

# Les représentations UML

## du logiciel portageMM2CPP

Les représentations UML de ce fichier « UMLmaj\_portageMM2CPP.ppt »  
(et « .pdf ») sont à lire en complément (parallèle)  
des représentations UML initiales qui se trouvent dans les fichiers :  
« UML1portageMM2CPP.pdf », ..., « UML7portageMM2CPP.pdf ».

N.Rousse - modelia -

### Les représentations UML du logiciel portageMM2CPP :

#### Pour la version initiale du logiciel (version 060203) :

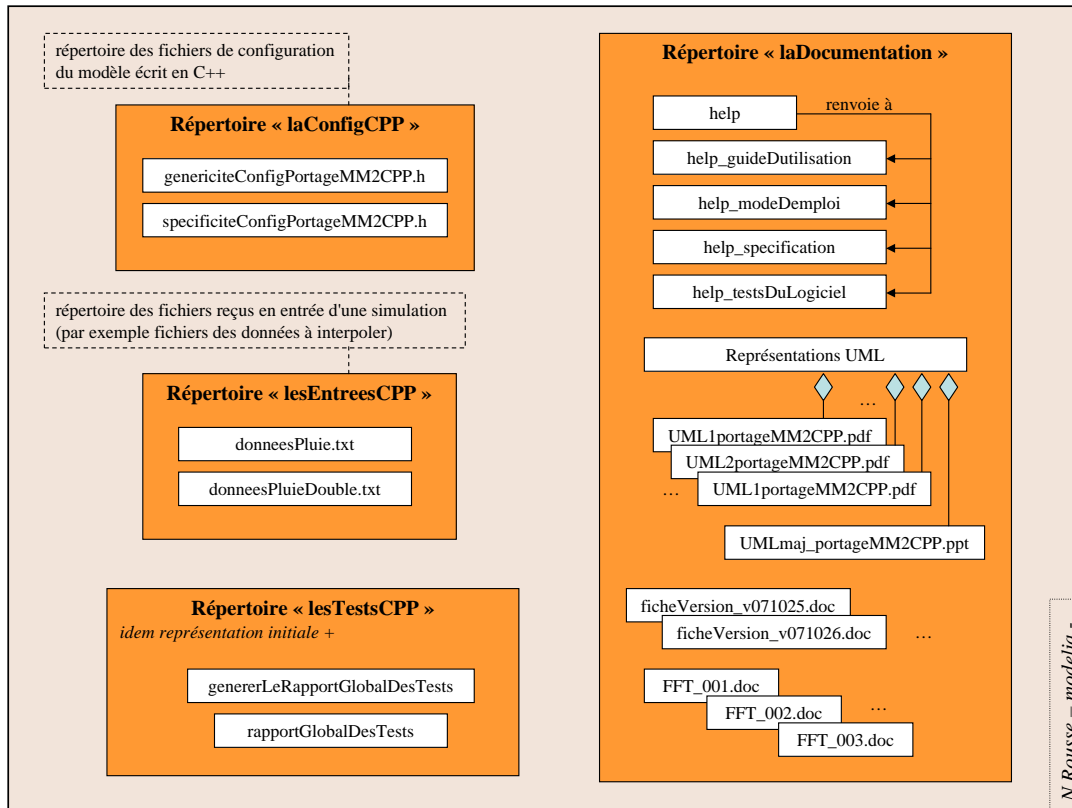
Les représentations UML se trouvent dans les fichiers « UML1portageMM2CPP.pdf »,  
« UML2portageMM2CPP.pdf », ..., « UML7portageMM2CPP.pdf », qui sont issus du fichier  
« UMLportageMM2CPP.zuml ».

Ces représentations UML (« UMLportageMM2CPP.zuml » et .pdf associés) avaient été produites avec le  
logiciel « Poseidon For UML » (Community Edition 4.x) de Gentleware (<http://www.gentleware.com>). Comme  
celui-ci n'existe plus autrement qu'en licence payante, « UMLportageMM2CPP.zuml » n'est plus maintenu.

