Virginie GALTIER
Stéphane VIALLE

simulations et co-simulations distribuées

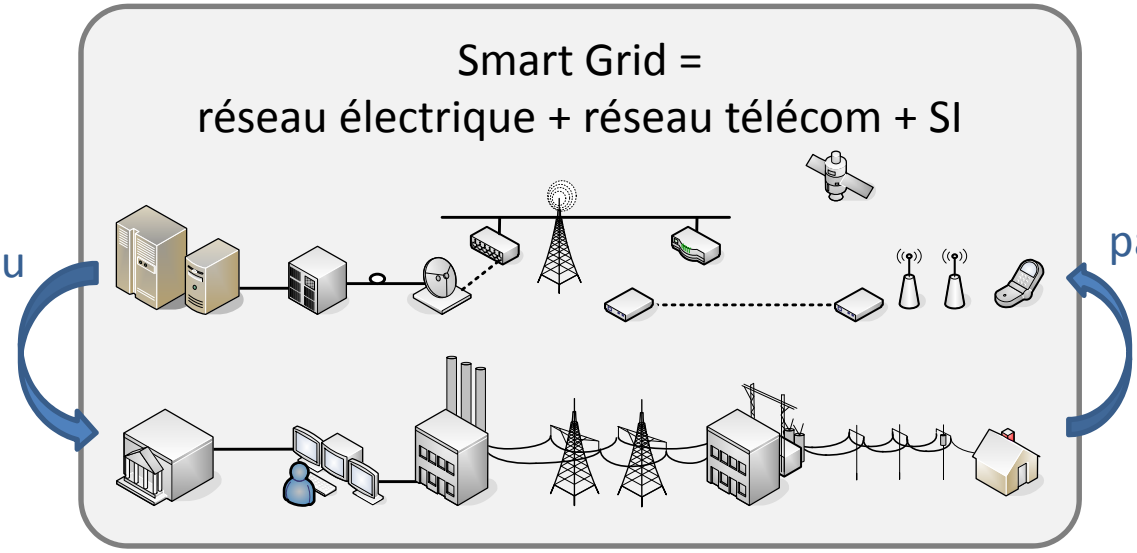
Séminaire « Réseau SI R&D » – EDF R&D – La Défense – 5 décembre 2013

Smart-Grid = « système complexe »

éléments très nombreux, dynamiques et hétérogènes, en interaction

→ difficile de répondre aux questions « aux frontières »

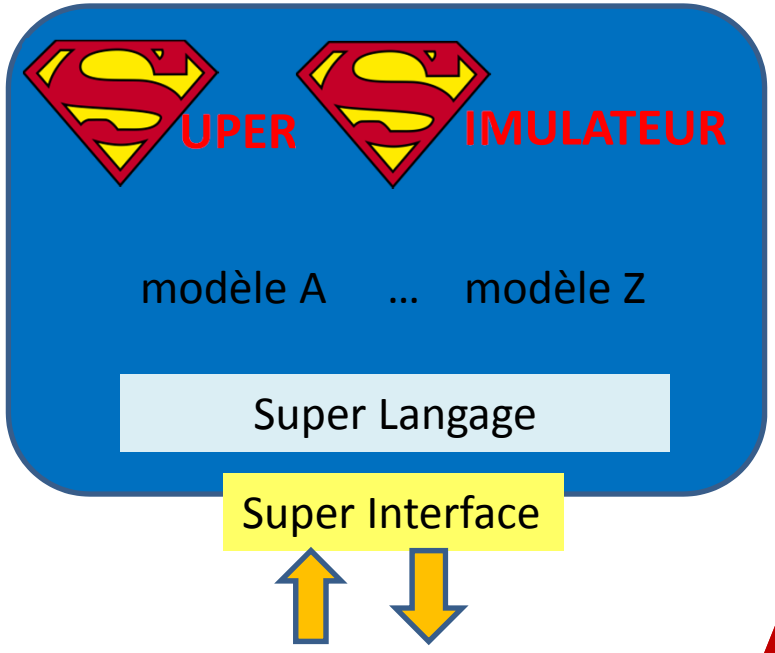
conséquence du piratage du réseau télécom sur la distribution d'électricité ?



conséquence de la panne d'alimentation électrique d'une antenne sur la transmission des données ?

étudier les influences inter-domaines : tests réels ?

