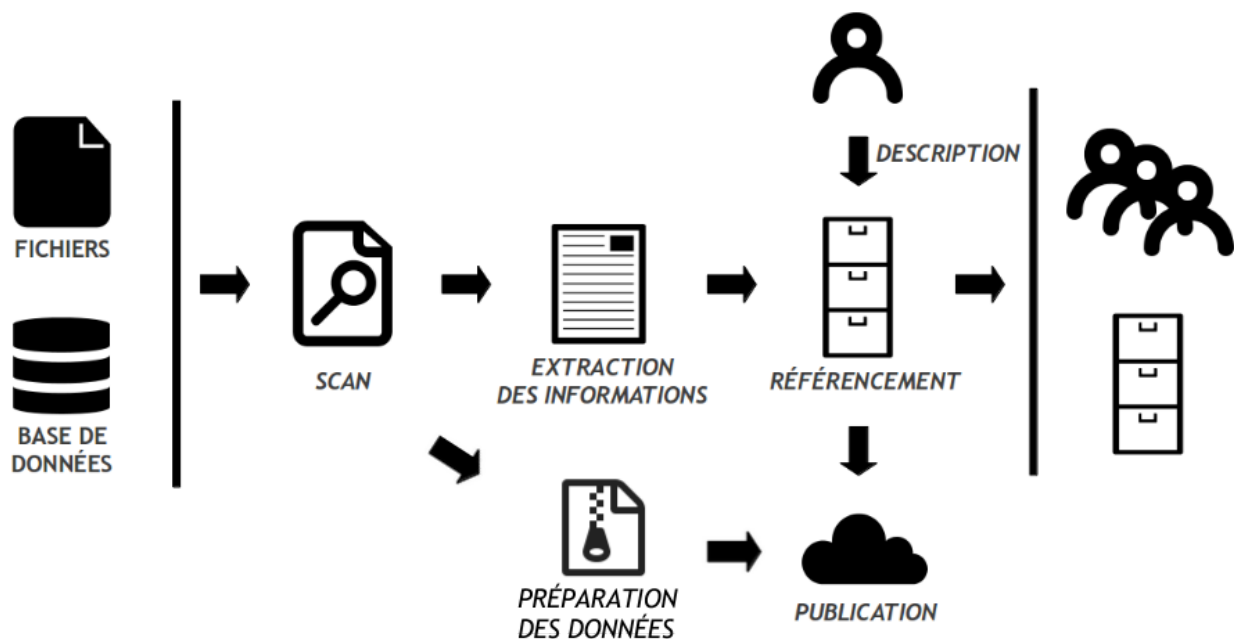


Inventaire automatique du patrimoine d'information géographique avec Talend Spatial

Synchroniser les données avec leur métadonnée est un point clé des IDS modernes. Dans cet atelier, vous apprendrez comment un ETL spatial permet d'analyser de multiples sources d'information géographique et d'inventorier ces ressources dans un catalogue sur Internet.



[Auteurs & contributeurs](#)

[Licence](#)

[Principe](#)

[Téléchargements & installation](#)

[Télécharger JAVA](#)

[Télécharger GDAL/OGR](#)

[Configurer GDAL/OGR](#)

[Ajouter des formats spécifiques](#)

[Installer TDI](#)

[Premier lancement](#)

[Importer les jobs dans l'espace de travail](#)

[Valider la bonne installation du module Spatial](#)

[Valider l'installation GDAL/OGR](#)

[Chaîne de traitements](#)

[Les formats supportés](#)

[L'extraction des propriétés par analyse des fichiers](#)

[Format PostGIS : filtrage](#)

[Règles générales](#)

[Le jeu de test](#)

[Configurer les jobs](#)

[Répertoire de fichier de propriétés](#)

[Configuration des fichiers de propriétés](#)

[Lancement des jobs depuis le Studio](#)

[Job "run"](#)

[Job "scan_vector"](#)

[Job "scan_raster"](#)

[Job "publish_metadata"](#)

[Job "pack_data"](#)

[Job "publish_files"](#)

[Export du job](#)

[Lancer le job](#)

[Lire le fichier journal](#)

[Lancement des jobs en mode autonome](#)

[Lancer le job en tant que service web](#)

[FAQ](#)

[Conclusions](#)

[Annexes](#)

[Autres travaux similaires & références](#)

[Corrections & améliorations apportées à TDI](#)

[Corrections & améliorations apportées au module Spatial](#)

Ce document présente la procédure d'installation des jobs Talend permettant de réaliser un inventaire automatique de données SIG. Le job est réalisé avec Talend Open Studio et plus particulièrement le module Data Integration (nommé TDI par la suite).

Auteurs & contributeurs

[François Prunayre](#) ([titellus](#))

Mathieu Coudert (Astrium GEO-Information Services)

Bruno Lafage ([Parc Nationaux de France](#))

Juan Luis Rodríguez Ponce ([geocat.bv](#))

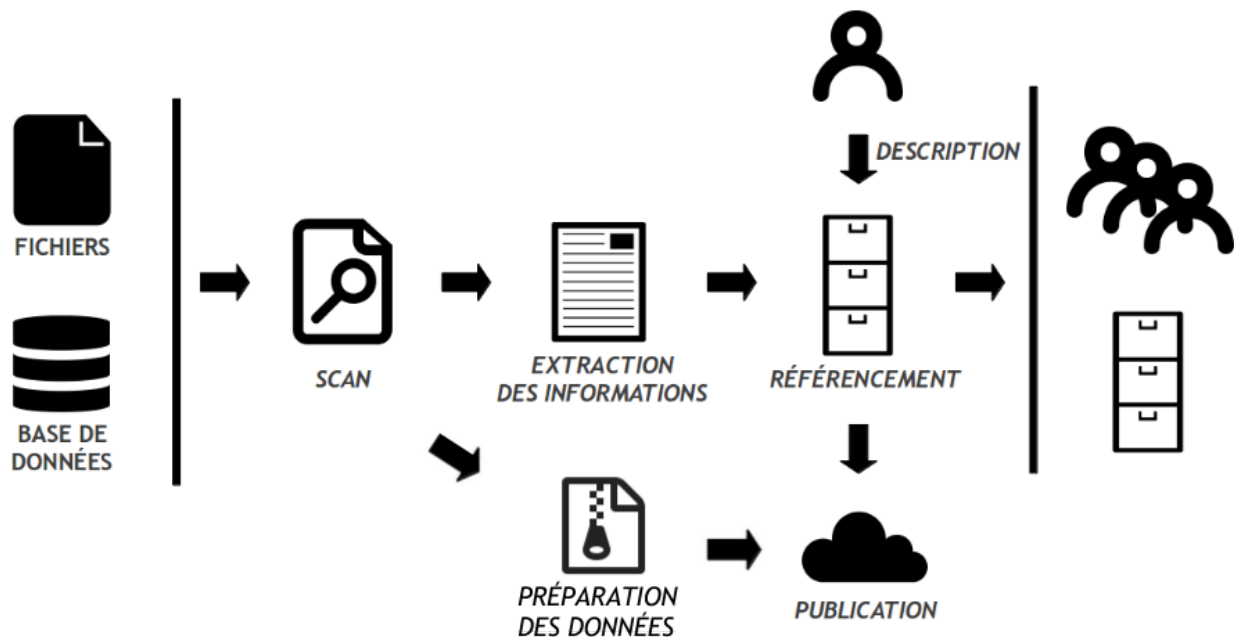
Licence

Cette oeuvre, création, site ou texte est sous licence Creative Commons Attribution - Partage dans les Mêmes Conditions 4.0 International. Pour accéder à une copie de cette licence, merci de vous rendre à l'adresse suivante <http://creativecommons.org/licenses/by-sa/4.0/> ou envoyez un courrier à Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.



Principe

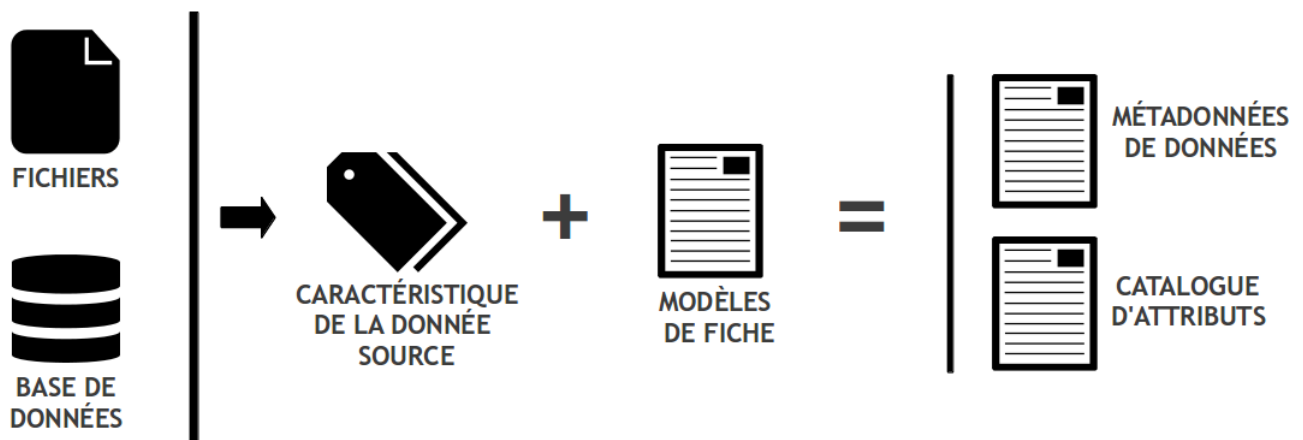
L'inventaire de données est réalisé par scan de répertoires, de bases de données, de service web ou de cartes. Pour chaque données SIG identifiées, les caractéristiques de la donnée sont extraites puis des fiches descriptives sont créées à partir d'un modèle de saisie. Une fois créées, les fiches peuvent être publiées dans un catalogue et les données packagées pour publication sur Internet.



La phase d'extraction repose sur l'analyse des données SIG. Pour les données vecteurs, 2 fiches sont créées :

- une métadonnée de donnée au format ISO19139
- un catalogue d'attributs au format ISO19110

Pour les raster et les cartes, seule la fiche de métadonnée de données est générée.



Video: <http://youtu.be/N3NhA5DNnCc>

Qu'allez vous apprendre ?

- Installer Talend et son module Spatial¹
- Créer et importer un espace de travail
- Lancer les jobs réalisant
 - un scan automatique des vos données SIG et créant les métadonnées et les imagerie correspondantes
 - publiant les informations dans votre catalogue

Téléchargements & installation

Avant de commencer l'installation, vérifier le type d'architecture pour la machine sur laquelle l'installation est faite (32 ou 64b). Les composants suivants DOIVENT être dans l'architecture ciblée :

- Java JDK
- GDAL

Pour TDI, l'installateur contient l'application pour les 2 types d'architecture. Cependant le bon exécutable DOIT être utilisé ie. pour Windows 32b utiliser TOS_DI-win32-x86.exe:



Télécharger JAVA

¹ <http://www.youtube.com/watch?v=vt3q5-3rOcA>

Vérifier la présence de Java : ouvrir une console et saisir `java -version`

```
java -version
java version "1.7.0_65"
OpenJDK Runtime Environment (IcedTea 2.5.3)
(7u71-2.5.3-0ubuntu0.14.04.1)
OpenJDK 64-Bit Server VM (build 24.65-b04, mixed mode)
```

<http://www.oracle.com/technetwork/java/javase/downloads/index.html?ssSourceSiteId=ocomen>
Installer Java en suivant la procédure d'installation.

Pour linux, installer JAVA avec

```
sudo apt-get install openjdk-7-jdk
```

Téléchargement de l'application

Télécharger Talend Data Integration (nommer TDI dans la suite du document)
<http://www.talend.com/download/data-integration> (version 5.6.2 recommandée)

Télécharger le module spatial

<http://sourceforge.net/projects/sdispatialet/files/sdispatialet/>

- version recommandée 6.0.1 :
<https://sourceforge.net/projects/sdispatialet/files/sdispatialet/TOS.spatial.6.0.1/>
- version recommandée 5.4.1 :
<https://sourceforge.net/projects/sdispatialet/files/sdispatialet/TOS.spatial.5.4.1/>

Télécharger GDAL/OGR

Pour GDAL, il est nécessaire d'avoir GDAL/OGR ainsi que le "binding" Java pour que Talend accède à la librairie GDAL. Par exemple, GDAL installé avec QGIS ne contient pas le binding Java. Il est alors nécessaire de faire l'installation ci-dessous.

