

Apport de l'approche objet pour capitaliser et faire coopérer des codes de calcul

Michel Lemoine

***ONERA-Centre de Toulouse / DPRS / SAE
Michel.Lemoine@cert.fr***

Les codes de calcul à l'ONERA

- **Développeurs :**
 - des physiciens,
- **Pourquoi :**
 - résolution d'équations : Navier-Stokes pour l'aérodynamique, de Maxwell pour l'électromagnétique, ...
- **Comment :**
 - écriture de solveurs : 50.000 à 300.000 lignes de Fortran 77
- **Pour qui :**
 - des utilisateurs finals qui combinent manuellement les solveurs avec des mailleurs en entrée (SIMAIL, IDEAS, ...) et des visualiseurs en sortie (UNIRAS, TECPLOT, ...)
- **Constat :**
 - un solveur \Leftrightarrow un développeur
 - les solveurs d'un domaine partagent au moins 95% de leurs fonctionnalités
 - pour un domaine donné : des dizaines de solveurs totalement indépendants

Problèmes

- **Des clients (internes ou externes) de plus en plus exigeants :**
 - les prototypes des années 70 / 80 sont satisfaisants :
 - fonctionnalités conformes aux besoins
 - performances excellentes
 - mais un prototype même très opérationnel n'a pas les qualités actuellement requises pour les logiciels :
 - interopérabilité : pouvoir combiner avec d'autres logiciels
 - adaptabilité : faire évoluer facilement les solveurs
 - facilité d'utilisation : avoir des interfaces plus conviviales
- **Des physiciens-chercheurs non satisfaits :**
 - des non informaticiens transformés en développeurs de logiciels
 - et surtout en mainteneurs

Pourtant les solveurs constituent savoir et savoir-faire de l'ONERA

Solutions envisageables et envisagées

- **Componisation à la Corba**

- **avantages :**

- pas de modification de l'existant : gain de temps
 - robustesse complétée
 - combinaison plus facile avec d'autres codes
 - introduction sans problème de nouvelles interfaces

- **inconvénients :**

- code monolithique figé, adapté à une machine et pas susceptible d'évolution
 - nécessaire intervention de spécialistes informatiques

- **Refonte totale**

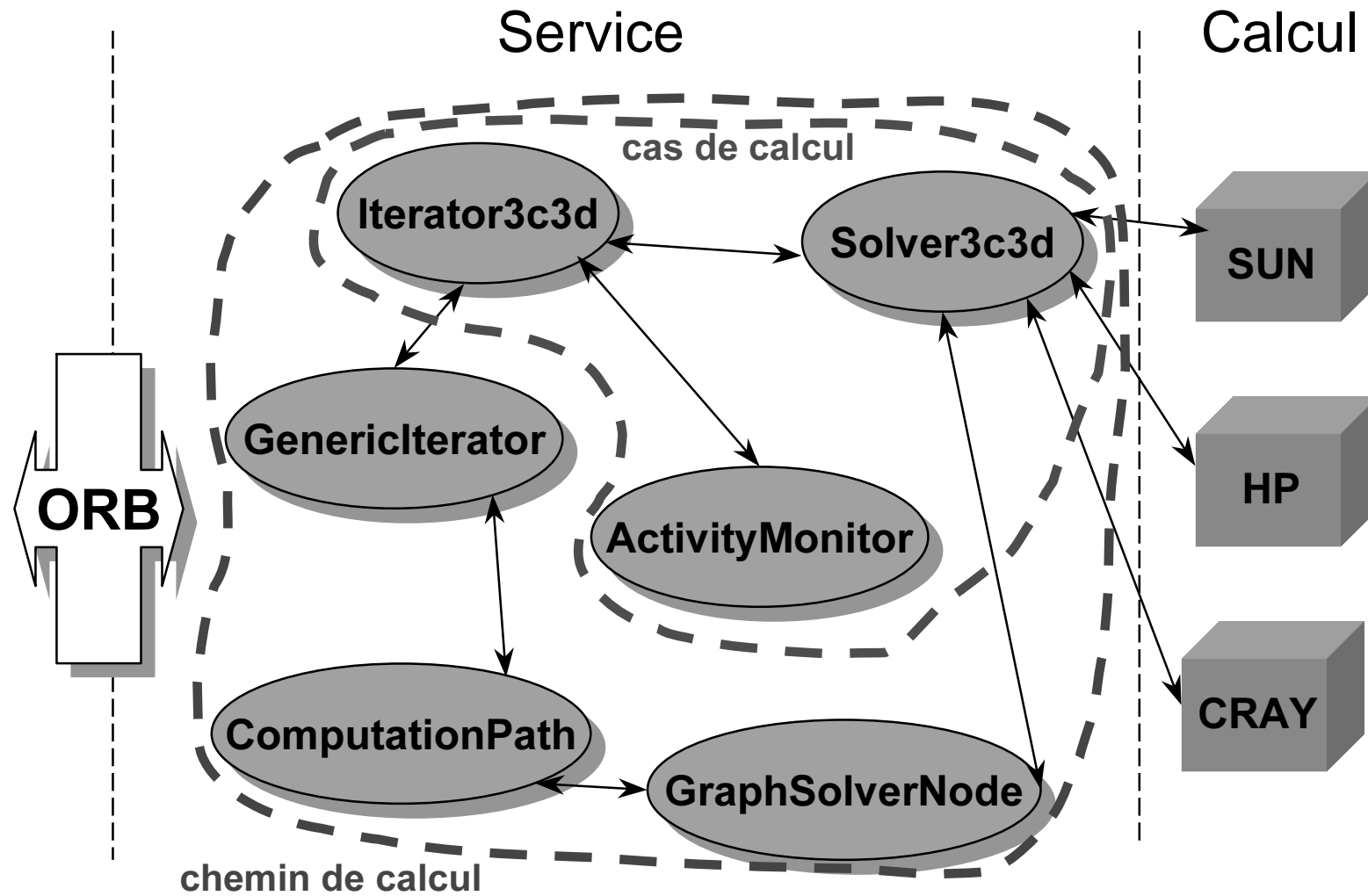
- **avantages : reconception complète visant à satisfaire de nouveaux besoins et de nouvelles qualités**

