

Présentation générale

Utilisation d'OpenFLUID-Engine

JC. Fabre, X. Louchart, F. Colin, C. Dagès, R. Moussa, M.
Rabotin, D. Raclot, P. Lagacherie, M. Voltz

LISAH - Laboratoire d'étude des Interactions Sol-Agrosystème-Hydrosystème

Plan

- 1 **Présentation générale**
 - Contexte & Historique
 - Approche de modélisation
 - Description de la plate-forme
- 2 **Exemples d'applications**
 - Echelle parcellaire
 - Echelle du bassin versant
- 3 **OpenFLUID-Engine**
 - Fonctionnalités générales
 - Fichiers d'entrée
 - Mise en oeuvre de simulations
 - Buddies
- 4 **Référentiel de fct. simu.**
 - Présentation

Plan

- 1 **Présentation générale**
 - Contexte & Historique
 - Approche de modélisation
 - Description de la plate-forme
- 2 Exemples d'applications
 - Echelle parcellaire
 - Echelle du bassin versant
- 3 OpenFLUID-Engine
 - Fonctionnalités générales
 - Fichiers d'entrée
 - Mise en oeuvre de simulations
 - Buddies
- 4 Référentiel de fct. simu.
 - Présentation

Modélisation intégrative du paysage

- Modélisation et simulation de processus spatialisés, interagissant fortement dans le temps et l'espace
i.e. flux de matière (eau, terre, énergie, ...), activités humaines, ...
- Approche pluri-disciplinaire
- Approche équilibrée entre processus et espace
- Approche complexe

⇒ Nécessité d'outils-support de la modélisation intégrative du paysage

Historique LISAH

Développement du modèle MHYDAS depuis 1995

- concepts de modélisation spatio-temporelle
- représentation des éléments du paysage
- hydrologie de surface et souterraine, polluants, érosion, agronomie, ...

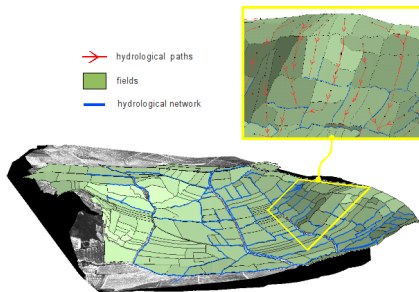
Développement de la plate-forme OpenFLUID depuis 2005-2006, opérationnelle en 2007

- formalisme de couplage dynamique
- support spatio-temporel
- standardisation des développements de modèles
- dynamique, capitalisation et partage autour de la modélisation spatialisée

Représentation Numérique du Paysage (RNP)

Représentation sous la forme d'unités spatiales connectées, issues

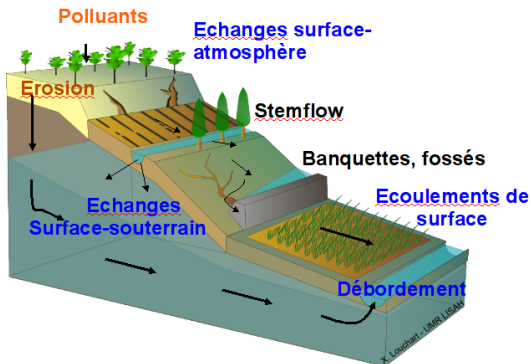
- du croisement de couches d'information spatiales (géométrie, relief, propriétés pertinentes, ...)
- des contraintes liées aux modèles qui vont être appliqués.



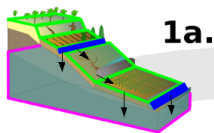
- Représentation finale sous forme d'un graphe orienté
- Classification des unités (surface, fossés, souterrain, ...)

Modélisation de processus spatialisés

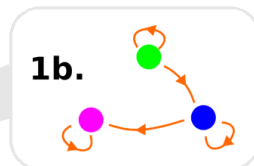
- Les processus sont associés à des unités spatiales
- Ces processus temporels donnent lieu à des modifications de l'état des unités spatiales \Rightarrow variables d'état des unités
- L'ensemble des processus couplés forme un modèle couplé



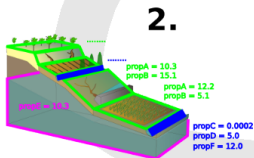
Démarche



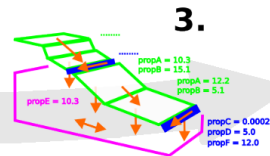
Représentation numérique du paysage sous la forme de graphes connectés, où les **noeuds** sont des éléments du paysage et les arcs les relations entre ces éléments
(Lagacherie et al., 2010)



Modèles couplant des processus à l'intérieur et entre objets du paysage
(e.g. Moussa et al., 2002)



Paramétrisation distribuée
Association de propriétés aux éléments du paysage



Exécution de la simulation

Une plate-forme "flux dans les paysages"?

