Une stratégie pour la simulation dans les collaborations Mathématiques et Entreprises.

Illustration au travers de Feel++, une Librairie parallèle Open Source pour le Calcul Haute Performance

Christophe Prud'homme

AMIES
Centre de Modélisation et Simulation de Strasbourg(Cemosis)
Université de Strasbourg

2/7/2014

Mathématiques, Entreprises et Calcul

AMIES

Objectif d'AMIES sur l'interface maths-entreprises

- **Sensibiliser** : labos, entreprises, étudiants, décideurs
- **Inciter à et soutenir** la recherche partenariale
- **Rendre visible** : compétences, formations, success stories

Quelques faits sur AMIES

- Labex Labellisé en 2010, et UMS CNRS-UJF
- Budget : 500 keuros / an
- Réseau : tous labos, toutes maths
- Pilote: INSMI/CNRS,
- Partenaires : INRIA et Université de Grenoble

Les Programmes d'AMIES

Soutien / Incitation

- PEPS: projets exploratoires maths-entreprises
- Organisation / Soutien de manifestations/forums impliquant étudiants-labos-entreprises
- Mise en contact labos-entreprises à travers un réseau de facilitateurs

Vitrine

- Communiquer sur les évènements M-E, structurer la communauté M-E (site web, newsletter , groupe de discussions sur réseaux sociaux ..)
- Cartographier les expertises dans les laboratoires

Contexte National I

AMIES

- tête de pont de l'interface mathématiques-entreprises
- laboratoires de maths intégrés aux universités →**lien naturel avec les autres disciplines**

Structures locales/régionales Modélisation et Simulation

Des structures émergent au niveau local/régional pour l'animation de l'interface Modélisation/Simulation

- MaiMoSiNE (Grenoble) `http://www.maimosine.fr`
- CeMoSiS (Strasbourg) `http://www.cemosis.fr`
- ICS (Paris) `http://www.ics.upmc.fr/`
- Maison de la simulation `http://www.maisondelasimulation.fr/` (vocation nationale)

Contexte National II

Moyens de calcul

- Les mathématiciens ont accès aux structures de calcul
  - locales (Mesocentre)
  - nationales (GENCI et France-Grilles)
  - européennes (PRACE)

Logiciels Modelisation/Simulation

- Communauté mathématique très active dans le développement de **logiciels** (souvent libres)
- Logiciels libres à portée internationale développés en France (du laptop à Prace) : *e.g.* FreeFem++, GetFem++, Feel++...
- AMIES $\rightarrow$ mise en avant des expertises dans les laboratoires dans le domaine du logiciel et logiciel libre.

Voir également l'offre AMIES autour des logiciels libres

cemosis Centre de modélisation et de simulation de Strasbourg

amies

Contexte National III

Groupement de Recherche(GDR) Calcul

- animé par un réseau d'ingénieur de recherche sur toute la France
- anime un réseau d'experts en calcul scientifique et calcul haute performance
- propose des formations variées autour des différentes aspects du HPC en collaboration avec la maison de la simulation et les mésocentres

http://calcul.math.cnrs.fr/

Offre Autour de la Simulation et du Logiciel Libre

Un constat sur le calcul scientifique et le libre

- Cout prohibitif des logiciels commerciaux pour un grand nombre de types d'utilisation (PME, monde éducatif...)
- De nombreuses entreprises dont des grands groupes font le choix stratégique du logiciel libre en France et à l'étranger (e.g. EADS, EDF, AUDI, Mercedes, BMW...)
- La complexité croissante des problèmes de simulation et du besoin d'ouverture, de transparence et de reproductibilité et que le libre apporte des solutions dans chacun de ces domaines.
- Le logiciel libre en calcul scientifique est très présent en mathématique (développement et utilisation)
- Difficultés d'utilisateur des logiciels libres :
  - Hétérogénéité des outils
  - Absence de couche experte et de documentation

Besoin de développer une offre à l'interface math-entreprises en simulation autour du logiciel libre

cemosis
Centre de modélisation et de simulation de Strasbourg
amies

5 raisons pour lesquelles le logiciel libre va dominer dans le calcul scientifique I

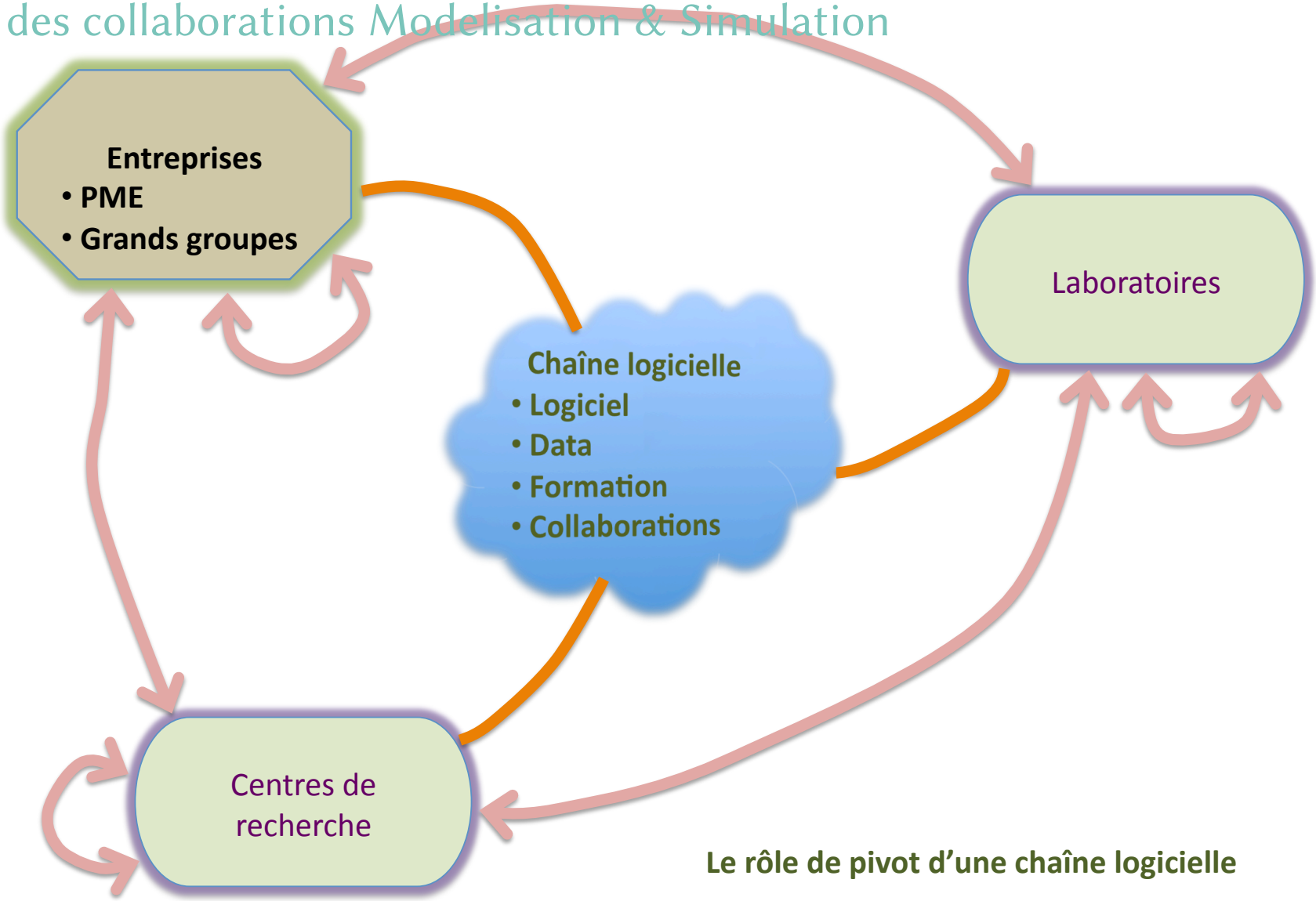
Paraphrasing W. Shroeder (Kitware)