Vérifier la présence de gdal : ouvrir une console et saisir `gdalinfo`

```
gdalinfo
Usage: gdalinfo [--help-general] [-mm] [-stats] [-hist] [-nogcp]
[-nomd]
                [-norat] [-noct] [-nofl] [-checksum] [-proj4]
                [-listmdd] [-mdd domain|`all`]*
                [-sd subdataset] datasetname

FAILURE: No datasource specified.
```

Pour windows : télécharger et lancer l'installer gdal-110-1600-core.msi "Generic installer for the GDAL core components" :

- pour windows 32b
<http://www.gisinternals.com/query.html?content=filelist&file=release-1600-gdal-1-11-1-mapserver-6-4-1.zip>
- pour windows 64b
<http://www.gisinternals.com/query.html?content=filelist&file=release-1600-x64-gdal-1-11-1-mapserver-6-4-1.zip>

Sous Windows, si vous n'avez pas les privilèges pour installer via l'installer (.msi), utiliser "Compiled binaries in a single .zip package".

Pour linux :

- compiler GDAL/OGR à partir des sources ou
- installer via les dépôts eg. ubuntu². Par exemple, installer qgis et libgdal-java :

```
sudo apt-get install qgis libgdal-java
```

Configurer GDAL/OGR

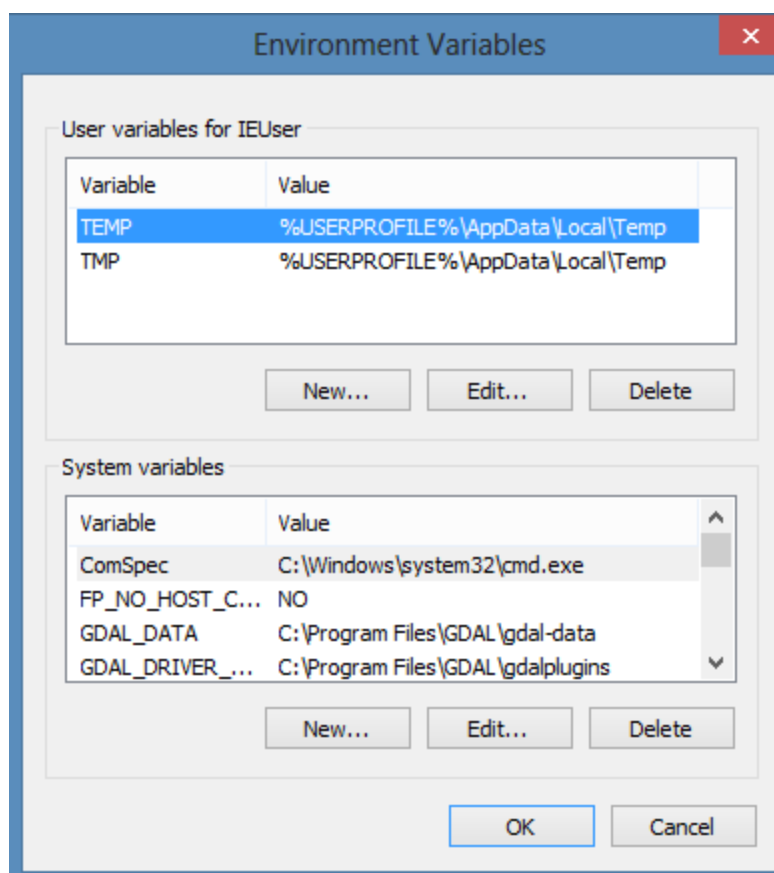
GDAL/OGR est la librairie utilisée pour analyser les fichiers SIG.

Installer GDAL/OGR à partir de l'installer (fichier .msi téléchargé). Choisir l'installation "Typical".

Pour accéder aux variables systèmes, réaliser les actions suivantes :

- dans l'explorateur Windows, clic droit sur Ordinateur
- sélectionner propriétés pour accéder à la configuration du système.
- sélectionner paramètres système avancés.
- sélectionner variables d'environnement.

² <https://launchpad.net/~ubuntugis/+archive/ubuntugis-unstable>



Ajouter les variables suivantes dans les variables du système :

Nom de la variable	Valeur (avec installer MSI)	Valeur (avec ZIP)
GDAL_PATH	C:\Program Files\GDAL	C:\Program Files\GDAL\bin\gdal
GDAL_DATA	%GDAL_PATH%\gdal-data	C:\Program Files\GDAL\bin\gdaldata
GDAL_DRIVER_PATH	%GDAL_PATH%\gdalplugins	C:\Program Files\bin\gdal\plugins
PROJ_LIB	%GDAL_PATH%\projlib	
PATH (séparer les valeurs par des “,”)	%GDAL_PATH%;%GDAL_PATH%\Java;...	%GDAL_PATH%;%GDAL_PATH%\Java;...

Pour chaque variable, cliquer sur “Nouvelle ...” et saisir le nom et la valeur.

Modifier la variable PATH existante en ajoutant les 2 chemins suivants séparés par des “,” :

- C:\Program Files\GDAL\java
- C:\Program Files\GDAL\

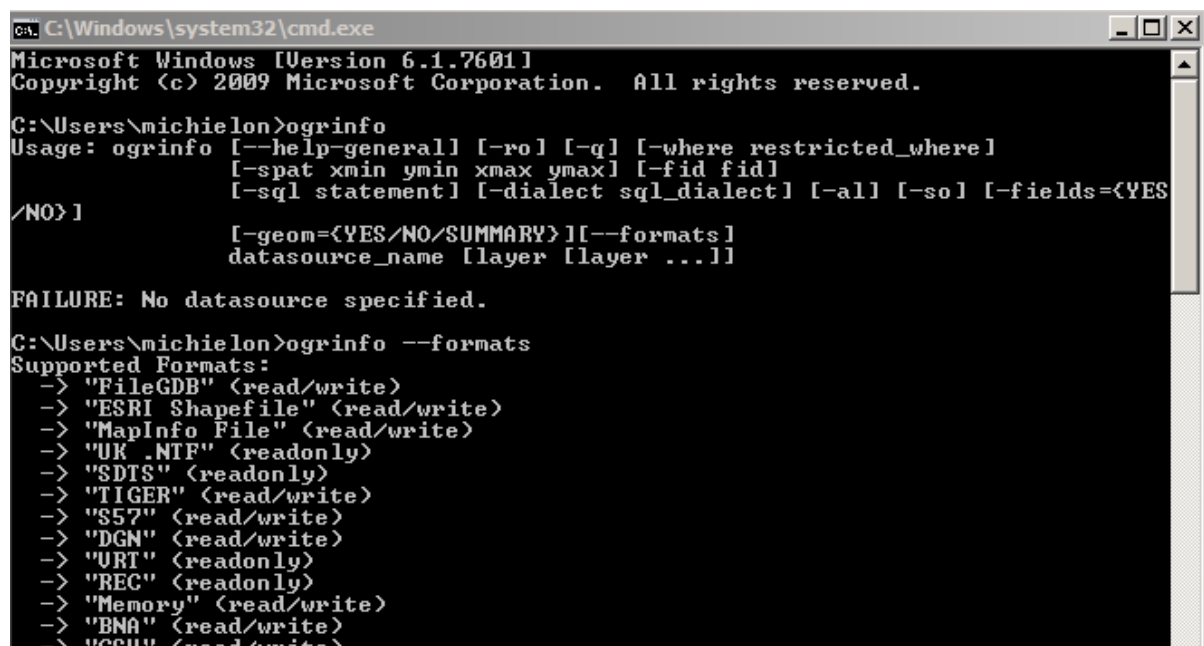
La variable PATH devrait alors être du type

“...;%GDAL_PATH%;%GDAL_PATH%\GDAL”.

Vérifier qu’il n’y a plusieurs installations de GDAL dans le PATH (la première est utilisée).

Comment vérifier la bonne install de gdal/ogr ?

Lancer une commande windows, et taper “ogrinfo --formats”.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\michi_\ogrinfo
Usage: ogrinfo [--help-general] [-ro] [-ql] [-where restricted_where]
              [-spat xmin ymin xmax ymax] [-fid fid]
              [-sql statement] [-dialect sql_dialect] [-all] [-sql] [-fields={YES
/NO}]
              [-geom={YES/NO/SUMMARY}] [--formats]
              datasource_name [layer [layer ...]]

FAILURE: No datasource specified.

C:\Users\michi_\ogrinfo --formats
Supported Formats:
-> "FileGDB" (read/write)
-> "ESRI Shapefile" (read/write)
-> "MapInfo File" (read/write)
-> "UK .NIF" (readonly)
-> "SDTS" (readonly)
-> "TIGER" (read/write)
-> "S57" (read/write)
-> "DGN" (read/write)
-> "VRT" (readonly)
-> "REC" (readonly)
-> "Memory" (read/write)
-> "BNA" (read/write)
-> "CSV" (read/write)
```

Ajouter des formats spécifiques

Pour lire d’autres formats (eg. Oracle, ECW), installer les plugins correspondants.

Vérifier la présence d'un format RASTER via gdal ou vecteur via OGR : ouvrir une console et saisir

```
gdalinfo --formats
Supported Formats:
  VRT (rw+v): Virtual Raster
  GTiff (rw+vs): GeoTIFF
  NITF (rw+vs): National Imagery Transmission Format
...

