



**HAL**  
open science

## HPCML - Un langage dédié au calcul scientifique

Marc Palyart

► **To cite this version:**

Marc Palyart. HPCML - Un langage dédié au calcul scientifique. Journées sur l'Ingénierie Dirigée par les Modèles(IDM'11), Jun 2011, Lille, France. pp.1. hal-00759801

**HAL Id: hal-00759801**

**<https://hal.science/hal-00759801>**

Submitted on 2 Dec 2012

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

---

# HPCML - Un langage dédié au calcul scientifique

**Marc Palyart<sup>\*,\*\*</sup>**

*\* CEA / CESTA  
33114 Le Barp  
marc.palyart@cea.fr*

*\*\* IRIT - Université de Toulouse  
118, rte de Narbonne  
31062 Toulouse*

---

*RÉSUMÉ. Avec l'augmentation soutenue du nombre de paradigmes de programmation et d'architectures matérielles fondamentalement différentes, il devient de plus en plus difficile d'exploiter de manière performante et durable la puissance de calcul offerte par les nouvelles générations de supercalculateurs. Cette article présente HPCML un DSML (Domain Specific Modeling Language) au cœur de l'approche MDE4HPC qui, afin de dompter cette complexité, vise à appliquer les principes de l'Ingénierie Dirigée par les Modèles (IDM) au développement d'applications de simulation numérique.*

*ABSTRACT. With the increasing number of programming paradigms and hardware architectures, high performance computing is becoming more and more complex in exploiting efficiently and sustainably supercomputers resources. Our thesis, through the MDE4HPC approach is that Model Driven Engineering (MDE) can help us taming the complexity of numerical simulation software development. This paper presents HPCML, a DSML (Domain Specific Modeling Language) at the center of this approach.*

*MOTS-CLÉS : calcul haute performance, IDM, simulation numérique, calcul scientifique, supercalculateur, Eclipse, MDE4HPC, HPCML, ArchiMDE.*

*KEYWORDS: High Performance Computing, MDE, numerical simulation, scientific computing, supercomputer, Eclipse, MDE4HPC, HPCML, ArchiMDE.*

---

## 1. Introduction

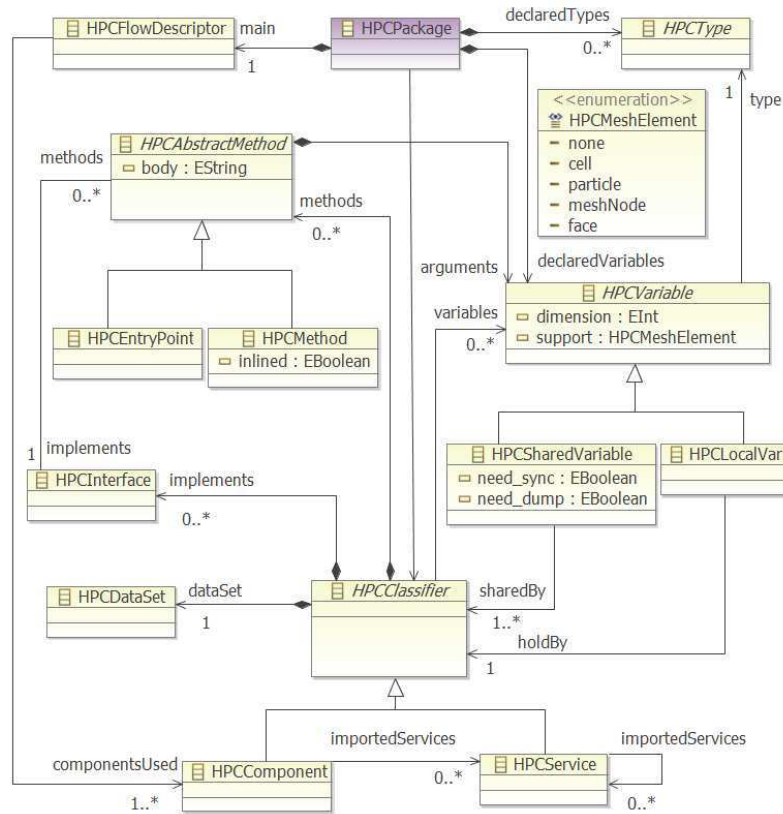
En 1965, Gordon Moore fit l'une des prédictions les plus visionnaires de toute l'histoire de l'informatique (Moore, 1965) lorsqu'il énonça que la performance des ordinateurs doublerait tous les dix-huit mois. De nos jours, cette remarquable prédiction reste toujours aussi pertinente. Cependant, alors que cette amélioration des performances a longtemps permis de conserver un modèle de programmation séquentiel, ces dernières années ont vu apparaître des architectures parallèles au sein même des microprocesseurs (multicoeurs). Ce changement de conception radical au niveau du matériel, oblige à revoir les méthodes de développement logiciel afin de tirer pleinement parti de la puissance de ces nouveaux processeurs.

Un des principaux objectifs de la communauté du calcul haute performance est de produire des applications de simulation numérique efficaces. En raison des importants besoins en puissance de calcul que requiert une simulation de phénomènes physiques, les développeurs ont du depuis longtemps abandonner le paradigme de programmation séquentiel afin d'exploiter les architectures parallèles des supercalculateurs. Malheureusement les pratiques actuelles de programmation parallèle et spécialement celles utilisées dans le monde HPC (High Performance Computing) sont de bas niveau et donc dépendantes de l'architecture. Bien qu'un bon niveau de performance puisse être atteint avec la plupart de ces approches, un certain nombre d'inconvénients apparaissent en termes de dépendances fortes vis-à-vis de l'architecture (portages complexes), de mélange des préoccupations et de complexité de programmation.

