

*Journée de rencontre des utilisateurs du pôle
de calcul intensif pour la Mer.
IFREMER – 15 mai 2008*

OASIS & Palm

2 coupleurs développés
au CERFACS



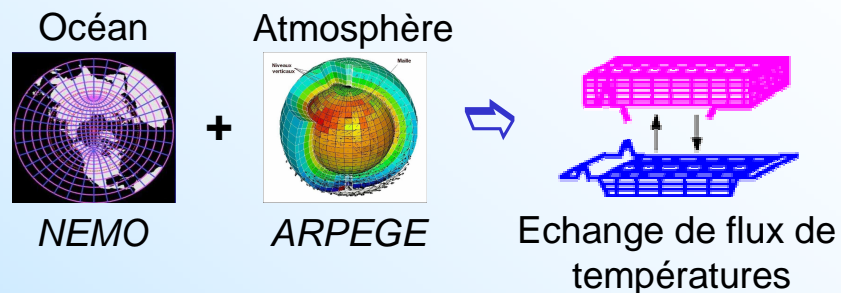
Anthony Thevenin (thevenin@cerfacs.fr)

-
- ① Le couplage de codes
 - ② OASIS : un coupleur spécialisé pour l'étude de systèmes climatiques
 - ③ PALM : un coupleur dynamique de codes parallèles
 - ④ Eléments de comparaison entre OASIS et PALM

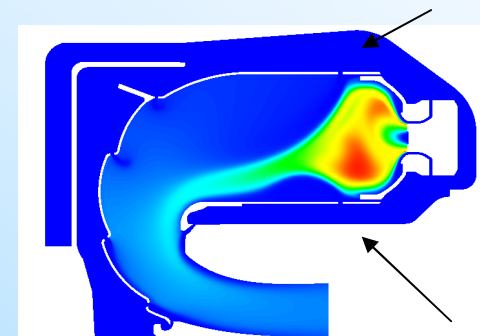
→ Pourquoi coupler des codes de calculs ?

↪ Traiter un système dans sa globalité à partir de différents modèles

ex : couplage océan-atmosphère



↪ Créer des applications avec des codes existants
ex : optimisation dans le cadre d'un couplage fluide - structure : recherche de la position optimale des entrées d'air dans une chambre de combustion



↪ Décomposer un système complexe en plusieurs éléments simples et ensuite coupler ces éléments (ex : assimilation de données)

→ Première solution : fusion de deux codes de calcul

Modèle d'océan

```
Program prog1
...
Call sub_prog2(in, out)
...
end
```

Modèle d'atmosphère

```
Program prog2
Subroutine sub_prog2(in,out)
...
end
```

⇒ Très efficace en temps de calcul (passage des données par adresse, ...)

⇒ **MAIS :**

- ⇒ Problèmes d'intégration (pas d'indépendance informatique des codes)
- ⇒ Degré zéro de la flexibilité (maintenance, évolution, ...)
- ⇒ Problème éventuel de mémoire

⇒ C'est de l'assemblage et non du couplage (pas de parallélisme)

→ Deuxième solution : utiliser un protocole de communication

Modèle d'océan ↔ Modèle d'atmosphère

```
Program prog1
...
Call XXX_send(prog2, data)
...
end
```

```
Program prog2
...
Call XXX_rcv(prog1, data)
...
end
```

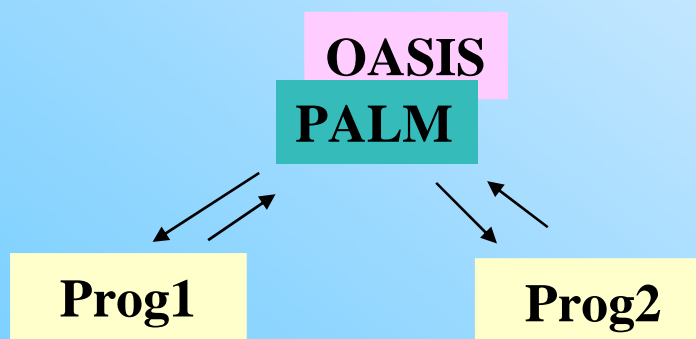
↪ Plus ou moins efficace selon le mécanisme

↪ **Mais :**

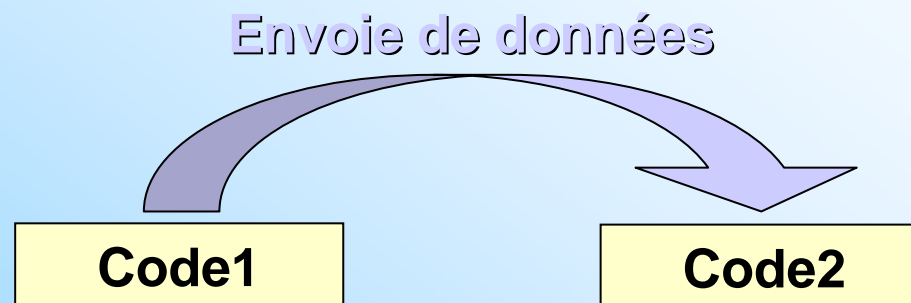
- ↪ Couplage non générique
- ↪ Demande une bonne expertise en calcul parallèle
- ↪ Couplage non flexible (parallélisme, interpolation, ...)
- ↪ Pas toujours portable selon le mécanisme utilisé
- ↪ Trop complexe avec plus de deux codes et de nombreux échanges

→ Troisième solution : utilisation d'un coupleur

- ⇒ Un coupleur est un logiciel qui permet d'échanger des informations entre plusieurs codes et de gérer leur exécution
- ⇒ Le coupleur lance les exécutables en **parallèle**
- ⇒ Quand on veut échanger des données on appelle des primitives PUT/GET génériques
- ⇒ Le coupleur utilise un protocole de communication portable et efficace (MPI)
- ⇒ Le coupleur propose des outils :
 - ⇒ **interpolation** : généralement, les modèles n'ont pas la même grille
 - ⇒ **redistribution automatique** : les modèles ne tournent pas sur le même nombre de processeurs