ogrinfo --formats
Supported Formats:
  -> "ESRI Shapefile" (read/write)
  -> "MapInfo File" (read/write)
...
```

Si le format souhaité n'est pas disponible installer le plugin correspondant.

Installer TDI

Dézipper TDI.

Dézipper ensuite le module spatial³ et copier le contenu du répertoire plugins dans le répertoire plugins de TDI. Cette étape installe le module spatial dans TDI. Cela permet d'ajouter à TDI :

- le type Geometry dans TDI
- la famille "geo" dans la palette
- les assistants de création de schéma générique

Le composant tAdvancedFileOutputXML de TDI doit être mis à jour afin de corriger un problème de génération de document XML (dû à <https://jira.talendforge.org/browse/TDI-26630>). Pour cela, dézipper le composant **tAdvancedFileOutputXML** du répertoire tos-components-patch du ZIP du module spatial. Copier tous les fichiers contenus dans le répertoire

tAdvancedFileOutputXML dans le répertoire du composant dans plugins/org.talend.designer.components.localprovider_5.x.x/components/tAdvancedFileOutputXML de l'installation de TDI. Cette opération doit mettre à jour les fichiers de configuration du composant (ie. javajet, xml, properties, ...). En fonction de la version de TDI, choisir le fichier correspondant :

- pour la version 6.x, utiliser tAdvancedFileOutputXML-6.x.zip
- pour la version 5.4.x, utiliser tAdvancedFileOutputXML-5.4.x.zip
- pour la version 5.5.x ou 5.6.x, utiliser tAdvancedFileOutputXML-5.5.x-or-5.6.x.zip

³ Attention : ajouter le module spatial après avoir lancé TDI ne fonctionne pas sur toutes les plateformes. Il est donc recommandé de dézipper TDI puis le module Spatial et ensuite lancer pour la première fois TDI.

Premier lancement

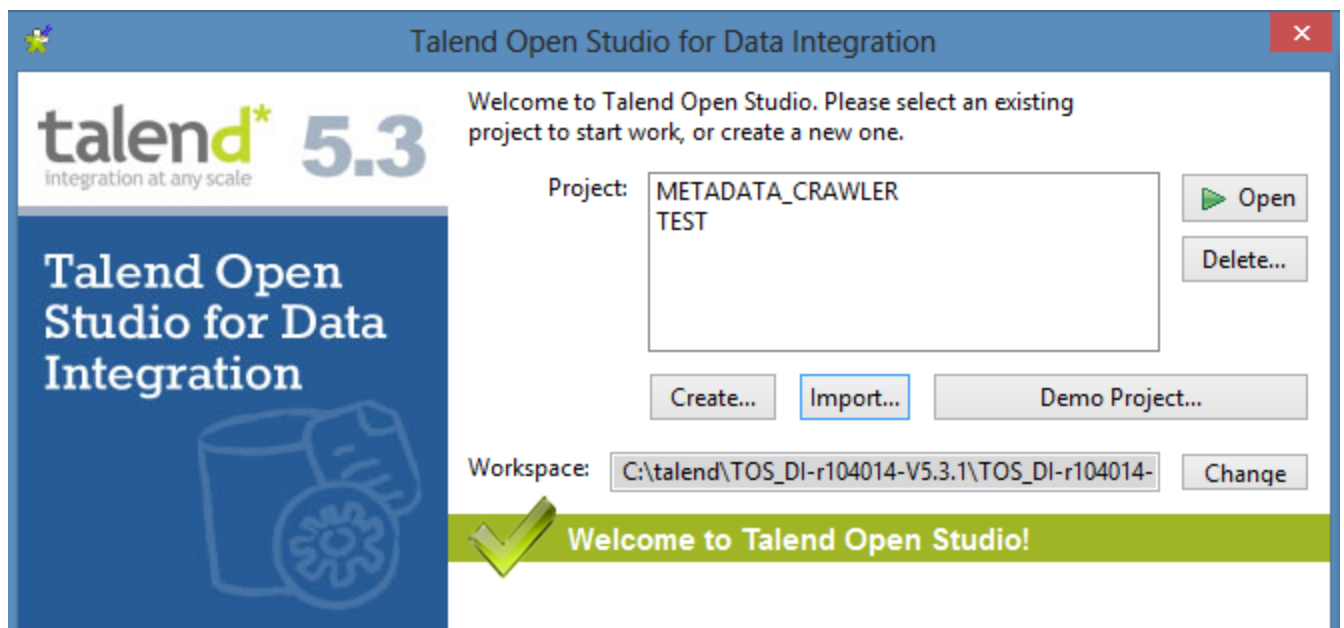
Démarrer TDI (eg. lancer TOS_DI-win32-x86.exe sous Windows 32b).

Pour ajouter TDI au menu Windows, créer un raccourci et l'ajouter au menu de lancement des programmes.

Cf. TalendOpenStudio_DI_IG_5.5.1_FR.pdf

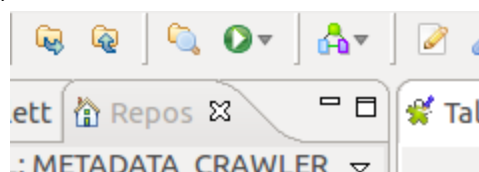
Importer les jobs dans l'espace de travail

A partir de l'écran d'accueil, sélectionner "créer", saisir le nom du projet (ci-dessous METADATA_CRAWLER) afin de créer un nouvel espace de travail.



Une fois créé, cliquer sur "ouvrir" et optionnellement renseigner le formulaire d'enregistrement Talend si nécessaire.

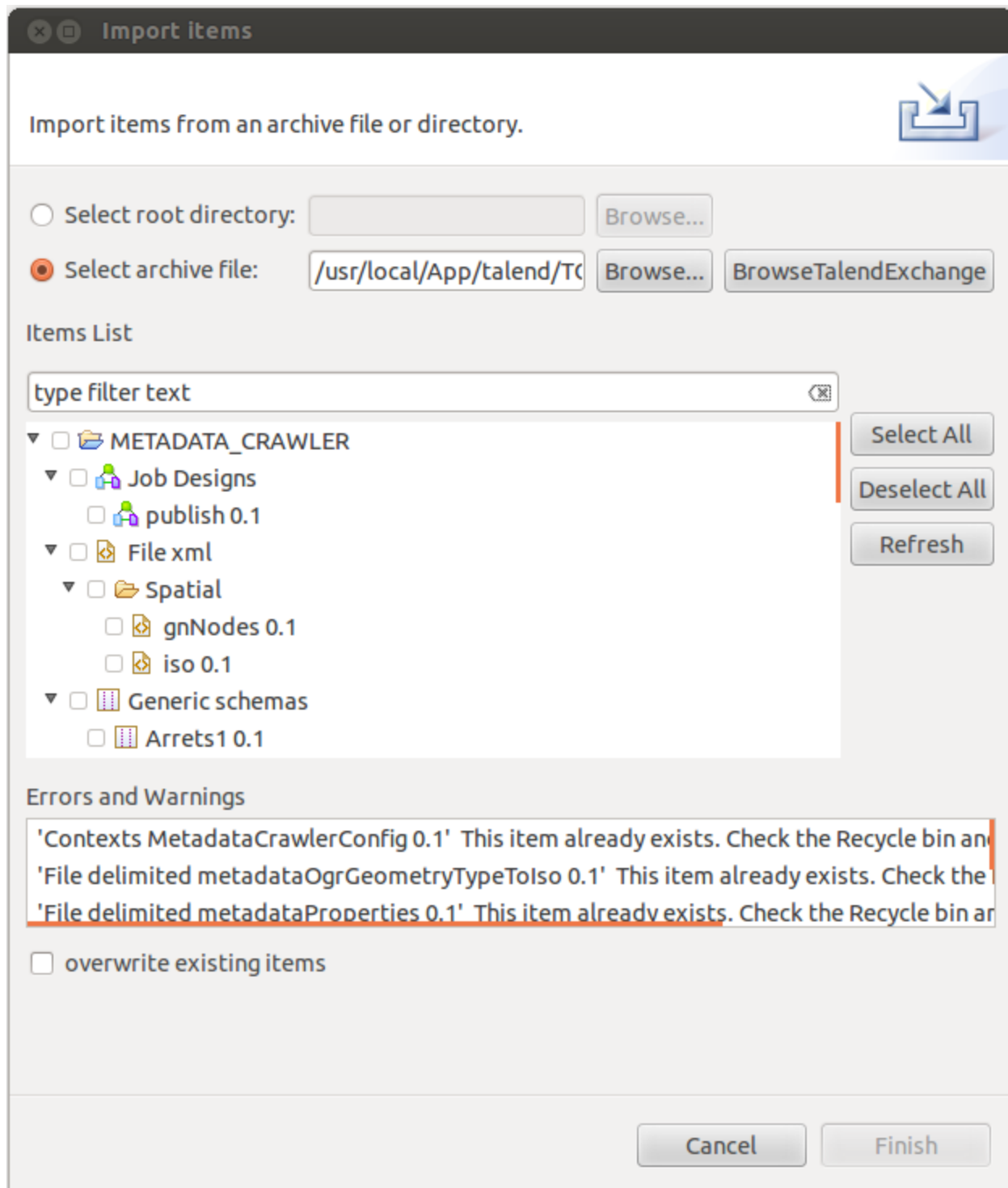
Ouvrir le projet puis cliquer sur l'icône "Importer des éléments" (ou "import Items") de la barre d'outil (2ème icône à gauche).



Télécharger les jobs depuis :

- <https://github.com/talend-spatial/workspace-metadata-crawler/archive/6.x.zip> pour TDI 6.x
- <https://github.com/talend-spatial/workspace-metadata-crawler/archive/5.x.zip> pour TDI 5.4.1
- <https://github.com/talend-spatial/workspace-metadata-crawler/archive/wk-for-5.3.1.zip> pour TDI 5.3.1

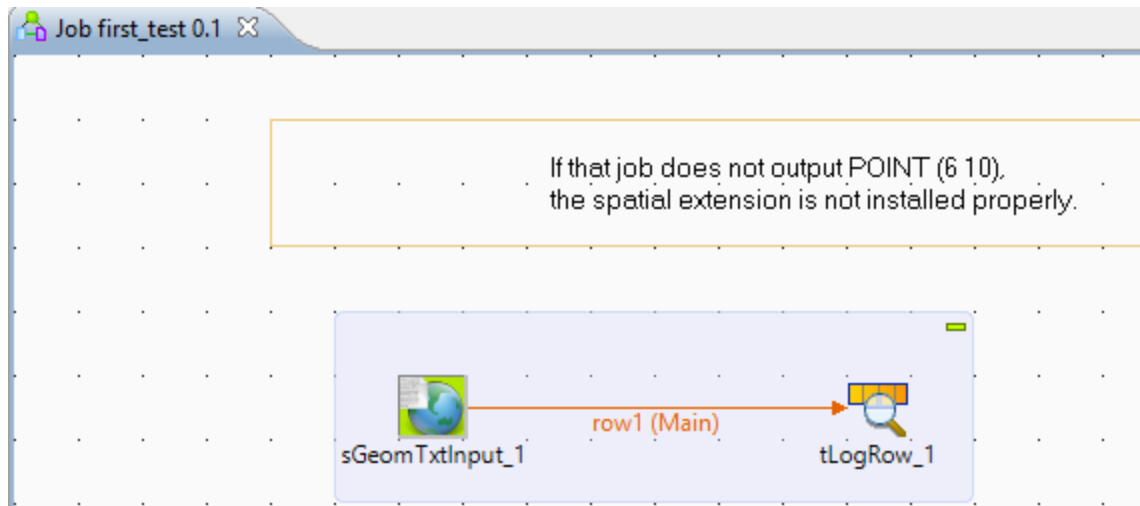
Sélectionner le fichier archive ZIP “METADATA_CRAWLER.zip”, sélectionner tous les éléments à importer et cliquer sur “Terminer”.



L'espace de travail pourra être exporter et importer via les outils d'import/export de Talend pour sauvegarder ou partager les jobs.

Valider la bonne installation du module Spatial

Ouvrir le job “first_test” et vérifier son bon fonctionnement. **Ce job permet de s’assurer que le module Spatial est bien installé.** Dans l’onglet “exécuter” et cliquer sur “exécuter”.

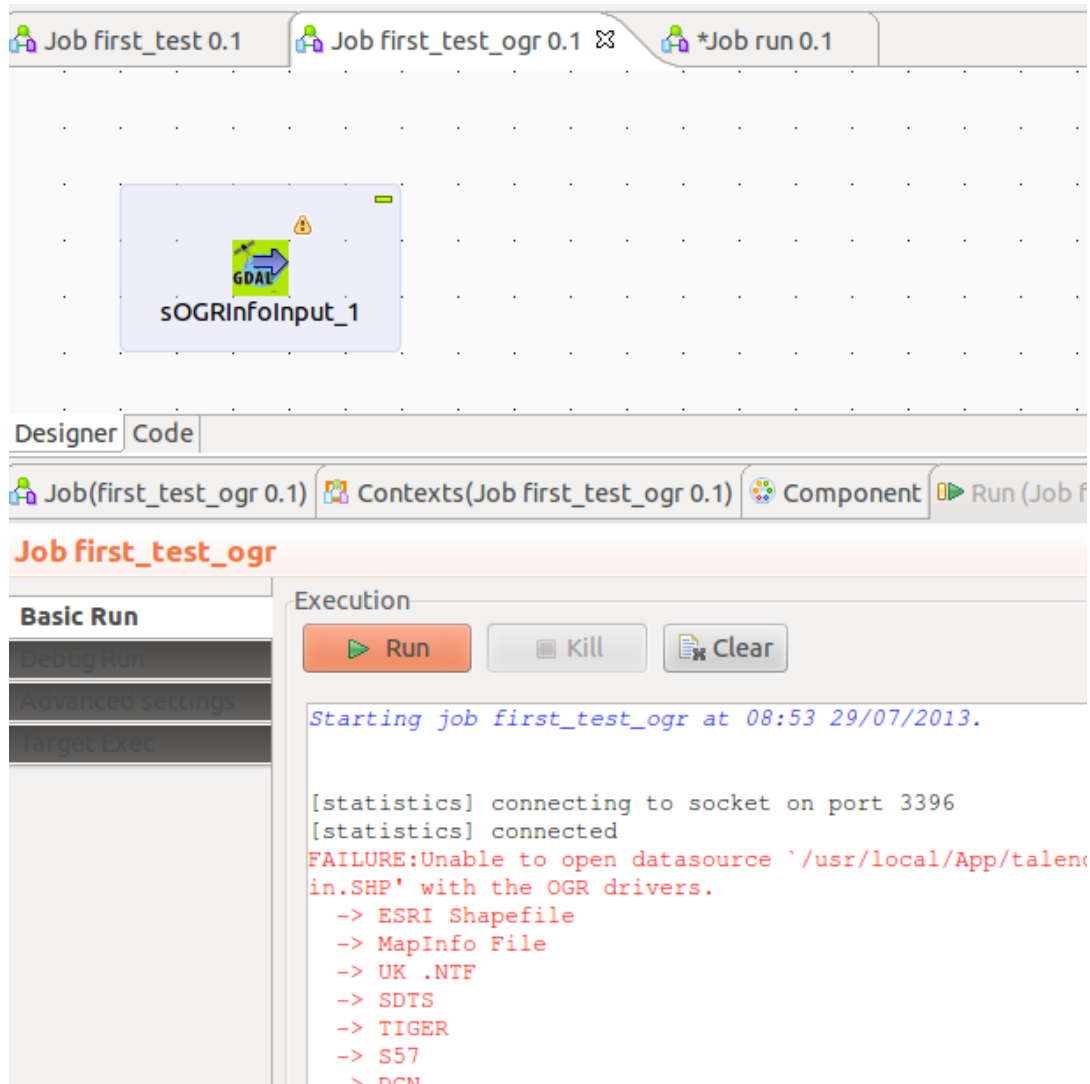


Le job doit retourner “POINT (6 10)” dans la fenêtre d’exécution du job.

Valider l’installation GDAL/OGR

Relancer TDI.

Ouvrir le job “first_test_ogr” et vérifier que la liste des drivers OGR apparaît dans la fenêtre d’exécution du job.



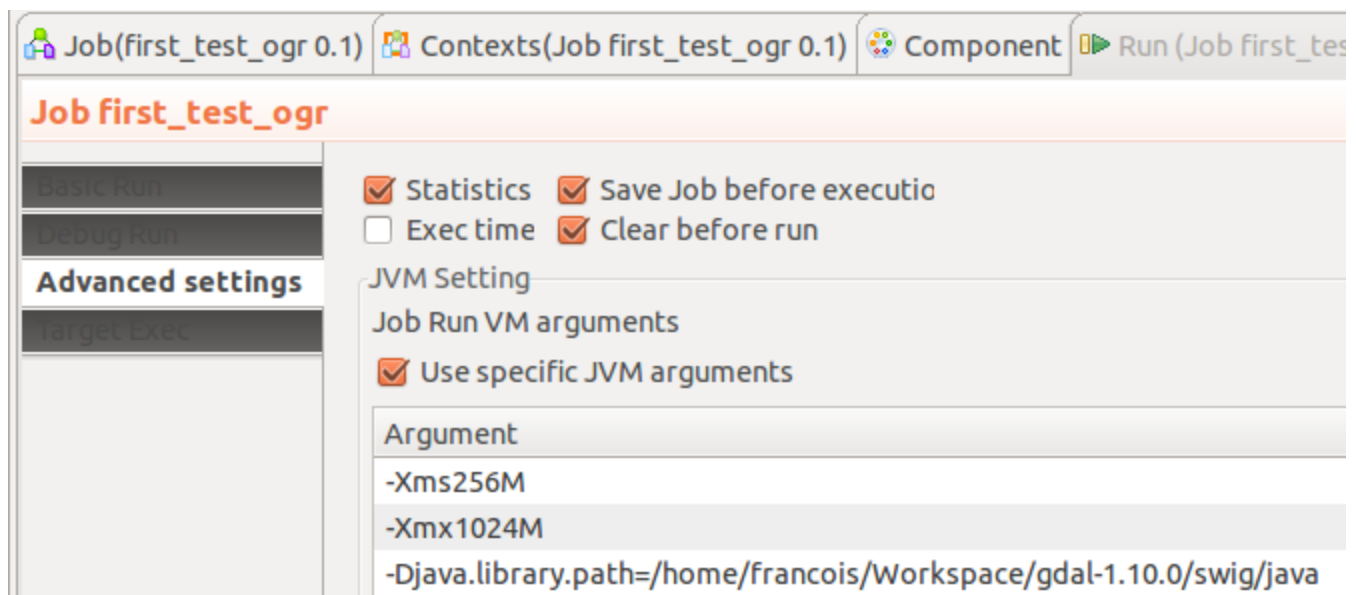
Si ce n'est pas le cas, vérifier les paramètres de lancement du job.

Le message d'erreur fréquent est UnsatisfiedLinkError :