Face à ce constat, nous pensons que l'application des techniques issues de l'ingénierie dirigée par les modèles permettrait de faire évoluer cette situation. Notre travail (Palyart *et al.*, 2011) s'inscrit dans ce cadre, en proposant la définition d'un modèle indépendant de la plateforme cible qui est raffiné par des transformations injectant progressivement les détails de la plateforme. Compte tenu de la taille l'article nous nous contentons ici d'aborder la description du modèle indépendant et des choix retenus concernant l'expression du parallélisme.

## 2. HPCML

HPCML (High Performance Computing Modeling Language) est un langage dédié développé au sein du CEA/CESTA pour la modélisation des codes scientifiques de simulation numérique. La Figure 1 présente de façon simplifiée les concepts disponibles au sein d'HPCML pour modéliser le PIM (Platform Independent Model). Certains de ces concepts sont dérivés de technologies très présentes dans la communauté du calcul scientifique, tel que le langage Fortran. En effet, avec l'introduction d'HPCML, nous souhaitons élever le niveau d'abstraction dans la conception sans changer radicalement les habitudes de développement.

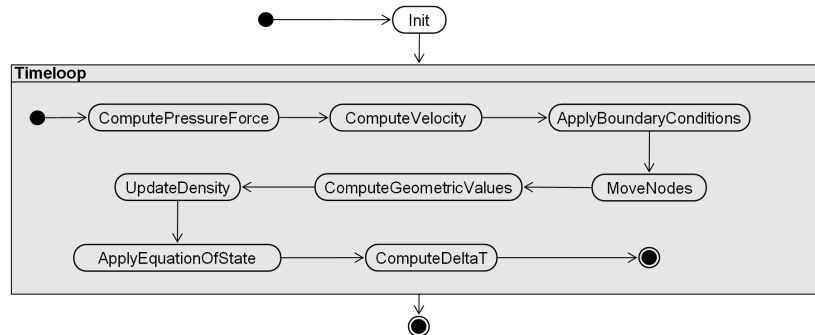


**Figure 1.** Version simplifiée du métamodèle d’HPCML

La brique logicielle structurelle de base d’HPCML est l’*HPCClassifier*. Cette unité logicielle permet de décrire un ensemble de méthodes effectuant des calculs sur des *HPCVariable*.

En règle générale, lors d’une simulation numérique, on souhaite simuler l’évolution d’un ou plusieurs phénomènes physiques dans le temps. On exécute jusqu’à la satisfaction de la condition d’arrêt (durée d’évolution, état physique atteint, ...) une boucle dont chaque itération fait avancer la simulation dans le temps, d’où le nom parfois donné à cette dernière de boucle en temps. Un *HPCFlowDescriptor* permet grâce à un formalisme proche des diagrammes d’activités UML de décrire l’ordre d’exécution des méthodes qui doivent être enchaînées lors d’une itération de la boucle en temps. La Figure 2 montre comme exemple, l’*HPCFlowDescriptor* d’un code simplifié d’hydrodynamique lagrangienne.

Les variables partagées (*HPCSharedVariable*) sont aussi un élément important de modélisation car elles permettent d’exprimer le parallélisme. Définir une variable en



**Figure 2.** Exemple d'*HPCFlowDescriptor*

tant que partagée, indique que cette dernière devra être synchronisée en cas de décomposition du domaine. En effet, dans les modèles de haut niveau, nous avons opté pour l'expression d'un parallélisme de données à travers la décomposition du domaine de calcul tel que présenté dans (Gropp, 1992). L'*HPCFlowDescriptor* donne cependant la possibilité d'exprimer un parallélisme de tâches. Finalement une variable peut être associée à un élément du maillage (noeud, face, maille, particule) afin d'indiquer sa position dans l'espace de calcul.

### 3. Perspectives

Dans cet article, nous avons surtout abordé la syntaxe abstraite d'HPCML. Les possibilités concernant sa syntaxe concrète sont toujours en cours d'évaluation. D'ailleurs plusieurs syntaxes concrètes sont envisagées afin de s'adapter au mieux aux différents métiers (physicien, numéricien, architecte logiciel, architecte matériel) de la simulation numérique en proposant différents points de vue. Afin d'évaluer les différentes syntaxes concrètes d'HPCML, nous avons développé un outil nommé ArchiMDE basé sur l'*Eclipse Modeling Project* (Gronback, 2009).

Des résultats encourageants concernant l'approche MDE4HPC laissent entrevoir un grand potentiel dans l'application de l'IDM pour le développement d'applications de simulation numérique.

Nous venons d'évoquer une application de l'IDM aux problématiques de la communauté du calcul scientifique. A l'avenir, on peut espérer que la communauté des modèles puisse également s'enrichir des spécificités du HPC.

#### 4. Bibliographie

- Gronback R., *Eclipse Modeling Project : A Domain-Specific Language (DSL) Toolkit*, Addison-Wesley Professional, 2009.
- Gropp W. D., « Parallel Computing and Domain Decomposition », in , T. F. Chan, , D. E. Keyes, , G. A. Meurant, , J. S. Scroggs, , R. G. Voigt (eds), *Fifth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, SIAM, Philadelphia, PA, USA, 1992.
- Moore G. E., « Cramming More Components onto Integrated Circuits », *Electronics*, vol. 38, n° 8, p. 114-117, April, 1965.
- Palyart M., Lugato D., Ober I., Bruel J.-M., « MDE4HPC : An approach for using Model-Driven Engineering in High-Performance Computing », *15th System Design Languages Forum (SDL 2011)*, 2011.