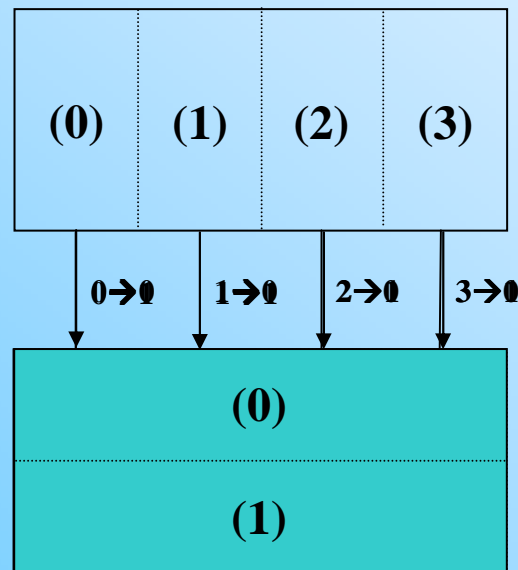


→ Schéma de communication "end point" :



- ① Lorsqu'un code de calcul produit une donnée potentiellement intéressante pour d'autres codes, il en informe le coupleur :
 - ⇒ appel à une primitive PALM_Put ou OASIS_Put
 - ② Lorsqu'un code de calcul a besoin d'une donnée, il informe le coupleur qu'il est en attente d'informations :
 - ⇒ appel à une primitive PALM_Get ou OASIS_Get
- ↪ Ces 2 phases sont entièrement décorréliées :
- ⇒ totale indépendance des codes
 - ⇒ possibilité de changer un code sans changer le reste de l'application

- Double niveau de parallélisme du coupleur :
 - ↪ Il lance les modèles de manières parallèles
 - ↪ Il peut prendre en charge des modèles parallèles
- Interpolation :
 - ↪ PALM utilise les algorithmes d'interpolation 2D d'OASIS
- Redistribution :
 - ↪ Redistribution automatique des objets dont la distribution est différente dans le code source et dans le code cible d'une communication (nombre de processeurs différents)



OASIS

Un coupleur spécialisé pour l'étude
de systèmes climatiques

E-Mail : valcke@cerfacs.fr

URL : <https://oasistrac.cerfacs.fr/>

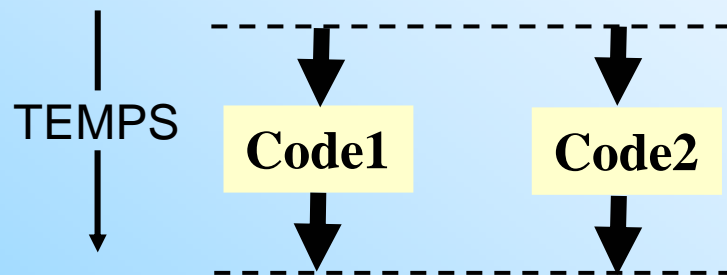
- ① Le couplage de codes
- ② **OASIS : un coupleur spécialisé pour l'étude de systèmes climatiques**
 - ↳ PRISM et OASIS
 - ↳ Concepts fondamentaux d'OASIS
 - ↳ OASIS 3
 - ↳ OASIS 4
 - ↳ Exemples d'utilisation
- ③ PALM : un coupleur dynamique de codes parallèles
- ④ Eléments de comparaison entre OASIS et PALM

- PRISM : Partnership for Research Infrastructure in Earth System Modeling
 - ↔ Consortium Européen
 - ↔ Partager le développement, la maintenance et le support d'outils pour la modélisation de systèmes climatiques
 - ↔ But : aider les modélisateurs à passer plus de temps sur les sciences

- OASIS est développé au CERFACS depuis 1991
 - ↔ OASIS : coupleur autorisant les échanges d'informations entre des codes représentant différents composants climatiques
 - ↔ Deux versions disponibles : OASIS 3 et OASIS 4
 - ↔ OASIS est développé dans le cadre de PRISM depuis 2001
 - ↔ OASIS 3 : ~ 25 groupes d'utilisateurs au niveau international
 - ↔ Aujourd'hui, OASIS4 est en beta test chez quelques utilisateurs

- Les développeurs actuels d'OASIS sont :
 - ↔ CERFACS, NEC (NLE-IT), CNRS

→ Couplage statique :



→ Les codes de calcul sont lancés en début de simulation et se terminent ensemble à la fin de l'application (double niveau de parallélisme)

→ Propriétés du coupleur :

→ OASIS3 :

- ⇒ Stable, utilisé par une large communauté climat
- ⇒ 2D uniquement
- ⇒ Communications partiellement parallèles / Interpolation non parallèle
- ⇒ La dernière version a été livrée en Septembre 2006 (prochaine version en été 2008)

→ OASIS4 :

- ⇒ Meilleure efficacité
- ⇒ Champs 3D
- ⇒ Totalement parallèle
- ⇒ Encore en développement

→ Configurer le coupleur :

- ↪ Identifier les modèles à coupler
- ↪ Identifier les champs de couplage à échanger
- ↪ Ecrire un fichier de configuration
- ↪ Instrumenter les codes des modèles : librairie de communication du coupleur

→ Ecriture du fichier de configuration (fichier texte) :

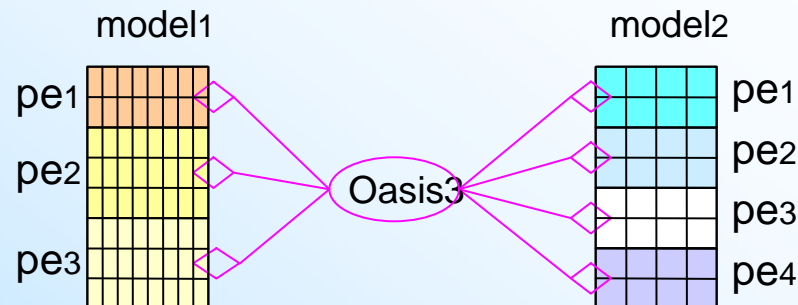
- ↪ Temps total de la simulation
- ↪ Nombre de champs de couplage
- ↪ Pour chaque échange : source et cible, interpolation

→ Utilisation d'outils spécifiques :

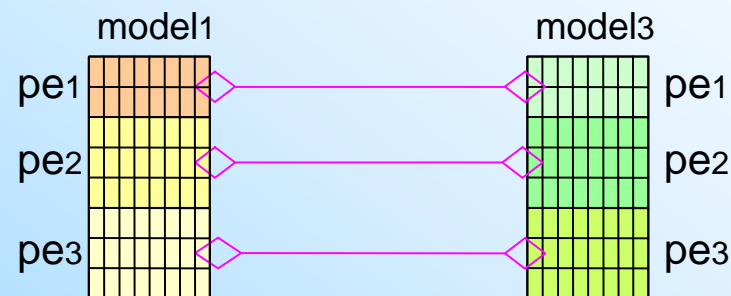
- ↪ Choix entre 4 types d'interpolations spatiales (de maillages)
- ↪ Moyenner ou accumuler les données automatiquement
- ↪ Lecture / écriture automatique de fichier de restart au début ou à la fin des runs

