

TP0 : Découverte et prise en main de la plateforme OpenFLUID

Objectifs:	Prendre en main la documentation utilisateur et technique, connaître les outils.
	Savoir utiliser la plateforme OpenFLUID
Pré-requis:	Aucun

1 Documentation

1.1 Site web du projet

Le site web OpenFLUID rassemble les informations générales sur le projet : objectifs, vue d'ensemble, applications, téléchargements, ... C'est le point d'entrée pour découvrir le projet. Le site web OpenFLUID est accessible à l'adresse http://www.openfluid-project.org.

1.2 Espace web Community

L'espace web OpenFLUID Community rassemble des contenus pour la communauté des acteurs du projet : les développeurs, les modélisateurs, les utilisateurs. Le tout est accessible dans un espace communautaire, qui comprend notamment :

- La documentation d'installation
- La manuel de référence pour les utilisateurs
- Le guide de développement des simulateurs
- La documentation de l'API de développement des simulateurs
- Le guide de bonnes pratiques
- Des exemples de code source
- Les informations de migration de version

- ...

L'espace OpenFLUID Community est accessible sur http://www.openfluid-project.org/community/

2 Les outils

2.1 Outils de développement

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 information d'installation du plugin OpenFLUID pour Eclipse sont disponibles sur http://www.openfluid-project.org/openfluid/community/index.php/Setting_up_the_development_environment

Les développements s'appuient également sur la suite de compilation/construction/test/packaging CMake.

CMake est disponible sur http://www.cmake.org .

2.2 Outils communautaires

Un canal IRC pour OpenFLUID est disponible sur freenode.net. Un canal IRC est un espace de discussion à plusieurs, où chacun peut poser des questions, proposer un sujet de discussion, ... et où chacun peut participer. C'est le moyen privilégié pour obtenir de l'aide sur OpenFLUID. Les modalités pour rejoindre le canal IRC sont disponibles sur

http://www.openfluid-project.org/openfluid/community/index.php/Community_support .

La mailing-list openfluid est également disponible pour des échanges et annonces autour du projet.

Les modalités pour s'abonner à la liste sont disponibles sur http://www.openfluid-project.org/openfluid/community/index.php/Community_support .

OpenFLUID propose également un outil de rapport de bug et de demande de nouvelle fonctionnalité, via son hébergement sur GitHub.

Les modalités pour signaler un bug ou déposer une demande de nouvelle fonctionnalité sont disponibles sur

http://www.openfluid-project.org/openfluid/community/index.php/Community_support.

3 Pour ce TP ...

3.1 Généralités

Les travaux pratiques qui vont suivre durant cette session de formation seront réalisés au travers d'une machine virtuelle Linux Ubuntu 12.04 32bits sous VirtualBox. Le nom d'utilisateur présent sur cette machine virtuelle est *openfluid*. La connexion avec ce nom d'utilisateur est automatique au lancement de la machine virtuelle.

Le répertoire personnel de l'utilisateur *openfluid* est /home/openfluid. Dans ce répertoire personnel, se trouve un répertoire formation dans lequel seront déposées toutes les données et codes-sources nécessaires ou créés pendant cette formation. Il est organisé comme suit :

- sous-répertoire datasets : jeux de données d'entrée

- sous-répertoire outputs : sorties des exécutions en ligne de commande avec openfluid
- sous-répertoire projects : projets OpenFLUID-Builder
- sous-répertoire src : codes sources fournis
- sous-répertoire workspace : code-source des simulateurs développés
- sous-répertoire datasets-solutions : solutions des projets réalisés pendant les TPs

3.2 Exemple d'application

Le jeu de données utilisé au cours de ce TP est dénommé "Manhattan". Il est constitué d'un extrait du tissu routier de la presqu'île de Manhattan, NY, USA.

Il comporte 39 unités spatiales réparties comme suit :

- 33 unités spatiales de classe RU (Road Unit) représentant des tronçons de rue,
- 6 unités spatiales de classe TLU (Traffic Light Unit) représentant des feux de circulation.