1. **Open Science (Open Source+Open Data)** dans différents domaines (e.g. médical, sociétal, ...)
   - Ouverture, transparence, reproductibilité mais aussi
   - Innovation rapide, Comparaison juste des technologies, Source de formation pour les nouvelles generations d'ingénieurs
2. **La recherche de l'authenticité**: les entreprises peuvent tester les logiciels libres par leurs propres ingénieurs, développer leur propre expertise et éviter les blocages dues aux technologies propriétaires
3. **Collaboration de haute qualité et agile autour du développement du logiciel** grace à un processus de développement logiciel éprouvé et des outils de développement ouverts

5 raisons pour lesquelles le logiciel libre va dominer dans le calcul scientifique II

4. **Scalabilité** en termes de methodologies, technologies, données, ressources informatiques mais aussi en termes de processus de développement. Le modèle open source a montré toute sa force en terme de scalabilité a contrario des modèles fermés (bazaar versus cathédrale).
5. **Business Model** collaboratif et amical plutôt que des semi-monopoles avec des licences très chères (fréquent en calcul scientifique) dont le coût est très souvent associé au nombre de CPU utilisés (licence par CPU)

Modalités d'une collaboration Math-Entreprise : Chaines Logicielles au coeur des collaborations Modelisation & Simulation

