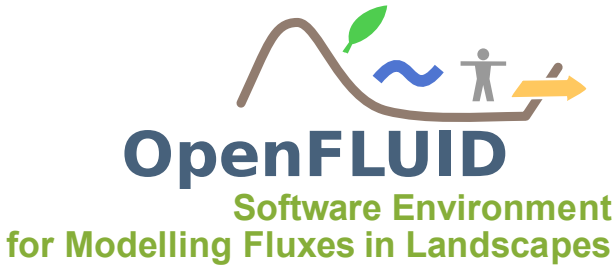


TP1 : Création d'un simulateur

Objectifs:	Créer un simulateur et sa signature, construire et compiler pour une utilisation avec le moteur
Pré-requis:	TP0

Note: L'environnement de développement préconisé est Eclipse, accompagné de l'extension CDT. L'ensemble est disponible sur http://www.eclipse.org/downloads/ .
En complément, un plugin OpenFLUID pour Eclipse a été développé pour accompagner la création d'un simulateur et générer un squelette du code source.
Les développements s'appuient également sur la suite de compilation/construction/test/packaging CMake. CMake est disponible sur http://www.cmake.org .

1 Créer un simulateur

Il existe 2 façons de créer le code source d'un simulateur :

- Ecrire le code source "à la main", à partir d'un code vierge : long, fastidieux, source d'erreurs, ...
- utiliser le plug-in OpenFLUID pour Eclipse : facile, assisté, intégré à l'environnement de développement, ...

Nous allons utiliser cette 2ème possibilité pour cet exercice.

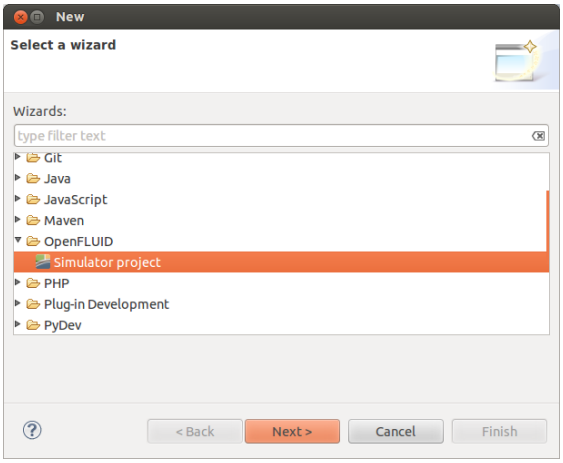
1.1 Lancement d'Eclipse

Pour lancer Eclipse, cliquez sur l'icône `Eclipse` présente sur le Bureau. Au lancement d'Eclipse, il peut vous être demandé de choisir le chemin du workspace que vous souhaitez utiliser. Il est conseillé d'utiliser celui proposé par défaut (sous Linux : `/home/openfluid/Bureau/formation/workspace`). Le workspace est le répertoire qui contiendra l'ensemble des projets de simulateurs développés dans cet exercice ainsi que les suivants.

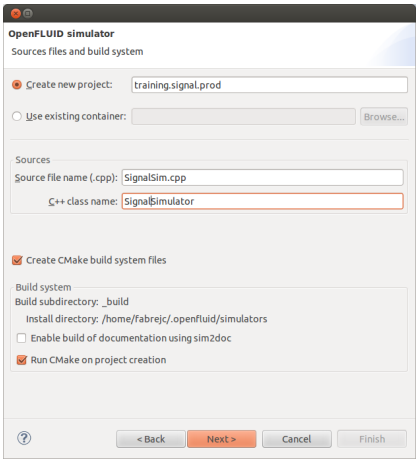
This document is licensed under Creative Commons license

1.2 Création du squelette du simulateur

Nous allons utiliser le plug-in OpenFLUID pour Eclipse afin de générer le code source "vide" d'un simulateur. Aller dans le menu d'Eclipse *File > New > Other...*. Dans la fenêtre qui s'ouvre, choisir *OpenFLUID > Simulator project* et cliquer sur *Next*.