- **inconvénients :**

- coûts
 - formation des physiciens à des techniques de Génie Logiciel

Composant 3C3D

ENST, 03/02/2000



Bénéfices du modèle composant et de l'architecture 3 parties

- **Calculs :**
 - codes inchangés, conservation des compétences.
- **Services :**
 - isolation des codes clients des codes de calcul,
 - encapsulation \Rightarrow réutilisation de l'existant,
 - composition \Rightarrow réutilisation d'autres composants.
- **Clients :**
 - IHM pour répondre aux besoins utilisateurs,
 - portabilité grâce à l'utilisation de JAVA,
 - simplicité de mise en œuvre.

Refonte complète : eLSA

- **eLSA : ensemble Logiciel de Simulation en Aérodynamique**
 - 10 personnes sur 5 ans
 - débuté en 97
- **Clients : Aérospatiale-BA, SNECMA, ...**
- **Qui : des aérodynamiciens + des informaticiens**
- **Quoi :**
 - version V0 : écoulements de fluides incompressibles visqueux mono-espèces, stationnaires et instationnaires, sur des maillages tridimensionnels fixes structurés par domaines
 - modèle général statique objet
- **Comment :**
 - reprise de l'existant par encapsulation C++
 - interfaces C++
- **Inconvénients : le temps de redéveloppement**