```
[statistics] connecting to socket on port 3337
[statistics] connected
Native library load failed.
java.lang.UnsatisfiedLinkError: no ogrjni in java.library.path
In order to use GDAL/OGR in Talend, the java.library.path variable should point to GDAL
library.
To set this property, go to the Run view > Advanced settings > Use specific JVM Arguments
and add a new argument:
-Djava.library.path=/path/to/gdal/swig/java
```

```
java.lang.UnsatisfiedLinkError: org.gdal.ogr.ogrJNI.RegisterAll()V
  at org.gdal.ogr.ogrJNI.RegisterAll(Native Method)
  at org.gdal.ogr.ogr.RegisterAll(ogr.java:115)
```

Dans ce cas, dans l'onglet "Run" > "Advanced settings", vérifier le paramètre "Use specific JVM arguments". Sous Windows cette option n'est pas nécessaire (car les variables d'accès à GDAL/OGR sont dans les variables systèmes). Sous linux, cette option peut être nécessaire si vous avez compiler une version en particulier de GDAL/OGR. Dans ce cas, pointer vers ajouter le paramètre -Djava.library.path=/path/to/gdal/swig/java.



Ne pas utiliser "Use specific JVM arguments" sous Windows

Si le problème persiste vérifier l'installation de GDAL (cf. [Configurer GDAL/OGR](#)).

Chaîne de traitements

Les formats supportés

Ces formats sont supportés par GDAL/OGR qui est la librairie utilisée par les composants Talend. Tous formats supportés par GDAL/OGR devraient l'être également pour le job (cf. http://www.gdal.org/ogr/ogr_formats.html).

Les formats suivant ont été testés :

- Vecteur
 - ESRI Shapefile

- Mapinfo Tab
- Table PostGIS
- GeoDatabase
- GPX
- KML
- DXF
- Interlis
- WFS
- RASTER
 - TIFF
 - ASC
 - PDF
 - ECW
 - AIG
- Maps
 - QGS project

L'extraction des propriétés par analyse des fichiers

A partir de ces fichiers, les propriétés suivantes sont extraites :

- le format,
- l'étendue dans la projection du fichier et stockée dans les métadonnées en WGS84,
- la projection,
- le type de données (eg. point, ligne, polygone),
- un lien vers le fichier ou table analysé (ie. lien vers le fichier sur le réseau local ou URL de la base de données),
- le nombre d'objets,
- pour les données vecteurs
 - liste des colonnes et leur typage (eg. nombre, texte, date)

Format PostGIS : filtrage

Dans certaines configurations, il peut être nécessaire de filtrer uniquement les tables d'un schéma. Pour cela, il est nécessaire d'indiquer des options supplémentaires pour la connexion à la base de données :

- scan de toute la base de données : "PG:dbname=talend host=localhost port=5432 user=www-data password=www-data"
- scan d'un schéma : "PG:dbname=talend host=localhost port=5432 user=www-data password=www-data schemas=schema1"
- scan d'une liste de table : "PG:dbname=talend host=localhost port=5432 user=www-data password=www-data tables=table1,table2"

Pour plus d'information sur ces paramètres consulter :

http://www.gdal.org/drv_pg_advanced.html

Règles générales

Lors de la création des fiches de métadonnées, un identifiant est créé de la façon suivante :

- métadonnées de données <filename>#<layer_name>
- catalogue d'attribut <filename>#<layer_name>#iso19110

Cela permet :

- de faire des mises à jour si la métadonnée est déjà présente dans le catalogue
- de ne pas tenir compte du changement de répertoire d'un fichier

Note : une limitation est que si 2 fichiers portent le même nom dans 2 répertoires différents, l'identifiant généré sera le même.

Le jeu de test

Les données de test portent sur quelques villes. Chaque répertoire contient 4 types de données :

- Données vecteur Urban Atlas⁴ ((c) Directorate-General Enterprise and Industry (DG-ENTR), Directorate-General for Regional Policy) au format ESRI Shapefile
- Données vecteur OSM⁵ ((c) OpenStreetMap contributors) au format ESRI Shapefile
- Données RASTER Corine Land Cover ((c) European Environment Agency (EEA)) au format TIFF
- Données vecteur sur la ville de Rennes au format KML.

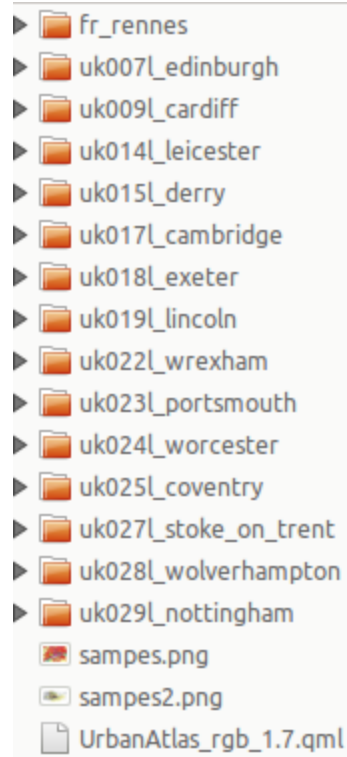
Vous pouvez également tester avec vos propres données !

Les villes disponibles sont les suivantes :

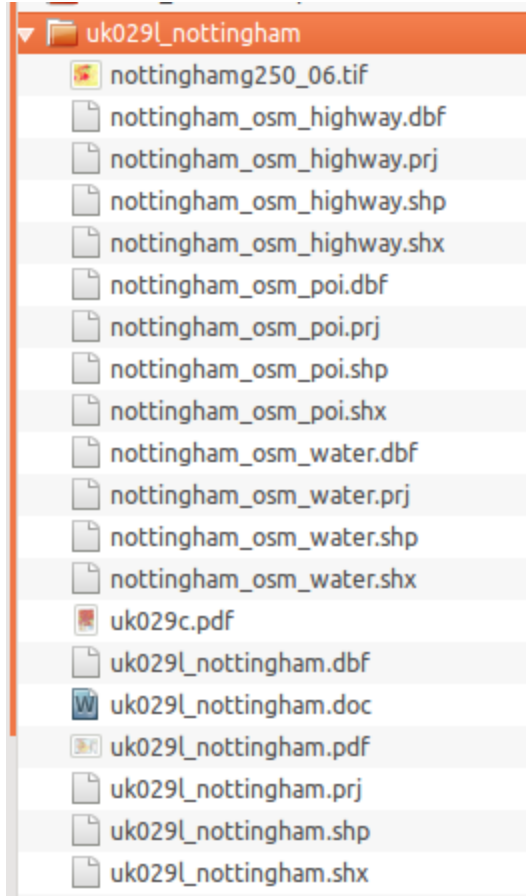
⁴

http://sdi.eea.europa.eu/catalogue/srv/eng/search?s_search=&s_E_any_OR_identifieur=urban%20atlas&hl=en

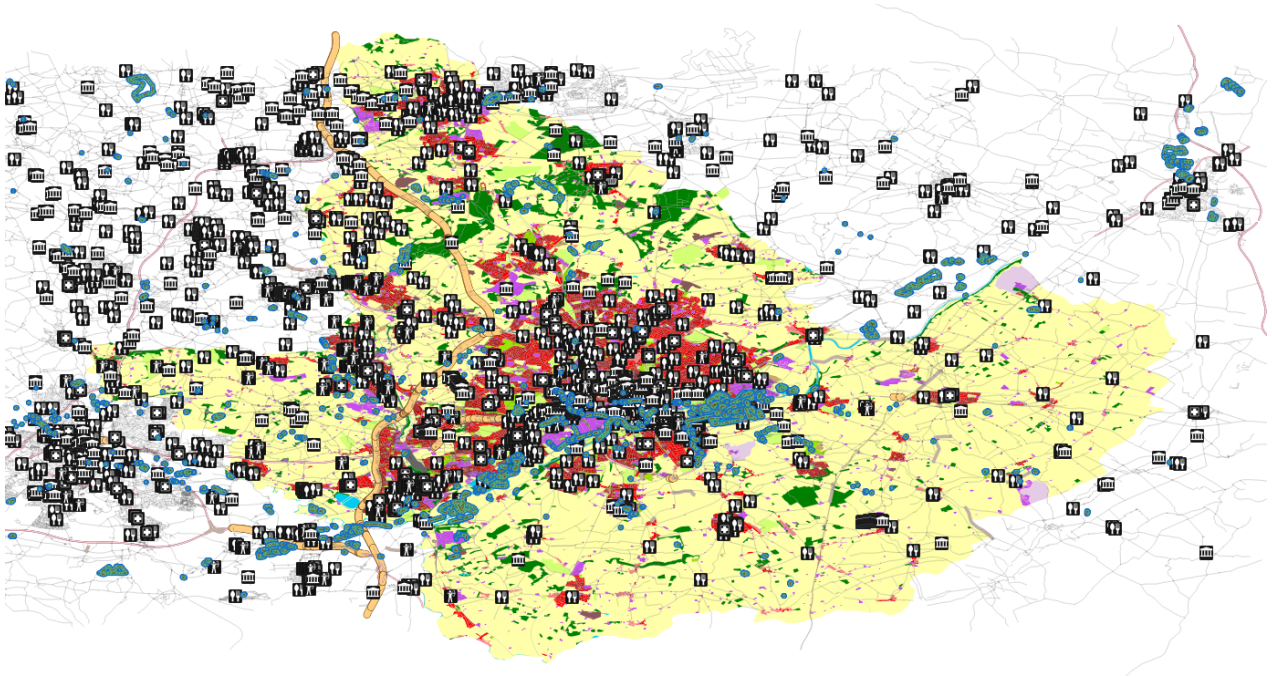