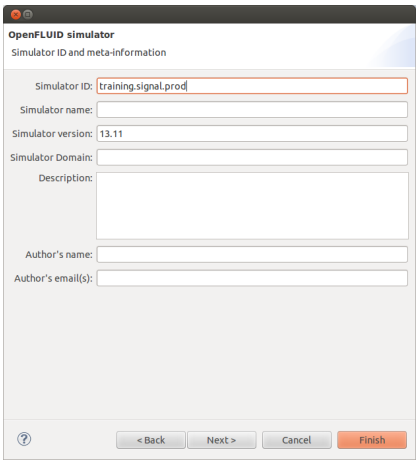


La fenêtre suivante est la 1ère étape de la création du simulateur. Cette première étape consiste à définir les caractéristiques du code source du simulateur.
Cliquer sur *Create new project* et indiquer le nom `training.signal.prod`. Choisir `SignalSim.cpp` comme nom de fichier (*Source file name*), `SignalSimulator` comme nom de classe C++ qui contiendra le simulateur (*c++ class name*). Laisser coché *Create Cmake build system files* ainsi que *Run Cmake on project creation* Cliquer sur *Next* une fois cette étape complétée.

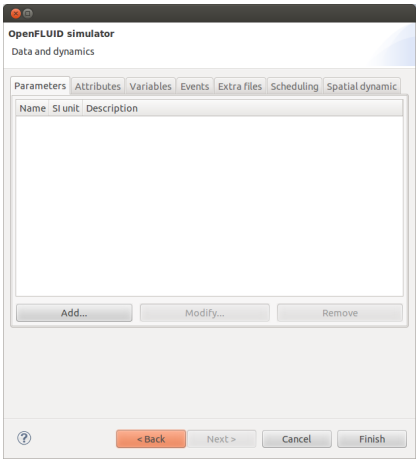


La deuxième étape consiste à renseigner les informations sur le simulateur, et notamment son identifiant (*Simulator ID*) par lequel il va être reconnu. Taper `training.signal.prod` comme identifiant de simulateur. Les autres champs sont facultatifs, vous pouvez y renseigner un nom plus détaillé du simulateur, votre nom et email. Cliquer sur *Next* une fois cette étape complétée.

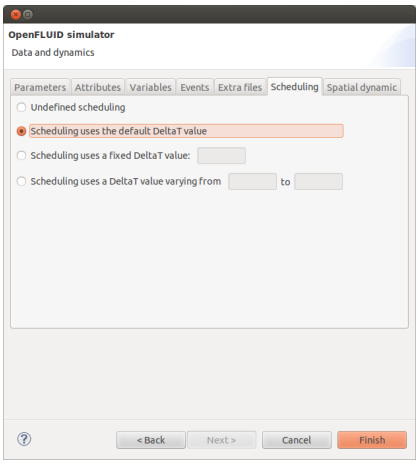
This document is licensed under Creative Commons license



La troisième et dernière étape consiste à renseigner les informations sur les variables/-paramètres/attributs/évènements qui sont produits/utilisés/mis à jour par le simulateur.



Aller dans l'onglet *Scheduling* et cocher l'option *Scheduling uses the default DeltaT value*.



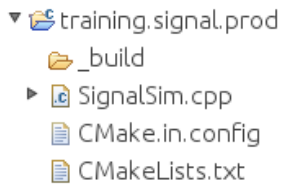
Enfin cliquer sur *Finish*.

This document is licensed under Creative Commons license

Si tout s'est bien passé, vous devriez obtenir un dossier `training.signal.prod` contenant 3 fichiers dans le *Project Explorer* d'Eclipse :

- `SignalSim.cpp` : fichier source du simulateur,
- `CMakeLists.txt` : fichier du système de construction du simulateur (à priori, à ne pas modifier) ,
- `CMake.in.config` : fichier de configuration de la construction du simulateur.