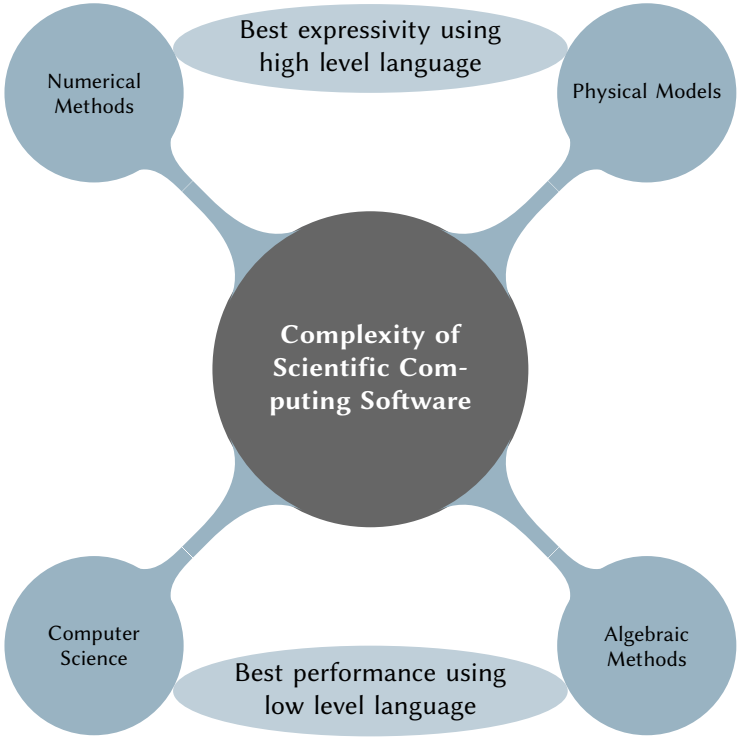


Le rôle de pivot d'une chaîne logicielle

Offre Math-Entreprises Simulation

Illustration au travers du project Feel++

Generative Programming and DS(E)L

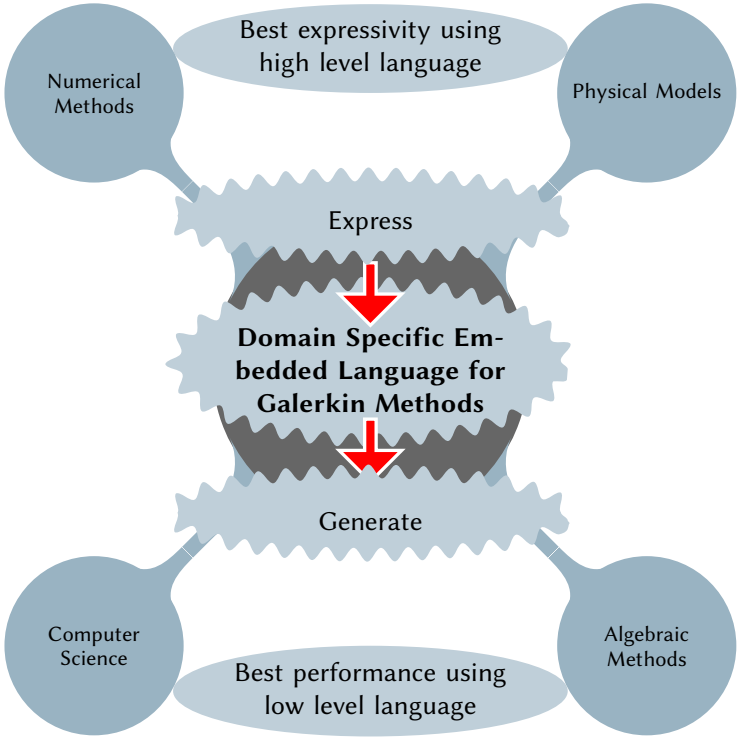


Complexity Types

- Algebraic
- Numerical
- Models
- Computer science

- Numerical and model complexity are better treated by a high level language
- Algebraic and computer science complexity perform often better with low level languages

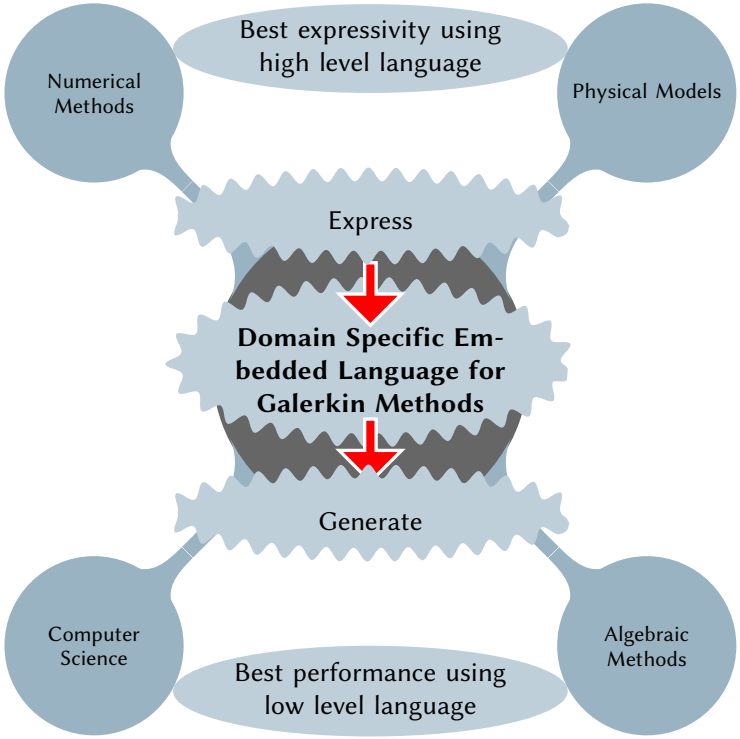
Generative Programming and DS(E)L



Generative paradigm

- distribute/partition complexity
- developer: The computer science and algebraic complexity
- user(s): The numerical and model complexity

Generative Programming and DS(E)L



Definitions

- A *Domain Specific Language* (DSL) is a programming or specification language dedicated to a particular domain, problem and/or a solution technique
- A *Domain Specific Embedded Language* (DSEL) is a DSL integrated into another programming language (e.g. C++)

A Concrete Solution : Feel++

http://www.feelpp.org

Finite Element Embedded Library and Language in C++

A Domain Specific Language for PDEs embedded in C++ providing a syntax very close to the mathematical language

Features

- Supports generalized arbitrary order Galerkin methods (cG, dG) in 1D, 2D and 3D
- Supports simplex, hypercube and high order meshes
- Supports seamless parallel computing
- Supports large scale parallel linear and non-linear solvers (PETSc/SLEPc)

cemosis
Centre de modélisation et de simulation de Strasbourg

A Concrete Solution : Feel++

http://www.feelpp.org

Current Funding

ANR-Cosinus-HAMM,
ANR-MONU-VIVABRAIN,
ANR-MONU-CHORUS,
FRAE-RB4FASTSIM,
Idex-Strasbourg-Cemosis
Labex-Strasboug-IRMIA

Dependencies

- C++11
- MPI (parallel comp.), HARTS
- Boost (C++ lib.)
- PETSc (solvers), SLEPc
- Gmsh (mesh gen.), VTK, ANN, GLPK

Feel++ Consortium

- Université de Strasbourg: IRMA
- Université de Grenoble: LIPHY and LJK
- University of Coimbra: Dept. of Mathematics
- CNRS: LNCMI

Process

- GitHub public(Feel++) and private(Collaborations) repositories
- Mailing lists per project
- Continuous Integration : Travis
- Nightly Tests: CDash
- Doxygen
- Dissemination: multiple media (Youtube, ...)

Licensing

Free software/GPL & LGPL-3 Available in Debian/Ubuntu, Redhat based, MacOSX(Macports,Homebrew), and some PRACE supercomputers : Curie, Fermi, Supermuc

Example of DS(E)L – $-\nabla \cdot (k\nabla u) = f$

Applied Mathematicians favorite equation: find u such that

$$-\nabla \cdot (k\nabla u) = f, \quad u = g \text{ on } \partial\Omega$$

which, using a Galerkin method, reads find $u \in X_h$ such that $\forall v \in X_h$

$$\int_\Omega \nabla u \cdot \nabla v = \int_\Omega fv$$

which leads to solving $Au = b$ where

$$\begin{aligned} A &= \left(\int_\Omega \nabla\phi_i \cdot \nabla\phi_j \right)_{i,j}, \\ f &= \left(\int_\Omega f\phi_i \right)_i \end{aligned}$$

