Optimization tools and applications developed during the OMD & OMD2 projects

Denis Caromel (Univ. Nice and INRIA)

Régis Duvigneau (INRIA)

Rodolphe Le Riche (CNRS and Ecole des Mines St-Etienne)

General context

OMD = Optimisation MultiDisciplinaire (French) = MDO (English).

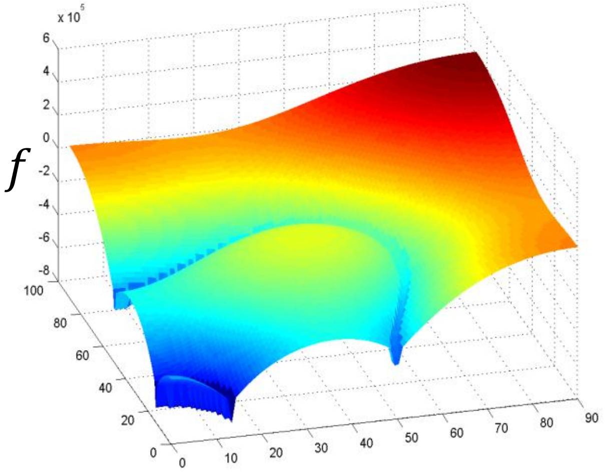
OMD2 = a continuation of OMD + Distributed.

- Numerical simulation has reached a high level of maturity and is increasingly turning to optimization.
- The encountered bottlenecks (see later) are not the ones mathematicians would spontaneously consider.

optimize a complex object

versus

$$\min_x f(x)$$

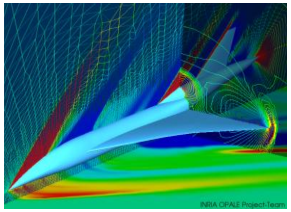


Context : uncertainties & multidiscipline

$f(x)$ is in fact $f(y(x,U))$ where

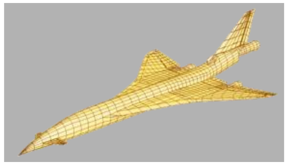
y is the output of the multidisciplinary simulation

a
e
r
o



pressure →

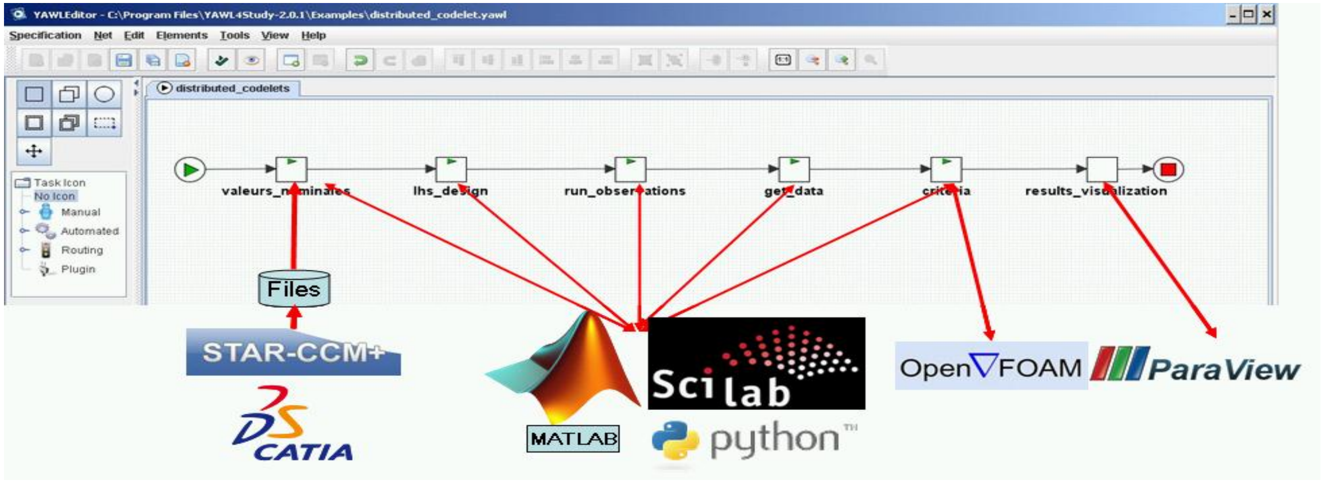
← deformed shape



s
t
r
u
c
t

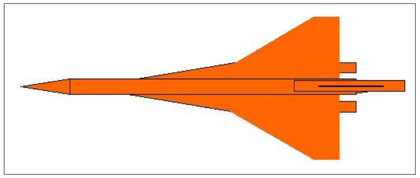
U are uncertain parameters

$f(y(x,U))$ is calculated through a complex workflow

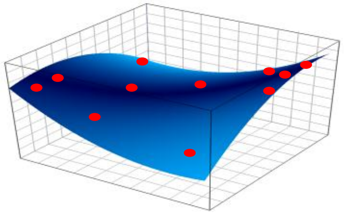


Context : computational time & multi-fidelity

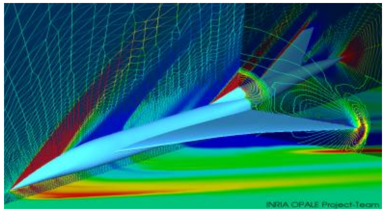
Many simulations are possible to describe the same object :



engineering model



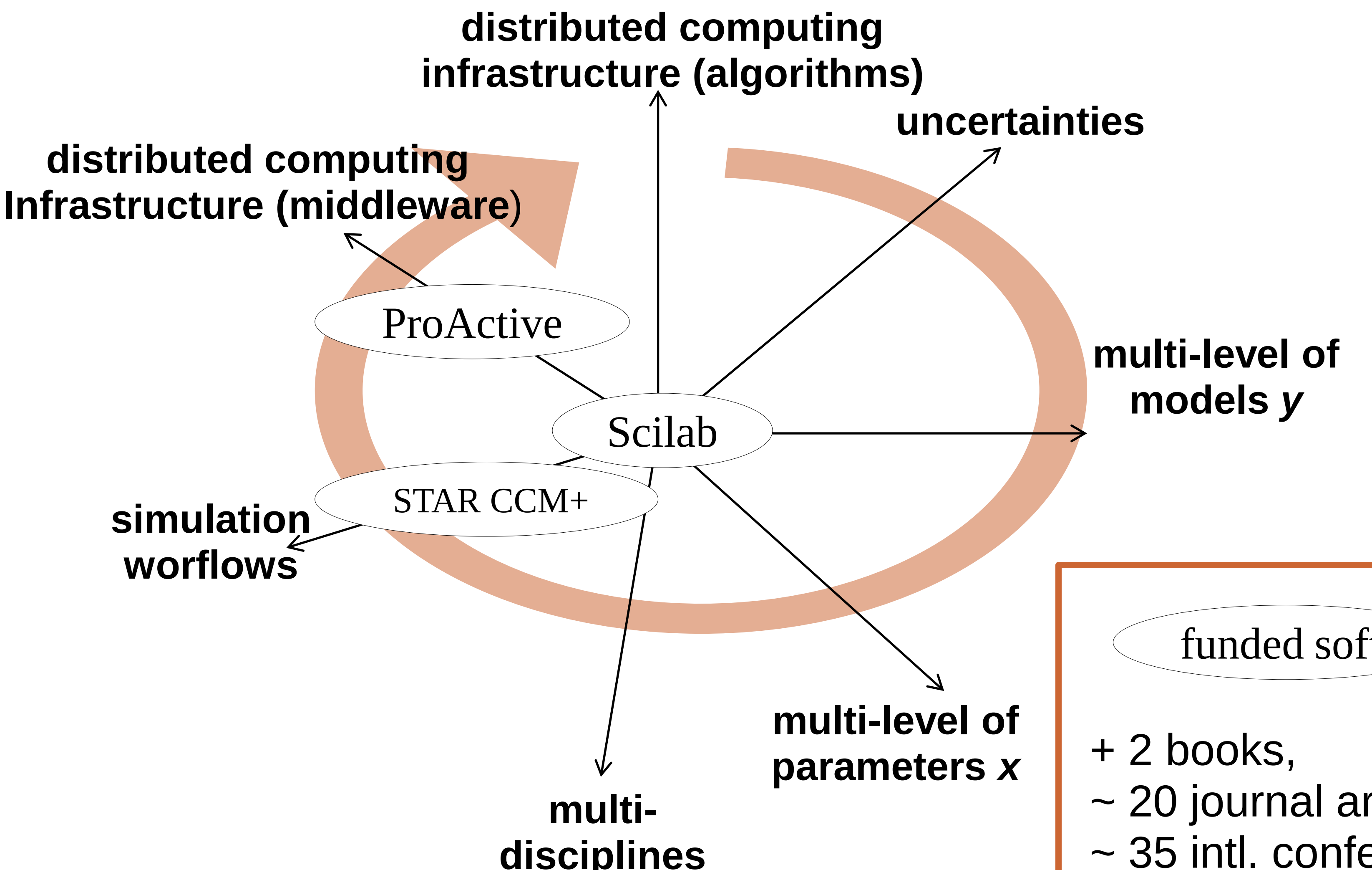
statistical model
(or metamodel)



discretized PDEs

They represent various compromises between simulation fidelity and computational cost.