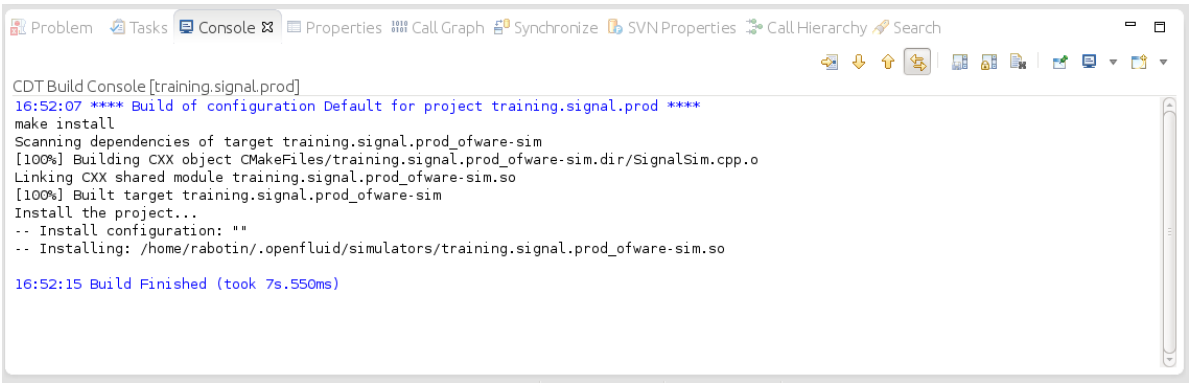
ainsi qu'un sous-répertoire `_build` qui contiendra les résultats de compilation/construction du simulateur.



2 Construire et installer le simulateur

La construction/installation du simulateur peut se faire de deux manières :

- Soit en utilisant la ligne de commande en exécutant la commande `make install` depuis le sous-répertoire `_build`
- Soit depuis Eclipse, dans le menu *Project > Build Project* ou cliquer sur l'icône `Marteau` de la barre de menu Outils. La méthode depuis Eclipse est à privilégier.



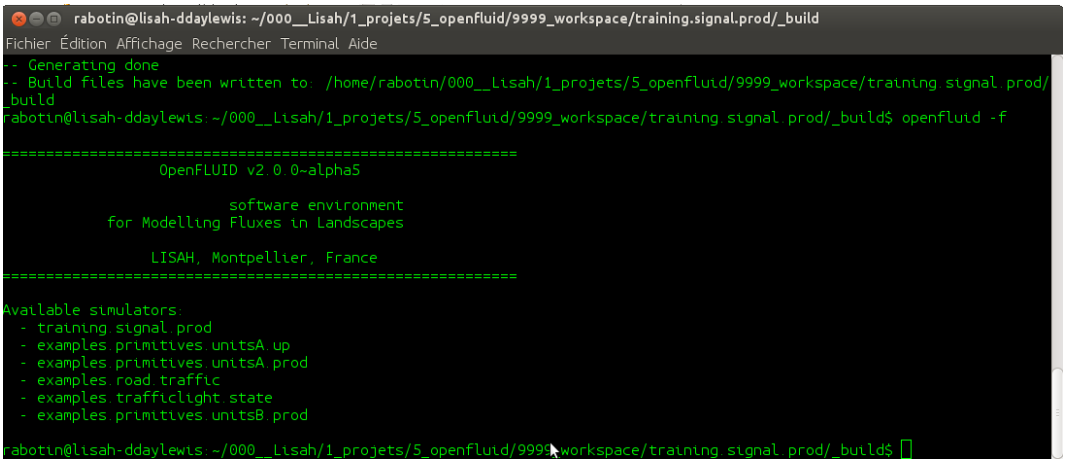
Note: Cette construction/installation est à exécuter à chaque fois que le code source du simulateur est modifié.

Pour vérifier que le simulateur a été correctement construit et installé, exécuter la commande suivante dans un terminal :

```
openfluid -f
```

Le simulateur devrait alors apparaître dans la liste.

This document is licensed under Creative Commons license



This document is licensed under Creative Commons license