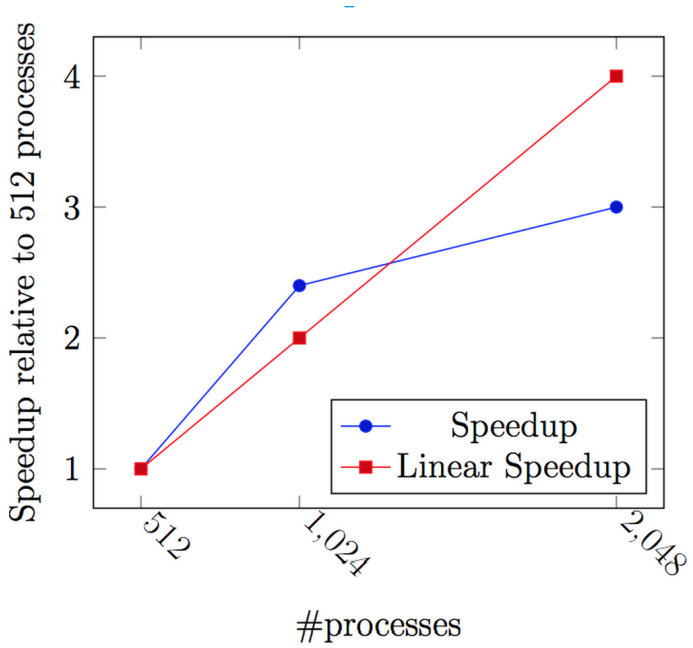
```
auto mesh = Mesh<Simplex<2>>::New();
loadGMSHMesh( _mesh=mesh, _filename="mesh.msh" );
// ℙ3 finite element space on
// triangular elements
auto Xh = Pch<3>(mesh);
auto u = Xh->element();
auto v = Xh->element();
// f a function e.g. f = 2π^2 sin(πx) * cos(πy)
auto f = expr("2 π^2 sin(πx)*cos(πy):x:y");
auto g = expr("sin(πPx())*cos(πy):x:y");
// ∫_Ω fv
auto l = form1(_test=Xh);
l= integrate( elements(mesh), f*id(v) );
// ∫_Ω ∇u · ∇v
auto a = form2(_test=Xh,_trial=Xh);
a = integrate( _range=elements(mesh),
     _expr=gradt(u)*trans(grad(v)) );
a+=on( boundaryfaces(mesh), u, l, g );
// solve a( u, v ) = l( v )
a.solve( _solution=u, _rhs=l );

std::cout << "‖u − g‖_L2 ="
          << normL2(_range=elements(mesh),
                    _expr=(idv(u)-g) )
          << "\n";
```

Consider the following incompressible Navier-Stokes solver,

- many variants can be implemented : oseen, fully-nonlinear, steady vs unsteady, turbulence(e.g. RANS), stabilisation, couplign with other physics...
- parallelism is seamless: scalability to thousands of cores

```
auto Xh = THch<2>( mesh ); // Taylor-Hood velocity + pressure
auto U = Xh->element(), V = Xh->element();
auto u = U.element<0>(), v = V.element<0>();
auto p = U.element<1>(), q = V.element<1>();
...
// test strain tensor
auto def = 0.5*(grad(v)  + trans(grad(v)));
// trial strain tensor
auto deft = 0.5*(gradt(u) + trans(gradt(u)));
// oseen
form2( _test=Xh, _trial=Xh) =
  // automatic quadature
  integrate( elements(Xh->mesh()),
             alpha*trans(idt(u))*id(v)
             + 2.0*nu*trace(trans(deft)*grad(v))
             + trans(gradt(u)*idv(beta))*id(v)
             - div(v)*idt(p) - divt(u)*id(q) );
```



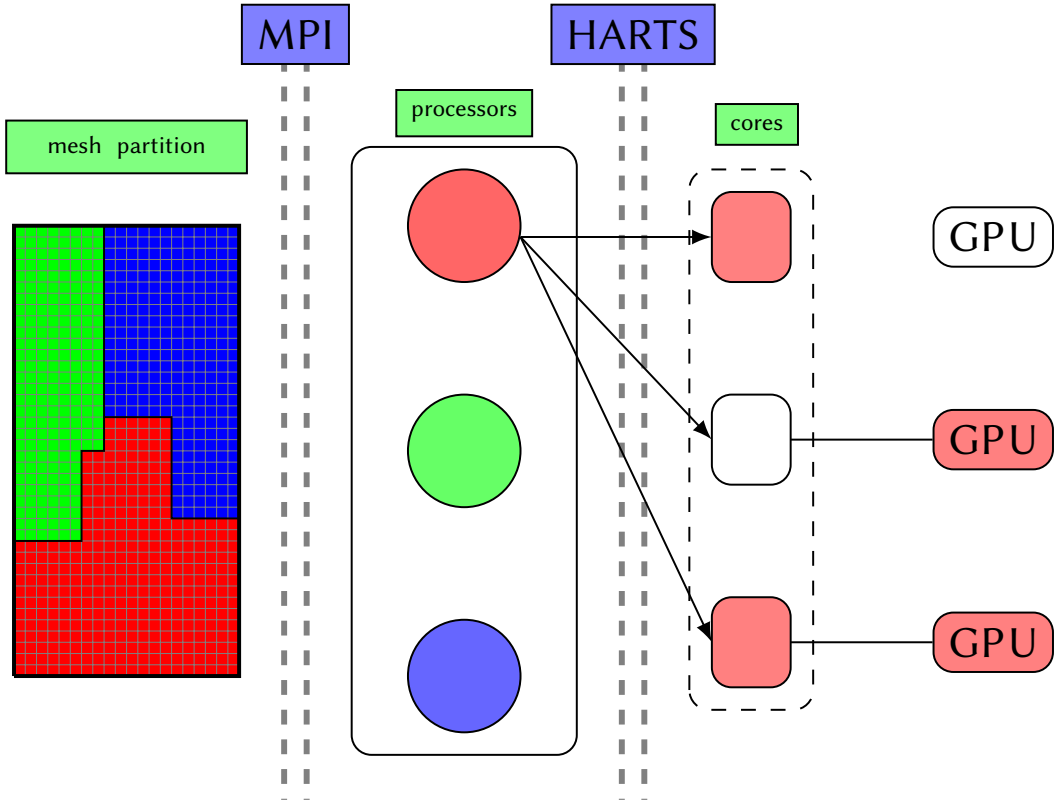
Seamless parallelisation

Hybrid architectures

- many nodes, many cores, hybrid nodes
- MPI, multi-threads, GPU

MPI implementation :