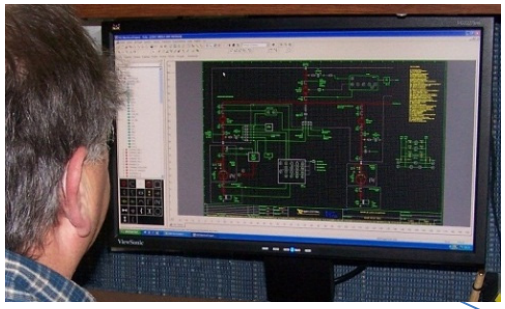
étudier les influences inter-domaines : simulateur monolithique ?



- conception, développement, maintenance ?
- intégration de toute l'expertise
- acceptabilité ?
- performances ?

EXCLU !

étudier les influences inter-domaines : couplage de simulateurs existants !



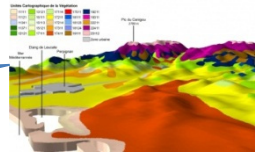
NetLogo

Dymola

MODELICA

MATLAB
SIMULINK

Glue



C-Learn
Climate Simulation
CLIMATE INTERACTIVE
Tools for a thriving future
Embeddable Web simulation

ns-3
NETWORK SIMULATOR

OMNeT++ 4
THE OPEN SIMULATOR
Academic Edition - not for commercial use.

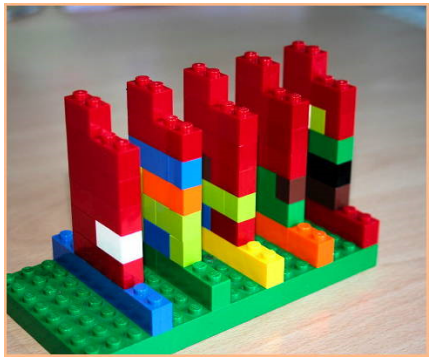
normes pour la multi-simulation

- couplage ad-hoc de simulateurs 2 à 2 :
  - non évolutif, non réutilisable…
- solution : interfaces standard
  - remplacement facilité d'un composant par un autre

  → interopérabilité, réutilisabilité

HLA : *High-Level Architecture*

FMI : *Functional Mockup Interface*



historique et communautés

FMI

- issue d'un consortium européen industriel
- v1 : 2010, v2 : fin 2013
- déjà supportée par de nombreux produits (Dymola, SIMPACK…)

HLA

- développée par les militaires américains
- issue de DIS et ALSP (début 90's), inspirée par CORBA
- 1ère version stable : 1998
- version civile IEEE en 2000 puis 2010
- passerelles pour OMNeT++, NS-3…

concepts

FMI

- composant = « ***FMU*** » = **bibliothèque**
  - donnant accès aux équations du modèle (mode « ME »)

    getDerivatives, setContinousState
  - ou implémentant un solveur manipulant les équations du modèle (mode « CS »)

    doStep, getReal…
- pour exploiter ces FMUs : écrire un « master » qui :
  - charge la FMU
  - init
  - ordonne l'exécution d'un pas
  - récupère les données (et les transfère à une autre FMU)

HLA

- composant = « ***federate*** » = **processus**
- federates connectés à un bus central (RTI)
  - échange de données par publication/abonnement
  - synchronisation : avance de temps demandée par le federate, accordée par le RTI

HLA et FMI ensemble : pourquoi ?

- certains simulateurs sont compatibles HLA, et d'autres FMI

→ intégrer les 2 normes

HLA

simulations event-based, temps réels
ex : composants du réseau télécom simulés par NS3 ou OMNeT++

