

<i>Identification</i>	
Identification version	FV_080917
Identification version précédente	FV_071026
Date	17/09/08
Auteur (rédacteur de la FV)	Nathalie Rousse
<i>Description des modifications introduites</i>	
Enrichissement m_linearInterpolation (prise en compte de la FFT_003), Modification approuvée	<p>m_linearInterpolation : modification majeure, m_linearInterpolation est la prise en compte de la FFT_003 (FFT_003 : Ajout de la prise en compte des entités ModelMaker qui sont calculées par interpolation linéaire à partir de données lues dans un fichier de données).</p> <p>Description de la modification effectuée :</p> <p>Les points majeurs des ajouts sont :</p> <ul style="list-style-type: none"> - ajout des classes entityInterpolee et controlled et de leur traitement (+ classe listeEntitesInterpolees...) - ajout fichierDentree et donneesAinterpoler qui servent à gérer le fichier contenant les données à interpoler. - ajout d'entités "controlled" dans le système particulier. - et aussi ajout de classe gestionAvancementSimulation. - et aussi mise à jour de la documentation en conséquence, découpage du fichier help (qui grossit) en plusieurs fichiers. <p>Pour plus de précisions : voir commentaires du fichier code source genericitePortageMM2CPP.h ; voir fichier help de documentation ("SPECIFICATION").</p> <p>Tests : les tests écrits pour vérifier la modification m_linearInterpolation sont les scénarios scnInterpolationX (X=1,2,3...), voir lesTestsCPP/scnInterpolationX/rapportTests.</p> <p>La modification est approuvée dans le sens où elle a été testée/vérifiée.</p>
Ajout m_integrale, Modification approuvée	<p>m_integrale : modification mineure.</p> <p>Description de la modification effectuée : des compartiments ont été ajoutés au système/modèle (compartiment C_constante, C_monte, C_descend, C_sinusoidale) pour mettre en œuvre/tester le calcul d'intégrale. Ceci a permis de repérer une correction (marquée « m_integrale ») nécessaire au niveau de l'initialisation de l'horloge, cette correction a été effectuée.</p> <p>Tests : les tests écrits en lien avec m_integrale sont les scénarios scnIntegrale et scnPrecisionCalculs, voir lesTestsCPP/scnIntegrale/rapportTests et lesTestsCPP/scnPrecisionCalculs/rapportTests.</p> <p>La modification est approuvée dans le sens où elle a été testée/vérifiée.</p>
Modification m_repertoires, Modification approuvée.	<p>m_repertoires : modification mineure de réorganisation des répertoires pour une organisation mieux adaptée et plus souple.</p> <p>Description de la modification effectuée :</p> <ul style="list-style-type: none"> - ajout du répertoire laConfigCPP : du code qui était jusque là dans leSourceCPP a été déplacé dans laConfigCPP. Le fait d'isoler ainsi la partie configuration permet lors des

	<p>tests d'adapter la configuration du logiciel au contexte du test sans toucher à leSourceCPP (le code testé).</p> <ul style="list-style-type: none"> - ajout de lesEntreesCPP : dédié aux données/fichiers d'entrée de la simulation (les fichiers des données à interpoler...). - réorganisation du répertoire lesTests pour que chaque scénario fonctionne de manière complètement autonome/locale. Nouvelle organisation : Le code source du test (répertoire leSourceTest) est compilé avec le code source du logiciel testé (copié/sauvegardé dans svg_leSourceCPP) et la configuration adaptée au test (répertoire laConfigCPP). Si le test ne requiert pas une configuration spécifique, alors laConfigCPP pointe simplement svg_laConfigCPP qui est une copie/sauvegarde de la configuration du logiciel testé. Le scénario est exécuté dans le répertoire executionTest. L'exécution produit des sorties dans le répertoire lesSortiesCPP, à partir d'entrées adaptées au test (répertoire lesEntreesCPP). Si le test ne requiert pas d'entrées spécifiques, alors lesEntreesCPP pointe simplement svg_lesEntreesCPP qui est une copie/sauvegarde des entrées du logiciel testé. Les vérifications reposent sur les informations des répertoires d'entrée/sortie propres au test : le répertoire lesSortiesTest (où sont récupérées les données de lesSortiesCPP) et le répertoire lesEntreesTest (contenant fichiers de résultats attendus...). Pour plus de précisions : voir fichier help de documentation ("TESTS DU LOGICIEL"). <p>La modification est approuvée dans le sens où elle a été vérifiée.</p>
Ajouts m_scenarios, Modification approuvée.	M_scenarios : de nouveaux tests ont été écrits : scnCasDerreur1,2,3, scnInitialisation1,2,3,5, scnIntegrale, scnInterpolation1,2,3,4,5,6,7, scnPrecisionCalculs (en plus de scn1nominal, scnTestV071025, scnTestV071026).
Description/identification des problèmes et limitations connues	
	FFT_004 (demande d'évolution mineure non réalisée).
Description autre (compléments)	
	Cette version a été développée sous nrousse@badet : ~/myworkspace/WI_portageMM2CPP.
Inventaire du produit logiciel (en précisant les versions et éditions)	
	<p>Le logiciel dans sa version 080917 est rassemblé dans son répertoire d'archivage : nrousse@zongo : ~/archiveEI_portageMM2CPP/080917_portageMM2CPP.</p> <p>En particulier, liste des composants logiciels nécessaires à la génération de cette version du logiciel :</p> <ul style="list-style-type: none"> - code source : *.cpp et *.h de logiciel_portageMM2CPP/leSourceCPP ; code de configuration : *.h de logiciel_portageMM2CPP/laConfigCPP. - outil de génération dans logiciel_portageMM2CPP/executionCPP.
	<p>Liste des documents associés à version 080917 du logiciel :</p> <p>~/archiveEI_portageMM2CPP/080917_portageMM2CPP/logiciel_portageMM2CPP/laDocumentation.</p>
Les instructions de production et d'installation du logiciel	
	<p>Procédure de génération du logiciel : dans ~/archiveEI_portageMM2CPP/080917_portageMM2CPP/logiciel_portageMM2CPP voir laDocumentation/help pour les explications d'utilisation de executionCPP/compiler.</p>