⁵ <http://www.openstreetmap.org>



Exemple de fichiers pour la ville de Nottingham :



Aperçu des données Urban Atlas et OSM :



Configurer les jobs

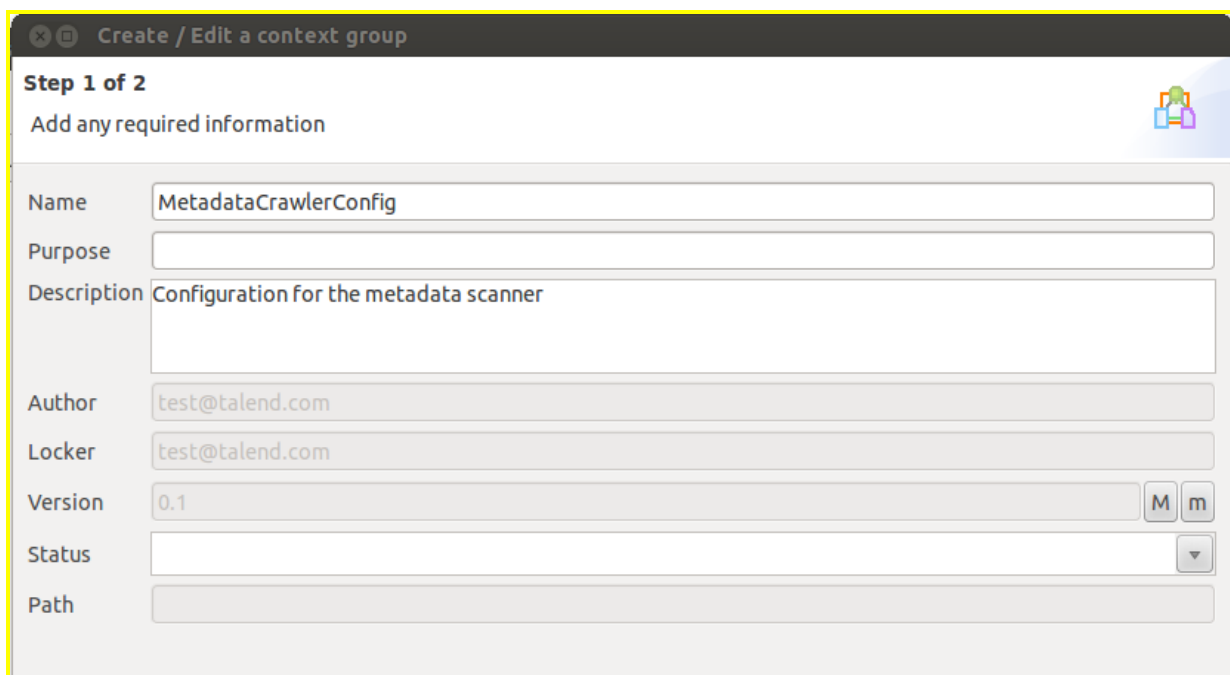
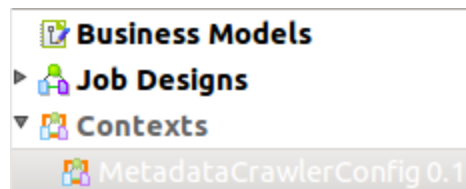
Répertoire de fichier de propriétés

Copier le répertoire “config” contenu dans le ZIP de l’espace de travail dans un répertoire (eg. “c:/geodata/inventaire/config”). Ce répertoire contient

- des fichiers de propriétés pour les jobs
- des scripts utilitaires pour le packaging des données

Paramètres du job

Dans le “Repository”, ouvrir le context “MetadataCrawlerConfig” pour définir les variables globales des jobs.

A screenshot of the 'Create / Edit a context group' dialog box, showing 'Step 1 of 2'. The dialog has a title bar with a close button and the text 'Create / Edit a context group'. Below the title bar, it says 'Step 1 of 2' and 'Add any required information'. The main area contains several input fields: 'Name' (filled with 'MetadataCrawlerConfig'), 'Purpose' (empty), 'Description' (filled with 'Configuration for the metadata scanner'), 'Author' (filled with 'test@talend.com'), 'Locker' (filled with 'test@talend.com'), 'Version' (filled with '0.1' and 'M' and 'm' buttons), 'Status' (empty with a dropdown arrow), and 'Path' (empty). There is a small icon of a folder with a plus sign in the top right corner of the dialog.

Cliquer sur suivant pour consulter la liste des variables.

Le context est un ensemble de variable utilisé par les jobs. Par exemple, le répertoire contenant les données à analyser (variable “data_dir”).

Dans la gestion du context, les 3 onglets ont pour fonction :

- Variables : liste des variables
- Values as tree : non utilisé
- Values as table : permet de définir les valeurs de chaque variable

Les variables sont les suivantes :

Step 2 of 2

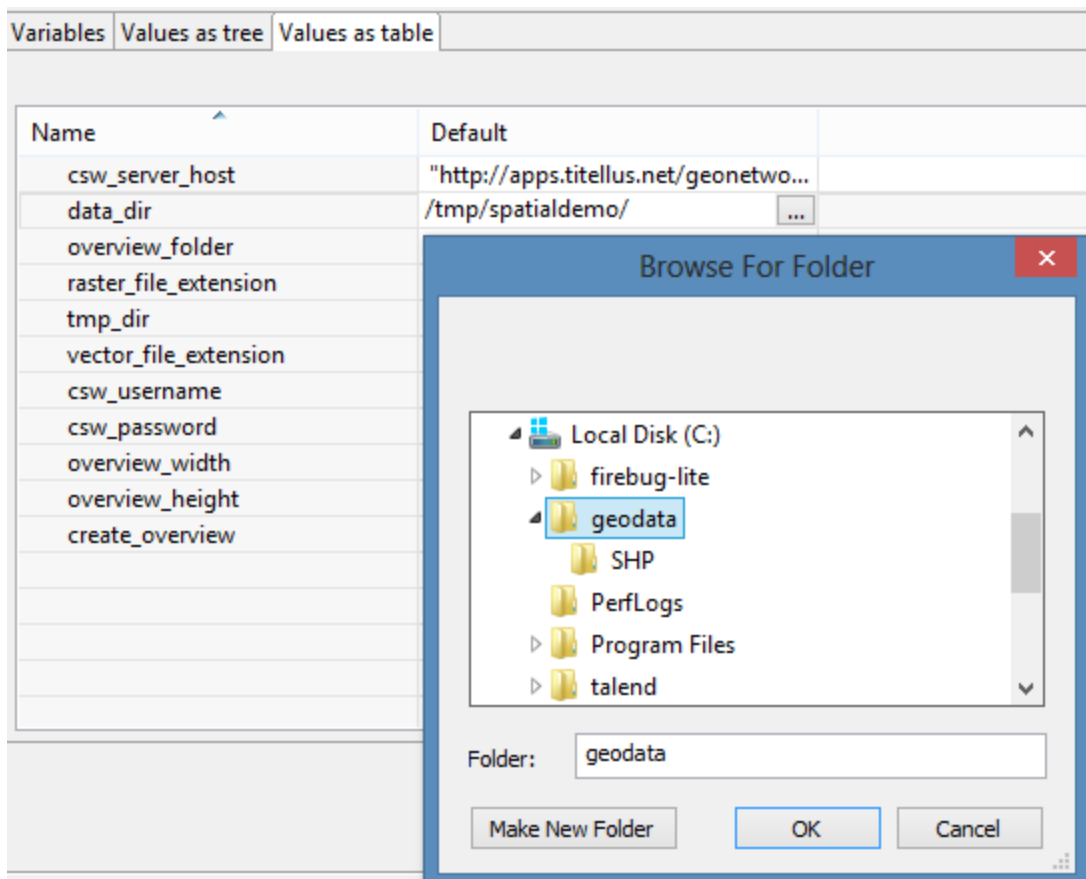
Define the contexts, variables and values



Variables	Values as tree	Values as table
Name	Type	Comment
create_overview	boolean Boolean	Create or not overview files
csw_password	String	The CSW user password
csw_server_host	String	The CSW server to push metadata records into
csw_username	String	The CSW user name
data_dir	Directory	The folder to scan
db_list	List Of Value	List of database to scan
default_projection	String	An EPSG code of a projection to use in case the projection could not be defined from the file
default_xmax	String	Default bounding box in case of failure extracting it from the file
default_xmin	String	Default bounding box in case of failure extracting it from the file
default_ymax	String	Default bounding box in case of failure extracting it from the file
default_ymin	String	Default bounding box in case of failure extracting it from the file
ftp_dir	String	The remove directory in the FTP server
ftp_host	String	The FTP host (if copy_mode is ftp)
ftp_password	String	The user password
ftp_port	int Integer	The FTP port (ie. probably 21)
ftp_username	String	The user name to log in the FTP server
geoserver_password	String	The GeoServer password
geoserver_server_host	String	The GeoServer node to publish data into using the REST API
geoserver_username	String	The GeoServer username
geoserver_namespace_prefix	String	The GeoServer namespace to publish data into
metadata_dir	Directory	The folder to create metadata records into
metadata_file_with_source	boolean Boolean	Store the generated metadata files and overviews next to the analyzed file
overview_copy_mode	String	Define the way to copy overview (possible values are ssh, ftp, copy)
overview_folder	Directory	The folder to copy overview into (if copy_mode is copy)
overview_height	String	The height of the overview to generate
overview_width	String	The width of the overview to generate
publish_metadata	boolean Boolean	Publish metadata or not in the CSW catalogue defined
raster_file_extension	String	RASTER file extensions
raster_file_extension_for_geopublication	String	RASTER file extensions supported for publication in mapservers
resource_dir	Directory	The folder with property files
ssh_dir	String	SSH remote directory
ssh_host	String	SSH host (if copy_mode is ssh)
ssh_password	String	SSH user password
ssh_port	int Integer	SSH port (ie. probably 22)