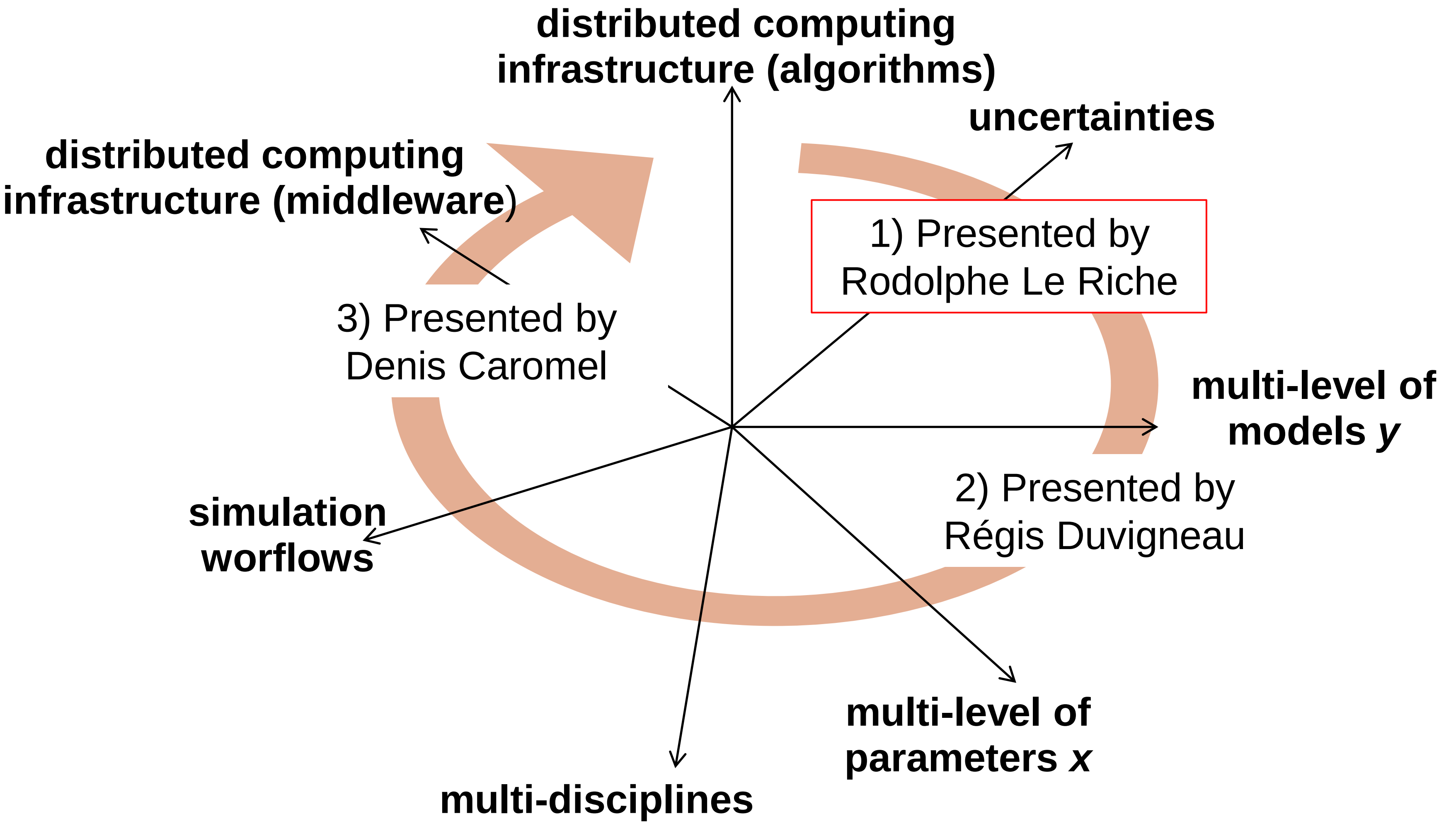
The OMD and OMD2 projects aim at solving realistic, simulation based OPTIMIZATION problems



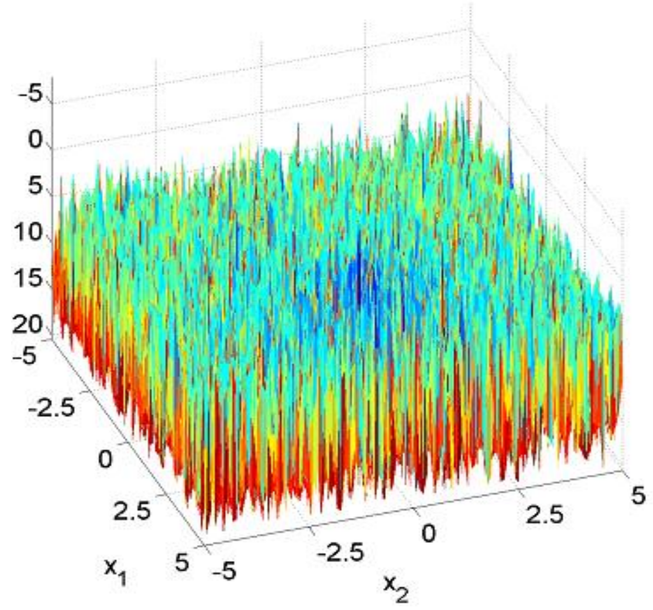
funded soft.

+ 2 books,
~ 20 journal articles,
~ 35 intl. conferences.

Outline of the talk



Optimization with uncertainties (1)



f(x) is noisy → *f(x,U)* , where *U* is random.
U used to represent uncertain environnement, model error, noise.

How to « optimize » *x* ?
What does it mean ?

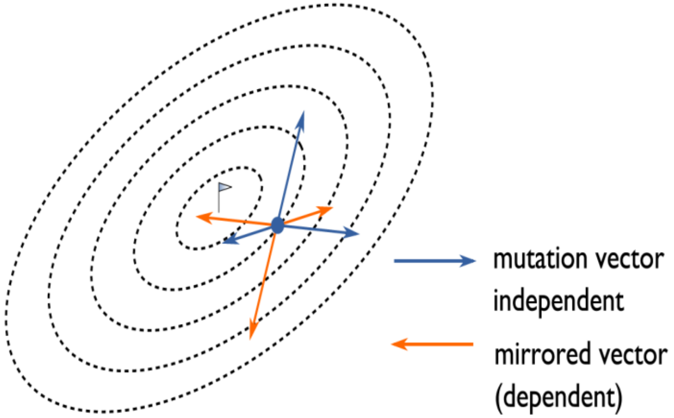
Optimization with uncertainties (2) : when noise cannot be controlled

Problem : $min_x\ f(x)$ but evaluations of *f(x)* are perturbed as *f(x,u)*, where *u* is not controlled.

Method : a stochastic optimization algorithm which strikes a good compromise between efficiency and robustness, **CMA-ES with mirrored sampling and sequential selection.** (CMA-ES available in Scilab)

D. Brockhoff, A. Auger, N. Hansen, D. V. Arnold, and T. Hohm. *Mirrored Sampling and Sequential Selection for Evolution Strategies,* PPSN XI, Springer, 2010

A. Auger, D. Brockhoff, N. Hansen, *Analysing the impact of mirrored sampling and sequential selection in elitist Evolution Strategies,* FOGA 2011.



Optimization with uncertainties (3) : when noise can be controlled

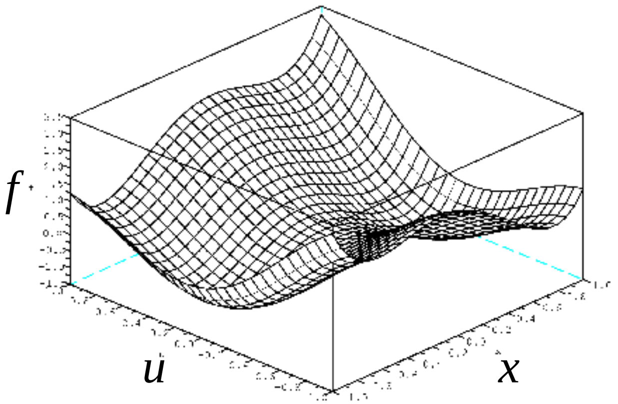
The optimization of a statistical measure of the performance is costly

Optimization : loop on *x*

Estimation of the performance (average, std dev, percentile of *f(x,U)*) : loop on *u* , Monte Carlo

Problem : $\min_x E_U f(x,U)$, *U* normal and calls to *f(x,u) are* controlled, *f* calls expensive.
Method : approximate $E_U f(x,u)$ by a kriging process used to simultaneously sample *U* and optimize *x*. No double loop.
Scilab implementation.

J. Janusevskis and R. Le Riche, Simultaneous kriging-based sampling for optimization and uncertainty propagation, ROADEF 2011 and report hal-00506957 2010.



Optimization with uncertainties (4) : when noise can be controlled

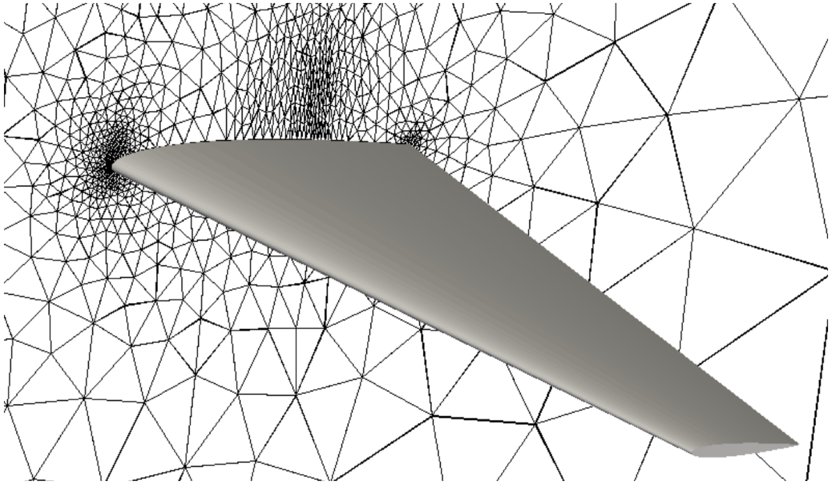
Problem : $min_x \, E_U \, f(x,U)$ and / or $min_x \, VAR_U \, f(x,U)$

Method 1 : The gradients and Hessian of f w.r.t. u are known (adjoint method, automatic differenciation) → **method of moments**, $E_U \, f(x,U)$, $VAR_U \, f(x,U)$ can be estimated from $grad_U \, f$, $Hess_U \, f$, $E(U)$ and VAR(U) .

Method 2 : Only f evaluations known → Monte Carlo and metamodels (radial basis functions, kriging). Scilab implementation.

- R. Duvigneau et al., *Uncertainty Quantification for Robust Design*, Multidisciplinary Design Optimization in Computational Mechanics, Wiley, 2010.

- G. Pujol et al., *L'incertitude en conception* , Optimisation multidisciplinaire en mécanique, Hermes, 2009.