- mesh partitioning
- dof table partitioning
- PETSc interface

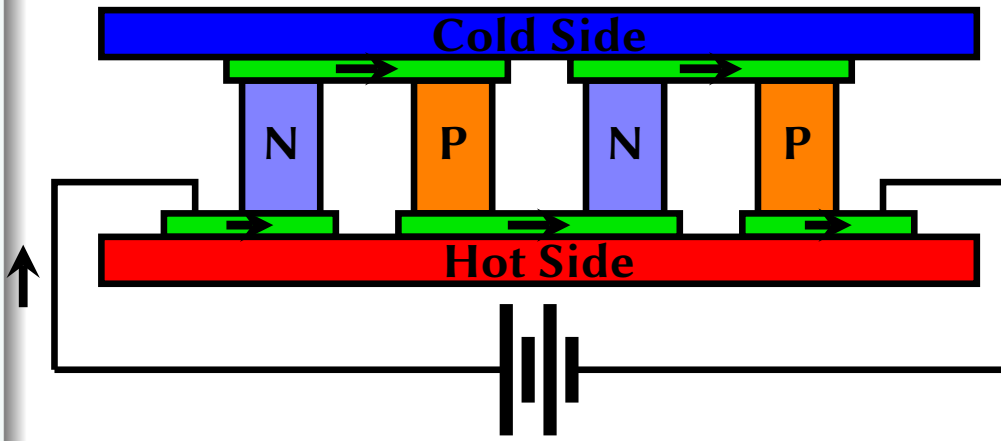


- The parallelism is completely transparent (implicit use)
- Parallelism can be also made explicit (control communications)

A SME in Alsace

Objectives

- Modeling and simulation of a detection device
- Need control of device temperature, use of Peltier modules (non-linear electro-thermal models)
- Uncertainty quantification, sensisitivity analysis, Control...



$$\rho C \frac{\partial T}{\partial t} + \nabla \cdot \mathbf{q} = Q \qquad \text{sur } \Omega_i, \quad \forall\, 1 \leq i \leq N_{mat} \tag{1}$$

$$\nabla \cdot (\mathbf{j} + \epsilon \partial \mathbf{E}/\partial t) = 0 \qquad \text{sur } \Omega_i, \quad \forall\, 1 \leq i \leq N_{mat} \tag{2}$$

$$T = T_D \qquad \text{sur } \Gamma_D^T \tag{3}$$

$$V = V_D \qquad \text{sur } \Gamma_D^V \tag{4}$$

$$\mathbf{q} \cdot \mathbf{n} = T_N \qquad \text{sur } \Gamma_N^T \tag{5}$$

$$\mathbf{j} \cdot \mathbf{n} = V_N \qquad \text{sur } \Gamma_N^V \tag{6}$$

$$\mathbf{q} \cdot \mathbf{n}_1 = -\mathbf{q} \cdot \mathbf{n}_2 \qquad \text{sur } \partial\Omega_i \cap \partial\Omega_j,\ i \neq j \tag{7}$$

$$\mathbf{j} \cdot \mathbf{n}_1 = -\mathbf{j} \cdot \mathbf{n}_1 \qquad \text{sur } \partial\Omega_i \cap \partial\Omega_j,\ i \neq j \tag{8}$$

A SME in Alsace: Electro-Thermal Benchmark with Comsol

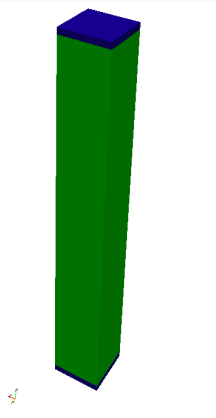
At the top :

- $T = 273.15K$
- $V = 0V$

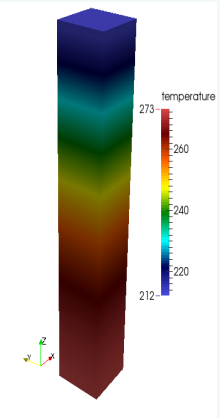
At the bottom :