➔ Communications basées sur MPI :

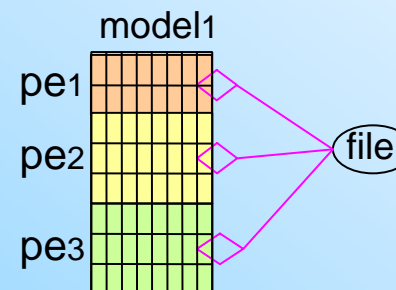
➔ Communication parallèle entre les modèles parallèles et le processus d'interpolation



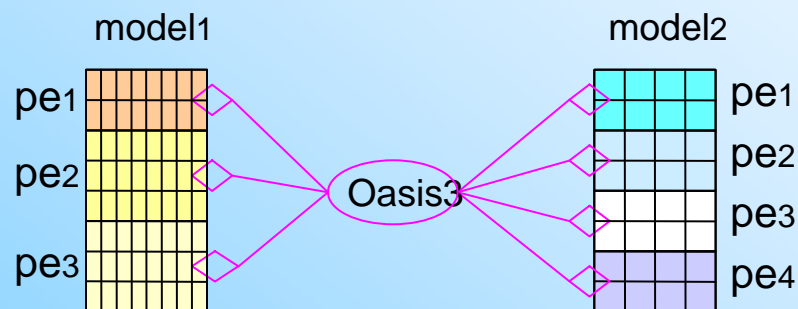
➔ Communication directe entre les modèles qui ont la même grille et la même partition



➔ Possibilité de lire et / ou d'écrire dans un fichier



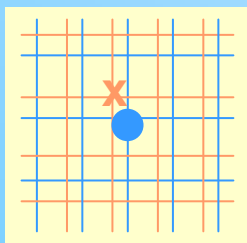
→ Une transformation des champs de couplage a lieu dans le cas :



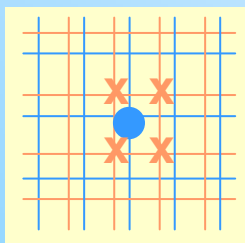
↪ Maillages différents : interpolation

↪ Nombre de processus source et cible différents : redistribution

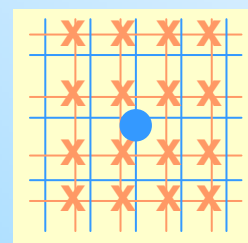
→ Différents types d'interpolation :



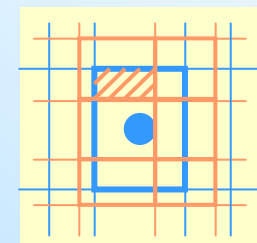
plus proche voisin



bilinéaire



bicubique



conservative

→ Différentes grilles supportées :

Structurées : lat-lon, étirée ou déformée (grille ORCA)

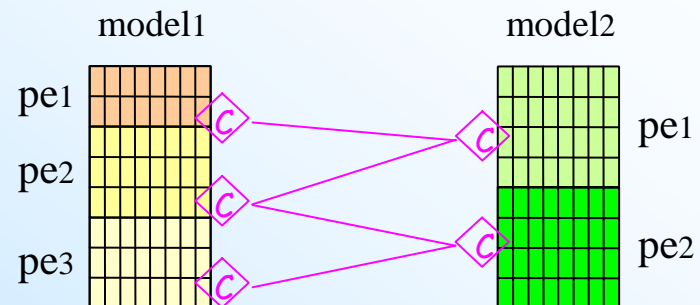
Atmosphérique gaussienne réduite, non structurée

- Nouvelle version du coupleur en développement depuis 2003 :
- Version beta disponible
- Aussi flexible qu'OASIS3 mais ENTIEREMENT parallèle :
 - ↪ Communications parallèles
 - ↪ Interpolation parallèle (algorithme multi grilles)
- Instrumentation des modèles :
 - ↪ Même type d'instrumentation qu'OASIS3
 - ↪ Possibilité de définir une grille 3D
- Ecriture du fichier de configuration :
 - ↪ Ecriture dans un fichier XML (Extensive Markup Language)
 - ⇒ Respect d'un schéma de configuration propre à OASIS4
 - ⇒ Possibilité de valider le contenu avant la simulation

→ Communications parallèles incluant les redistributions :

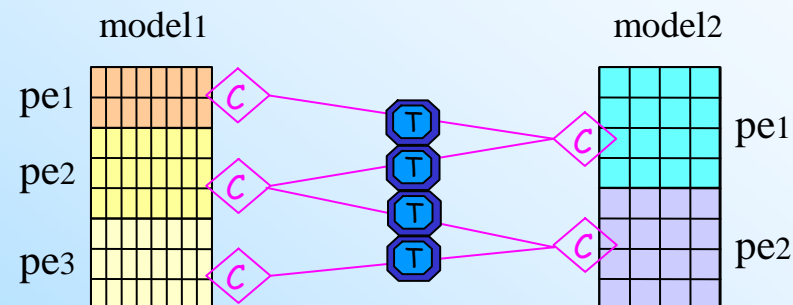
↪ Grilles identiques et partitionnement différents :

↪ Repartitionnement direct

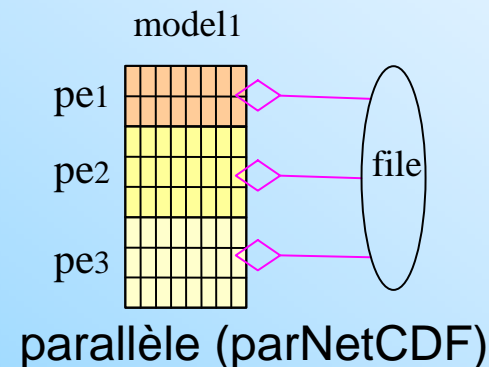
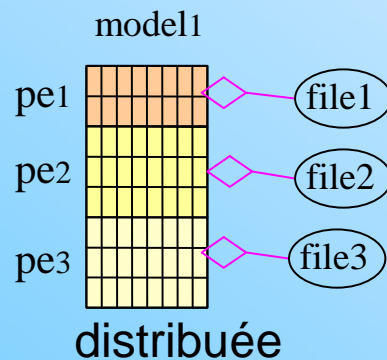
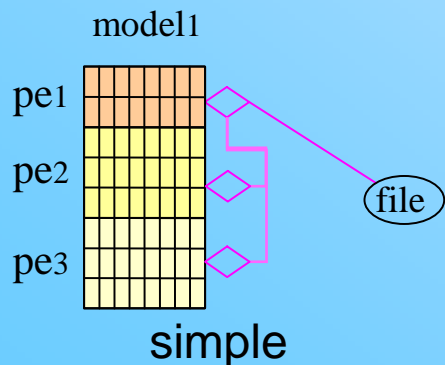