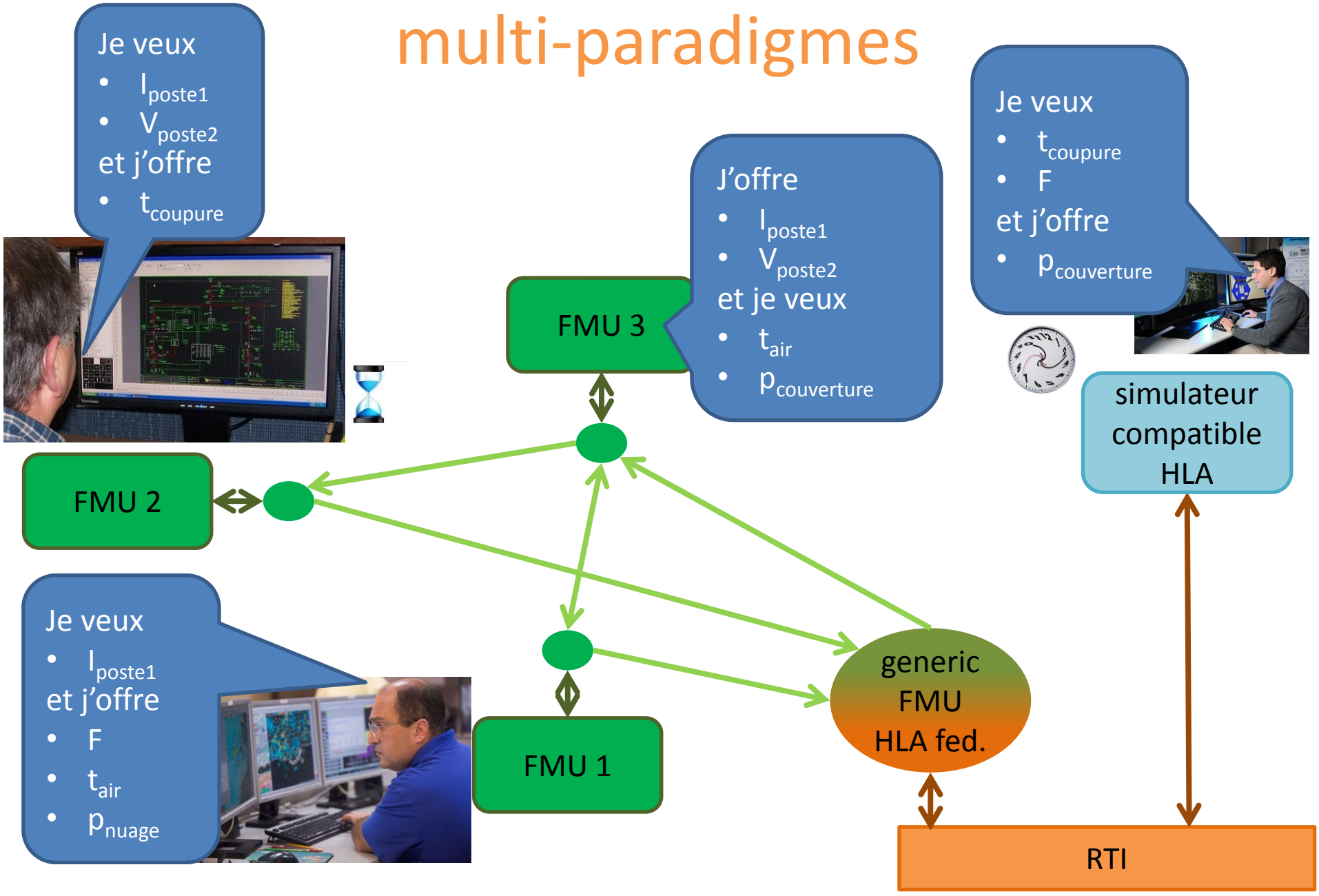
FMI

simulations temps discret, modèles physiques
ex : composants du réseau électrique modélisés avec le langage Modelica, exportés comme FMUs

- FMI ne propose aucun middleware pour la distribution de la multi-simulation

→ le RTI d'HLA peut-il fournir ce niveau ?