Un environnement logiciel pour la modélisation intégrative

- Ouvrir à la pluri-disciplinarité, proposant des **formalismes "aussi génériques que possible"**
- Construire des **modèles adaptés** aux objectifs donnés

Un outil pour la recherche

- **Utiliser** et **réutiliser** des modèles
- Partager un **environnement logiciel unique**
- Définir des **standards** pour le développement et les données
- Etre dans la lignée Open-Source

Une démarche d'ingénierie logicielle

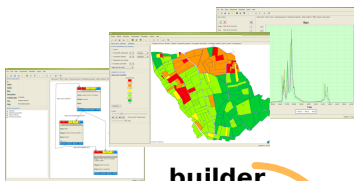
- Profiter des technologies récentes ou émergentes
- Améliorer la maintenance, l'évolutivité, la qualité
- Introduire des méthodes de projet adaptées (AGILE)

Un lien pour une communauté (petite ou grande)

- **Partager** des **compétences** et des **connaissances**

La plate-forme OpenFLUID

construction de modèles et d'espaces, paramétrisation, exécution, exploitation des résultats, visualisation



builder

engine

web

simulation functions

segmentation functions

**modélisation
simulation
des flux**

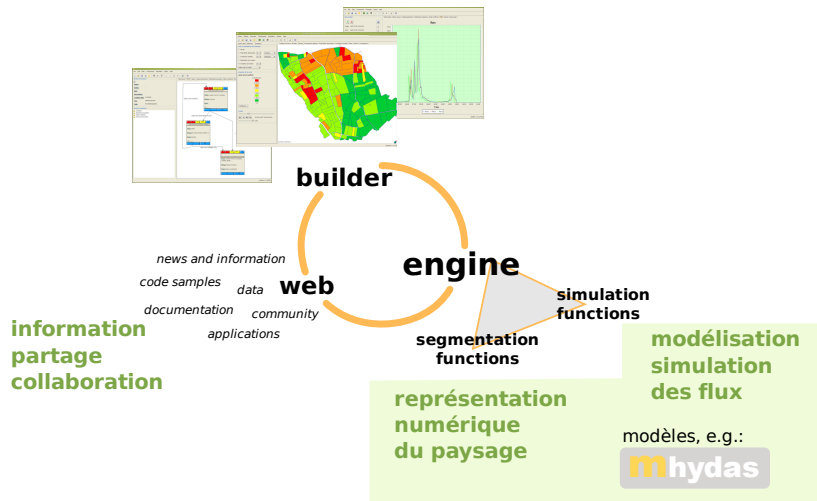
**représentation
numérique
du paysage**

**information
partage
collaboration**

news and information
code samples data
documentation community
applications

La plate-forme OpenFLUID

construction de modèles et d'espaces, paramétrisation, exécution, exploitation des résultats, visualisation



OpenFLUID-Engine

Composé d'un **noyau**

- structuration et gestion des données spatio-temporelles,
- gestion du branchement des codes de calcul,
- exécution et suivi de la simulation, contrôles de cohérence,
- gestion des entrées/sorties,

... de **fonctions de simulation** pour construire des modèles couplés

- codes de calcul à exécuter, branchés dynamiquement au noyau ("plug-ins")
- SDK pour le développement de nouveaux codes

... de **procédures de représentation numérique** de l'espace étudié (segmentation)

- intégration de sorties de scripts sous GRASS 6.2+

OpenFLUID-Engine

La construction de modèles, adaptés au contexte et objectifs de simulation, se fait à partir **d'une ou plusieurs fonctions de simulation**

Les fonctions de simulation peuvent être développées en utilisant le **SDK OpenFLUID** (Source Development Kit)

Les scientifiques/modélisateurs peuvent développer leurs propres fonctions de simulation