↪ Grilles et partitionnement différents :

↪ Interpolation parallèle



↪ Ecritures dans des fichiers :



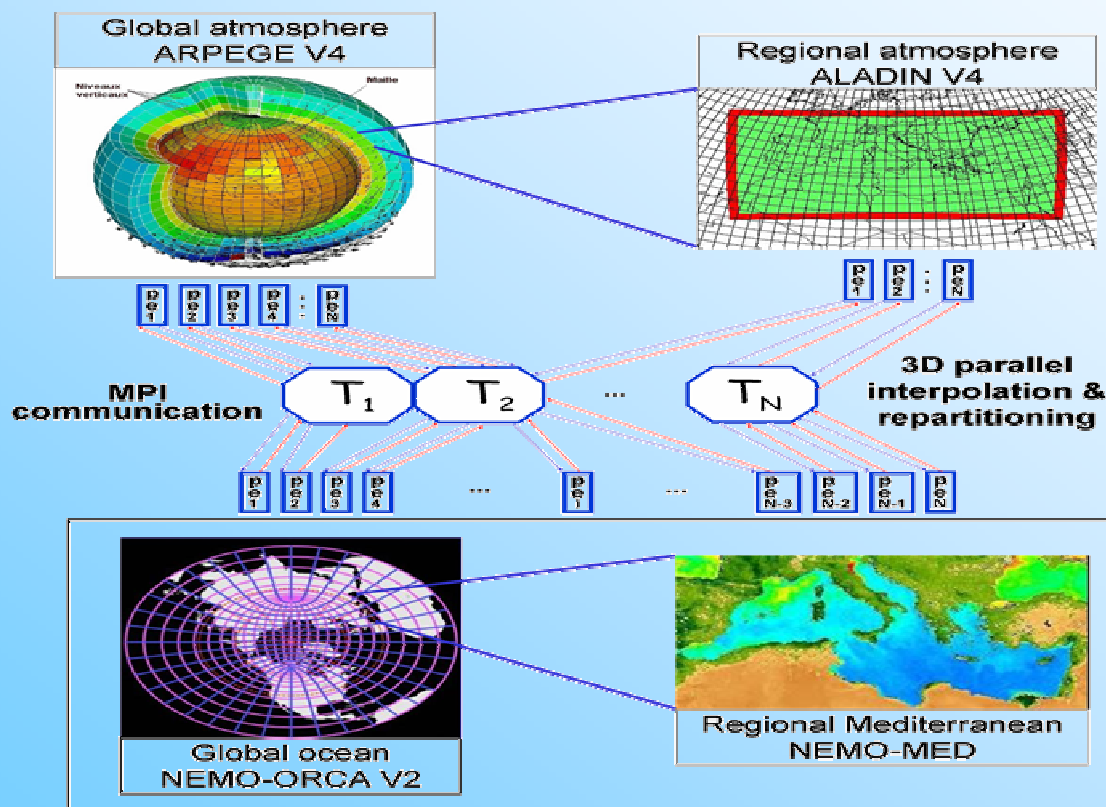
- Différentes interpolations :
 - ↔ En 2D : comme OASIS 3
 - ↔ En 3D : plus proche voisin, trilinéaire

- Transformations locales :
 - ↔ Addition et multiplication par un scalaire
 - ↔ Moyennes et accumulations (ex : échange d'un champs de pluie)

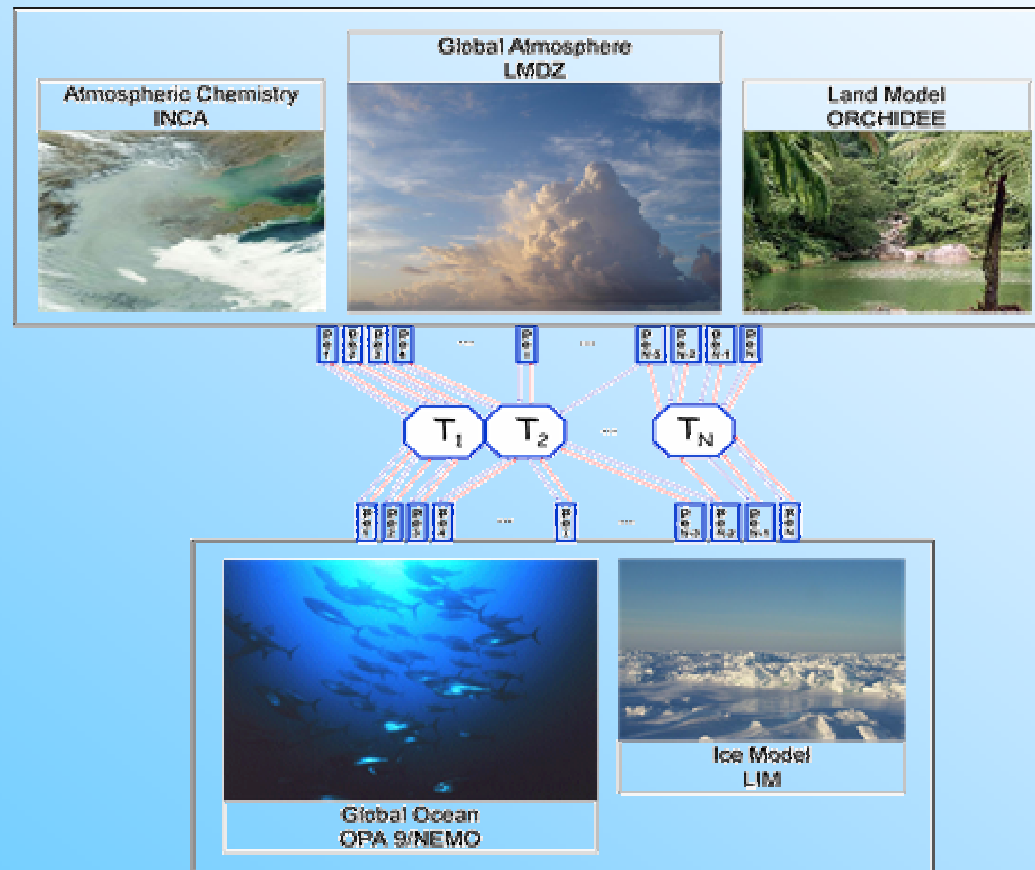
- Sur différentes grilles :
 - ↔ Régulières et irrégulières lat-lon
 - ↔ Etirées, déformées
 - ↔ Gaussiennes réduites