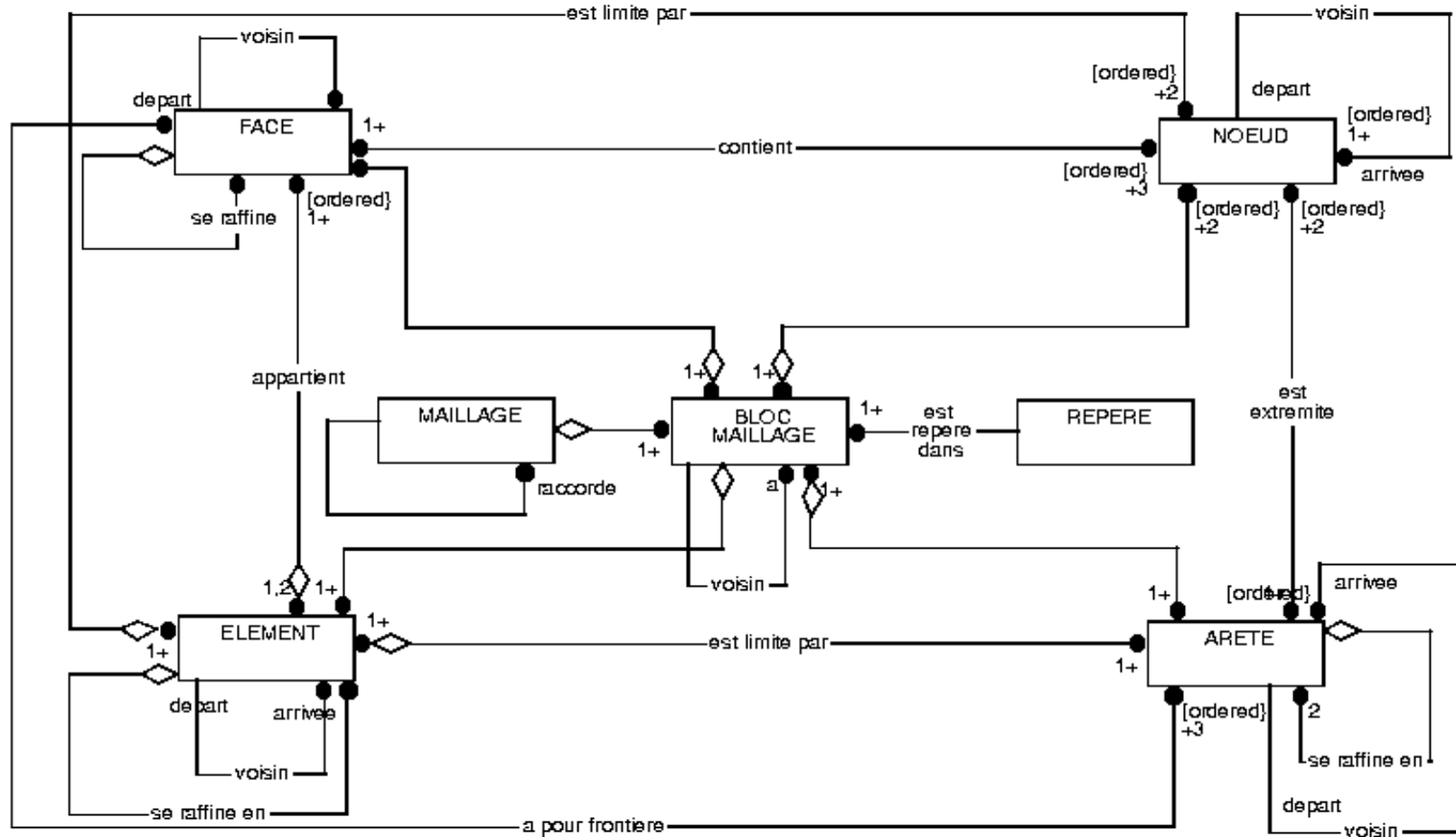
Nouvelle approche : Données Métier, Encapsulation et Glu

- **Principe :**

- **demander aux physiciens de repenser (sur le papier) leurs solveurs en adoptant un vocabulaire commun**
- **établissement d'un document de référence des données métier**
- **élaboration d'un référentiel objet des données métier**
- **identification des services à offrir aux solveurs**
- **modélisation objet des services**
- **réalisation C++ et Java des services**
- **encapsulation des solveurs à la Corba**
- **glu avec HLA ou un outil ad hoc**

Le projet PAME

- **PAME : Projet d'Atelier de Modélisation Electromagnétique**
 - objectif ambitieux : l'avion complet
- **Données métier : une bible de 200 pages**
- **Les services : PPTM**
 - vérification des formats des fichiers de maillages
 - pré-traitements avec ou sans visualisation
 - post-traitement : uniquement de la visualisation
- **Modélisation UML**
- **Réalisation : C++**
- **Glu : OML**



En guise de conclusion

- **Apport incontestable de l'approche objet mais**
- **Convaincre des physiciens**
- **Nécessité de créer des équipes mixtes**
- **Nécessité de délimiter les compétences : qui fait quoi !**
 - Les données métier : un réceptacle pour le savoir et le savoir-faire
 - La modélisation OO : un moyen de dialoguer et de partager
 - L'implémentation Java ou C++ : une affaire d'informaticiens
 - La glu : un problème épineux !
 - Les performances : Fortran n'est pas mort !