ssh_username	String	SSH user name
tmp_dir	Directory	The output folder
vector_file_extension	String	Vector file extensions
vector_file_extension_for_geopublication	String	Vector file extensions supported for publication in mapservers

A minima, il est nécessaire de configurer les variables suivantes :

- le répertoire des données : variable “data_dir”. Ce répertoire contient les données à analyser (eg. “c:/geodata/”). **Note : Les espaces ne sont pas supportés dans les chemins des fichiers.**
- le répertoire de travail : variable “tmp_dir”. Ce répertoire contient les fichiers temporaires (eg. c:/temp).
- le répertoire de ressource : variable “resource_dir”. Ce répertoire contient des fichiers de paramétrage (eg. “c:/geodata/inventaire/config”).
- la variable pour le positionnement des fichiers générés (imagettes et métadonnées) :
 - à côté des fichiers analysés. Dans ce cas définir la variable “metadata_file_with_source” avec la valeur “true”.
 - regroupés dans un même répertoire (Attention: cela suppose que les noms des fichiers analysés sont uniques). Dans ce cas définir la variable “metadata_file_with_source” avec la valeur “false” et définir le répertoire de génération (variable “metadata_dir”).



Les répertoires sont configurés en pointant vers le répertoire correspondant sur le système

Sous windows remplacer les “\” par des “/” si vous faites des copier/coller de chemin de répertoire.

Configuration des fichiers de propriétés

Dans le répertoire “config”, le fichier metadata-properties.csv contient les valeurs par défaut à ajouter aux métadonnées générées. Les descripteurs sont les suivants :

Nom du descripteur	Définition
type	Type de données analysées (vector ou raster)
mdIndividualName	Contact pour la métadonnée : Nom
mdOrganisationName	Contact pour la métadonnée : Organisation
mdPositionName	Contact pour la métadonnée : Rôle
mdVoice	Contact pour la métadonnée : Téléphone
mdDeliveryPoint	Contact pour la métadonnée : Adresse
mdCity	Contact pour la métadonnée : Ville
mdAdministrativeArea	Contact pour la métadonnée : Région
mdPostalCode	Contact pour la métadonnée : Code postal
mdCountry	Contact pour la métadonnée : Pays
mdEmail	Contact pour la métadonnée : Mèl
mdRole	Contact pour la métadonnée : Rôle
individualName	Contact pour la ressource : Nom
organisationName	Contact pour la ressource : Organisation
positionName	Contact pour la ressource : Rôle

voice	Contact pour la ressource : Téléphone
deliveryPoint	Contact pour la ressource : Adresse
city	Contact pour la ressource : Ville
adminArea	Contact pour la ressource : Région
postalCode	Contact pour la ressource : Code postal
country	Contact pour la ressource : Pays
email	Contact pour la ressource : Mèl
role	Contact pour la ressource : Rôle
abstractText	Résumé Utiliser %n pour insérer des retours à la ligne
overviewUrlPrefix	Les imageries ne sont pas dans la métadonnée. Pour qu'elles soient accessibles lors de la publication sur Internet, il est nécessaire de les copier dans un répertoire accessible par Internet.
keyword	Premier mot clé
keyword2	Second mot clé
geokeyword	Un mot clé de type géographique
accessConstraints	Contraintes d'accès. Valeurs possibles : copyright patent patentPending trademark license intellectualPropertyRights restricted otherRestrictions
useConstraints	Contraintes d'utilisation. Valeurs possibles : copyright patent patentPending trademark license intellectualPropertyRights restricted otherRestrictions

useConstraintsInfo	Information sur les contraintes légales
useLimitation	Information sur les conditions d'utilisation
scaleDenominator	Niveau d'échelle (eg. 50000)
language	Langue de la donnée
charset	Jeux de caractères
topicCategoryCode	Catégorie ISO
linkage1URL	URL du premier lien
linkage1protocol	Protocol du premier lien
linkage1name	Nom du premier lien
linkage1description	Description du premier lien
lanFileDesc	Description pour le lien vers le fichier sur le réseau local
lanFileprotocol	Protocol du lien pour le fichier sur le réseau local
wmsURL	URL du lien vers WMS
wmsProtocol	Protocol du lien vers WMS
wmsDesc	Description du lien vers WMS

Les descripteurs peuvent contenir des valeurs nulles.

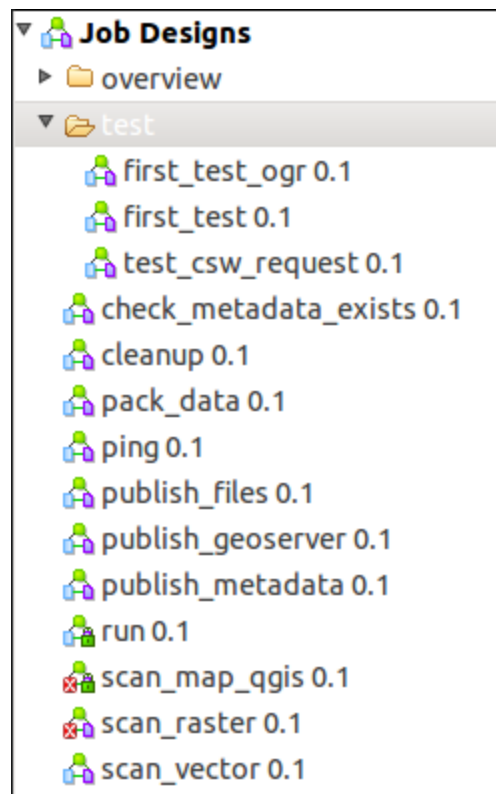
Autres variables :