- Utilisation de grilles non géographiques :
 - ↔ Pas d'interpolation possible
 - ↔ Echange de données qui ne sont pas définies sur une grille géographique

- ➔ Couplage de modèles à METEO FRANCE dans le projet CICLE:
 - ↪ Actuellement en développement avec OASIS 3



- Couplage de modèles à l'IPSL dans le projet CICLE:
 - ↔ Fonctionne actuellement avec OASIS 3
 - ↔ En cours d'implémentation avec OASIS 4





Un coupleur dynamique de codes parallèles

E-Mail : thevenin@cerfacs.fr
morel@cerfacs.fr

URL : <http://www.cerfacs.fr/~palm>

-
- ① Le couplage de codes
 - ② OASIS : un coupleur spécialisé pour l'étude de systèmes climatiques
 - ③ **PALM : un coupleur dynamique de codes parallèles**
 - Concepts fondamentaux de PALM
 - Comment "palmer" un code
 - L'interface graphique PrePALM
 - Les outils
 - Exécuter une application avec PALM
 - Exemples d'utilisations
 - ④ Eléments de comparaison entre OASIS et PALM

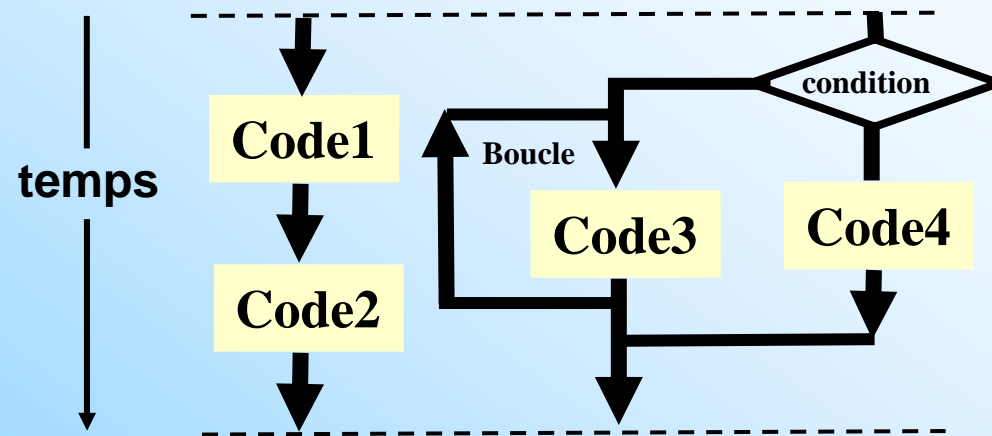
- Couplage dynamique :
 - ↳ Gestion dynamique des ressources (processeurs et mémoire)

- Interface graphique PrePALM :
 - ↳ La mise en place des applications se fait au travers de l'IHM PrePALM

- Boîtes d'algèbres :
 - ↳ Boîte à outil interfaçant les opérations d'algèbre linéaire

- Outils de supervision
 - ↳ Monitoring de l'application en cours d'exécution
 - ↳ Analyse des performances

→ Couplage dynamique :



- ① Un composant peut être lancé au cours de la simulation et rendre ses ressources lorsqu'il a terminé
 - ⇒ Les ressources informatiques (mémoire et nombre de processeurs) sont gérées par le coupleur
- ② Lancement parallèle ou séquentiel
 - ⇒ Parallélisme de tâche : via l'interface graphique
 - ⇒ Parallélisme des composants
- ③ Lancement en boucle est sous conditions
 - ⇒ Description d'algorithmes complexes

→ Pré requis :

- ↪ Disposer des codes sources (une réflexion est en cours afin de se débarrasser de cette contrainte)
- ↪ Le code doit être écrit dans un langage compilé (FORTRAN, C ou C++)

→ Modifications à apporter aux codes :

- ↪ Pour un code en FORTRAN : remplacer "PROGRAM" par "SUBROUTINE"
- ↪ Pour un code en C ou C++ : remplacer "MAIN" par un nom de fonction
- ↪ Si le code est parallèle :
 - ↪ Retirer les appels à MPI_INIT() et MPI_FINALIZE()
 - ↪ Remplacer MPI_COMM_WORLD par PL_COMM_EXEC
 - ↪ Remplacer les appels à STOP, EXIT par PALM_ABORT
- ↪ Créer une carte d'identité
- ↪ Instrumenter le code (utilisation des fonctions de la bibliothèque PALM)

→ Créer une carte d'identité :

!PALM_UNIT ¥			}	unité
!	-name	<i>nom de l'unité</i> ¥		
!	-functions	{ <i>fonction à appeler</i> }		
!	-comment	{ <i>un commentaire optionnel</i> }		
!				
!PALM_SPACE ¥			}	espace
!	-name	<i>nom de l'espace</i> ¥		
!	-shape	(<i>.,., .,., .,</i>) ¥		
!	-element_size	<i>PL_[INTEGER, REAL, DOUBLE_PRECISION, ...]</i> ¥		
!	-comment	{ <i>un commentaire optionnel</i> }		
!				
!PALM_OBJECT ¥			}	objet
!	-name	<i>nom de l'objet</i> ¥		
!	-space	<i>nom de son espace</i> ¥		
!	-intent	[<i>IN, OUT, INOUT</i>] ¥		
!	-time	[<i>ON, NO</i>] ¥		
!	-tag	[<i>ON, NO</i>] ¥		
!	-comment	{ <i>un commentaire optionnel</i> }		
!				
!PALM_DISTRIBUTOR			}	distributeur
!	-name	<i>nom du distributeur</i> ¥		
!	-type	[<i>regular, custom</i>] ¥		
!	-nbproc	<i>nombre de processus de la distribution</i> ¥		
!	-function	<i>nom de la fonction de distribution</i> ¥		
!	-comment	{ <i>un commentaire optionnel</i> }		