- heat load q
- current I

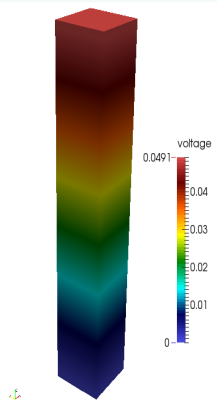
Others : adiabatic



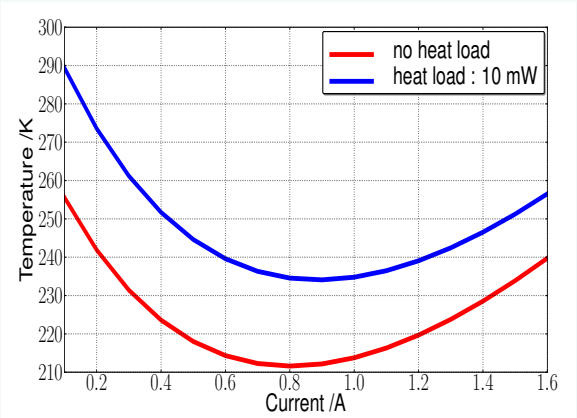
(a) Geo



(b) T

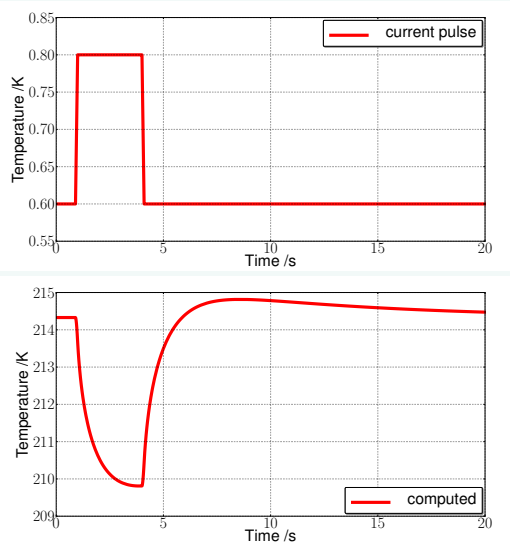


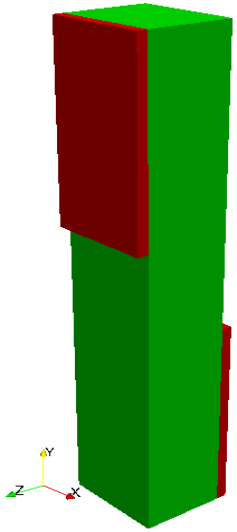
(c) V



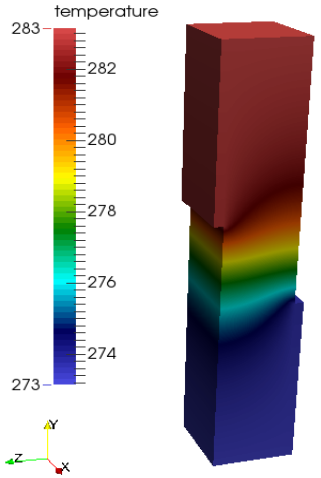
(d) Temperature cold side

Transient

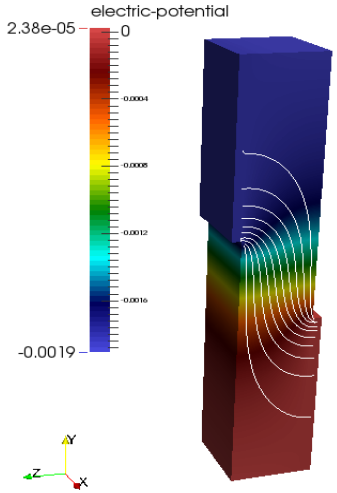




(e) Temperature

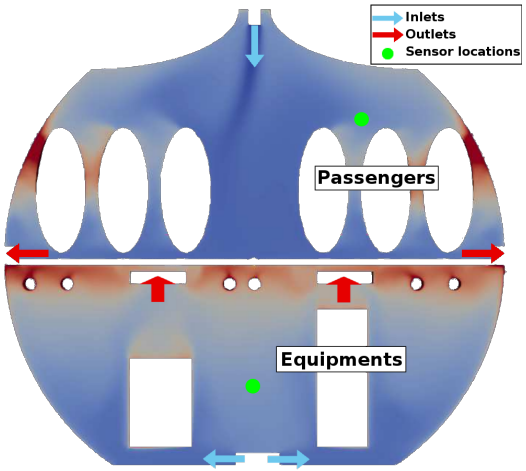


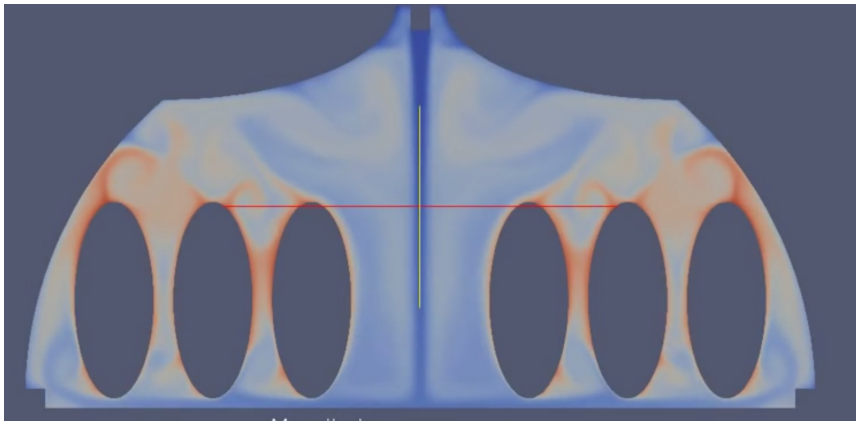
(f) Temperature



(g) Voltage

ANR MONU CHORUS: EADS, EDF, Orange, Phimeca, Imacs, ECN, UStrasbourg, UParis7





Objective

Apply reduced basis methods on an aerothermal simulation in an avionic bay

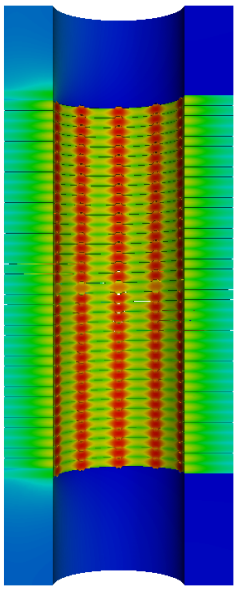
Applied Issues

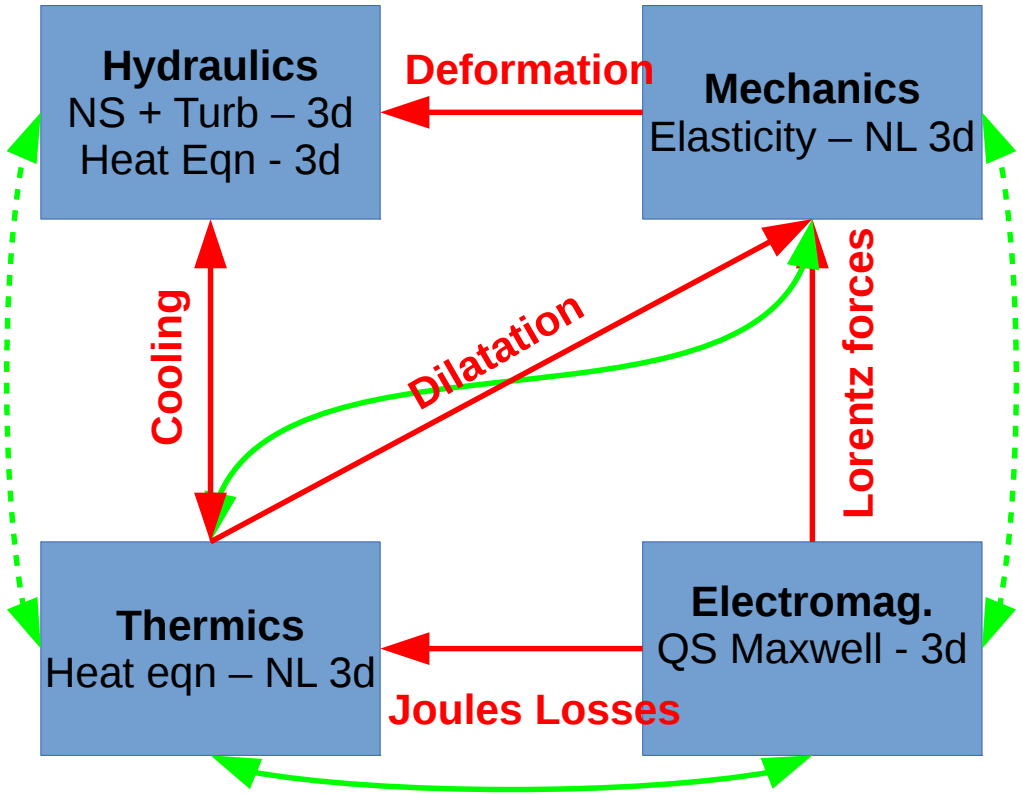
- Reduce costs
- Improve equipments safety
- Improve current models
- A step to a wide use of meta-models in industrial design cycle