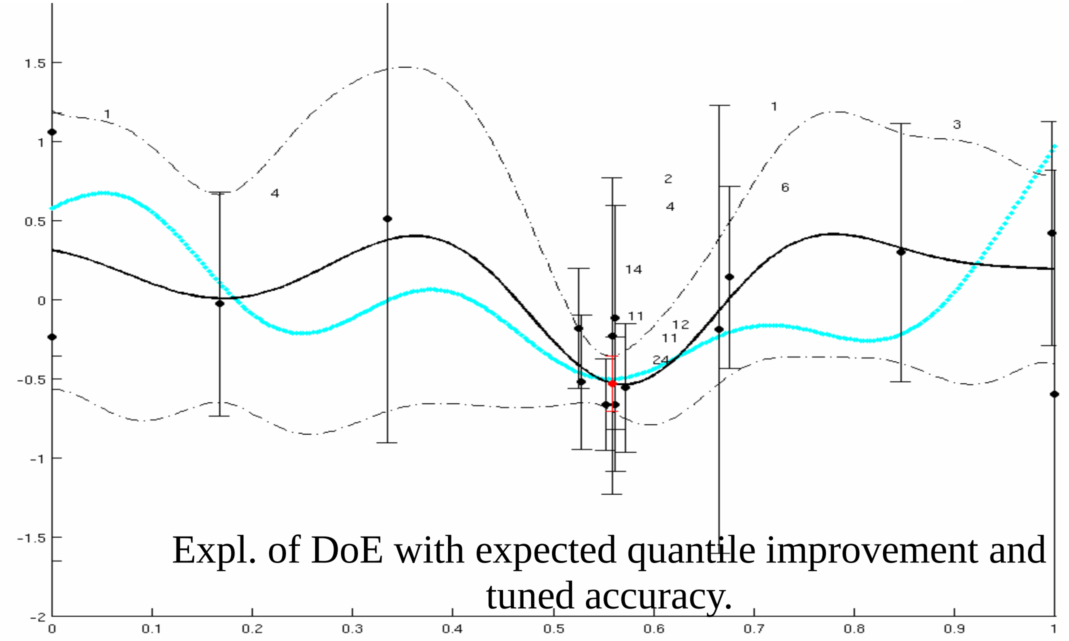
application to f = drag , Duvigneau et al. 2010.

Optimization with uncertainties (5) : adjustable fidelity

Adjustable fidelity = the noise of the simulations can be reduced at an increased computational cost, e.g., $t^2 \sim 1 / time$ (Monte Carlo).

Expected quantile improvement : a measure of potential improvement at unknown points accounting for the risk associated to noisy observations (based on kriging).

V. Picheny et al., *Optimization of noisy computer experiments with tunable precision*, Technometrics, 2011.

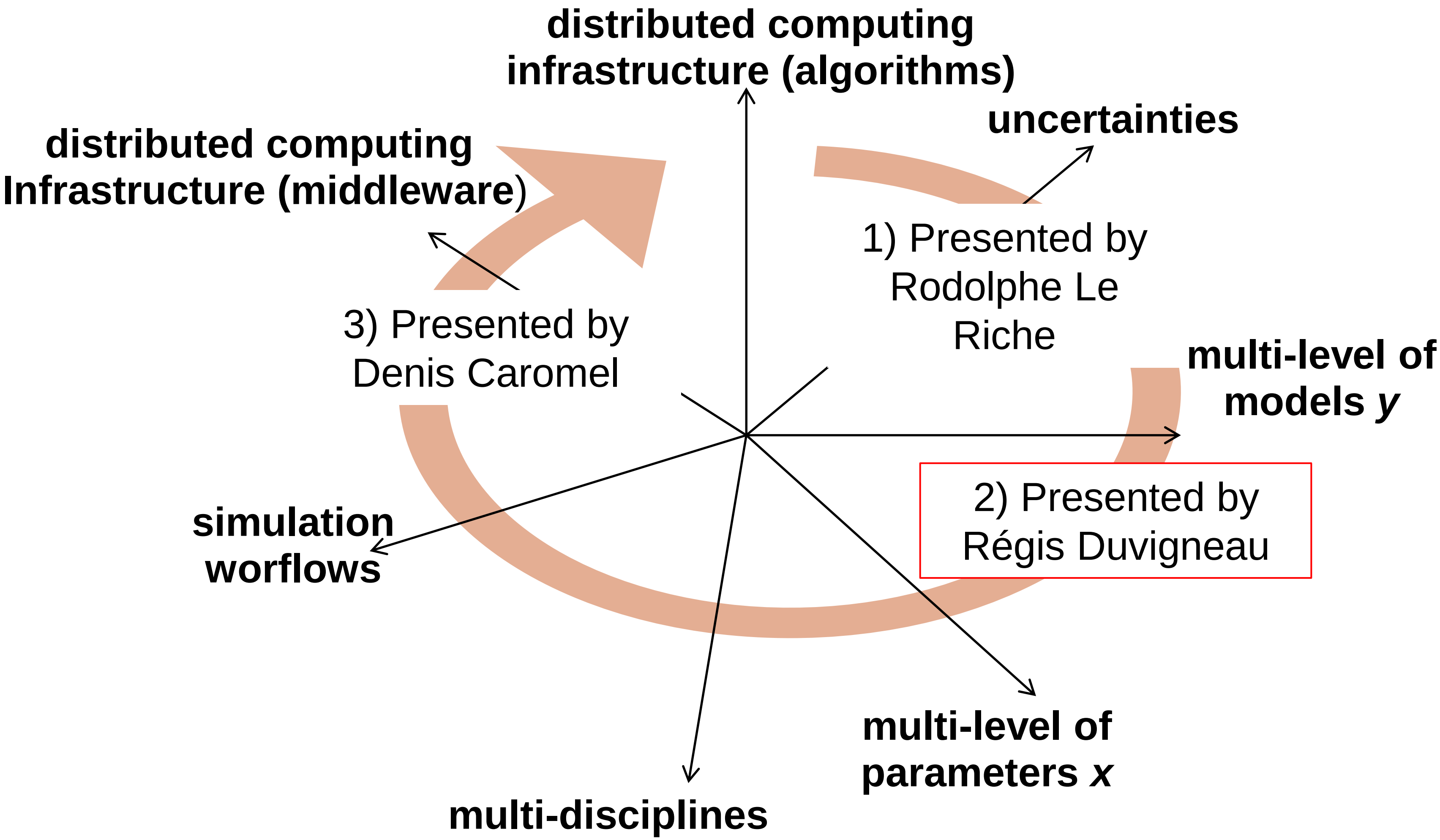


Expl. of DoE with expected quantile improvement and tuned accuracy.

Fidelity accounted for in the optimization algorithm : Scilab implementation.

D. Salazar et al., *An empirical study of the use of confidence levels in RBDO with Monte Carlo simulations*, Multidisciplinary Design Optimization in Computational Mechanics, Wiley, 2010.

Outline of the talk



Multilevel Algorithms : Motivations

Simulation-based optimization in industrial context is limited by the CPU time required, originating from :

- Complex physical systems ➔ expensive simulations
- Presence of several local minima ➔ large number of simulations
- Anisotropy of objective functions ➔ slow convergence
- Large number of design parameters ➔ slow convergence

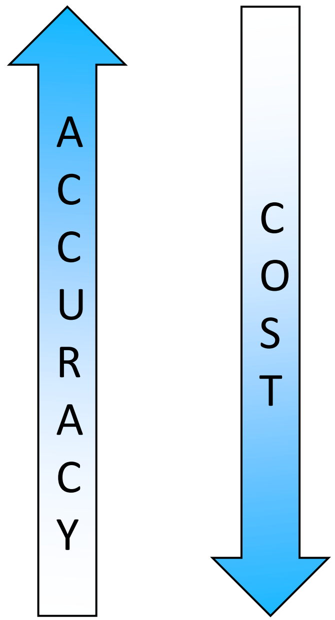
To address these issues, hierarchical approaches are proposed :

- **Multilevel modelling**
- **Multi-fidelity simulation**
- **Multilevel parameterization**

Multilevel Modelling : Principles

Physical systems can be represented by models of various accuracy levels:

	Cost
Non-linear PDEs (ex:Navier-Stokes)	Some hours on a cluster
Linear PDEs (ex: Laplace)	Some minutes on workstations
Reduced-order models (ex: POD)	Some seconds on workstations
Metamodels (Polynomial fitting, Kriging)	Less than one second

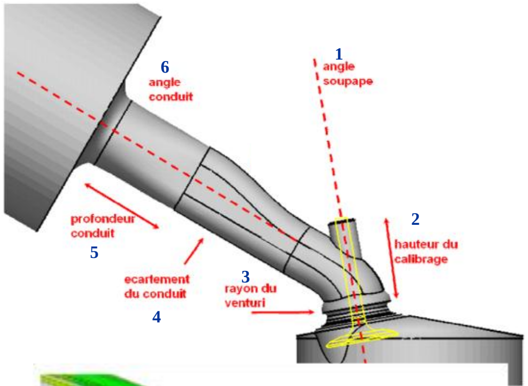


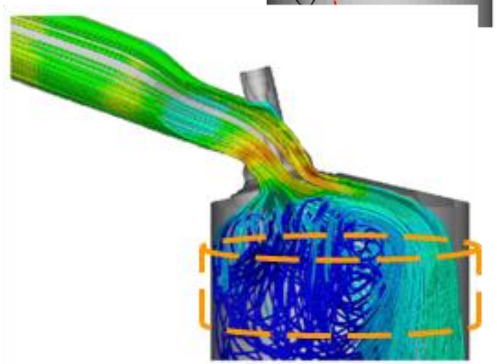
Multilevel modelling optimization : use **as much as possible low-level modelling**, but reach **high efficiency for high-level modelling**