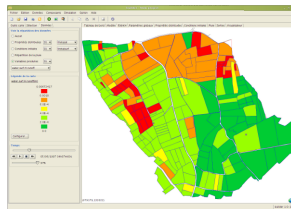
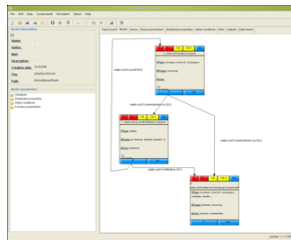
- en **encapsulant** des modèles/codes existants
- en **développant** des modèles/codes en partant de rien

OpenFLUID-Builder

En interaction avec OpenFLUID-Engine, en tant que GUI

- Import de RNP
- Editeur de modèle
- Paramétrisation
- Simulation
- Traitement des résultats
- Visualisations spatio-temporelles

Extensible par ajout de
Builder-extensions (plug-ins)



OpenFLUID-Web

<http://www.umr-lisah.fr/openfluid/>

Une présentation générale

- Projet
- Applications
- Téléchargements

Un espace collaboratif

- Documentations utilisateur, développeur
- Guides, conventions
- Exemples de codes
- ...

The screenshot shows a web browser displaying the OpenFLUID website. The browser's address bar shows the URL <http://www.umr-lisah.fr/openfluid/>. The website has a green and white color scheme. At the top, there is a navigation menu with tabs for 'Accueil', 'Projet de recherche', 'MIVIDAS', and 'Community'. Below this is a 'Présentation' section with the heading 'Le projet OpenFLUID'. The text describes the project as a major software program for modeling processes in landscapes, developed by the LISAH laboratory. It mentions the MIVIDAS model and the goal of creating a user-friendly modeling platform. There are three main axes of research: cognitive, operational, and spatial. The 'Segmentation de l'espace' section discusses the need to represent the landscape and the use of GIS data. The 'Modélisation des processus' section describes the goal of modeling processes in a modular way. At the bottom left of the page content, there is a 3D visualization of a terrain with a river and trees. The browser's status bar at the bottom shows 'Terminé' and 'en_GB'.

Etat d'avancement

OpenFLUID-Engine:

- Version 1.4.2, disponible en téléchargement
- ~ 40 fonctions de simulation développées (hydrologie, polluants, érosion)
- Licence libre GPLv3 (version $\geq 1.5.0$)

OpenFLUID-Builder:

- Version 1.1.x, compatible OpenFLUID-Engine 1.1.x uniquement
- Développements actuellement "en stand-by"

OpenFLUID-Web:

- Accessible en ligne
- Espace communautaire régulièrement étoffé

Ruissellement et propagation

Parcelle AW6 - Roujan (X. Louchart)



Parcelle AW6 - Roujan :

- 1200 m²
- 1070 unités spatiales

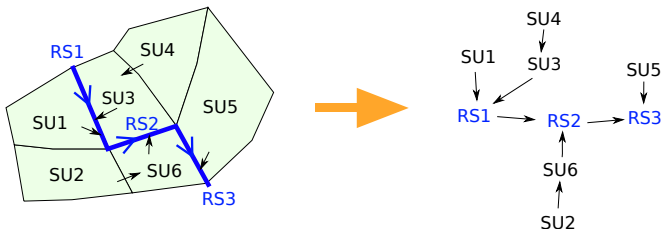
Simulation :

- sur 1 heure
- pas de temps : 10 s

Représentation numérique du paysage (RNP)

Le paysage est représenté par OpenFLUID-engine sous la forme d'un graphe

- Les noeuds sont les unités spatiales représentant le paysage
- Les arcs orientés sont les connexions entre les unités
- Chaque noeud porte des propriétés propres à l'unité qu'il représente



Simulation de processus

Les processus sont représentés par des **fonctions de simulation** qui utilisent et produisent des valeurs de **variables d'état** sur les unités spatiales.

Elles manipulent également des **données d'entrée** distribuées, des **paramètres globaux**, des **événements discrets**.

- Chaque fonction de simulation déclare les variables utilisées et produites, ses données d'entrée et ses paramètres globaux
- Les fonctions de simulation s'engagent à respecter ce qu'elles déclarent

Paramètres de fonctions

Les paramètres de fonctions sont des **paramètres globaux** qui s'appliquent à une **fonction en particulier**

- attachés aux fonctions
- nombre entiers ou décimaux, chaînes de caractères
- scalaires ou vecteurs

Fichiers d'entrée : organisation et formats

Un jeu de données d'entrée OpenFLUID-engine est constitué d'un ensemble de **fichiers d'entrées standardisés** au format **XML**.