- Pour construire un couplage, PALM dispose d'une interface graphique

- L'IHM PrePALM permet de décrire des algorithmes complexes :
 - ↪ Matérialiser les composants
 - ↪ Gérer le parallélisme
 - ↪ Décrire les boucles et les conditions
 - ↪ Décrire les communications

- PrePALM dispose d'outils d'optimisations :
 - ↪ Contrôles de cohérence des données échangées
 - ↪ Mise en relief de l'algorithme de couplage
 - ↪ Fonctionnalités d'analyse de l'application et de monitoring

Prepalm - /home/thevenin/presentation_PALM_OASIS_Brest_Mai_2008/ex_palm/exemple

File Settings Constants Step-actions Date conversion Analyse run Utilities Help

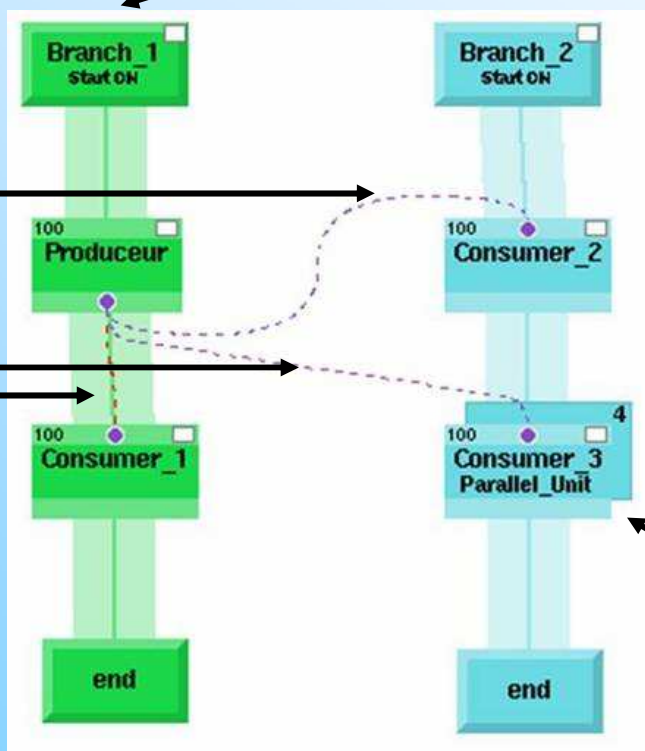
View: Communications Branches

- Branches
- Communications
- Units
- Objects
- Sub-object descriptor
- Steps
- Spaces
- Texts
- Distributors
- Localisations

Insert Delete Filter

no	source	target	
1	producteur_1	vecteur_print_1	vecteu
2	producteur_1	vecteur_print_2	vecteu
3	producteur_1	vecteur_print_3	vecteu

Branches :
séquence d'unités et
d'instructions :



Unité :
↳ Code de calcul
↳ Unité prédéfinie

Unité parallèle

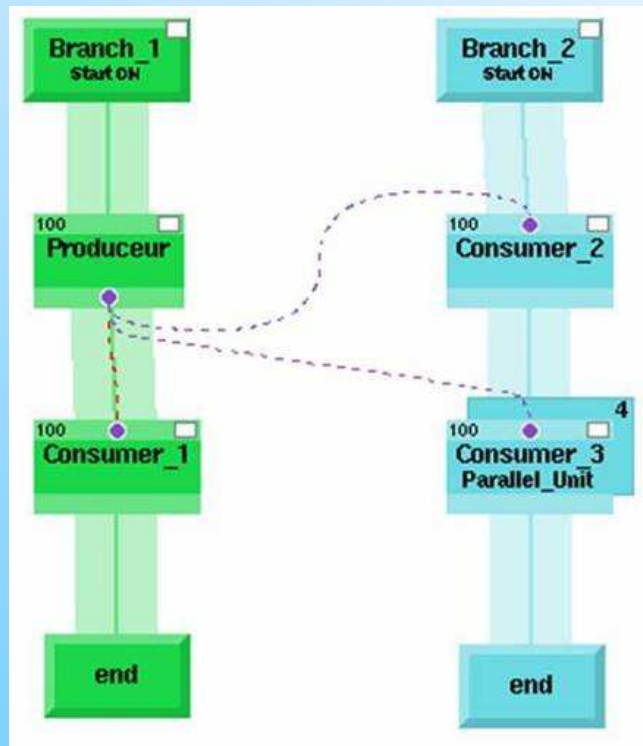
Communications :
elles sont décrites en
reliant les plots des unités

→ Double niveaux de parallélisme pour **exploiter au maximum** le parallélisme intrinsèque d'une application

① **Parallélisme de tâches : branches**

Calcul parallèle en "dessinant"

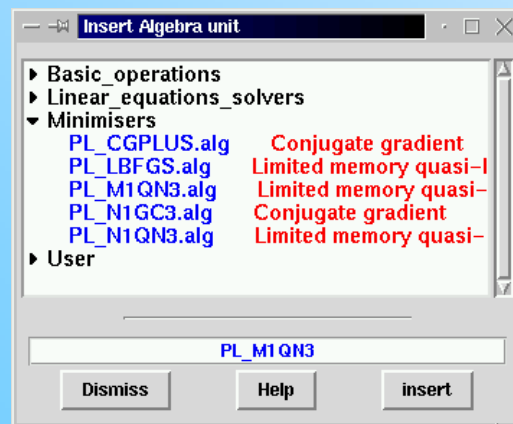
↑
TEMPS
↓



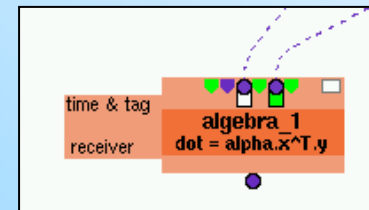
② **Composants distribués : unités parallèles**

- Définition : Unités pré-définies qui interfacent des bibliothèques mathématiques (BLAS, LAPACK, ...) et qui permettent d'effectuer des opérations algébriques sur les objets échangés entre les unités.