Les unités spatiales peuvent être visualisées sous GoogleEarth en ouvrant le fichier Manhattan.kml situé dans le répertoire /home/openfluid/Bureau/formation/datasets/pw0/

Elles sont également disponibles au format shapefiles dans le dossier

/home/openfluid/Bureau/formation/datasets/pw0/shapefiles/

- Manhattan_RU_wgs84.shp : fichier représentant les tronçons de routes,
- Manhattan TLU wgs84.shp : fichier représentant les feux de circulations.

Le format shapefile peut être visualisé à l'aide d'un logiciel SIG (Système d'Information Géographique) comme QGIS; il permet la visualisation des formes géométriques des entités ainsi que de leurs tables attributaires.



La simulation va modéliser la circulation routière pour la journée du 31 mai 2013 de 9h à 17h.



Les connectivités entre unités spatiales sont définies comme suit :



4 Objectifs et démarche du TP

4.1 Objectif

L'objectif de ce TP est de découvrir la plateforme OpenFLUID et les possibilités qu'elle offre dans le cadre de modélisations spatialisées. Un modèle simplifié de circulation de voitures servira d'exemple de modélisation.

4.2 Démarche

A partir d'un projet de modélisation déjà existant et de l'interface graphique OpenFLUID-Builder, les différentes fonctionnalités de la plateforme seront abordées dans un premier temps.

Puis dans un deuxième temps, le projet sera modifié en testant de nouvelles valeurs de paramètres et en ajoutant des simulateurs supplémentaires (afin de prendre en compte de nouveaux processus à simuler).

Note: Dans un premier temps de ce TP, seules les Road Units (RU) seront prises en compte dans le modèle proposé; la prise en compte des Traffic Light Units (TLU) n'interviendra que dans la deuxième partie du TP.

5 Prise en main de la plateforme OpenFLUID

5.1 Découverte de l'interface graphique OpenFLUID-Builder

OpenFLUID-Builder est l'interface graphique de la plateforme OpenFLUID ; cette interface a été développée afin de manipuler de manière aisée les projets de modélisation : elle permet ainsi de modifier des valeurs de paramètres, ajouter ou enlever des simulateurs, lancer une simulation...

Note: Pour informations, la plateforme OpenFLUID peut également être utilisée directement en ligne de commande, ceci permet par exemple d'utiliser la plateforme Open-FLUID sur des clusters de calcul. Cet aspect de la plateforme ne sera pas abordé pendant cette session de travaux pratiques.

Pour lancer l'application **OpenFLUID-Builder**, cliquer sur l'icône OpenFLUID-Builder sur le bureau.



La fenêtre qui s'ouvre permet d'accéder à des projets déjà existants et d'en créer de nouveaux. Le projet utilisé durant ce TP existe déjà et s'appelle projet_Manhattan.

Cliquer sur le bouton Open a project et sélectionner le projet projet_Manhattan, puis cliquer sur Open.

🖋 🔺 🙍 rabotin	OpenFLUID-Projects					
laces	Name	✓ Modified				
🕽 Search	🚘 kamech	16/07/2012	Project details			
Recently Used	kamech2	16/07/2012	i rojece decino			
rabotin	🚞 Manhattan	24/04/2013				
Desktop	📄 Manhattan2	24/04/2013	Name:			
File System	🚞 Manhattan3	24/04/2013	projet_Manhattan			
BackupMount	📄 old_tp_mhydas	27/06/2012	Description			
Lecteur de dis	📄 projet_exemple	26/09/2012	(none)			
/data	📄 projet_Manhattan	Yesterday at 16:18				
Documents	📄 projet_mhydas	24/09/2012	Authors:			
Musique	i recherche_bug	17/05/2013	(1010)			
Images	📔 rieutord	25/02/2013	Creation date:			
Vidéos	📔 Roujan_domain	07/06/2012	2013-Jun-02 16:06:34			
resultvirtual1	📔 roujan_trough	13/07/2012	Modification date:			
datastore	🚞 sat_bauzille	16/05/2013	2013-Jun-02 17:28:40			
dddddore	🚞 test1-7	08/03/2012				
	🚞 test2	04/12/2012				
	🚞 test2bis	09/01/2013				
	📄 test2ter	10/01/2013				
	🚞 testlandr	03/07/2012				
	iestmick	18/04/2012				
	📄 toto	08/03/2012				
	i toto2	03/04/2013				
	📔 toto3	03/04/2013				
	iii toto4	23/04/2013				
	📔 tp	21/05/2013				