#### Pour la version 080917 du logiciel :

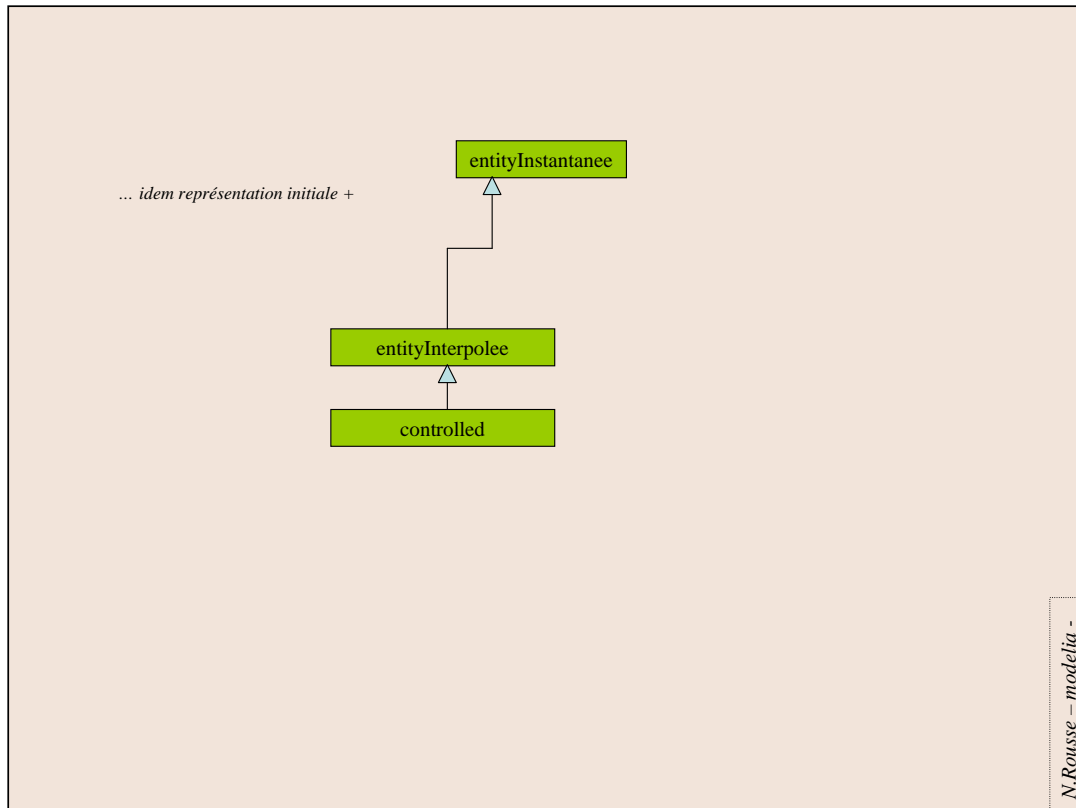
C'est ce fichier « UMLmaj\_portageMM2CPP.ppt » au format Microsoft Powerpoint (ou encore le fichier  
« UMLmaj\_portageMM2CPP.pdf » qui en est issu) qui contient les évolutions des représentations UML.

Les représentations UML du fichier « UMLmaj\_portageMM2CPP.ppt » (ou encore du fichier  
« UMLmaj\_portageMM2CPP.pdf ») sont à lire en complément (parallèle) des représentations UML initiales :  
« UML1portageMM2CPP.pdf », ..., « UML7portageMM2CPP.pdf ».