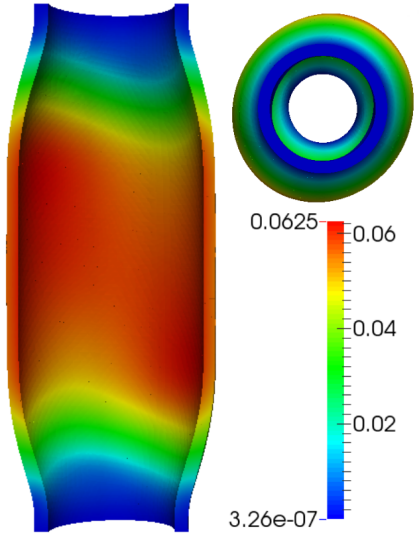
Some Scientific Issues

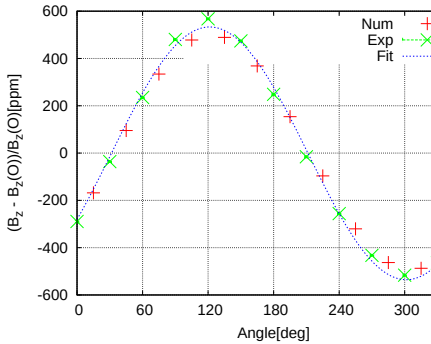
- Tturbulence
- Mixed forced and natural convection
- Boundary conditions coupled to an ECS (Environment Control System)
- Error prediction

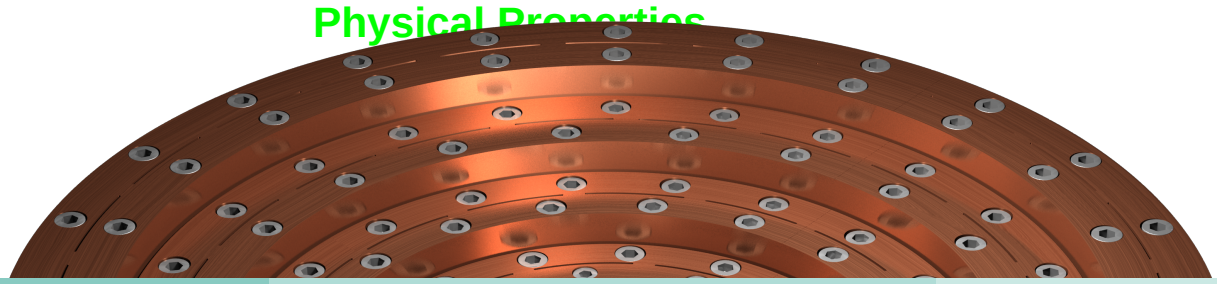
High Field Magnets: LNCMI, SigmaPhi, CEA, UDS, UBS











High Field Magnets: LNCMI, SigmaPhi, CEA, UDS, UBS

Objectives

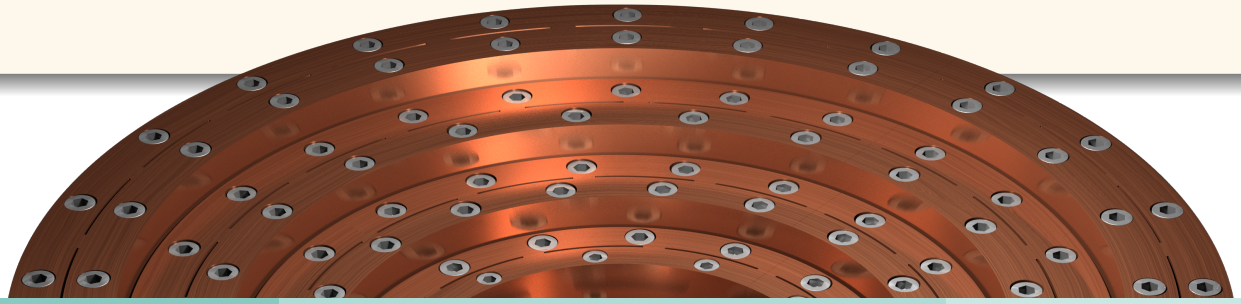
- Build Hierarchical Models : From simple analytical model to full 3D
- Simulation : from small part of the Magnets to full geometry
- Sensisitivity Analysis, Parameters identification, UQ
- Optimization under uncertainties