La fenêtre principale de OpenFLUID-Builder s'ouvre et se compose de plusieurs parties :

😣 🗐 🗊 OpenFLUID-B	uilder [proje	et_Manhat	tan]		
Project Edit Simulatio	n Help				
New Open	Save Sa	ive as	X Close	😻 Map View	R un
Project Explorer					
Model					
▼ Spatial domain					
RU (33 units)					
Datastore					
Monitoring					
Run configuration					
Output browser					
Project Dashboard					
 Model definition 	ok				
 Model parameters 	ok				
 Required files 	ok				
 Spatial representation 	ok				
 Inputdata 	ok				
 Project consistency 	ok				
 Datastore 	ok				
 Monitoring 	ok				
 Run configuration 	ok				
Current project path: /h	ome/rabotin/	/OpenFLUI	D-Projects	/projet Manhat	tan
				//··//	

- la barre d'icônes : elle permet la gestion d'un projet (créer un nouveau projet, sauvegarde du projet, ouverture/fermeture...). Elle permet également l'accès à MapView (fenêtre de visualisation des données cartographiques) et de lancer la simulation (bouton Run),
- la fenêtre Project Explorer : elle permet d'accéder au modèle du projet (simulateurs et leurs paramètres), au domaine spatial et à la gestion des paramètres de la simulation,
- la fenêtre Project Dashboard permet de visualiser la cohérence du projet : un système de feux vert, orange et rouge permet de connaître l'état du projet,
- la fenêtre de visualisation principale qui évolue en fonction des actions.

5.2 Fenêtre Project Explorer

La fenêtre Project Explorer permet de gérer les différents paramètres du modèle et de la simulation. Elle se décompose en six sous-parties : Model, Spatial domain, Datastore, Monitoring, Run configuration et Output browser.

Dans la fenêtre Project Explorer, cliquer sur Model, un nouvel onglet Model apparaît dans la fenêtre de visualisation principale.

See OpenFLUID-E Project Edit Simulati	Build on	der [projet_Manhattan] Help	
New Open	Sa	ve Save as Close Map View Run	
Project Explorer		Model 🗙	
Model			
 Spatial domain RU (33 units) Datastore 		► Global parameters	
Monitoring		Add simulator	Expand all Collapse all
Run configuration Output browser		examples.road.traffic	~ _
		Road Unit (RU) function transfert and stockage for cars	\vee
		Parameters and information	
		Information Parameters Variables Inputdata	
		ID: examples.road.traffic	
		MuttCapacity Name: Road Unit (RU) function transfert and stockage for cars	
Deale at Dealth and		Path: /home/rabotin/000_Lisah/1_projets/S_openfluid/9998_s build/dist/lib/openfluid/simulators	jit/rabotin/openfluid/
Project Dashboard		Description:	
 Model definition 	0	Status: Experimental	
 Model parameters 	e	Domain: examples	
		Process	
 Required files 	0	Author(s): Michael Rabotin <rabotin@supagro.inra.fr></rabotin@supagro.inra.fr>	
 Spatial representation 	0		
 Inputdata 	0		
 Project consistency 	0		
 Datastore 	0		
 Monitoring 	0		
 Run configuration 	0		
(*()	Þ		
Current project path: /h	ome	e/rabotin/OpenFLUID-Projects/projet_Manhattan	
	_		

Cet onglet permet la gestion du modèle couplé composé de 1 à plusieurs simulateurs. Il permet d'ajouter de nouveaux simulateurs, d'en retirer et de modifier leur ordre dans le modèle. Il permet également d'accéder aux informations de chaque simulateur (auteurs, description du rôle du simulateur, documentation sur les paramètres et les variables). La partie Global parameters permet d'accéder et modifier les paramètres globaux du modèle qui s'appliquent à tous les simulateurs.

Dans cette simulation, le modèle est composé d'un seul simulateur : le simulateur examples.road.traffic qui va réaliser le transfert de stocks de voiture entre chaque RU du domaine.

Dans la fenêtre Project Explorer, cliquer sur Spatial domain, un nouvel onglet Spatial domain apparaît dans la fenêtre de visualisation principale.