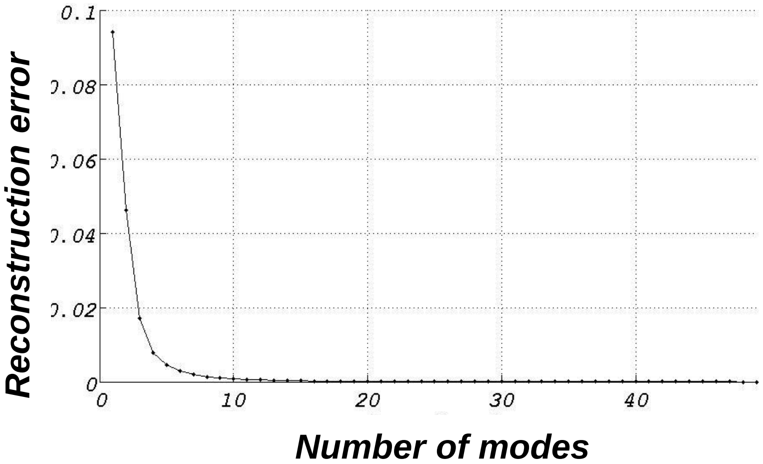
Example : POD

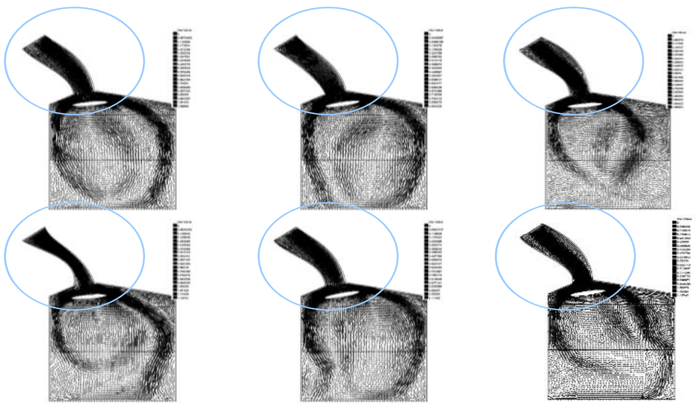
Proper Orthogonal Decomposition (POD) consists of constructing a reduced-order model based on a small number of eigenmodes using some high-level modelling simulations

Ex : Admission duct design









[Univ. Tech. Compiègne, Ec. Centrale Paris]

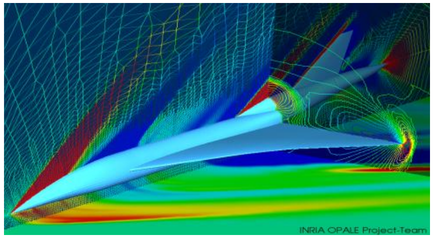
Example : Metamodels

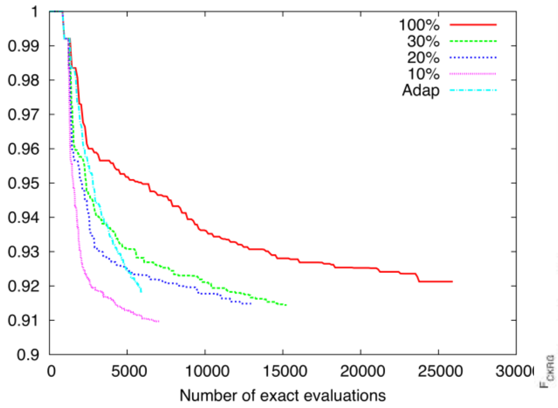
Replace some expensive simulations by a mathematical model that interpolates data previously computed

Two examples of strategy:

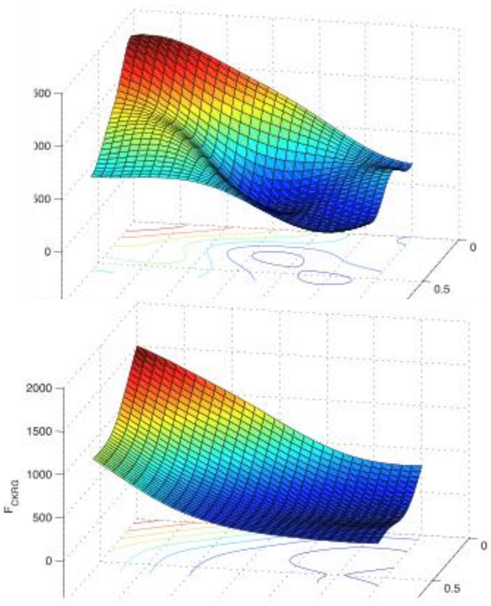
- Use metamodels only to select promising simulations that will be carried out with high-level modelling (inexact pre-evaluation method)
- Use metamodels to completely drive the search by determining designs with highest probability of improvement (EGO method)

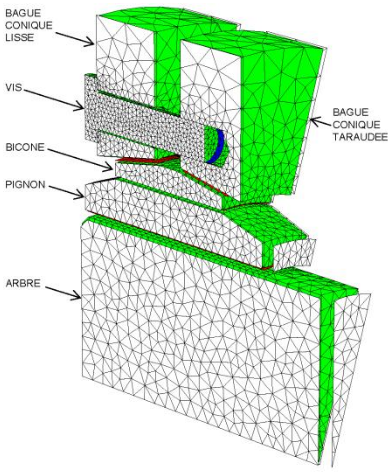
Ex 1 : aerodynamic wing design





Ex 2 : structural design





[INRIA Opale, LMT Cachan]

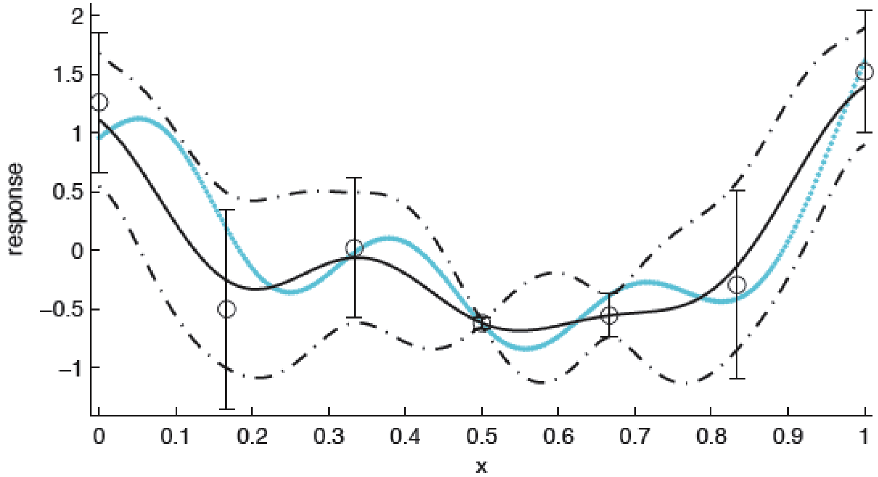
Multi-fidelity simulation : Principle

Reduce the computational cost by degrading the numerical accuracy of the simulations used during the optimization :

- Use coarser grids
- Use incomplete convergence
- Use low-order discretization schemes

Optimization algorithms have to take into account the errors !

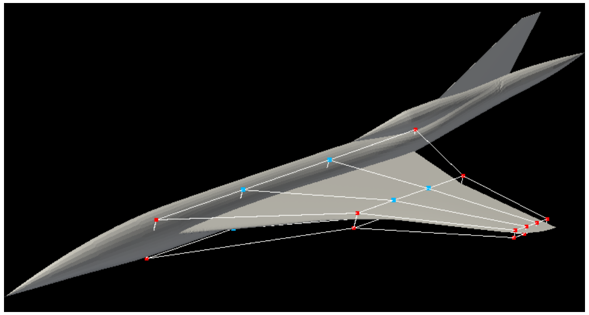
Ex : construction of metamodels taking into account confidence intervals

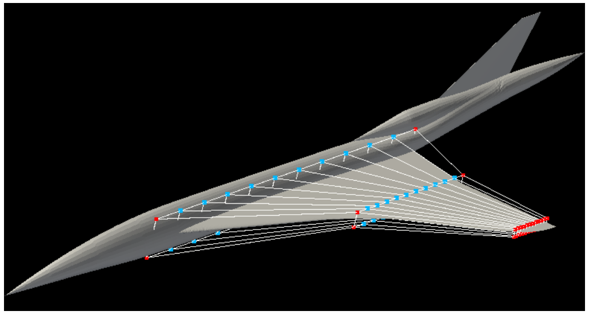


Multilevel parameterization : Principle

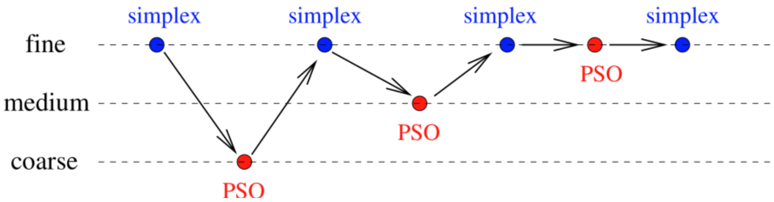
Accelerate the convergence (anisotropy and multimodality) by conducting the search in nested design spaces

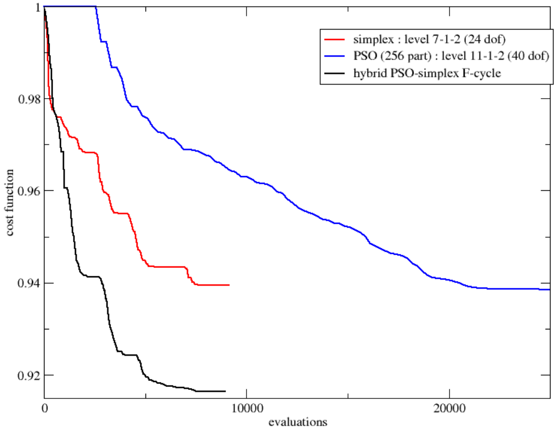
- Construct some parameterization levels (coarse level with small number of parameters to fine level with large number of parameters)
- Define hierarchical optimization strategy