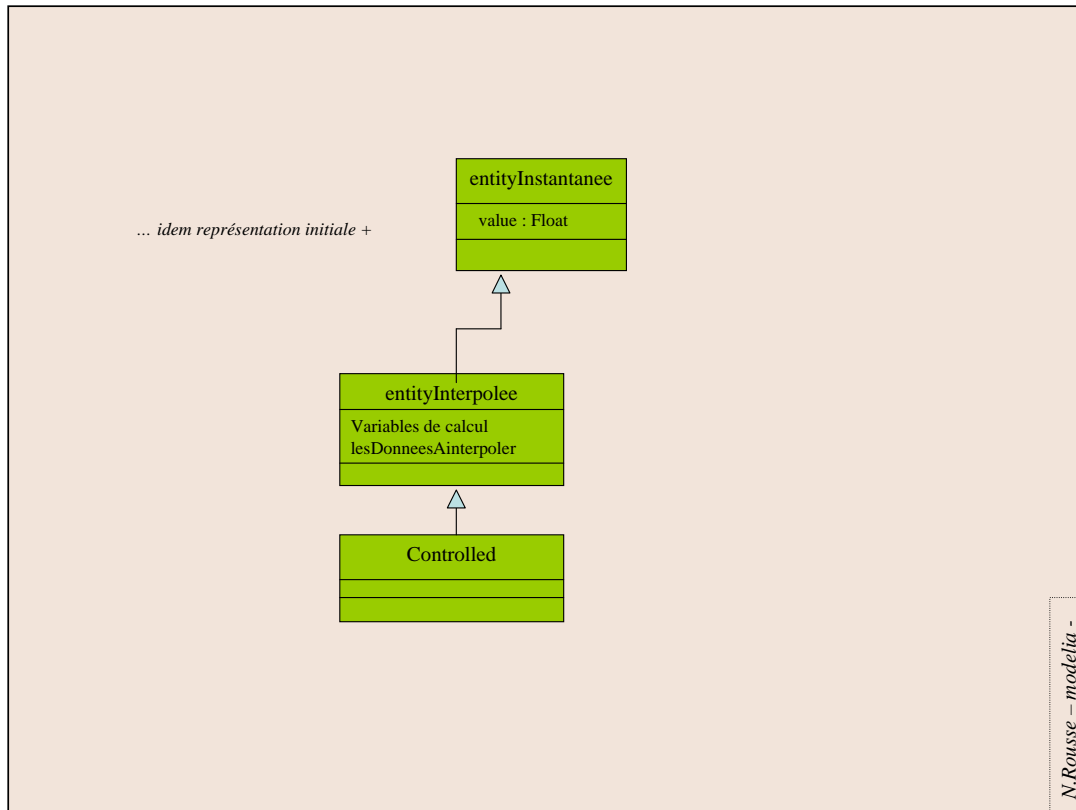
### Diagramme de déploiement portageMM2CPP

Cette représentation est à lire en complément de la représentation UML initiale « UML1portageMM2CPP.pdf ».



### Diagramme des classes du code C++

Cette représentation est à lire en complément de la représentation UML initiale  
« UML2portageMM2CPP.pdf ».



### Diagramme des classes génériques

Cette représentation est à lire en complément de la représentation UML initiale  
 « UML3portageMM2CPP.pdf ».

Page blanche

N.Rousse - modelia -

### Diagramme des classes spécifiques

**Ceci est à lire en complément de la représentation UML initiale « UML4portageMM2CPP.pdf » :**

La représentation « UML4portageMM2CPP.pdf » n'est pas mise à jour ici. Elle correspond au modèle traité alors qu'il provient – ni plus ni moins - du modèle modele2.mod écrit sous ModelMaker. Or le modèle traité n'est plus seulement issu de modele2.txt, il contient également des entités nouvelles, comme les entités « controlled » I1\_pluie, I20..., les entités « flow » Fpluie1, Fpluie20..., les « compartiment » C\_constante, C\_monte, C\_sinusoidale... (voir modifications m\_linearInterpolation et m\_integrale). Ces nouvelles entités ne sont pas illustrées, bien qu'à la lecture de la représentation « UML4portageMM2CPP.pdf », elles soient à prendre en considération à chaque fois que le sont les entités issues de modele2.txt.