Image: Second State Image: Second	uilder [proj on Help	et_Manhattan]			
New Open	Save Sa	ve as Close Map View	Run		
Project Explorer		Model 🗙 Spatial domain 🗙			
Model					
▼ Spatial domain		Unit classes	ID 🔻	Process order	
RU (33 units)		RU	1	11	
Datastore		110	2	10	
Monitoring			3	11	
Run configuration			4	18	<u>I</u>
Output browser			5	20	
			6	2	
			7	4	
			8	19	
Project Dashboard			9	5	
 Model definition 	ok		10	5	
 Model parameters 	ok		11	21	
 Required files 	ok		12	17	
 Spatial representation 	ok		13	12	
 Inputdata 	ok		14	6	
 Project consistency 	ok		15	1	
Datastore	ok		16	6	
 Monitoring 	ok		17	10	
 Run configuration 	OK		18	9	
			19	8	
			20	1	
				10	
Current project path: /h/	ome/rabotin	OpenFLUID-Projects/projet_Manh/	attan		

Cet onglet permet de visualiser les différentes unités spatiales du domaine (identifiants, Process Order).

Le domaine spatial de la simulation se compose pour le moment de 33 RU (Road Unit). Dans un premier temps, seules les RU seront considérées dans la simulation (les TLU seront ajoutés dans la suite du TP).

En double cliquant sur une unité, une fenêtre apparaît permettant de visualiser les liens de cette unité avec les autres unité du domaine spatial (lien de topologie orientée).

Unit classes	ID 🔻 Process order
RU	1 11
	2 10
	😣 Unit edition
	Unit class: RU
	ID: 6
	Process order: 2
	▼ "From" units
	Class ID 🔹 📑
	RU 15
	▼ "To" units
	Class ID 🔻 📇
	RU 24
	RU 26 💻
	"Parent" units
	"Children" units
	Cancel OK
	20 1

Dans la fenêtre Project Explorer, dans Spatial domain, double cliquer sur RU (33 unités), un nouvel onglet Spatial data [RU] apparaît dans la fenêtre de visualisation principale.

😵 🗐 🗊 OpenFLUID-Builder [projet_M	Aanhattan]	
Project Edit Simulation Help		
New Open Save Save a	as Close	Map View Run
Project Explorer Mo	del 🗙 Spatial	domain 🗙 Spatial data [RU] 🗙
Model		
▼ Spatial domain	putdata Events	5
RU (33 units)	D 🔻 capacity	stockini
Datastore 1	5	50
Monitoring 2	5	50 -
▶ Run configuration 3	4	30
Output browser 4	3	30
5	4	50
6	i 1	50
Project Dashboard 7	' 1	50
Model definition ok	1	30
Model parameters ok	2	30
Required files ok	0 6	30
Spatial representation ok	1 5	30
• Inputdata ok	2 7	50
Project consistency ok	4 6	30
Datastore ok 1	5 4	50
Monitoring ok 1	6 3	20
Run configuration ok	7 4	40
1	8 8	20
1	9 2	30
2	7 0	20
Current project path: /home/rabotin/Ope	enFLUID-Project	s/projet_Manhattan

Cet onglet permet de visualiser les données associés à chacune des unités spatiales de type RU ; certaines données peuvent être liées à la géométrie de l'unité spatiale, d'autres aux processus étudiés (capacity= valeur maximum de transfert de voitures, stockini : stock initial de voitures...)

Dans la fenêtre Project Explorer cliquer sur Run configuration, un nouvel onglet Run configuration apparaît dans la fenêtre de visualisation principale.

😣 🗐 🗉 OpenFLUID-B	uilder [proj	jet_Manhattan]
Project Edit Simulatio	on Help	
		🚨 🗴 🚙 🕨
New Open	Save S	ave as Close Map View Run
Project Explorer		Model 🗙 Spatial domain 🗙 Spatial data [RU] 🗙 Run configuration 🗙
Model		
* Spatial domain		Simulation period
Datastore		Period begin Period end
Monitoring		
Run configuration		Date 2013-05-31 09:00:00 Date 2013-05-31 17:00:00
Output browser		Time 9 \$ h 0 \$ m 0 \$ s Time 17 \$ h 0 \$ m 0 \$ s
Project Dashboard		Simulation scheduling
 Model definition 	ok	Delta T 60 \$ seconds
Pequired files	ok	
Spatial representation	ok	Constraint None *
 Inputdata 	ok	Constraint Trote ·
 Project consistency 	ok	
 Datastore 	ok	
 Monitoring 	ok	Memory management
 Run configuration 	ok	Values buffer 1 0 steps
Current project path: /h	ome/rabotir	n/OpenFLUID-Projects/projet_Manhattan

Cet onglet permet de gérer la valeur du pas de temps ainsi que les dates de début et de fin de la simulation. Il permet également une gestion de la mémoire en cas de simulation avec des volumes importants de données.