Insertion de l'unité d'algèbre dans l'application PALM

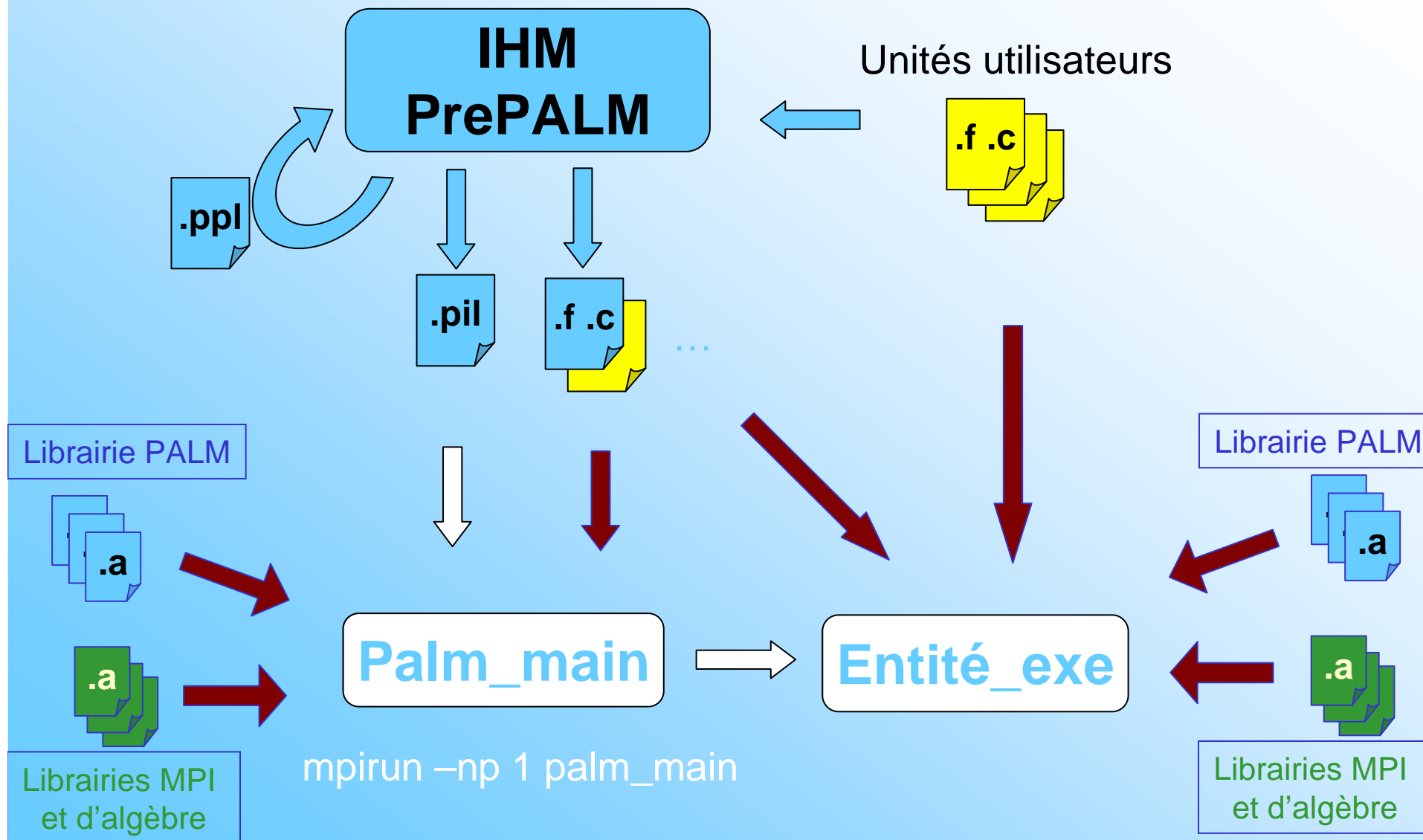


Le lancement des unités d'algèbre est identique à celui des unités utilisateurs

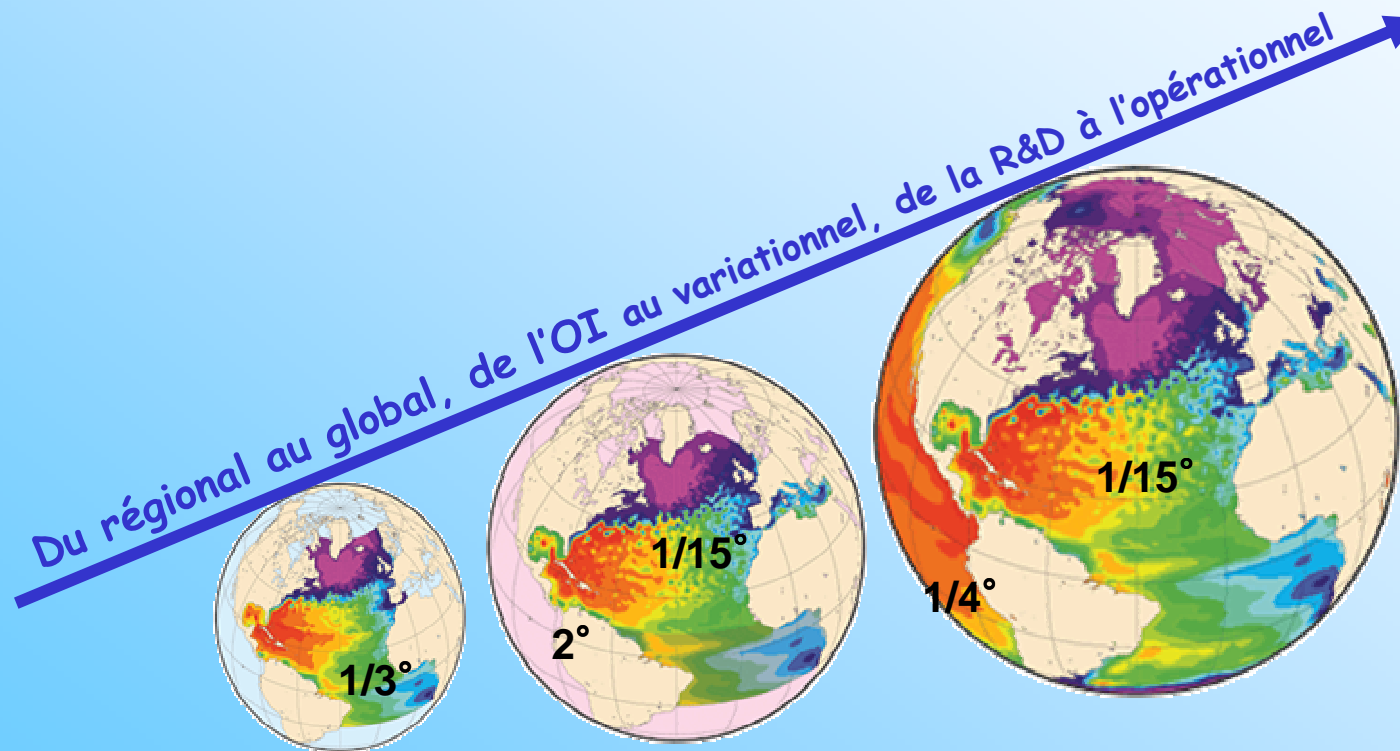


L'utilisateur peut composer ses propres unités d'algèbre.