objectif : architecture décentralisée multi-paradigmes



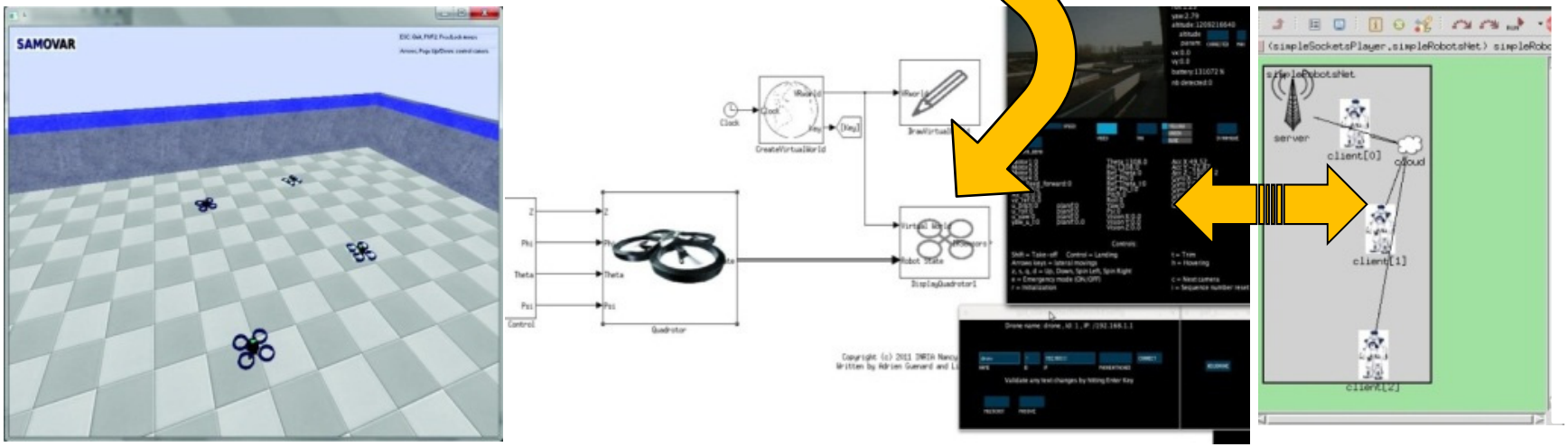
difficultés

- description de la multi-simulation
  - données échangées
  - graphe de connexion des composants
  - metadata (indication de convergence, marge de tailles de pas admissibles…)
- modèles de temps variés à faire co-exister (événementiel, temps discret, temps réel…)
- performances distribuées, capacité à fonctionner en temps réel
- co-initialisation (*bootstraping*)

autre cas d'étude

« société » de drones volant en formation

interaction avec un drone réel
« co-simulation » HIL
contraintes temps réel



modèle de quadrotor
Simulink, exporté en FMU

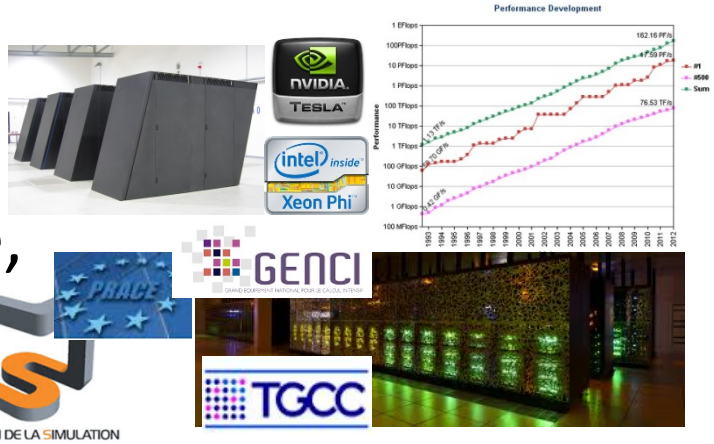
réseau sans fil
simulé avec OMNeT++
interfacé avec HLA

parallélisme pour la co-simulation

Deux facettes du parallélisme dans la co-simulation :

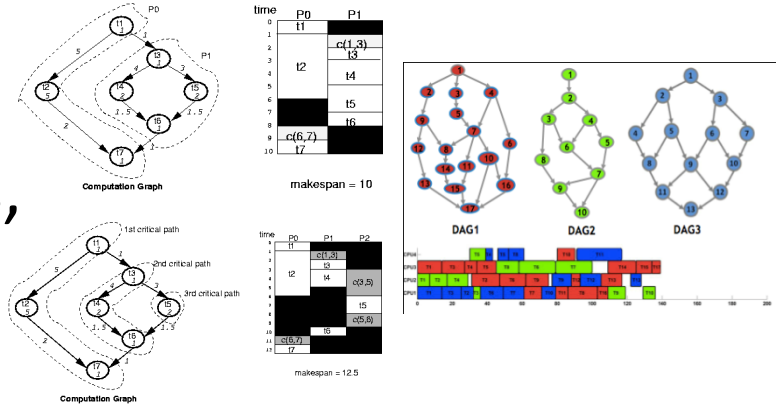
1. Le parallélisme de données

« Clusters, Massivement parallèle,
Large échelle, Top500… »



2. Le parallélisme de contrôle

« WorkFlow, Makespan,
Graphe de tâches hétérogènes,
Scheduling, Concurrence… »



→ **Expertises et problématiques différentes**

parallélisme de données en 2013

Architectures cibles classiques :

« **Hiérarchiques** » : (ex) cluster de nœuds multi-cœurs

« **Hybrides** » : nœuds CPU + accélérateur (ex : GPU)

Hiérarchique hybride : (ex) cluster de CPU+GPU 

Outils de programmation classiques :

Bibliothèques de noyaux de calculs (*BLAS/MKL/FFTW/IPP/…)*

Cumul de paradigmes de programmation

Envois de msgs + multithreading + calcul vectoriel

MPI + OpenMP/TBB/… + CUDA/Intrinsics/…

Extensibilité forte … ou faible !!