La simulation est paramétrée selon un pas de temps de 60 secondes avec une date de début à 9h00 le 31/05/2013 pour finir à 17h00 le même jour. Dans la fenêtre Project Explorer, cliquer sur Monitoring, un nouvel onglet Monitoring apparaît dans la fenêtre de visualisation principale.

800 OpenFLUID-E Project Edit Si <u>mulati</u>	Builder [pr on Hel <u>p</u>	ojet_Manhattar	י] 			_				
New Open	📩 Save	Save as Cli	ose Map View	Run						
Project Explorer		Model X S	ipatial domain 🗶 Spa	tial data [RU] 🗙	Run configura		Aonitorina 🗙			
Model					j	•••	5			
▼ Spatial domain		🔶 Add o	bserver					Expand	all Collaps	e all
RU (33 units)										TA.
Datastore		export.va	ars.files.csv						\sim	
Monitoring		Exports si	imulation variables to C	SV files					-	
Run configuration		▼ Parame	aters and information							
Oucput browser			cers one in ormation			Informati	ion			
								s filos seu		
		format.F	ormat1.commentchar	#	-	Name:	Exports sin	nulation variables to CSV files		
		format.F	ormat1.date	%Y %m %d %H %	6M %S 💻	Path:	/usr/lib/op	enfluid/observers		
		set.RU.fo	ormat	Format1	_	Descripti	ion: This obser	ver exports variables to CSV files		
		set.RU.u	nitclass	RU	_		format. <f< td=""><td>ormatname>.date : the date format u</td><td>singthe</td><td></td></f<>	ormatname>.date : the date format u	singthe	
Project Dashboard		cet PLLv:	200	evamples PLLS s	tock		standard C	date format		
 Model definition 	ok			examplesitions			comment l	ormatname>.commentchar : the char lines	acterror	
 Model parameters 	ok						format. <f< td=""><td>ormatname>.header : the header type</td><td>e</td><td></td></f<>	ormatname>.header : the header type	e	
 Required files 	ok						format. <h< td=""><td>ormatname>.precision : the precision</td><td>for real</td><td></td></h<>	ormatname>.precision : the precision	for real	
 Spatial representation 	ok						set. <setn< td=""><td>ame>.unitclass : the unit class of the s</td><td>et</td><td></td></setn<>	ame>.unitclass : the unit class of the s	et	
 Inputdata 	ok						set. <setn< td=""><td>ame>.unitsIDs : the unit IDs included in</td><td>n the set.</td><td></td></setn<>	ame>.unitsIDs : the unit IDs included in	n the set.	
 Project consistency 	ok						set. <setn< td=""><td>ame>.vars : the variable included in th</td><td>e set,</td><td></td></setn<>	ame>.vars : the variable included in th	e set,	
 Datastore 	ok						separated	by semicolons. Use * to include all var	iables	
 Monitoring 	ok						set. <setn< td=""><td>ame>.rormat : the <rormatname> use by a format parameter</rormatname></td><td>d, must</td><td></td></setn<>	ame>.rormat : the <rormatname> use by a format parameter</rormatname>	d, must	
 Run configuration 	ok					Version:	2.0.0~beta	1		
						Status:	Experimen	ital		
		export.va	ars.files.kml-anim						~	
		Exports si	imulation variables to ti	me indexed kml fil	es for animat	ion			~ -	
		▶ Parame	aters and information							
		Falalite	cors and in officiation							U V
Current project path: /b	ome/rabot	in/OpenELLID-E	Projects/projet_Maphal	tan						
con che project path. /h	ionne, rabot		rojecca/projec_Marina	con						đ

Cet onglet permet la gestion des sorties des résultats désirées pour la simulation que ce soit le format de sortie désiré (partie Add observer) ou les unités et les variables désirées (partie Add a parameter).