- `model.xml` : définition du modèle couplé
- fichiers `*.ddef.xml` : définition de la RNP
- fichiers `*.ddata.xml` : données d'entrée distribuées
- fichiers `*.events.xml` : définition des évènements distribués
- `run.xml` : configuration de la simulation
- `output.xml` : configuration des sorties des résultats

Il est possible pour les fonctions de simulation de gérer leurs propres fichiers d'entrée.

Fichier model.xml

Exemple

```
<?xml version="1.0" standalone="yes"?>
<openfluid>
  <model>
    <function fileID="water.atm-surf.rain-su.files" />
    <function fileID="water.surf-uz.runoff-infiltration.mseytoux" >
      <param name="resstep" value="0.000005" />
    </function>
  </model>
</openfluid>
```

- déclaration d'une fonction : balise `<function>` avec attribut `fileID`
- déclaration d'un paramètre de fonction : balise `<param>` avec attributs `name` pour le nom du paramètre et `value` pour sa valeur

Fichiers *.ddef.xml

Exemple

```
<?xml version="1.0" standalone="yes"?>
<openfluid>
  <domain>
    <definition>
      <unit class="SU" ID="1" pcsorder="1">
        <to class="SU" ID="5" />
      </unit>
      <unit class="SU" ID="5" pcsorder="2">
        <to class="RS" ID="9" />
      </unit>
      <unit class="RS" ID="9" pcsorder="1" />
    </definition>
  </domain>
</openfluid>
```

- déclaration d'une nouvelle unité : balise <unit> avec les attributs précisant la classe de l'unité, son ID et son ordre de traitement
- déclaration d'une connexion avec une autre unité : sous-balise <to> de la balise <unit>

Fichiers *.ddata.xml

Exemple

```
<?xml version="1.0" standalone="yes"?>
<openfluid>
  <domain>
    <inputdata unitclass="SU">
      <columns order="area;slope;flowdist" />
      <data>
1 4813.344 0.06265 14.366
2 293.982 0.27129 18.468
      </data>
    </inputdata>
  </domain>
</openfluid>
```

- données en colonnes entre balises <data> et </data> (1ère colonne = ID de l'unité)
- ordre des colonnes (après colonne ID) précisé au travers de la balise columns avec l'attribut order

Fichiers *.events.xml

Exemple

```
<?xml version="1.0" standalone="yes"?>
<openfluid>
  <calendar>
    <event unitclass="SU" unitID="1" date="1997-03-30_12:00:00">
      <info key="molecule" value="diuron"/>
      <info key="percent_area" value="100"/>
      <info key="rate_ha" value="1"/>
    </event>
  </calendar>
</openfluid>
```

- un évènement est déclaré au travers d'une balise event, et est associé à une unité, à une date donnée
- il porte de 0 à n informations déclarées par une balise info, chaque valeur (attribut value) étant identifiée par une clé (attribut key)

Fichier run.xml

Exemple

```
<?xml version="1.0" standalone="yes"?>
<openfluid>
  <run>
    <deltat>60</deltat>
    <period begin="1997-03-29_03:00:18" end="1997-04-01_16:23:21" />
    <progressout packet="50" keep="5" />
  </run>
</openfluid>
```

- la configuration de la simulation comprend obligatoirement une période de simulation (<period>) et un pas de temps d'échange (<deltat>)
- les sorties progressives (optionnel) peuvent être précisées au travers de la balise <progressout>

Fichier output.xml

Exemple

```
<?xml version="1.0" standalone="yes"?>
<openfluid>
  <output>
    <files colsep="_" dtformat="%Y-%m-%dT%H:%M:%S" commentchar="#">
      <set name="full" unitsclass="SU" unitsIDs="*" vars="*" />
      <set name="oneunit" unitsclass="SU" unitsIDs="1" vars="*" />
      <set name="onevar" unitsclass="RS" unitsIDs="*" vars="water.surf.H.level-rs" />
    </files>
  </output>
</openfluid>
```

- configuration de jeux de données de sortie (<set>)
- "customisation" des sorties : format de dates, choix des unités, choix des variables, ...