- `uuidCreatedFromFilePath` : L'identifiant de la métadonnée est calculé sur la base du nom du fichier ou du chemin vers le fichier. Par exemple si le chemin vers le fichier est `E:\DATA\3_ORTHO\_ORTHOPHOTO_SION\10CM\1233.tiff`, l'uuid de la fiche peut être créé à partir de "1233.tiff" (cela sous entend que le nom des fichiers est uniques dans l'arborescence à analyser) ou sur le chemin `"3_ORTHO\_ORTHOPHOTO_SION\10CM\1233.tiff"` après le `data_dir`.

Lancement des jobs depuis le Studio

Cette section décrit les jobs, le traitement et le résultat attendu.

Les jobs mis en place sont les suivants :

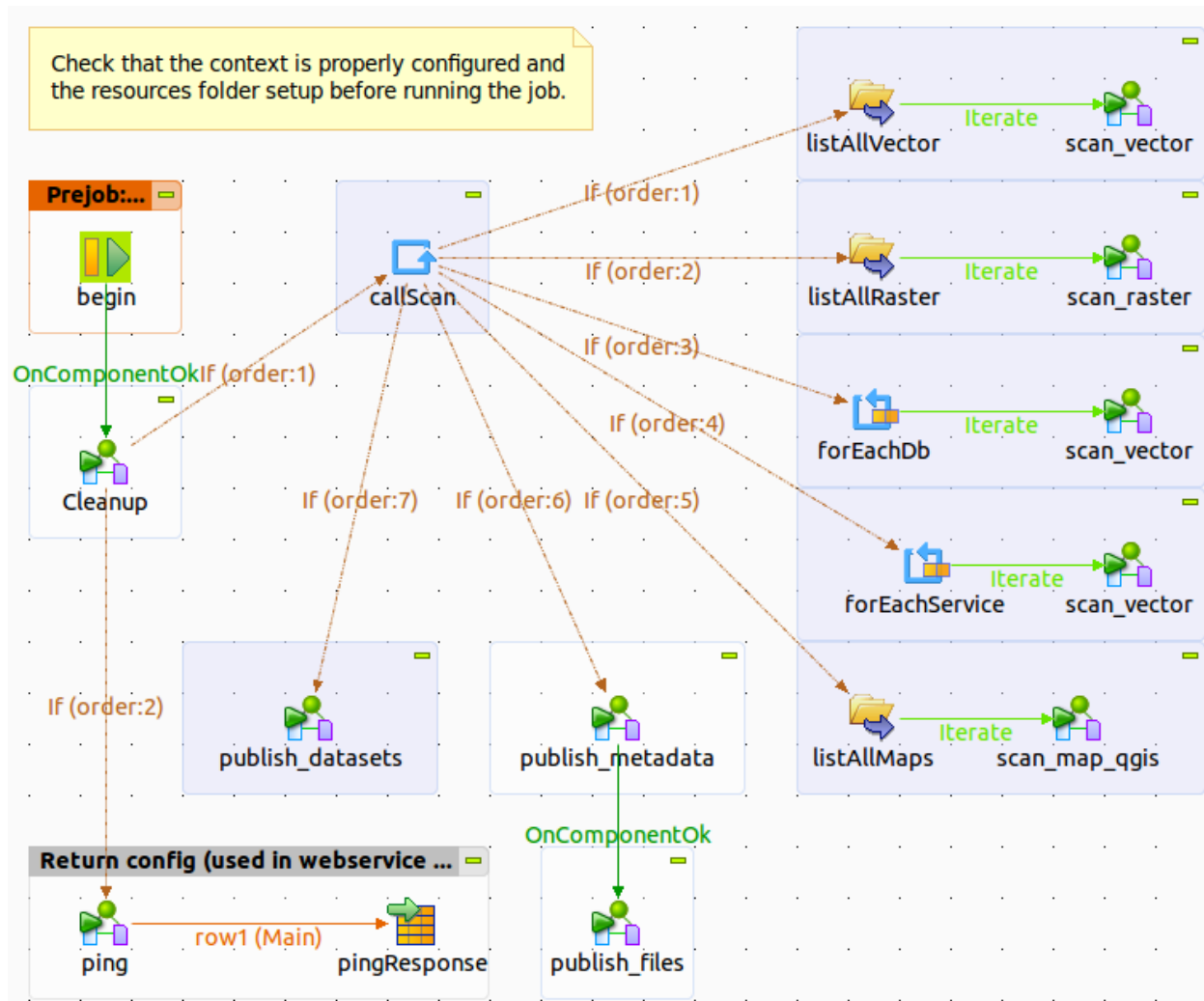


Job	Description
run	Job principal s'occupant de l'enchainement des différents jobs
scan_raster	Analyse un fichier raster
scan_vector	Analyse un fichier vecteur (ie. fichier, table, couche dans un service)
scan_map_qgis	Analyse une carte QGIS
overview/generateRaster Overview	Génération des imageries raster
overview/generateVector Overview	Génération des imageries vecteur
pack_data	A partir des données source, génère un fichier ZIP.
publish_metadata	Publie les métadonnées dans un catalogue via le protocole CSW

publish_geoserver	(Expérimental) Publie les données en WMS dans GeoServer via l'API REST
publish_files	Publie des fichiers (eg. données zippées ou imageries) dans un répertoire ou par copie FTP ou par copie SSH
check_metadata_exists	Vérifie si une métadonnée est déjà publiée dans le catalog distant à l'aide d'une requête CSW
Job de tests	
test/first_test	Vérifie que le module Spatial est correctement installé
test/first_test_ogr	Vérifie que la librairie GDAL/OGR est correctement configurée
test/test_csw_request	Teste une transaction CSW

Job “run”

Le job “run” s’occupe de rechercher les fichiers vecteur, raster, les tables des bases de données, les couches d’un service ou les cartes QGIS puis de lancer l’analyse. Une fois l’analyse faite, les données sont éventuellement packagées puis les métadonnées sont publiées dans le catalogue.



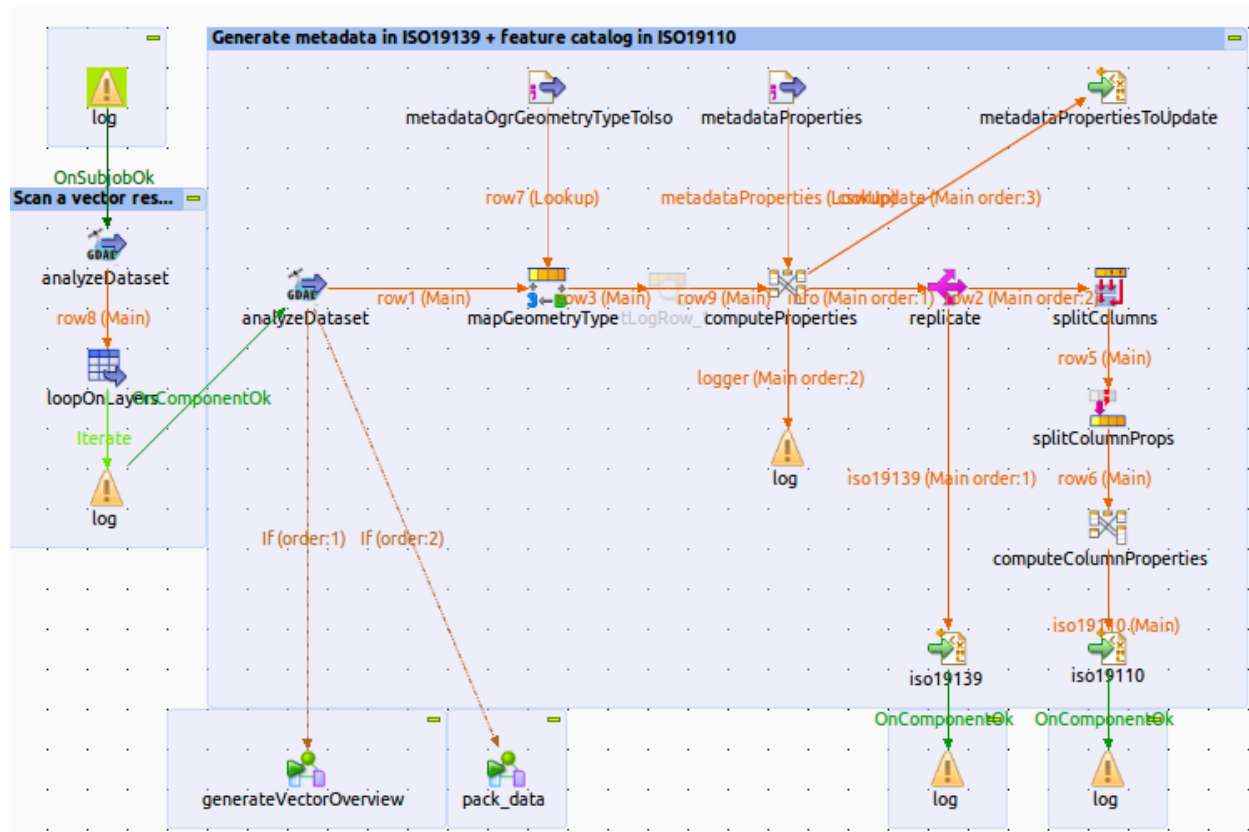
Paramètres :

Name	Default
▼ MetadataCrawlerConfig	/tmp/spatialdemo//<"PG:dbname=talend host=localhost port=5432 user=www-data passwo
data_dir	/tmp/spatialdemo/
db_list	<"PG:dbname=talend host=localhost port=5432 user=www-data password=www-data">
publish_metadata	true
raster_file_extension	".*\\.tif .\\.tiff .\\.ecw .\\.asc .\\.pdf"
vector_file_extension	".*\\.kml .\\.shp .\\.mif .\\.tab .\\.gml .\\.dxf"

- Le paramètre "publish_metadata" du contexte global permet d'activer ou pas la publication dans le catalogue
- Le paramètre "pack_data" du contexte global permet d'activer ou pas le packaging des données
- Le paramètre "pack_data_publish" permet d'activer ou pas la publication des données dans un répertoire pour la diffusion
- Le paramètre "db_list" contient une liste de base de données PostGIS à analyser
- Le paramètre "service_list" contient une liste des services WFS à analyser

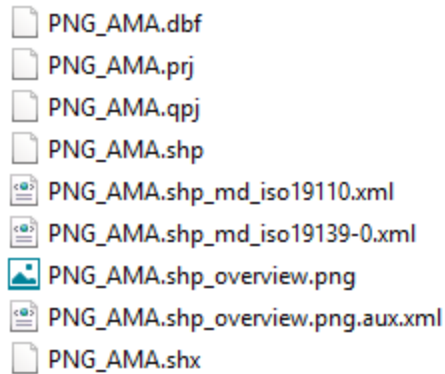
Job “scan_vector”

Ce job analyse un fichier SIG ou une base de données PostGIS.



Pour chaque fichier analysé, les fichiers suivants sont créés :

- <nom_du_fichier>_md_iso19139-<#>.xml : métadonnée de donnée. Une fiche par couche contenu dans la ressource. Pour un shapefile, une seule fiche est créée.
- <nom_du_fichier>_md_iso19110.xml : catalogue d'attributs
- <nom_du_fichier>_overview.png : Imagerie au format PNG
- <nom_du_fichier>.zip : Un ZIP contenant les données analysées (pour les tables PostGIS, les données sont converties en Shapefile).
- <nom_du_fichier>.zip.md5 : somme de contrôle pour valider le téléchargement du ZIP



Exemple pour un Shapefile

Paramètres :

Name	Default
file	/tmp/spatialdemo/SHP/PAG_AOA.shp
▼ MetadataCrawlerConfig	true//tmp/spatialdemo///usr/local/App/talend/TOS_DI-r104014-V5.3.1/workspace/METADATA_
create_overview	true
data_dir	/tmp/spatialdemo/
resource_dir	/usr/local/App/talend/TOS_DI-r104014-V5.3.1/workspace/METADATA_CRAWLER/_resources/
tmp_dir	/tmp/csw/
vector_file_extension	".*\\.kml .\\.shp .\\.mif .\\.tab .\\.gml .\\.dxf"

Pour filtrer sur les tables ou un schéma, ajouter les paramètres suivant dans la chaîne de connexion⁶ (cf. [Format PostGIS : filtrage](#)) :

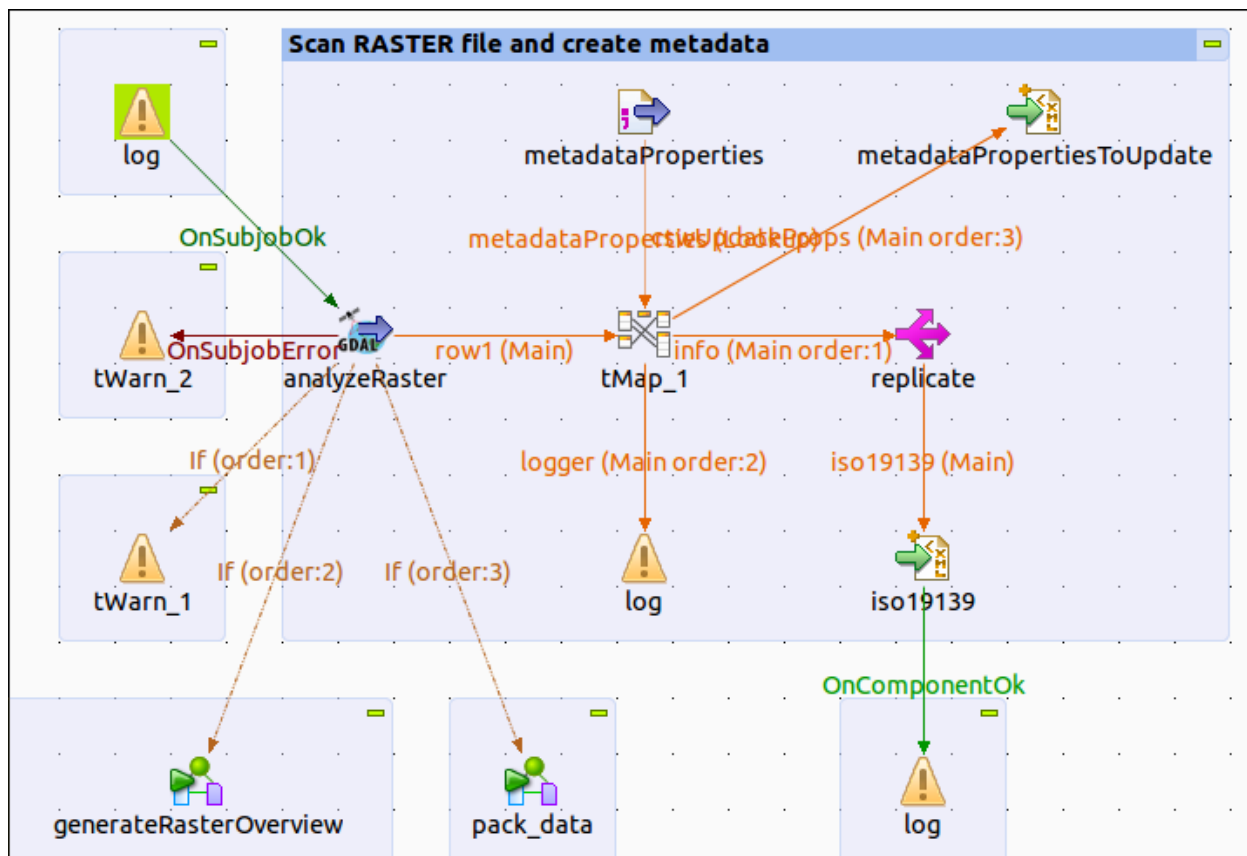