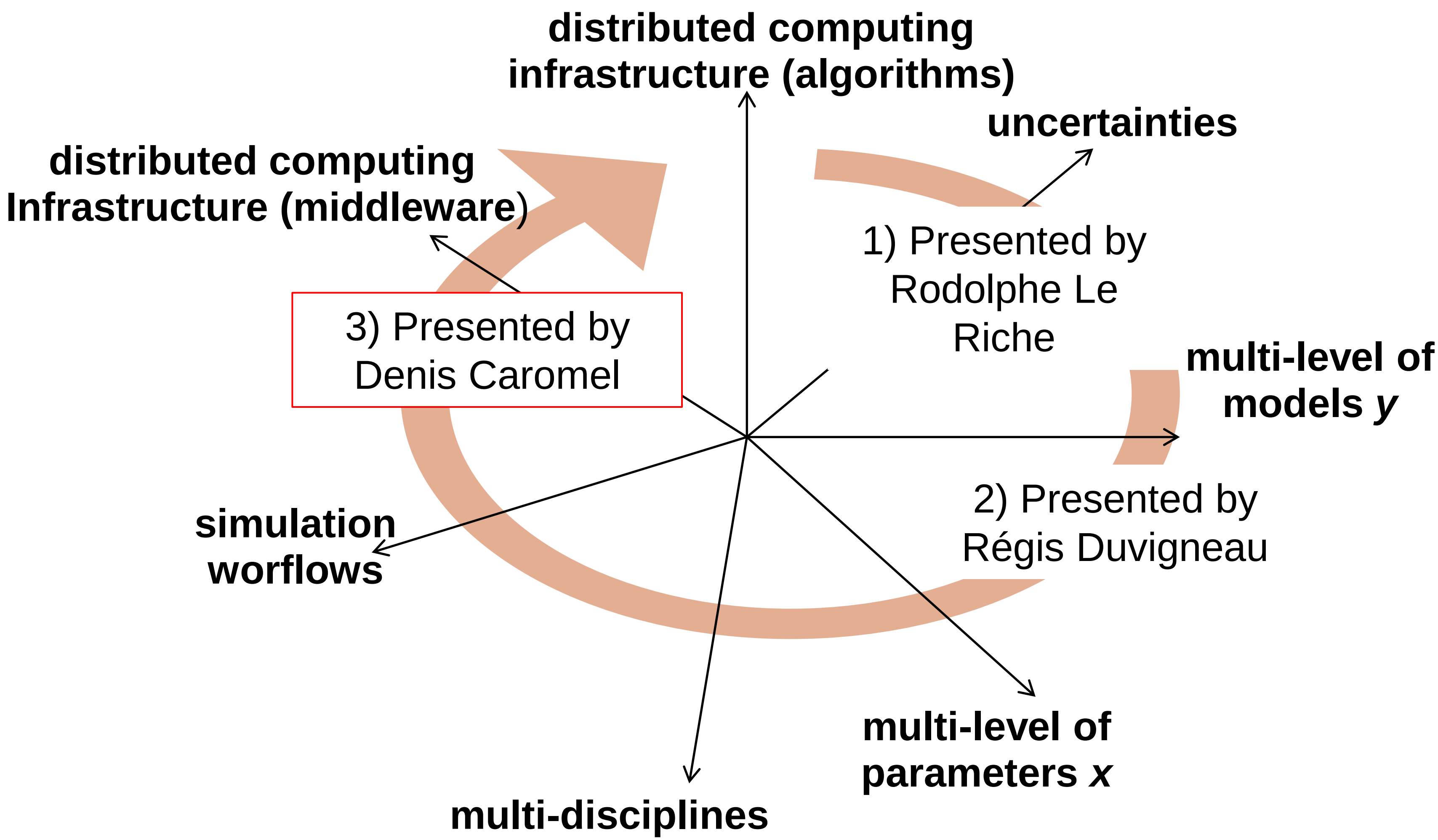
Ex : aerodynamic design with three levels and hybridization





[INRIA Opale]

Outline of the talk



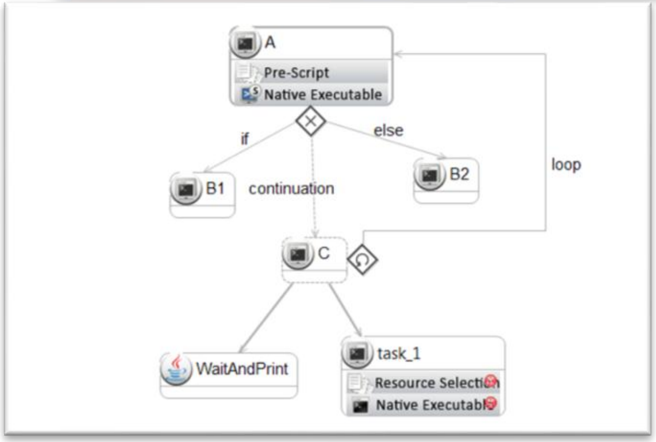
Distributed optimization : middleware, workflows and integration with Scilab

Workflow analysis : for complex applications, evolve over time, distributed environnement, resilient. (INRIA Opale)
T. Nguyên, et al. A Distributed Workflow Platform for Simulation, Intl. Journal on Advances in Intelligent Computing Systems, 2011.

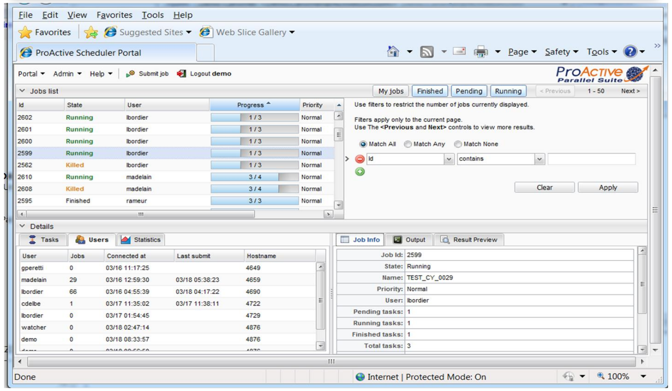
➡ **Distribution on heterogeneous infrastructures, workflow visualization** (Activeon)

Interface with the Scilab opensource scientific calculation plateform.

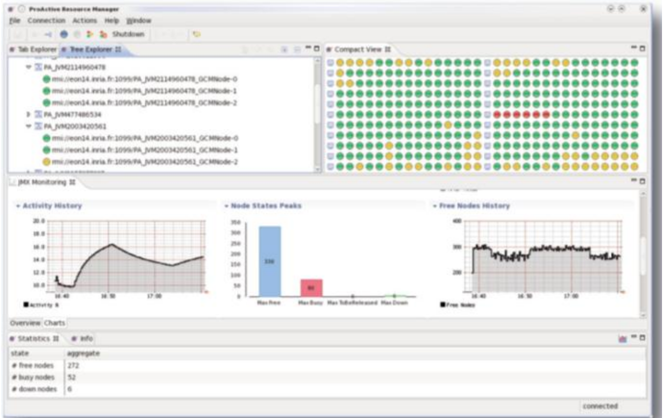
ProActive Parallel Suite



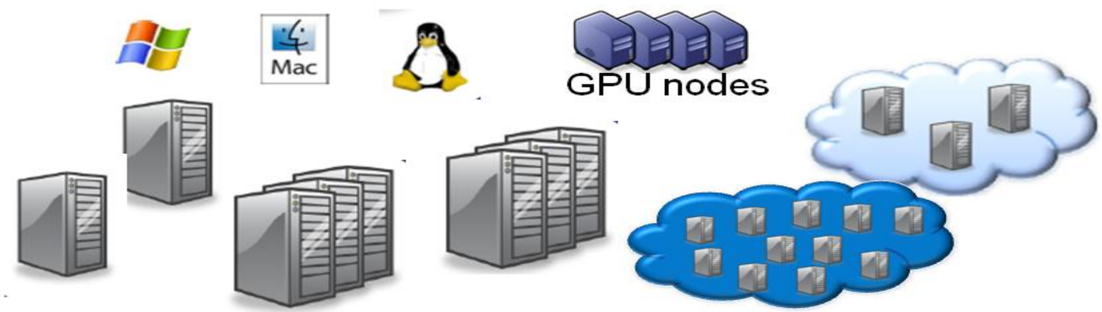
Workflow Execution
Studio Editor and Visualization
Parallel Programming in Java