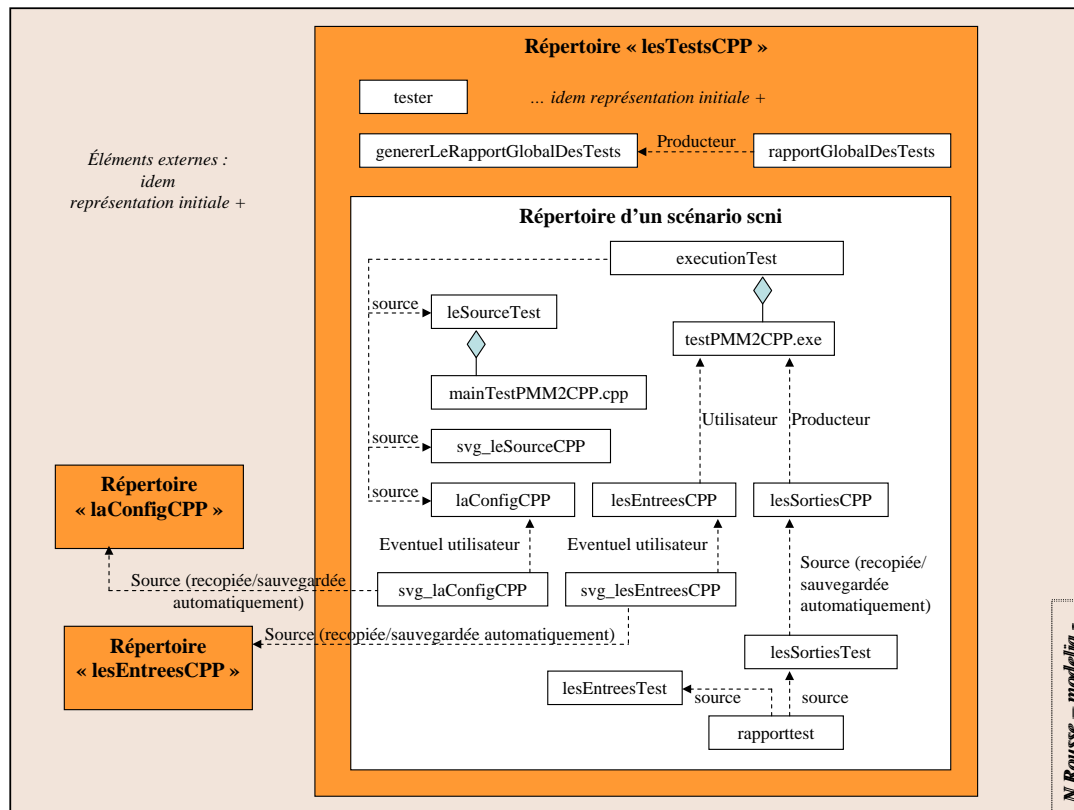
Page blanche

N.Rousse - modelia -

---

### Diagramme d'activité des méthodes

**Pas de complément pour la représentation UML initiale « UML5 portageMM2CPP.pdf » : inchangé.**



### Diagramme de déploiement Répertoire « lesTestCPP »

Cette illustration est à lire en complément de la représentation UML initiale « UML6 portageMM2CPP.pdf » et :

Les répertoires précédemment nommés « lesEntrees » et « lesSorties » (dans « UML6 portageMM2CPP.pdf ») sont renommés « lesEntreesTest » et « lesSortiesTest ».

Le répertoire « lesSortiesCPP » (dans « UML6 portageMM2CPP.pdf ») précédemment externe au répertoire d'un scénario scni se trouve maintenant à l'intérieur.

Des répertoires sont ajoutés au répertoire d'un scénario scni : `laConfigCPP`, `svg_lesEntreesCPP`...

Page blanche

N. Rousse - modelia -

### **Séquence de construction et déroulement d'un scénario de test**

**Ceci est à lire en complément de la représentation UML initiale « UML7 portageMM2CPP.pdf » :**

Pour les scénarios dont les vérifications s'appuient sur des résultats obtenus sous ModelMaker (ce qui n'est pas le cas de tous) : La représentation « UML7portageMM2CPP.pdf » est toujours valable, à condition d'y remplacer « lesEntrees » et « lesSorties » par respectivement « lesEntreesTest » et « lesSortiesTest ».

Pour des scénarios dont les tests ne s'appuient pas sur des résultats obtenus sous ModelMaker : on pourrait décrire une « séquence de construction et déroulement d'un scénario de test » à peu près similaire à la représentation « UML7portageMM2CPP.pdf », sauf que les entrées de tests (« lesEntreesTest ») qui y sont produites au moyen de l'environnement ModelMaker le seraient d'une autre manière (pré-calcul de résultats attendus...). Le rapport des tests reste produit par comparaison des résultats de « lesEntreesTest » et « lesSortiesTest ».