Dans ce projet, les sorties pour cette simulation vont être le nombre de voitures pour chaque RU (variable examples.RU.S.stock). Les sorties se feront à l'aide de deux observateurs et seront donc accessibles sous deux formats :

- l'observateur export.vars.files.csv exporte les données au format csv (format compatible avec la plupart des tableurs),
- l'observateur export.vars.files.kml-anim exporte les données au format Google Earth (format kml/kmz).
- **Note:** Pour plus d'informations sur les formats de sortie disponible avec OpenFLUID, se reporter au tutoriel sur l'utilisation des observateurs.
- **Note:** L'onglet Output browser ne sera accessible qu'après avoir lancé une première simulation. Il permet un accès facilité aux fichiers de sortie.

5.3 Fenêtre Project Dashboard

La plateforme OpenFLUID réalise automatiquement des contrôles de validité de la simulation, ce qui permet à l'utilisateur de vérifier rapidement si toutes les données et valeurs requises pour une simulation sont présentes.



Des contrôles automatiques sont ainsi effectués en temps réel sur la validité et le bon paramétrage du modèle, la validité du domaine spatial, la configuration des sorties et de l'exécution. Dans le cas où un point est au rouge, la simulation ne peut pas s'effectuer.

5.4 MapView

Cliquer sur l'icône Map View pour faire apparaître l'onglet de visualisation des données spatiales.



Ce visualisateur de type SIG permet une visualisation simple du domaine spatial et des différentes couches SIG des unités le composant.

Note: Le Map View est encore à l'état de développement et peut comporter encore quelques bugs d'affichage.

6 Première simulation

6.1 Pour lancer la simulation

Pour lancer la simulation cliquer sur l'icône Run, une fenêtre de progression de la simulation s'ouvre.

				-						
Project Edit Simulation	Builder [proj on Help									
New Open	Save S	Dave as	X Close	藆 Map View	Run					
Project Explorer		Model 🗙	Spatial do	main 🗙 Sp	atial data [RU] 🗙	Run configura	tion 🗙 M	onitoring 🗙	MapView 🗙	
Model										
Spatial domain		🚽 🖓		ij All 🔻 🔝	• 😑 🚺	l 😳 lools				
RU (33 units)								Ô	Layers	
Monitoring									KU (A 1
Run configuration					/	_ /			5	\sim
Output browser						7			♥ Style	
		😞 Simu	lation						Color:	
									Width:	1 ‡
			Simulatio	n from 2013	05-31 09:00:00	to 2013-05-31	17:00:00		Opacity:	100 ——
									Show	units IDs
Project Dashboard		-								
 Model definition Model parameters 	ok									
Required files	ok	Pre-si	mulation	Initializat	ion <u>si</u>	mulation	Finalizal			
 Spatial representation 	ok									
 Inputdata 	ok	▶ Detail	s							
 Project consistency 	ok									
 Datastore 	ok									
Monitoring Dup configuration	ok			7						
Karrconingaration	UK		Int y							
			0							
								ų		
		(4(
		X: -73.9609	904 Y:40.8	15499 (unkn	own coordinate	system)				
Current project path: /h	ome/rabotin	r/OpenFLUIC	D-Projects/p	projet_Manhi	attan					

6.2 Visualisation des résultats dans un tableur

Pour visualiser les résultats, cliquer dans Project Explorer sur Output browser.