Portal, Multi-Application & Multi-Tenant
Enterprise Orchestration

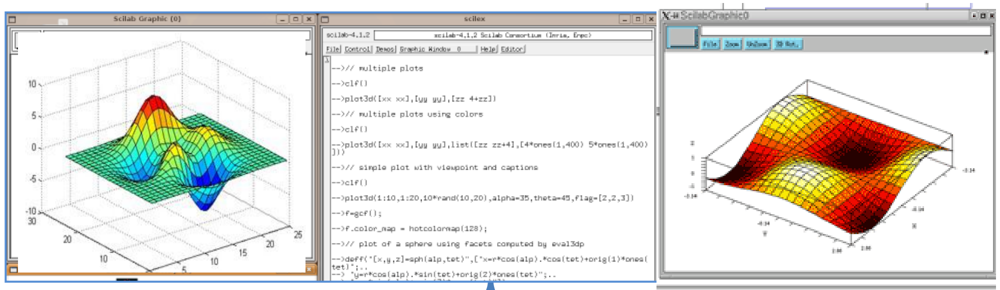


Physical and Virtual Machines Management



Integration with Scilab and Matlab

Integration with Scilab and Matlab



Scilab

MATLAB

The Language of Technical Computing

Static Policy	Timing Policy 12/24	Dynamic Workload Policy
LSF	Desktops	EC2

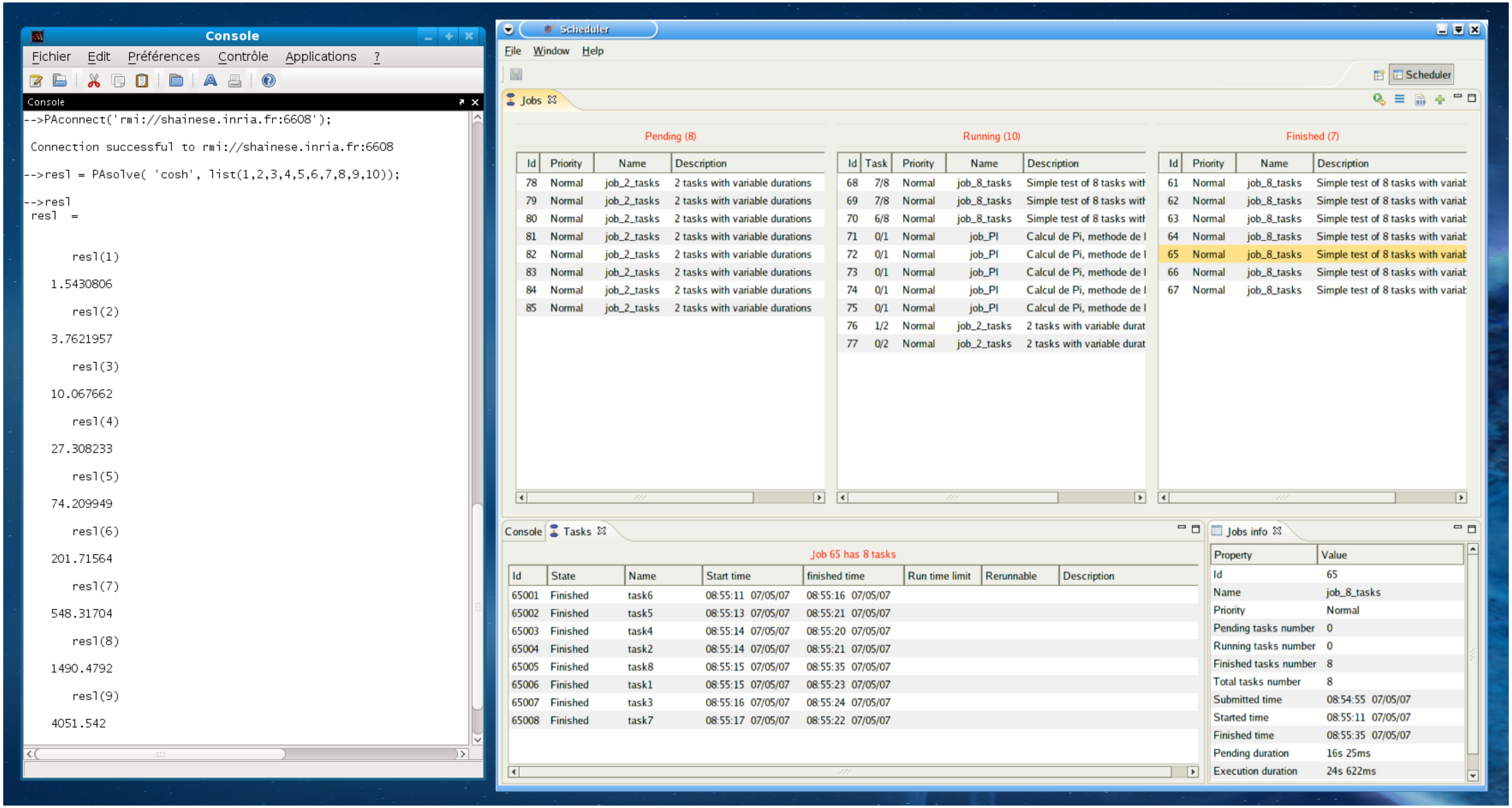
Dedicated resources

Desktops

Amazon EC2

ProActive Parallel Suite

Interface ProActive ⇔ Scilab



ProActive Parallel Suite

Demonstration

The ProActive PACA Grid Platform (4)

Use Cases and Demonstration on a Production Platform

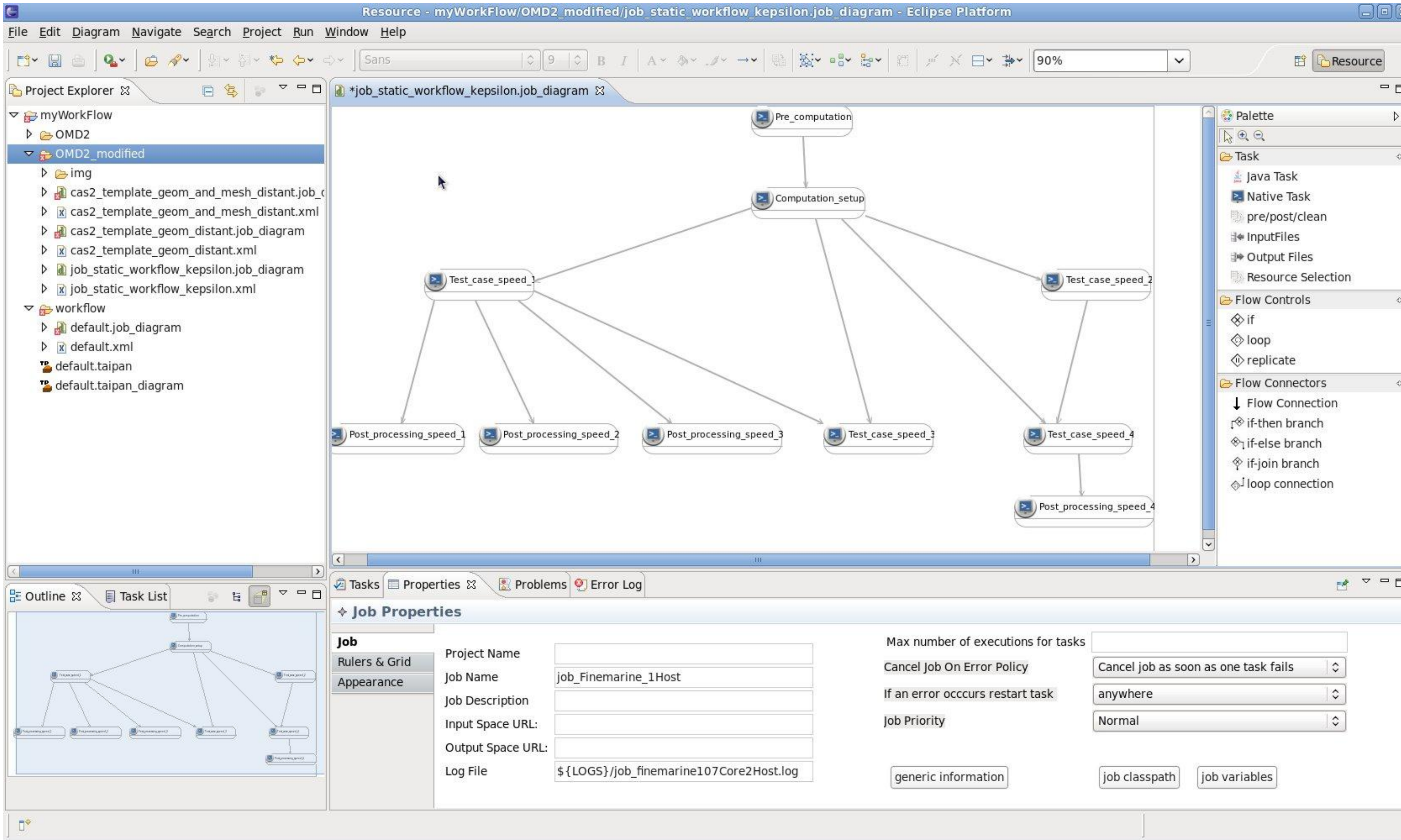
Total:

