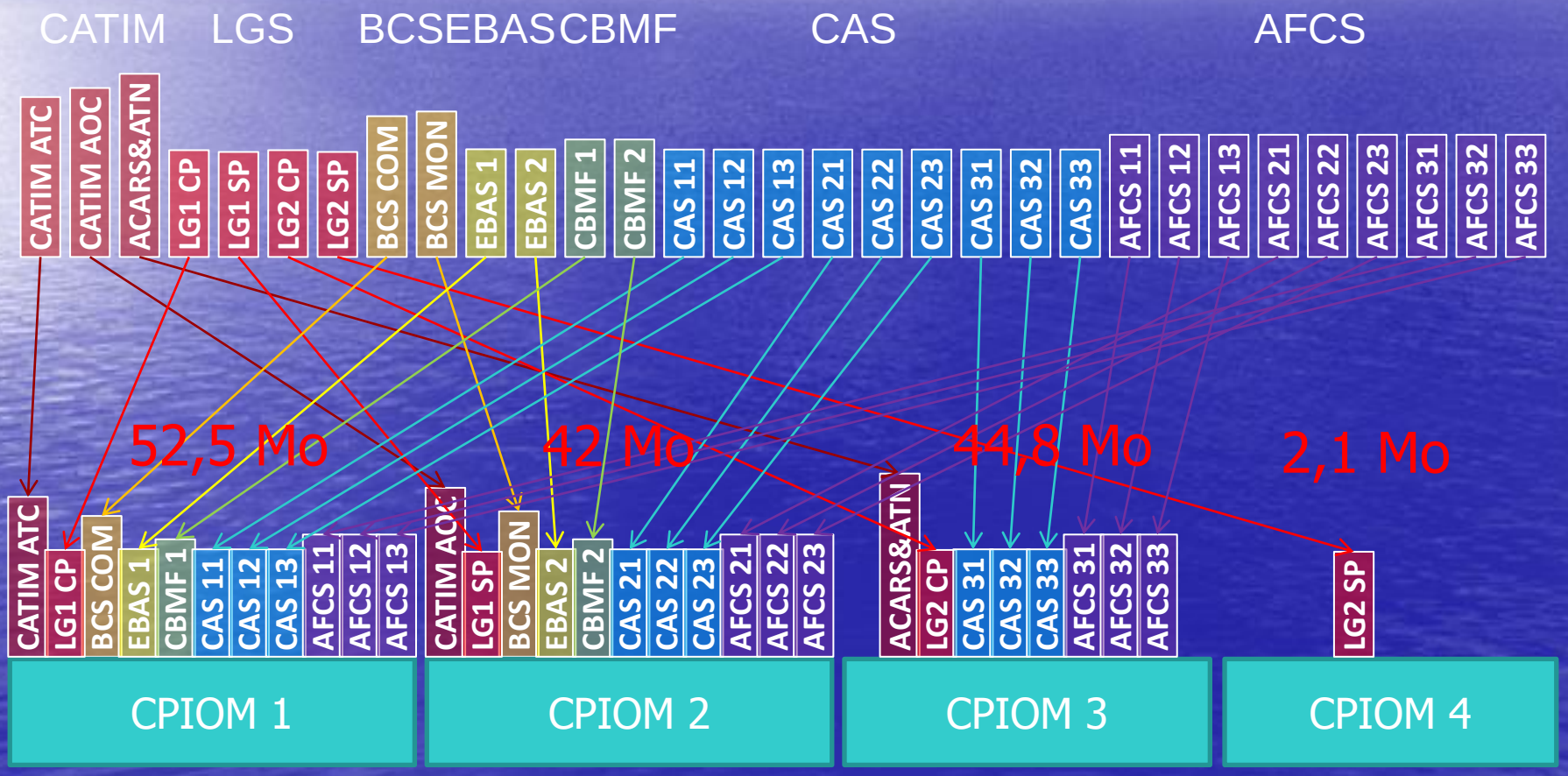


Déploiement de fonctions avion