⊗⊜⊜ OpenFLUID-E Project Edit Simulatio	Builder [p r on Help	ojet_Manhattan]	
New Open	📩 Save	Save as Close Map View Run	
Project Explorer		1 Spatial domain ¥ Spatial data [PLI] ¥ Pun configuration ¥ Monitoring ¥ ManView ¥ Output browser	*
Model			^ _
▼ Spatial domain		Output directory: /home/rabotin/OpenFLUID-Projects/projet_Manhattan/OUT	
RU (33 units)			
Datastore		Output directory	
Monitoring		Name	•
Run configuration		examples road traffic of ware-sim log	
Output browser		examples.trafficlight.state_ofware-sim.log	
		export.vars.files.csv ofware-obs.log	
		export.vars.files.kml-anim_ofware-obs.log	
		🗎 kmlanim.kmz	
		🗋 openfluid-messages.log	
		RU_RU10_examples.RU.S.stock.csv	
Project Dashboard		RU_RU11_examples.RU.S.stock.csv	
• Model definition	ok	RU_RU12_examples.RU.S.stock.csv	
Model parameters	ok	RU_RU13_examples.RU.S.stock.csv	
Required files	ok	RU_RU14_examples.RU.S.stock.csv	
 Spatial representation 	ok	RU_RU15_examples.RU.S.stock.csv	
 Inputdata 	ok	🖾 RU_RU16_examples.RU.S.stock.csv	
 Project consistency 	ok	RU_RU17_examples.RU.S.stock.csv	
 Datastore 	ok	RU_RU18_examples.RU.S.stock.csv	
 Monitoring 	ok	RU_RU19_examples.RU.S.stock.csv	
 Run configuration 	ok	RU_RU1_examples.RU.S.stock.csv	
		PUL PUL24_examples PULS stock.csv	
		P PLI PLI22 examples PLIS stock csv	
		PUL PUI22 examples PUIS stock csv	
		RU RU24 examples RUS stock csv	
		RU RU25 examples RU5 stock csv	
		RU RU26 examples.RU.S.stock.csv	
Current project path: /h	ome/rabot	in/OpenFLUID-Projects/projet Manhattan	

Double-cliquer sur le fichier de la RU 11 *RU_RU11_examples.RU.S.stock.csv.* Une fenêtre graphique s'ouvre et propose de choisir le logiciel avec lequel ouvrir ce fichier, sélectionner dans la liste *LibreOffice Calc.*

A l'aide d'un graphique de type ligne, visualiser la dynamique de la variable pendant la simulation.

Faites une copie du graphique et sauvegarder le fichier tableur.

Note: Les fichiers de résultat sont stockés dans le dossier OUT du projet et donc sont disponibles dans le dossier home/openfluid/ OpenFLUID-projects/nom du projet/OUT).

6.3 Visualisation des résultats avec Google Earth

Ouvrir le logiciel Google Earth (icone sur le Bureau). A l'aide de Fichier, Ouvrir, ouvrez le fichier /home/openfluid/OpenFLUID-projects/projet_Manhattan/OUT/kmlanim.kmz. Utiliser le curseur chronologique pour visualiser l'évolution du stock de voitures.



7 Modification de la simulation

L'interface OpenFLUID-Builder permet de modifier rapidement les paramètres d'un projet et de relancer une simulation.

7.1 Modification de la durée de simulation

Modifier la durée de la simulation pour que celle-ci s'arrête désormais à 11h30. Relancer la simulation et analyser de nouveau la sortie RU_RU11_examples.RU.S.stock.csv. Faites une copie d'écran.

7.2 Modification des paramètres

Le simulateur examples.road.traffic possède un paramètre qui permet de modifier la donnée d'entrée capacity qui représente le taux maximum de voitures transférables d'une RU à l'autre; c'est le paramètre MultiCapacity. Dans l'onglet Model, aller dans l'onglet du simulateur examples.road.traffic et modifier la valeur du paramètre MultiCapacity avec la valeur de 2. Relancer une simulation et comparer avec les résultats précédents.

7.3 Enregistrer le projet

Pour enregistrer le projet, cliquer sur l'icône Enregistrer.

8 Amélioration du projet initial

8.1 Objectif

Le modèle actuel ne prend en compte que les processus sur les RU. Le domaine spatial étant également composé de feux de croisement (Traffic Light Unit - TLU), un simulateur pour modéliser le changement d'état des feux va être ajouté.

8.2 Ajout d'un nouveau simulateur

Pour ajouter un nouveau simulateur, aller dans l'onglet Model, cliquer sur l'icône "+ Add simulator" et dans la liste ajouter le simulateur examples.trafficlight.state. Avec les flèches haut et bas, positionner le en première position du modèle.