- 1 368 Cores
- 480 CUDA Cores
- 30TB Storage

Publically Available Today

ActiveEon
SCALE BEYOND LIMITS

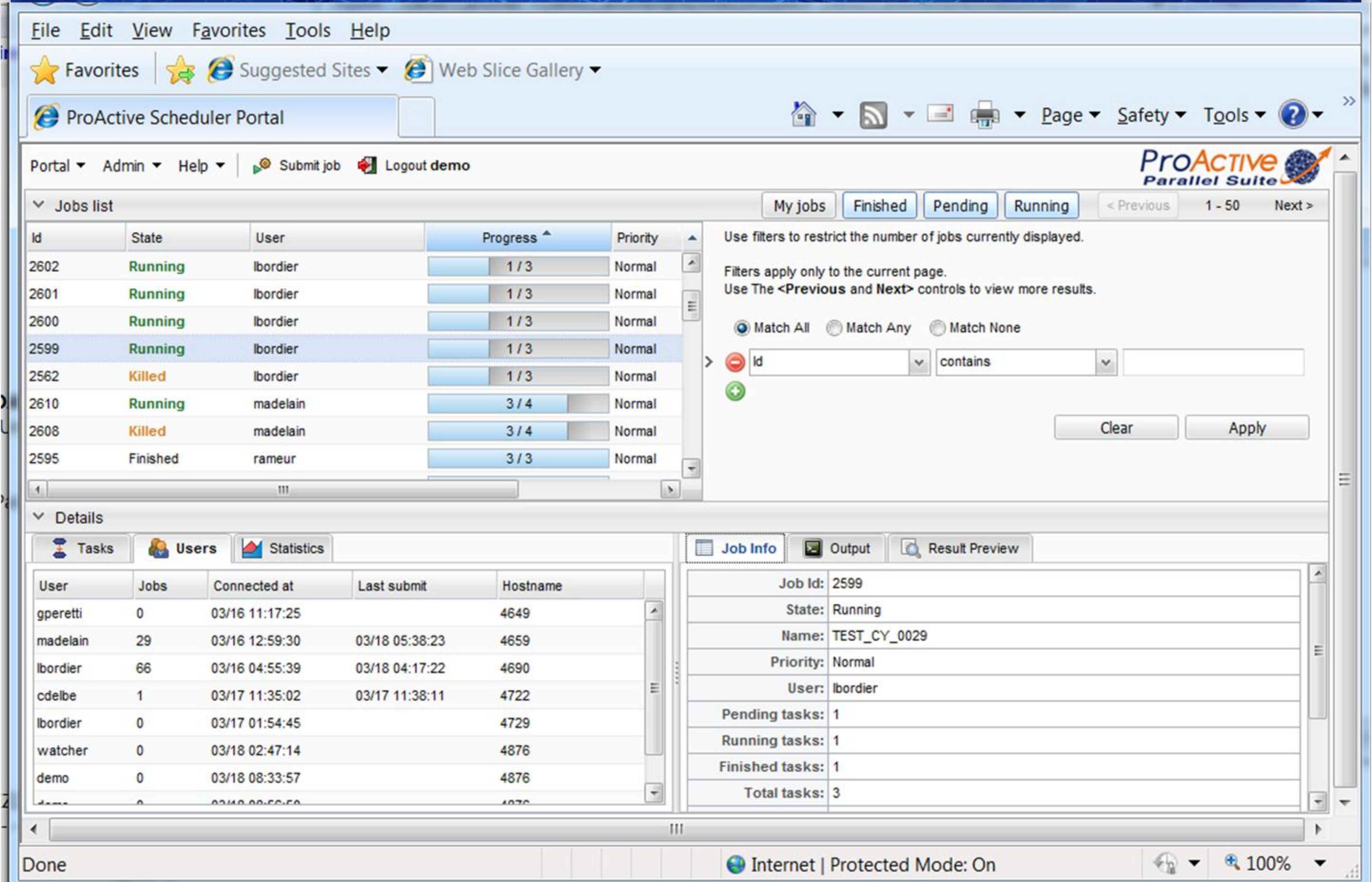
Workflow Studio



Activeeon SCALE BEYOND LIMITS

ProActive Parallel Suite

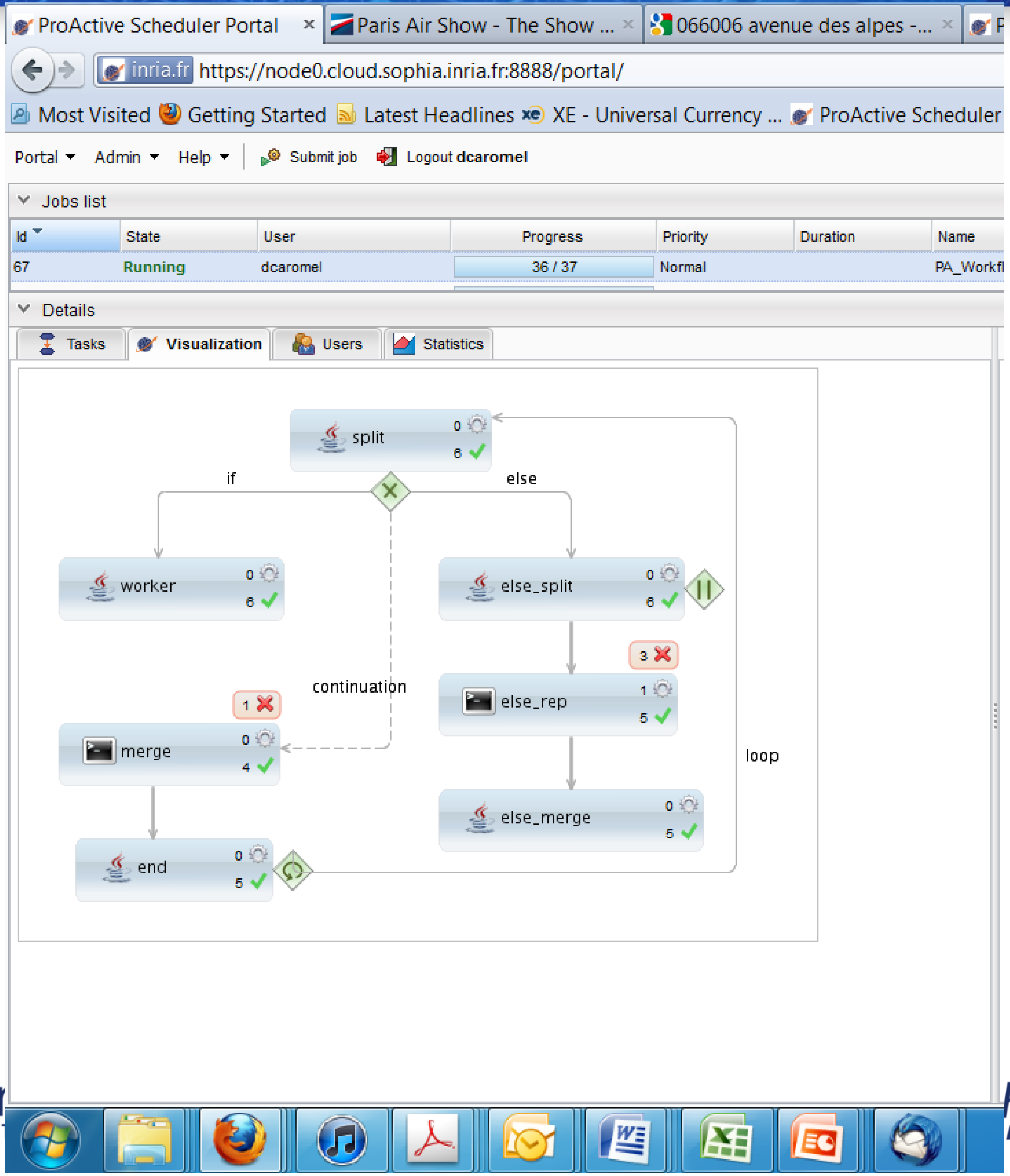
ProActive Orchestration Portal



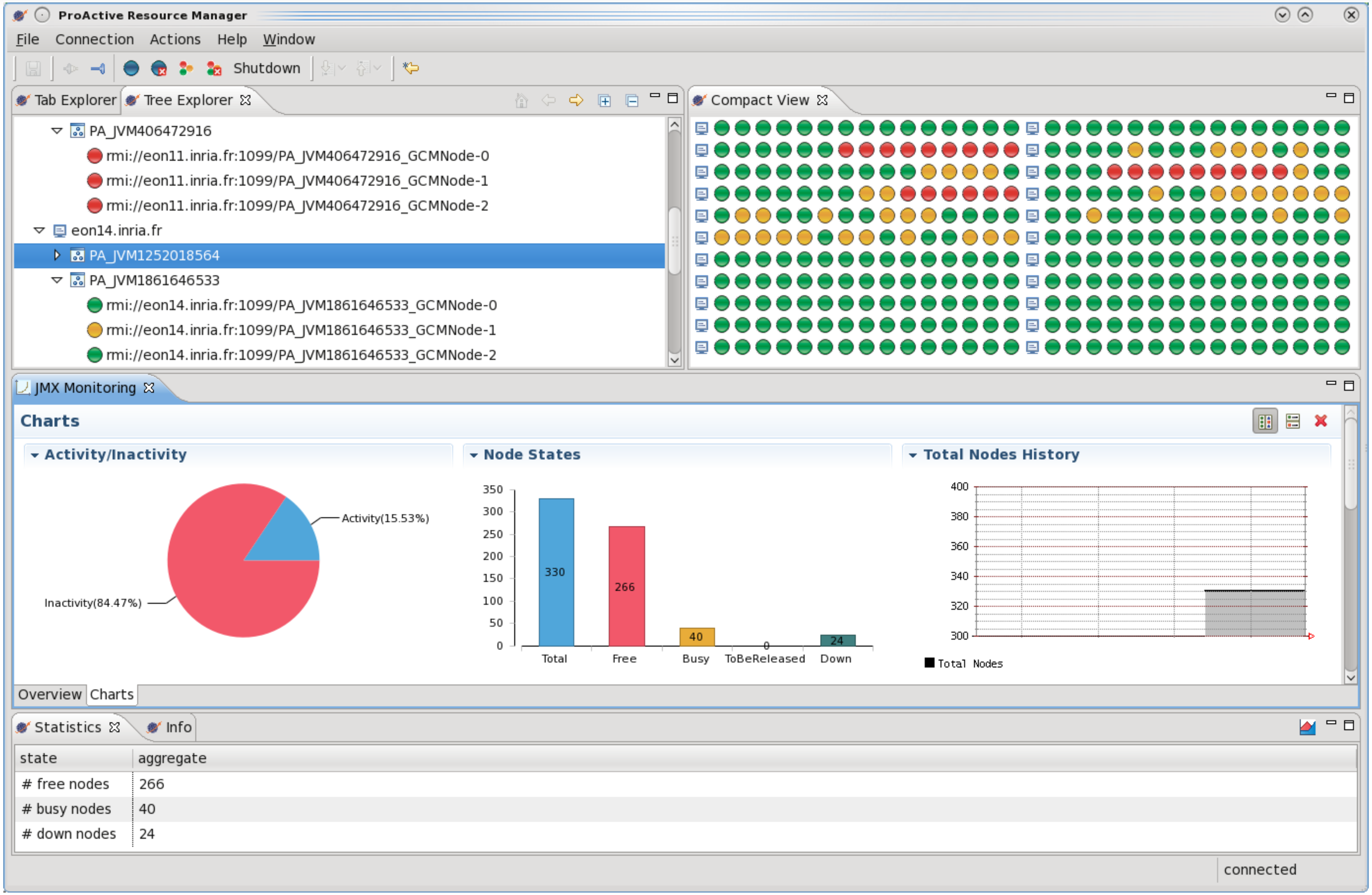
ActiveEon SCALE BEYOND LIMITS

ProActive Parallel Suite

Graphical Visualization of Workflow Execution



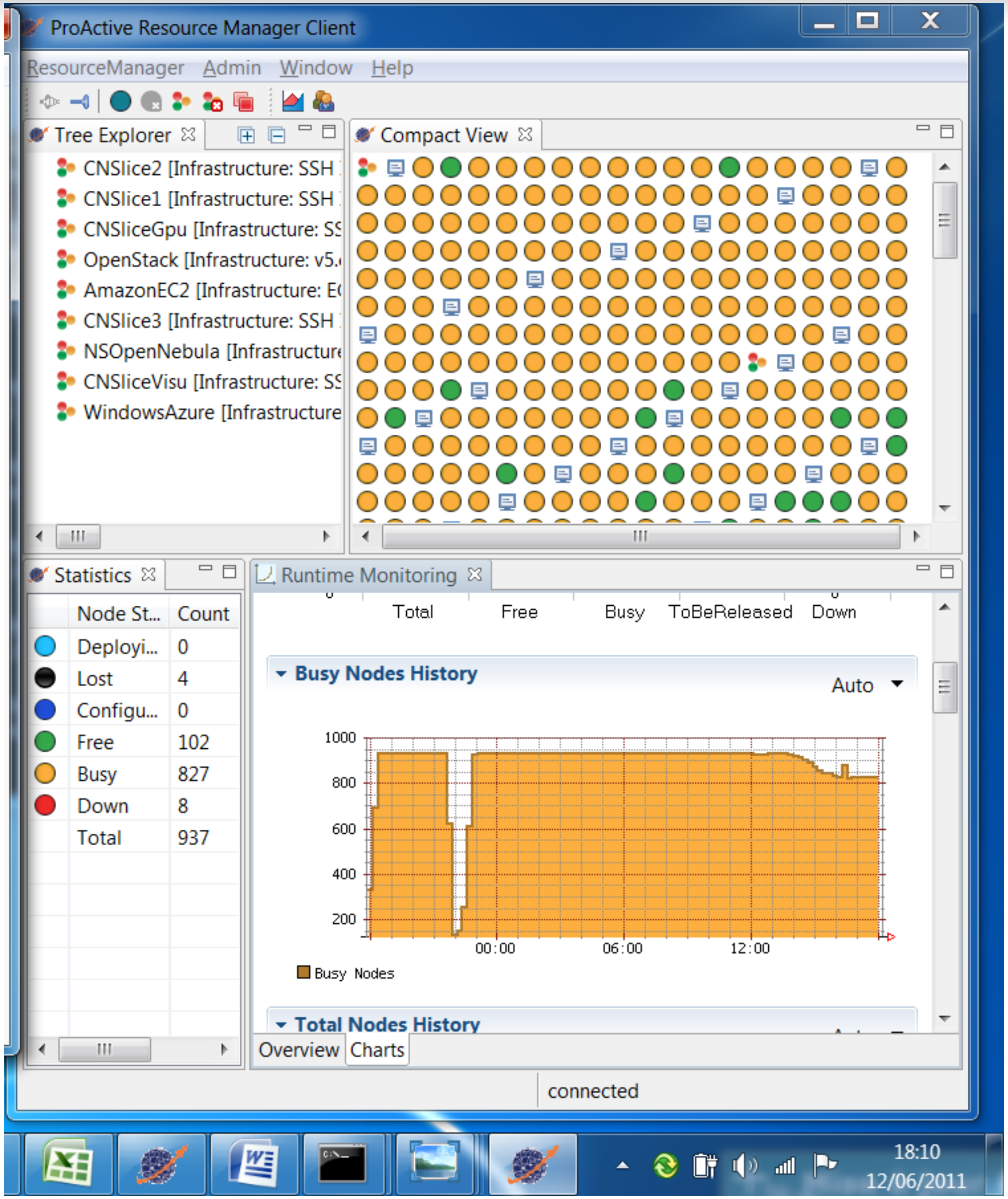
Heterogeneous Resource Management



ActiveEon SCALE BEYOND LIMITS

ProActive Parallel Suite

Sustained Load of 99 % over Long Periods



ActiveEon SCALE BEYOND LIMITS

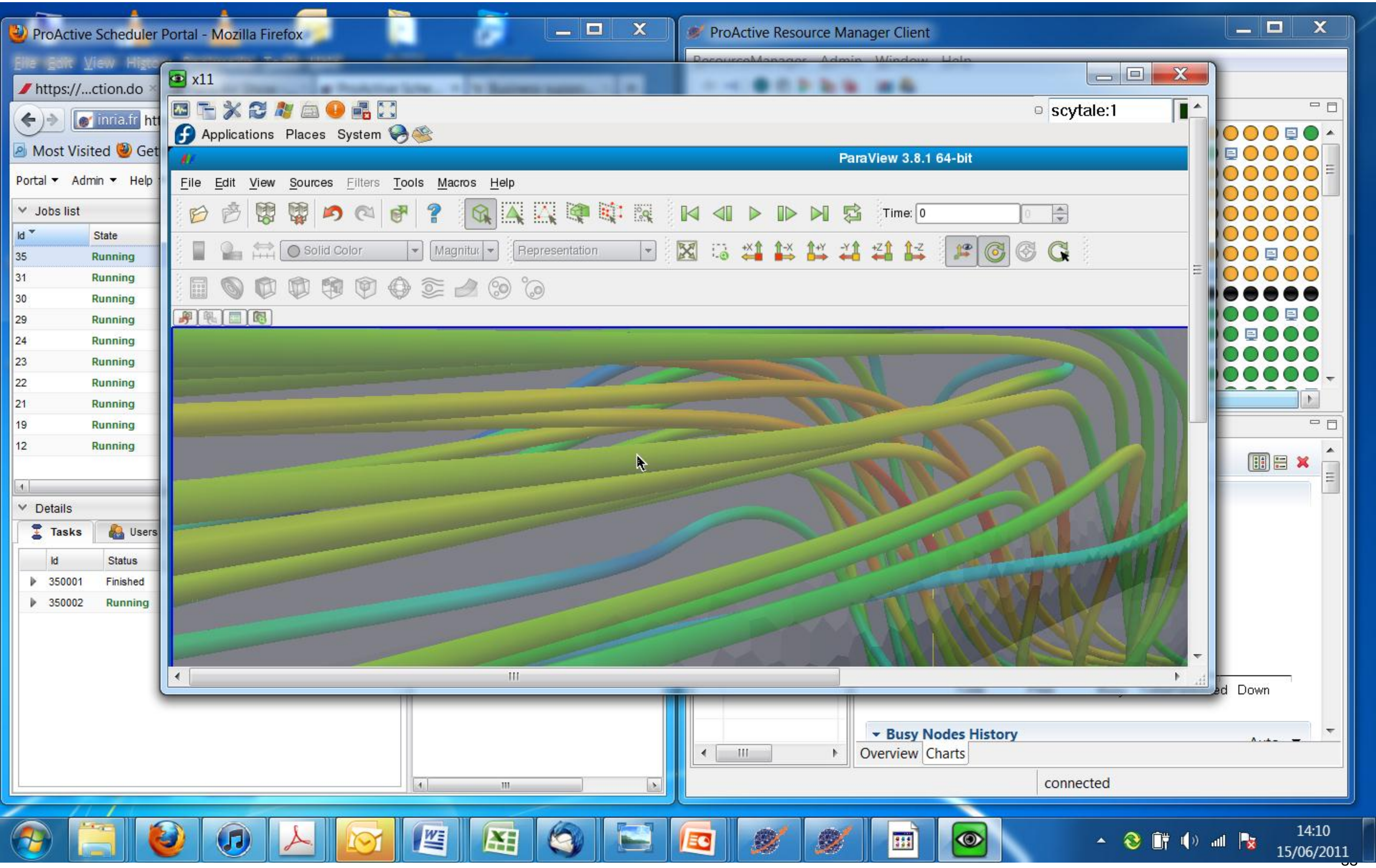
ProActive Parallel Suite

ProActive OMD2 Demo

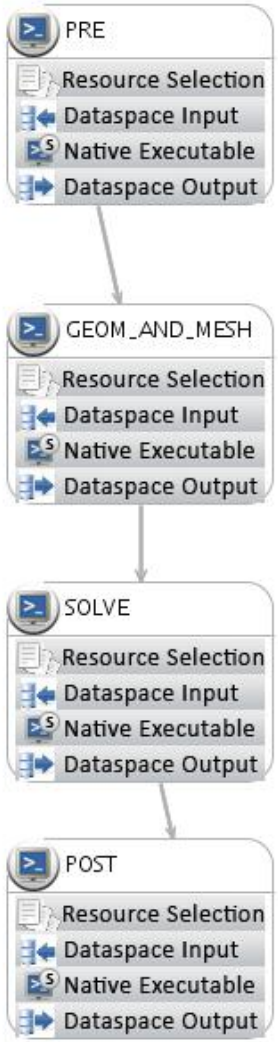
1000 Cores Production Cloud Portal

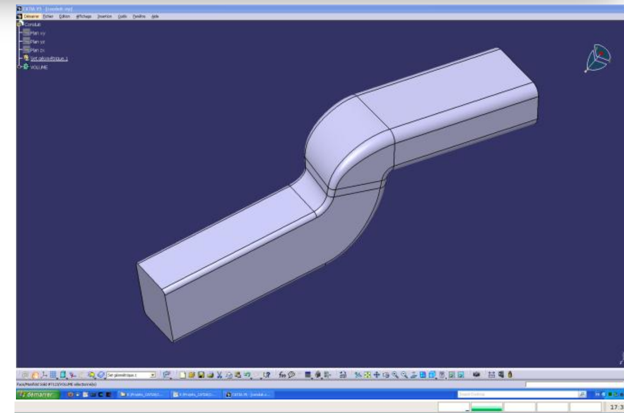
ProActive Parallel Suite

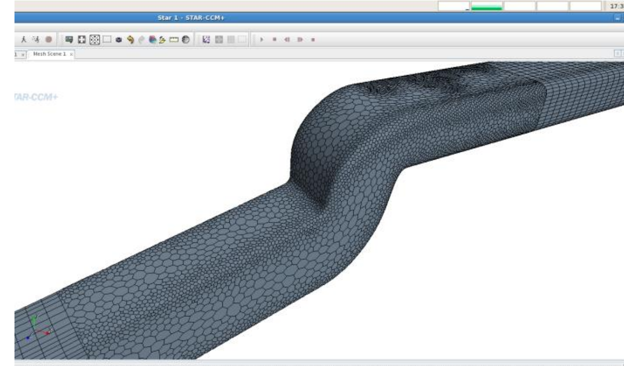
Remote Visualization Directly from Portal

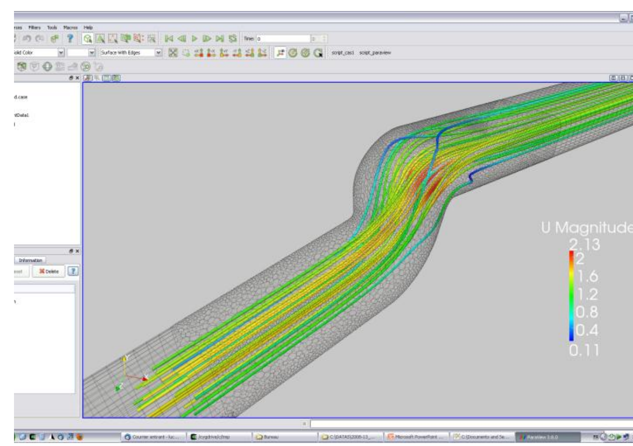


Engineering Optimizations: Renault UC









End Of Demonstration

ProActive Parallel Suite

Optimization algorithms for distributed computing infrastructures

CMA-ES (Covariance Matrix Adaptation Evolution Strategy) :
a state-of-the-art stochatic optimizer. λ calls to *f* are distributed on λ computing nodes at each iteration. What is the best parents population size ? (INRIA / TAO)

→ Default choice of CMA: $\mu=\lambda/2$ not optimal
→ $\mu=min(d,\lambda/4)$ close from optimal , $ln(\mu) \approx ln^2(ln(\lambda))$
→ Optimal convergence rate prop to $log(\lambda)$

M. Jebalia, A. Auger, Log-linear convergence of the scale-invariant (μ/μ_w, λ)-ES and optimal m for intermediate recombination and large population sizes, PPSN XI, 2010.

Distributed optimization based on kriging : use a kriging metamodel to summarize past and currently running simulations, decide new simulations based on multi-points expected improvements

D. Ginsbourger, et al., *Dealing with asynchronicity in kriging-based parallel global optimization*, WCGO-2011.

The OMD and OMD2 projects aim at solving realistic, simulation based OPTIMIZATION problems