Environnement

Répertoires par défaut (sous Unix/Linux) pour un utilisateur toto

- OpenFLUID-engine personnel :
/home/toto/.openfluid/engine
- Données en entrée :
/home/toto/.openfluid/engine/OPENFLUID.IN
- Données résultats :
/home/toto/.openfluid/engine/OPENFLUID.OUT
- Fonctions de simulation :
/home/toto/.openfluid/engine/functions

Variables d'environnement

- \$OPENFLUID_FUNCS_PATH : chemins(s) de recherche de fonctions de simulation

Principales options de la ligne de commande

Option	Description
-i	chemin du jeu de données d'entrée
-o	chemin de sauvegarde des résultats
-p	chemins de recherche de fonctions de simulation
-f	liste des fonctions disponibles
-r	rapport détaillé sur les fonctions disponibles
-u	rapport détaillé sur une(des) fonction(s) particulière(s)
-q	affichage silencieux pendant la simulation
-v	affichage détaillé pendant la simulation
--version	retourne la version d'OpenFLUID-Engine

- Les options de la ligne de commande peuvent être combinées

Buddies

Les **buddies** ou "compagnons" sont des outils périphériques, accompagnant OpenFLUID dans sa mise en oeuvre

4 compagnons sont intégrés à OpenFLUID-engine:

- **func2doc** : génération automatique de documentation scientifique
- **convert** : convertisseur de formats de jeux de données d'entrée
- **newfunc** : générateur de code source de fonctions de simulation
- **newdata** : générateur de jeux de données vides formatés

Utilisation en ligne de commande `openfluid-engine` avec les options `--buddy`, `--buddyopts`, `--buddyhelp`

Buddy func2doc

Utilisation du code source de la fonction de simulation pour en extraire les informations nécessaires

- Lecture de la signature de la fonction de simulation
- Insertion du contenu \LaTeX placé en commentaires du code source entre les balises `<func2doc>` et `</func2doc>` (optionnel)

Génère un fichier latex, pouvant être converti à la volée en PDF ou en HTML

Exemple

```
openfluid-engine --buddy func2doc --buddyopts \  
  inputcpp=MyFunc.cpp,outputdir=./doc,PDF=1
```

Buddy convert

Conversion d'un jeu de données d'entrée OpenFLUID-engine d'un format vers un autre

- Conversion des fichiers standards du jeu de données uniquement
- Conversions disponibles : 1.3 vers 1.4, 1.4 vers 1.5

Génère le jeu de données converti, et le place dans un répertoire donné

Exemple

```
openfluid-engine --buddy convert --buddyopts \  
  convmode=14_15,inputdir=./data14,outputdir=./data15
```

Buddies newfunc et newdata

- newfunc génère le(s) fichier(s) source(s) "squelette" d'une fonction de simulation

Exemple

```
openfluid-engine --buddy newfunc --buddyopts \  
  cppclass=MyFunction , funcid=my.test.function
```

- newdata génère les fichiers d'un jeu de données d'entrée vide, et formaté pour openfluid-engine

Exemple

```
openfluid-engine --buddy newdata --buddyopts outputdir=./data.IN
```

Plan

- 1** Présentation générale
 - Contexte & Historique
 - Approche de modélisation
 - Description de la plate-forme
- 2** Exemples d'applications
 - Echelle parcellaire
 - Echelle du bassin versant
- 3** OpenFLUID-Engine
 - Fonctionnalités générales
 - Fichiers d'entrée
 - Mise en oeuvre de simulations
 - Buddies
- 4** Référentiel de fct. simu.
 - **Présentation**

Référentiel de fonctions de simulation

Capitalisation et partage des fonctions de simulation développées au LISAH

- Partage et versionnement du code source, via gestionnaire de code source collaboratif (subversion)
- Construction et packaging automatisés
- Tests sur les fonctions et modèles, automatisés (toutes les nuits) ou manuels
- Stockage et sauvegarde

En interne LISAH uniquement, à utiliser avec un client Subversion :