Choices

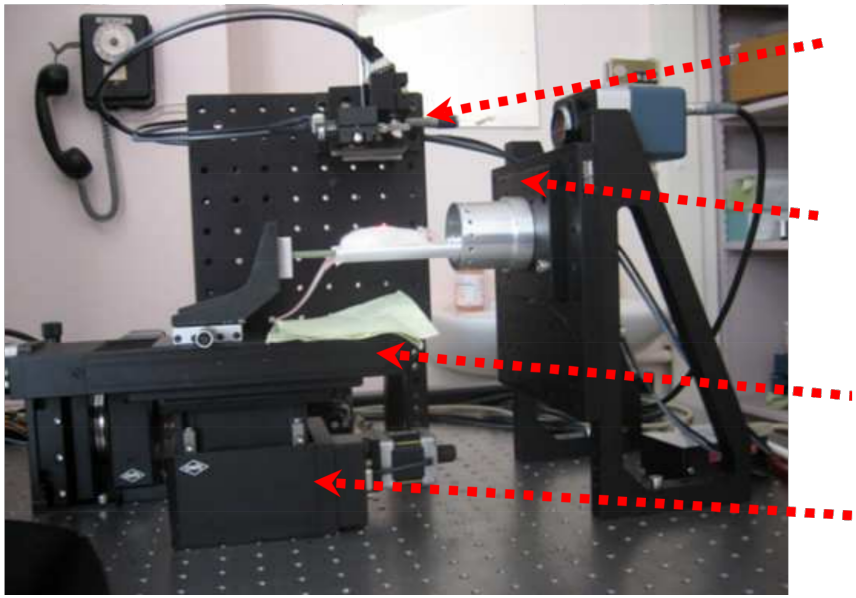
- State of the art Finite Element Methods (High Order)
- Use of Open Source sofware or library to keep control over our development

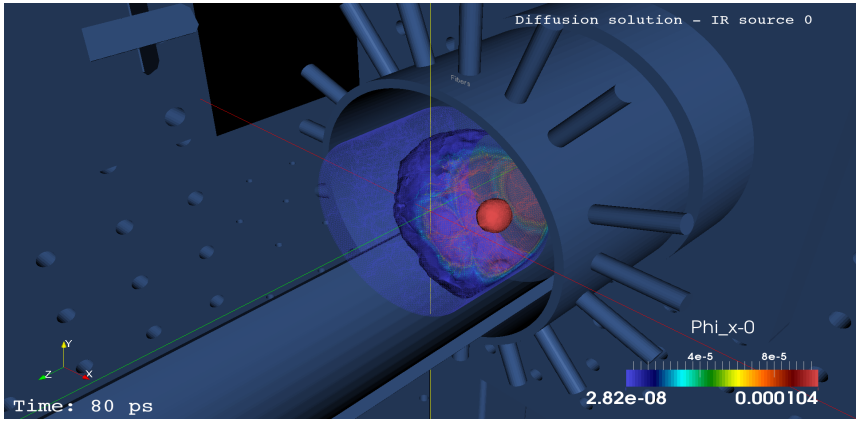
Requirements

- Generic and easily be extended
- Reduction Order Model to tackle large geometry and Multi-Physics
- HPC



Diffuse Optical Tomography: Hopital Civil de Strasbourg, UDS, UHA





Principle

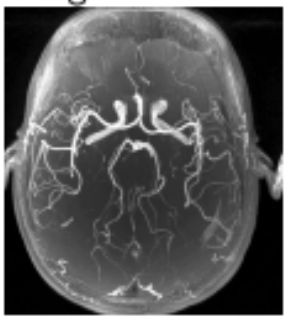
- Pulse an infrared light (IR) on the skin of a patient and measure at the opposite side the photon density diffused across the body.
- Add a fluorescent marker (Indocyanine green ICG) to improve the contrast.

Objectives

- Reconstruct the absorption and diffusion maps to understand the photons propagation and determine "opaque areas" assimilated to tumours.
- Provide a good image resolution to be put into practice in hospitals.
- Build a reconstruction software associated to the tomograph

ANR MONU Vivabrain: Kitware, URCA, UDS, UJF, ESIEE

Image ARM réelle



Traitement multi-modal d'image →

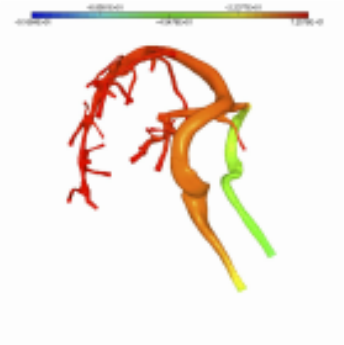
Modèle vasculaire 3D

↑

Simulation d'angiographies virtuelles

Génération de maillage 3D

↓

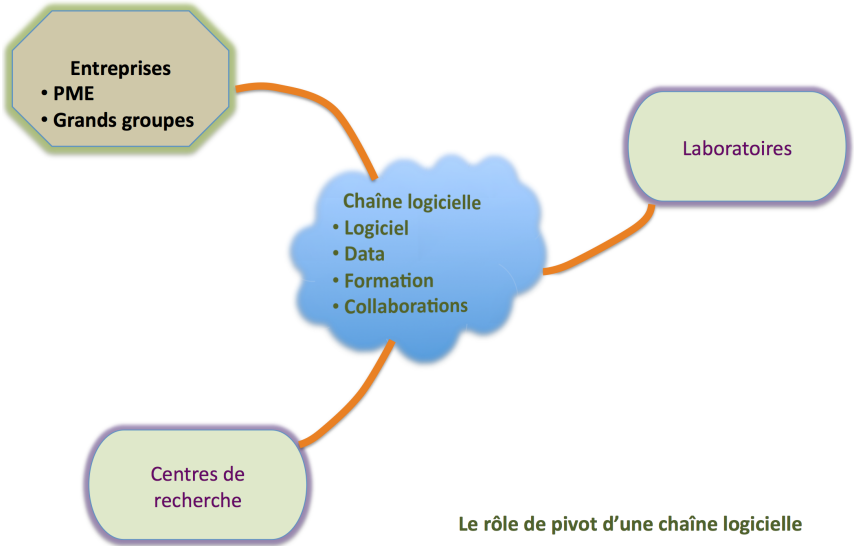


Simulation 3D+t d'écoulement sanguin ←

Champ de pression

Maillage 3D

Perspectives & Conclusion



Le rôle de pivot d'une chaîne logicielle

Autres projets Feel++ et Entreprises

- Holo3 : Metrologie
- PlasticOmnium : Mécanique des fluides
- Bull : Calcul Haute-Performance (ANR HAMM)
- De nouvelles collaborations à venir ...

Modèle de Collaboration

- Le modèle est en place sur un certain nombre de collaboration autour de Feel++,
- Les résultats sont maintenant attendus,
- Des ajustements en fonction des projets seront peut-être nécessaires.

cemosis
Centre de modélisation et de simulation de Strasbourg