Debugger !!

parallélisme de données

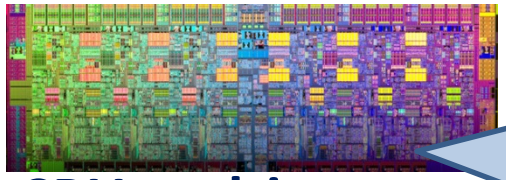
Intérêts pour la co-simulation :

Paralléliser et accélérer la simulation d'un composant sur beaucoup de cœurs

Speedup

Distribuer et rendre possible la simulation d'un composant dans beaucoup de RAM

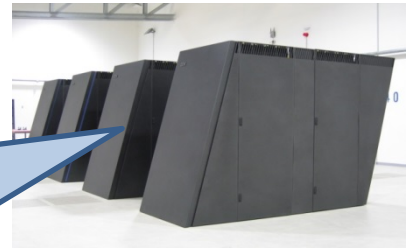
Size up



CPU multi-coeurs

UN simulateur

UN autre simulateur

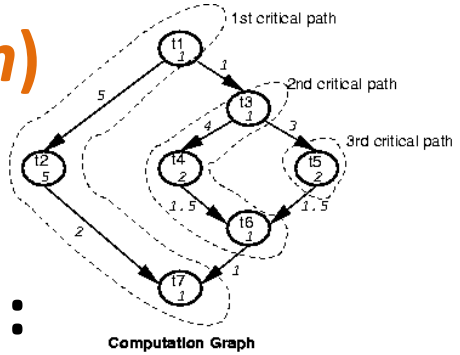


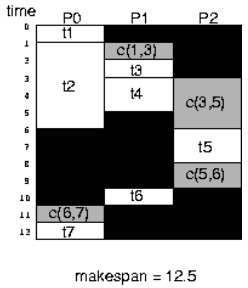
parallélisme de contrôle en 2013

Problématique classique :

- Ressources informatiques limitées
- Flot de tâches sans cycle (mais communicantes)

Placer et ordonnancer les tâches pour minimiser le temps d'exec. total (le *makespan*)





Difficultés à prévoir en co-simulation :

- Présence de cycles, voire « un vrai graphe de tâches »…
- Volumes/temps de communications entre tâches
- Optimiser l'exec. sur architecture hiérarchique (cluster de multi-cœurs)

parallélisme de contrôle

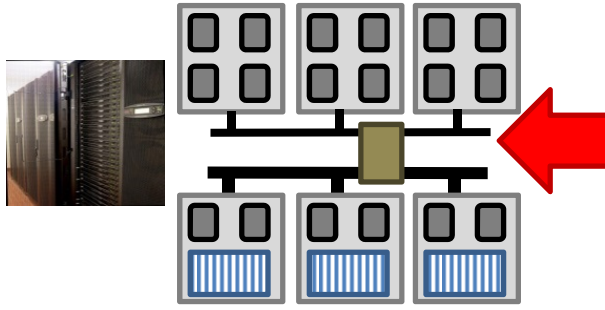
Intérêts pour la co-simulation :

Rendre possible et accélérer des co-simulations de systèmes complexes : speedup & size up

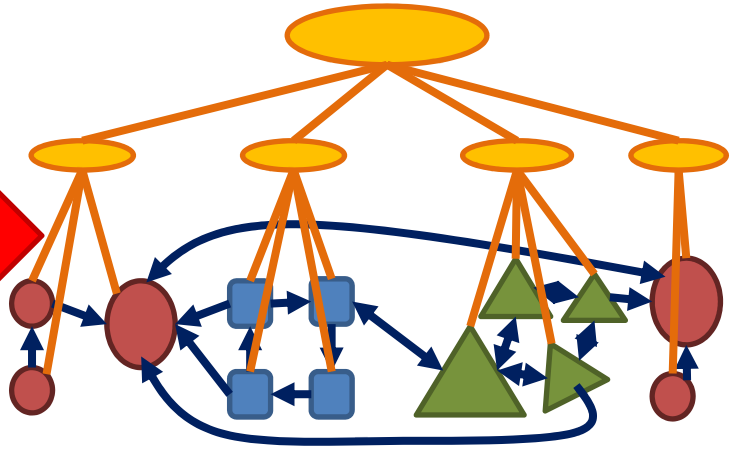
- Distribuer le graphe de tâches de la co-simulation
- Plus de cœurs de calculs
- Plus de RAM
- Plus de Bw réseau