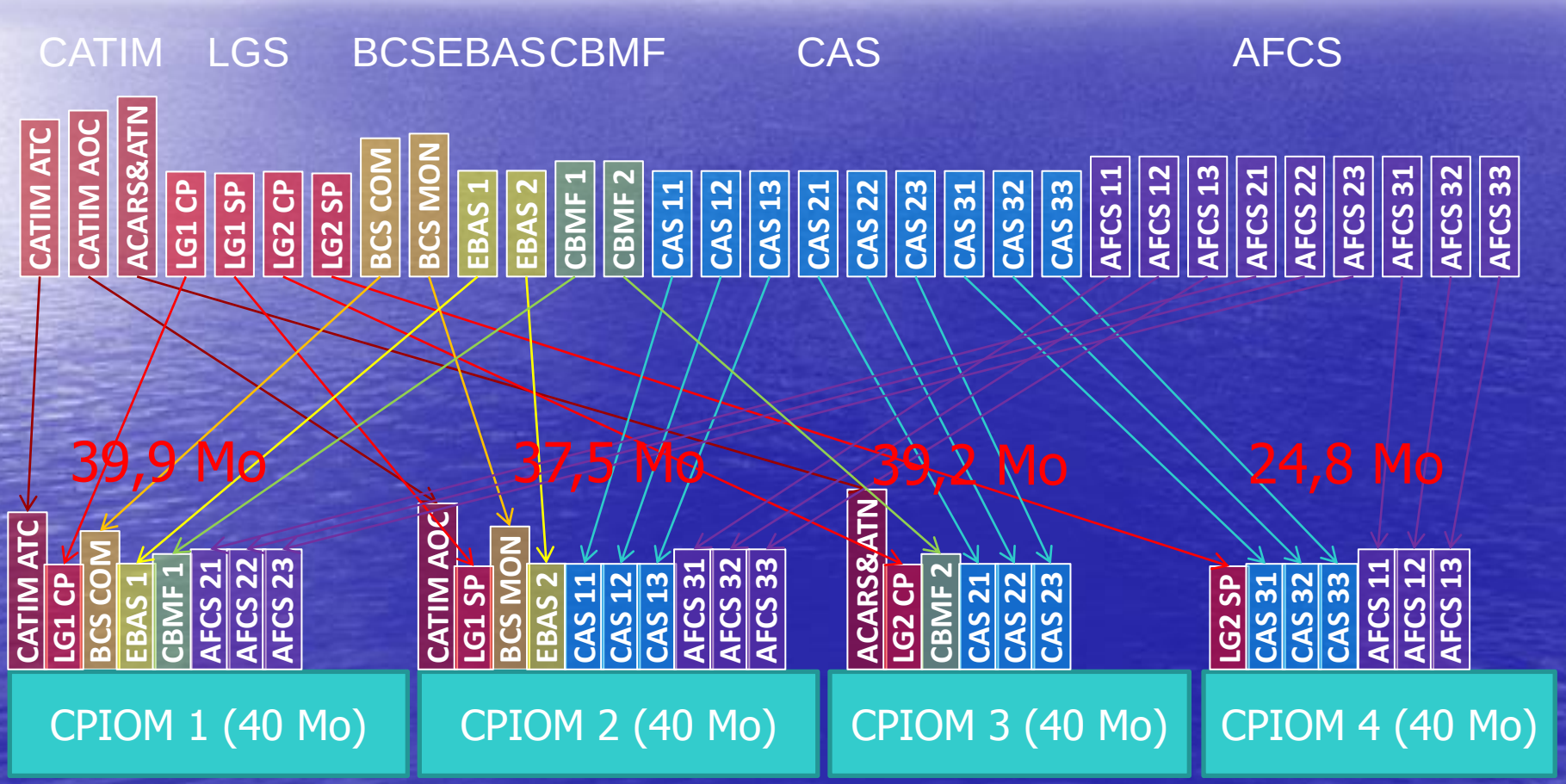
Avec contraintes de sûreté de fonctionnement