Project Explorer Model	Model X Run configuration X Output browser X Ma	pView 🗙			
 Spatial domain RU (33 units) TLU (6 units) Datastore 	 Global parameters Add simulator 				Expand all Collapse
Monitoring * Run configuration Output browser	openfluid-builder Available simulators	Domain	Information	Variables Inputdata	<u>-</u>
Project Dashboard Model definition o Model parameters o Spatial representation o Project consistency o Project consistency o Nonitoring o Run configuration o	Simulators examples primitives unitsA prod examples primitives unitsA up examples primitives unitsA up examples primitives unitsB prod examples primitives unitsB prod examples randt ranffe examples traffic light state tests base tests fortran water suf-rug-nunff-infiltration mseytoux water suf-rug-nunff-infiltration mseytoux water suf-transfer-rs.hayami water suf-transfer-rs.hayami vater suf-transfer-rs.hayami Generator liget Cenerator liget Cenerator liget Cenerator Random Cenerator Random Cenerator	examples examples examples examples (unknown) meteo hydrology Hydrology Hydrology	ID: Name: Path: Description: Status: Domain: Process: Method: Author(s):	examples traffic light state //tome/ /tome/ /tome/ /to	
				Cancel OK	

Note: Pour supprimer un simulateur du modèle, il suffit de sélectionner le simulateur et de cliquer sur l'icône "-".

8.3 Ajout d'un observateur

Ajouter un nouvel observateur qui va permettre de visualiser directement l'évolution de la variable examples.RU.S.stock sous forme de graphique au format pdf ou au format *Gnuplot*. Dans l'onglet Monitoring, ajouter un observateur de type export.vars.plot.gnuplot et paramétrer le pour obtenir le les sorties de la variable examples.RU.S.stock sur l'entité RU 11 comme sur la figure suivante :

export.vars.plot.g	nuplot		A
lots simulation var	iables using GNUplot		~
Parameters and	information		
+		Informatio	n
graph.g1.key	default	ID:	export.vars.plot.gnuplot
graph.g1.series	s1	Path:	/home/rabotin/000_Lisah/1_projets/5_openfluid/9998_git/rabotin/openfluid/_build/dist/lib/openfluid/observer
graph.g1.ylabel	values (xx.yy-1)	Description	n: This observer plots simulation variables using GNUplot formalism and tools It uses series of values (simulated or from existing data files) that can be combined into one or many graphs
output	examples.RU11.stock.pdF	-	Plotted variables can only be scalar variables (boolean, integer, double) Parameters for declaring series can be: serie-seriename>var: the name of the variable to plot (e.g. var.foo) serie-seriename>variables (be g. 17) serie-seriename>sourcefile : the file to plot (e.g. measured_data dat), if this serie is not based on simulation variables serie-seriename>sourcefile : the file to plot (e.g. measured_data dat), if this serie is not based on simulation variables serie-seriename>sourcefile : the GNUplot style to use to plot this serie (e.g. linespoint) serie-seriename>sourcefile : the label to use for this serie (e.g. foo) instead of the automatic label Parameters for declaring graphs can be: graph.sgraphname>series : a semicolon seprated list of seriename> (e.g. serie1;serie2) graph.sgraphname>key: the mode for key display graph.sgraphname>key: the mode for key display graph.sgraphname>key: the to the graph Global parameters are: terminal: the terminal mode to replace the default wxt mode (e.g. pdfcairo size 11.7,8.3) output: the output file name when terminal exports graphs to file (e.g. mygraph.grap
persistent	0		
serie.s1.label	RU#11-cars-stock	-	
serie.s1.style	line	-	
serie.s1.unitID	11	-	
serie.s1.unitclass	RU	-	
serie.s1.var	examples.RU.S.stock	-	
terminal	pdfcairo size 11.7,8.3	-	
tryopengnuplot	1	-	
		Version: Status:	2.0.0-beta1 Stable

Lancer la simulation puis aller dans l'onglet Output browser, double- cliquer sur le fichier examples.RU11.stock.pdf pour visualiser le graphique de l'évolution de la variable examples.RU.S.stock pour l'unité RU 11.



Ajouter un observateur de type csv sur les unités TLU afin de connaître leur changement d'état au cours de la simulation.

Note: Pour plus d'informations sur le paramétrage des observateurs, se reporter au tutoriel sur l'utilisation des observateurs.

8.4 Quitter OpenFLUID

Sauvegarder votre projet, pour quitter le projet en cours, cliquer sur l'icône Fermer et valider. Pour fermer la plateforme OpenFLUID, dans la barre de menu, cliquer sur projet/Quitter.