« *passage à l'échelle* »

Graphe de rsrcs informatiques



Hiérarchique (hybride)

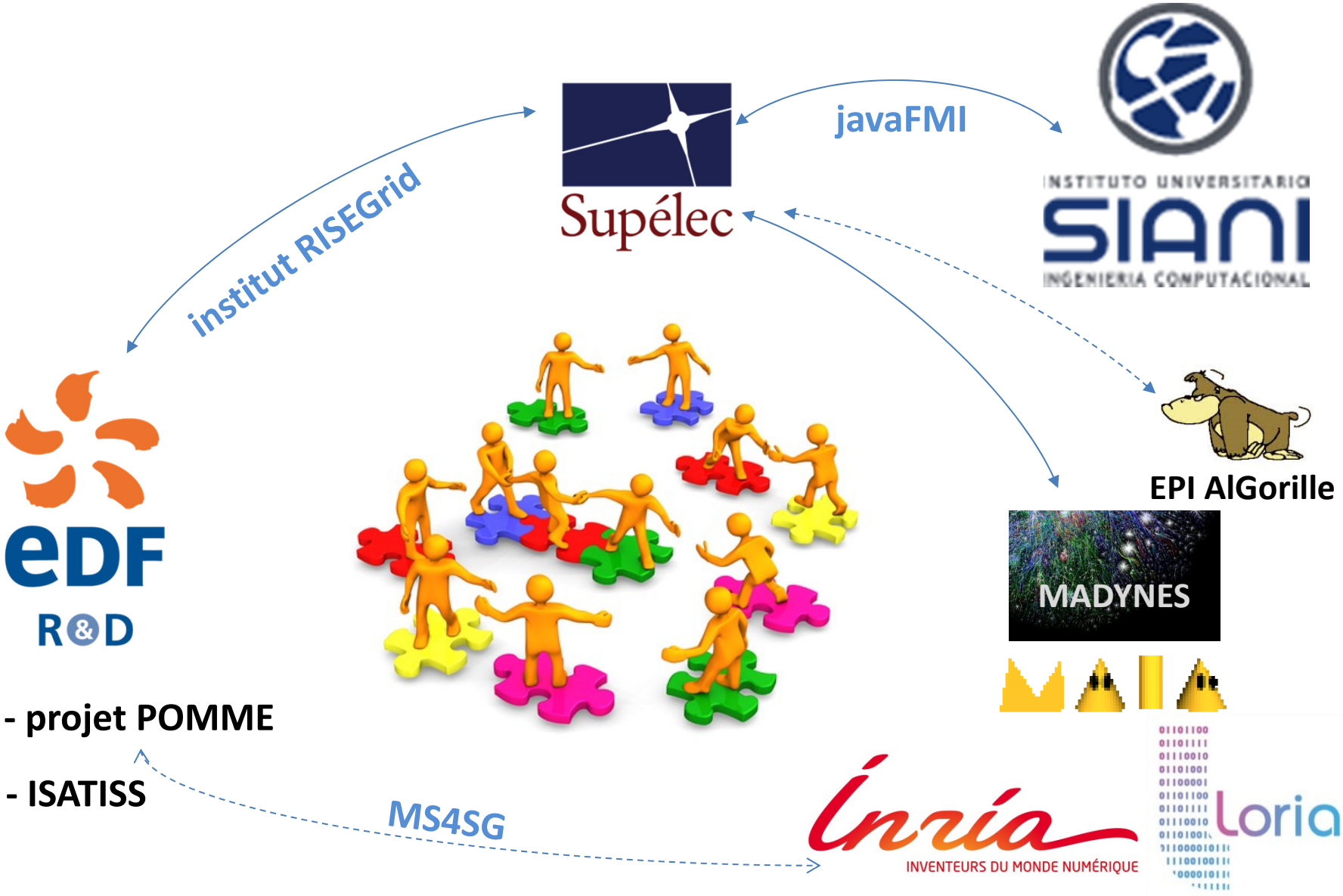


Hiérarchie de tâches de

Graphe de co-simulation

de tâches hétérogènes de simulations

collaborations



Supélec

Co-simulation

Smart Grids

Architecture décentralisée

Drones & Env. Réseau

HLA

FMI

Modèle de description de la co-simulation

Questions ?

Modèle de temps

Parallélisme de données

Parallélisme de contrôle

Speedup
Size up

Placement & Ordonnancement

Graphe de co-simulation