- ➔ Un fichier de sortie PALM par branche de l'application avec niveaux de détails fixés séparément selon les catégories de messages
- ➔ Une fonction de debug modifiable par l'utilisateur appelée lors des PALM_Put et PALM_Get pour vérifier le contenu des objets
- ➔ Fonctionnalité de monitoring en cours d'exécution
- ➔ Un analyseur de performance sur le temps "elapse" ou "cpu"
- ➔ Rejouer graphiquement l'application PALM

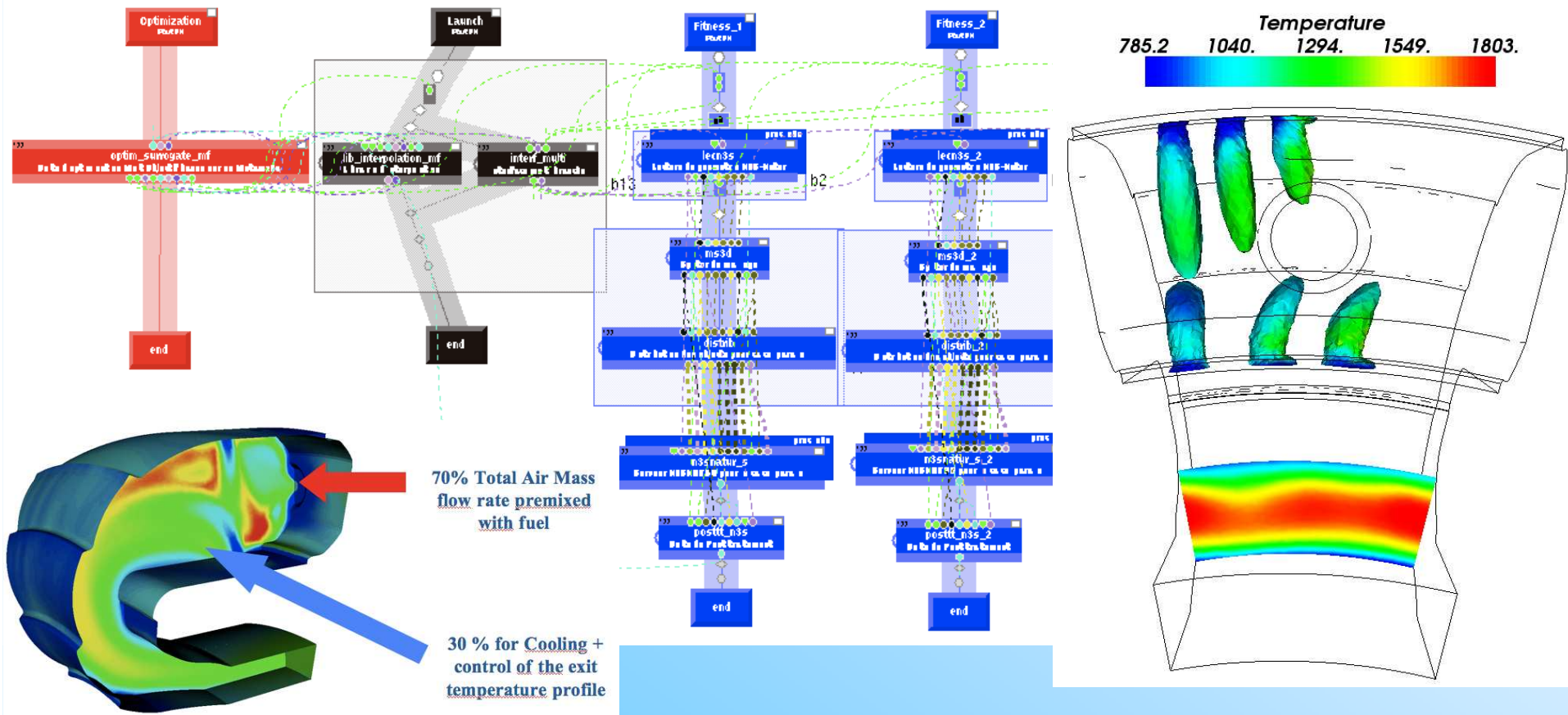


- ➔ PALM chez MERCATOR (opérateur en prévisions océaniques) :
 - ↪ Utilisateur historique du coupleur PALM
 - ↪ Utilisation en R&D et en opérationnel pour toutes ses chaînes d'assimilation de données
 - ↪ Les mêmes méthodes s'appliquent aux différentes résolutions



➔ PALM dans le projet MIPTO (Management of an Integrated Platform for auTomatic Optimization)

➔ Optimisation de forme des chambres de combustion aéronautiques



	OASIS3	OASIS4	PALM
Développement	Stable	Beta version ; en évolution	Stable ; en évolution
Type de couplage	Statique		Dynamique
Adaptation des modèles	Appel à des fonctions de la bibliothèque OASIS		Appel à des fonctions de la bibliothèque PALM
Configuration	Flexible ; dans un fichier texte	Flexible; dans un fichier XML ; validation automatique	Flexible ; via PrePALM
Communications	Partiellement parallèles	Complètement Parallèles	Complètement Parallèles
Interpolations	2D et non parallèles	3D et parallèles	2D et non parallèles
Domaines d'applications	Géophysique		Couplages génériques + assimilation de données
Interface graphique	Non		Oui
Support utilisateur	Oui		Oui
Formation	Possible		Oui

*Journée de rencontre des utilisateurs du pôle
de calcul intensif pour la Mer.*

IFREMER – 15 mai 2008

OASIS

&

Palm

E-Mail : valcke@cerfacs.fr
coquart@cerfacs.fr
URL : <https://oasistrac.cerfacs.fr/>

E-Mail : thevenin@cerfacs.fr
morel@cerfacs.fr
URL : <http://www.cerfacs.fr/~palm>

↳ Prochaine formation PALM :
25, 26 et 27 Juin 2008