Répartition des charges

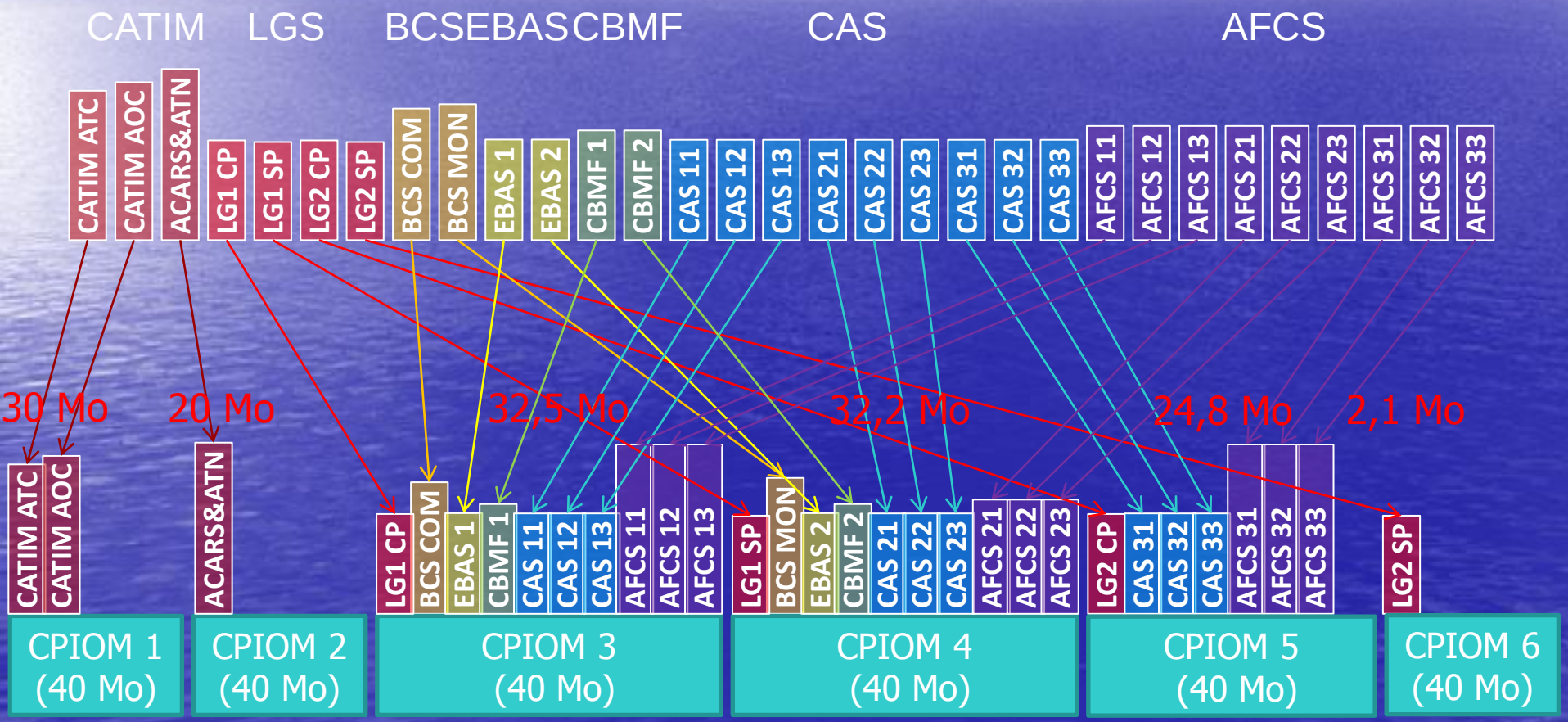


Avec contraintes de sûreté de fonctionnement et de capacité

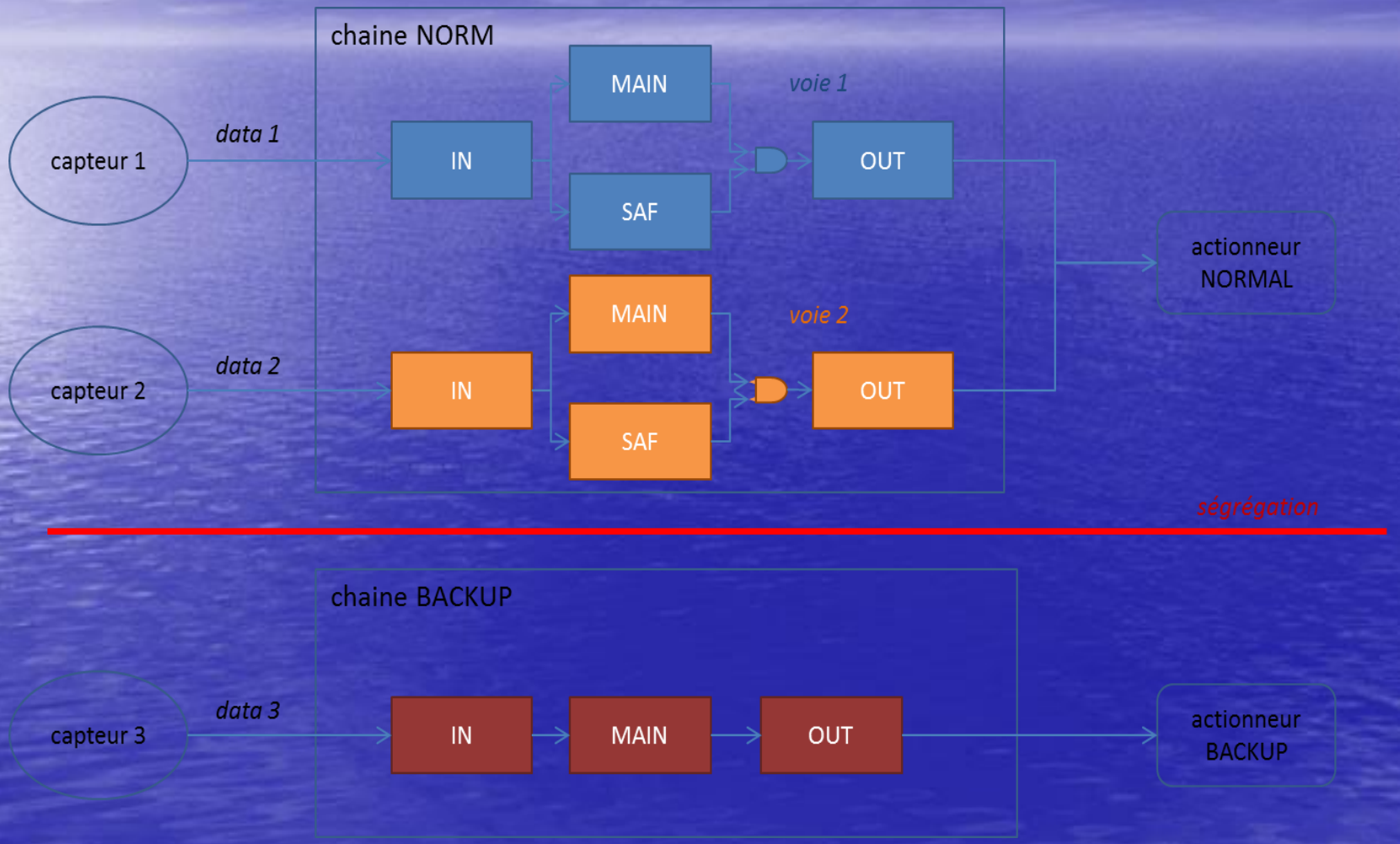


Avec contraintes de sûreté de fonctionnement, de capacité et de sécurité

et CPIOM de 40 Mo



Évolutions du modèle



Évolutions du modèle

- Chaînes de traitement complètes : capteurs, communications, actionneurs, alimentation électrique
- et encore :
  - Safety (FDAL, IDAL)
  - Security (SAL) …
- et aussi (pêle-mêle) : latences, compatibilité de fréquences, coûts, contraintes physiques (poids, encombrement), puissance électrique consommée, ségrégation matérielle, ségrégation des alimentations, fournisseurs …

Bilan DEPS/IMA

On dispose d'un formalisme de conception de haut niveau en rupture avec l'existant

- on peut envisager un modèle déclaratif d'architecture système, utilisable pendant tout son cycle de vie :
  - Génération d'architectures
  - Dimensionnement, Vérification d'une d'architecture
  - Modification incrémentale d' une architecture
    - Nouvelles fonctions avion, applications …
    - nouveaux composants
  - … Certification de l'architecture

Outlines

- Context
- The DEPS project
- The DEPS language
- The DEPS solver
- DEPS by example
- Use-case IMA
- Ongoing studies and developments

Etudes et développements en cours (1)

- Développement d'une nouvelle version de DEPS (L. Zimmer, P.A. Yvars)
  - enrichissement de l'ontologie et des types de base
  - collecteurs de modèles
  - sélecteurs de modèles
  - contraintes « catalogue »

Etudes et développements en cours (2)

- Coopération Dassault Aviation/SupMéca
  - Vérification de systèmes de génération et de distribution électrique
- Portage de modèles CE
  - Pile à Combustible, Transmission de puissance, Robot
- Thèse UTC-SupMéca :
  - Modèles de synthèse pour la conception optimale en génie électrique – Application à l'électrification des véhicules

Informations

- Le langage DEPS est supporté par l'association *DEPSLink*
- www.depslink.com

- Contacts

- P.A. Yvars pierre-alain.yvars@supmeca.fr
- L. Zimmer Laurent.Zimmer@dassault-aviation.com

Merci pour votre attention