<https://www.umar-lisah.fr/svn/openfluid-functions/>

FunctionsRepository2Web

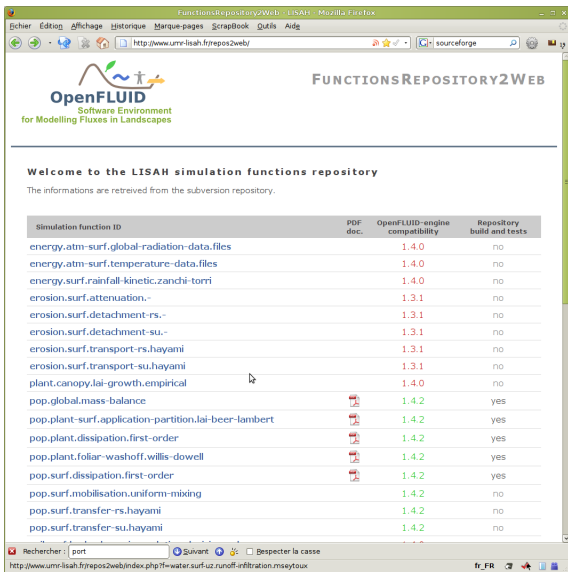
Consultation du référentiel de fonctions de simulation via une application web

- Liste des fonctions disponibles
- Versions de compatibilité OpenFLUID-Engine
- Documentation associée (générée par le buddy `func2doc`)
- Historique des modifications des fonctions
- ...

En interne LISAH uniquement, accessible avec un navigateur web :

<http://www.umn-lisah.fr/repos2web/>






FunctionsRepository2Web



OpenFLUID
Software Environment
for Modelling Fluxes in Landscapes

FUNCTIONSREPOSITORY2WEB

Welcome to the LISAH simulation functions repository
The informations are retrieved from the subversion repository.

Simulation function ID	PDF doc.	OpenFLUID-engine compatibility	Repository build and tests
energy.atm-surf.global-radiation-data.files		1.4.0	no
energy.atm-surf.temperature-data.files		1.4.0	no
energy.surf.rainfall-kinetic.zanchi-torri		1.4.0	no
erosion.surf.attenuation.-		1.3.1	no
erosion.surf.detachement-rs.-		1.3.1	no
erosion.surf.detachement-su.-		1.3.1	no
erosion.surf.transport-rs.hayami		1.3.1	no
erosion.surf.transport-su.hayami		1.3.1	no
plant.canopy.lai-growth.empirical		1.4.0	no
pop.global.mass-balance		1.4.2	yes
pop.plant-surf.application-partition.lai-beer-lambert		1.4.2	yes
pop.plant.dissipation.first-order		1.4.2	yes
pop.plant.foliar-washoff.willis-dowell		1.4.2	yes
pop.surf.dissipation.first-order		1.4.2	yes
pop.surf.mobilisation.uniform-mixing		1.4.2	no
pop.surf.transfer-rs.hayami		1.4.2	no
pop.surf.transfer-su.hayami		1.4.2	no

Rechercher : port Suivant Respecter la casse
http://www.umr-lisah.fr/repos2web/index.php?water.surf.uz.runoff-infiltration.mseytoux fr_FR

FunctionsRepository2Web

The screenshot shows a Mozilla Firefox browser window displaying the FunctionsRepository2Web website. The page title is "FUNCTIONSREPOSITORY2WEB". The logo for "OpenFLUID" is visible, with the tagline "Software Environment for Modelling Fluxes in Landscapes". The main content area displays the package name "water.surf-uz.runoff-infiltration.mseytoux" and a list of metadata:

- ID: water.surf-uz.runoff-infiltration.mseytoux
- Contacts: (none)
- Short description: (none)

Below this, three subversion information blocks are shown, each corresponding to a different OpenFLUID-engine version:

- With OpenFLUID-engine version 1.4.2**
 - Repository build : yes
 - Subversion information
 - Last Changed Rev: 481
 - Last Changed Author: lalauze
 - Last Changed Date: 2010-04-08 16:02:15 +0200 (Thu, 08 Apr 2010)
 - View log
- With OpenFLUID-engine version 1.4.1**
 - Repository build : yes
 - Subversion information
 - Last Changed Rev: 445
 - Last Changed Author: lalauze
 - Last Changed Date: 2010-01-29 15:59:06 +0100 (Fri, 29 Jan 2010)
 - View log
- With OpenFLUID-engine version 1.4.0**
 - Repository build : yes
 - Subversion information
 - Last Changed Rev: 445
 - Last Changed Author: lalauze
 - Last Changed Date: 2010-01-29 15:59:06 +0100 (Fri, 29 Jan 2010)
 - View log

The browser's search bar at the bottom contains the text "port". The status bar at the bottom indicates "Terminé" and "fr_FR".