- tables=table1,table2'
- schemas=schema1,schema2

Un fichier de log listOfLayersOutOfExtent.csv est généré pour les couches en dehors de l'emprise de vérification.

Job "scan_raster"

Ce job s'occupe de générer une métadonnée de données et une imagerie pour le fichier raster analysé.

⁶ http://www.gdal.org/ogr/drv_pg_advanced.html



Pour chaque fichier analyser, les fichiers suivants sont créés :

- <nom_du_fichier>_md_iso19139-<#>.xml : métadonnée de donnée. Une fiche par couche contenu dans la ressource. Pour un geotiff, une seule fiche est créée.
- <nom_du_fichier>_overview.png : Imagette au format PNG
- <nom_du_fichier>.zip : Un ZIP contenant les données analysées
- <nom_du_fichier>.zip.md5 : somme de contrôle pour valider le téléchargement du ZIP

 geotif_MNT_06.prj
 geotif_MNT_06.tfw
 geotif_MNT_06.tif
 geotif_MNT_06.tif_md_iso19139-0.xml

Exemple pour un geotiff

Job “scan_map_qgis”

Ce job analyse des projets QGIS (fichier QGS) afin de créer des métadonnées de cartes.

Pour activer l’analyse des cartes, modifier le context et modifier les paramètres :

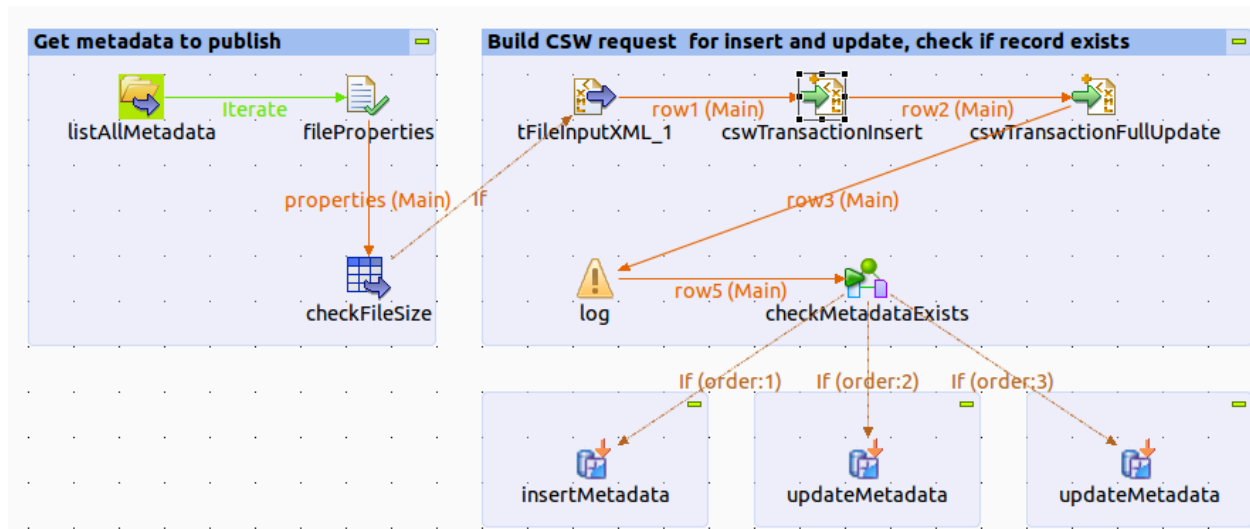
- scan_map : true

- scan_map_and_datasource : false pour ne pas créer les métadonnées pour les données utilisées dans la carte, true sinon
- map_data_dir : répertoire contenant les projets QGIS

Note : il est envisageable d'analyser des cartes ArcGIS en utilisant <https://github.com/titellus/Mxd2Qgs> pour la conversion mxd vers qgs.

Job “publish_metadata”

Ce job publie les métadonnées générées dans un catalogue en utilisant le protocole CSW.



Le job recherche les métadonnées dans le répertoire de génération, vérifie la présence de la fiche dans le catalogue et joue une transaction CSW afin d'insérer ou de mettre à jour l'enregistrement.

Job “pack_data”

Ce job s'occupe de préparer les données au format ZIP pour une éventuelle publication dans un répertoire accessible sur Internet.

Les paramètres du job sont les suivants :

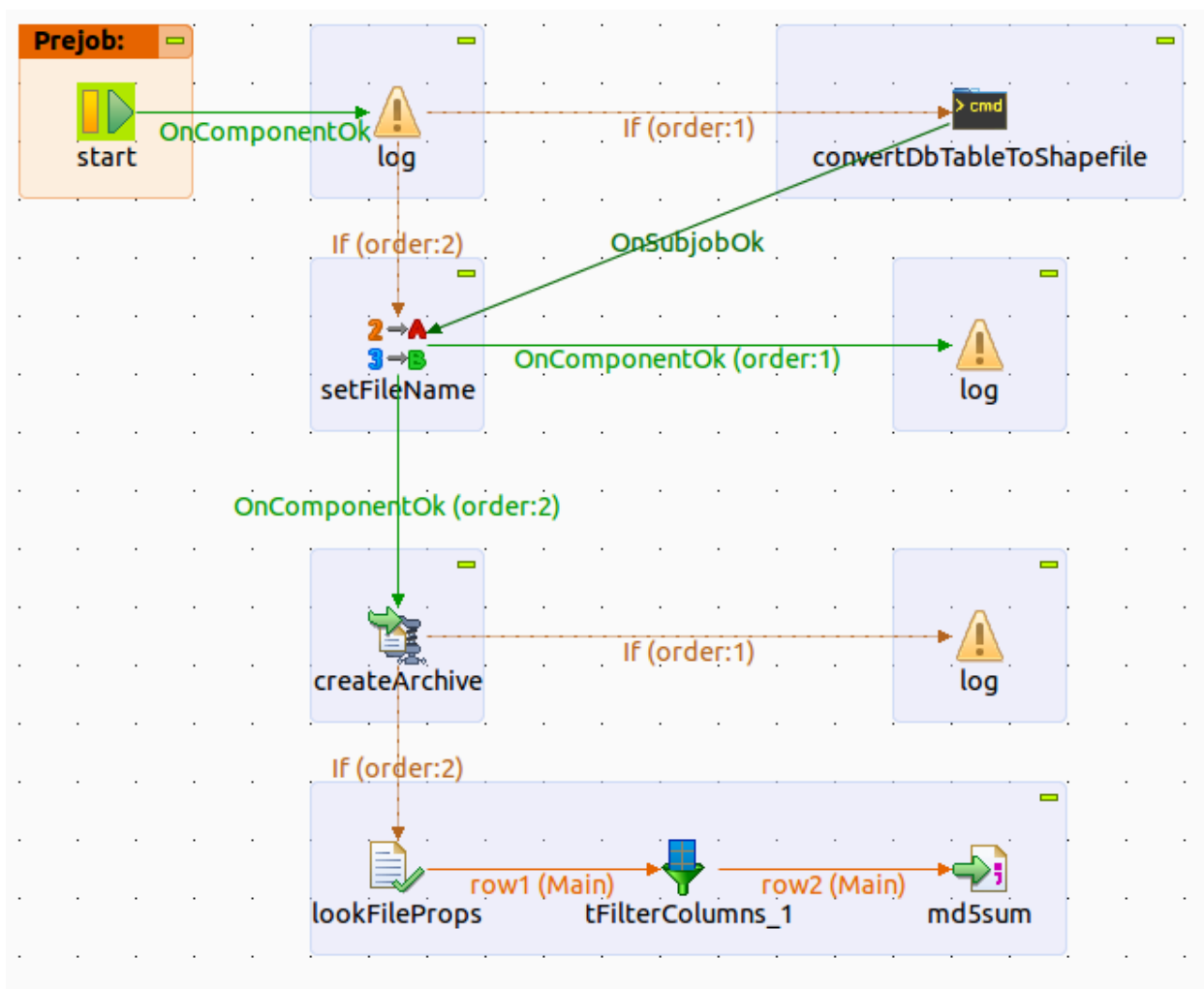
1	fileToPack	File	PG:dbname=talend host=localhost port=5432 us
2	layerName	String	"country"
3	MetadataCrawlerConfig (from repository context)		
4	pack_data_dir	Directory	/tmp/talend/down
5	resource_dir	Directory	/home/francois/data/tesl
6	pack_data_create_checksum	boolean Boolean	true
7	log_dir	Directory	/tmp/talend/loc

- fileToPack : Nom du fichier à packager (ou URL de la base de données)

- layerName : Nom de la couche à traiter
- pack_data_dir : Répertoire où placer les données générées
- resource_dir : Répertoire des ressources où les scripts de conversion convert_db_to_shapefile sont placés
- pack_data_create_checksum : Création de la somme de contrôle ou pas
- log_dir : Répertoire pour les logs

Nore : Les tables PostGIS sont converties en Shapefile par le composant convertDbTableToShapefile qui utilise les scripts du répertoire resource convert_db_to_shapefile.sh ou convert_db_to_shapefile.bat.

Le processus est le suivant :



L'ensemble des variables de configuration du packaging des données sont les suivantes :

54	pack_data	boolean Boolean ▾	true
55	pack_data_dir	Directory ▾	/tmp/talend/download/
56	pack_data_create_checksum	boolean Boolean ▾	true
57	pack_data_publication_dir	Directory ▾	/var/www/html/data/
58	overview_url_prefix	String ▾	"http://localhost/overview/"
59	pack_data_url_prefix	String ▾	"http://localhost/data/"
60	pack_data_link_template	String ▾	"Download data in {{format}} format"
61	pack_data_publish	boolean Boolean ▾	true

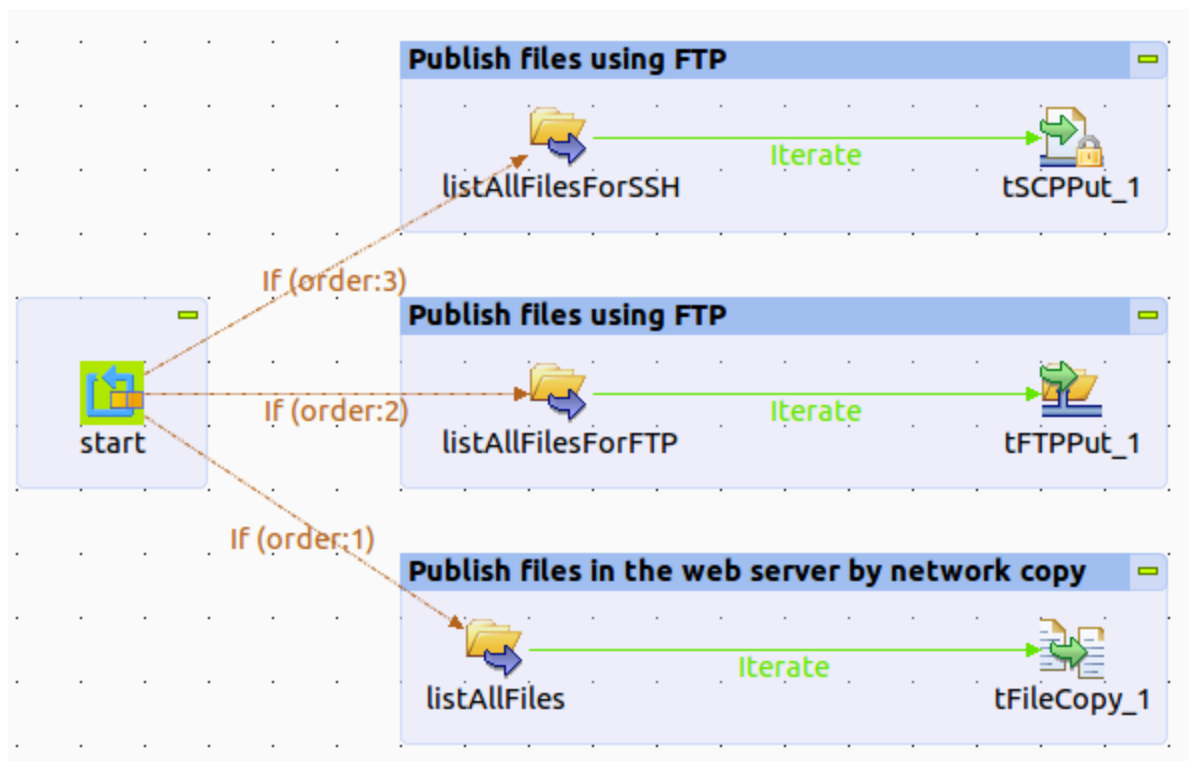
Job “publish_files”

Ce job publie les données zippées ou les imagerie dans un répertoire de la machine locale ou sur un serveur distant via SSH ou FTP. Le but est de rendre accessible les imagerie sur le web afin que les références présentent dans les fiches soient résolues (cf. configuration du fichier metadata-properties.csv, variable “overviewUrlPrefix”).

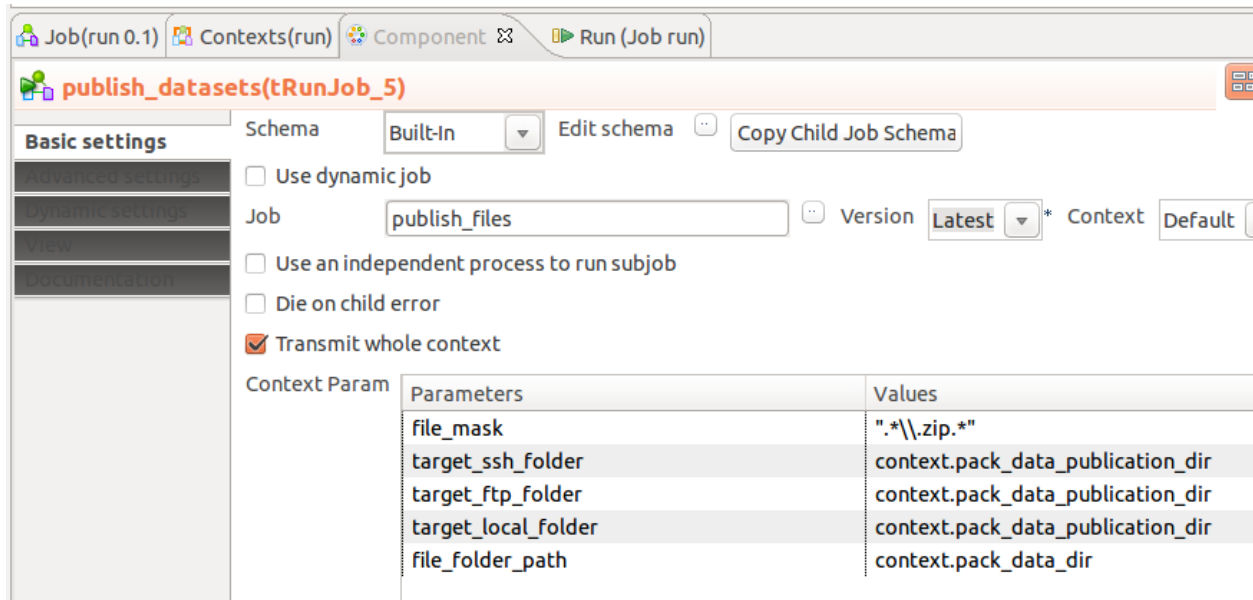
3 modes de copie sont disponibles. Définir la variable overview_copy_mode avec une des 3 valeurs possibles :

- ssh
- ftp
- copy

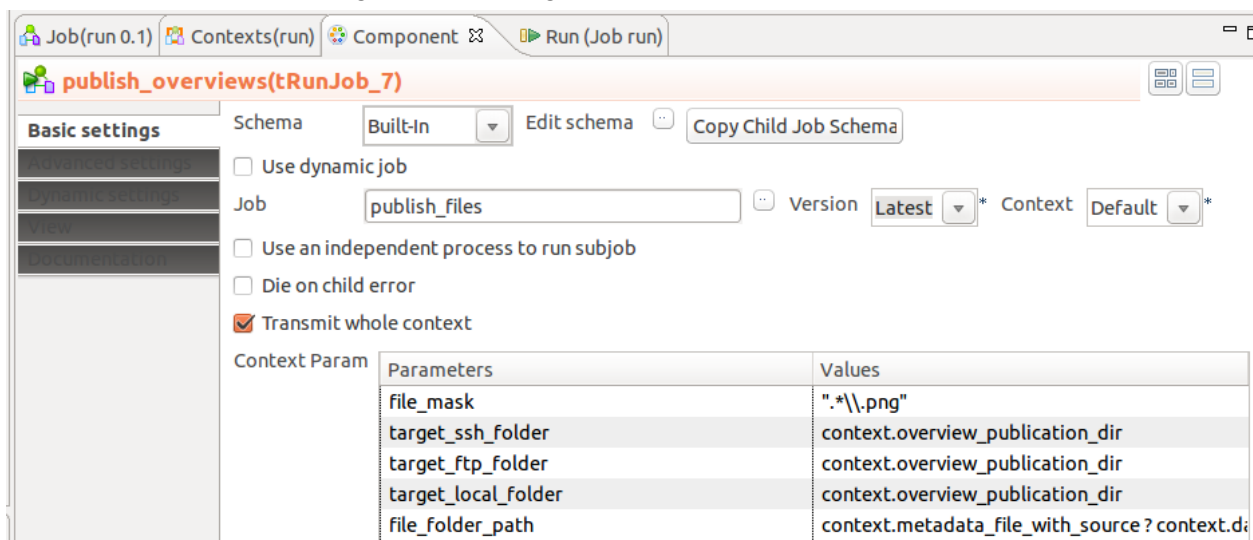
Ensuite configurer les variables de connexions pour les serveurs de type SSH ou FTP.



Pour la publication des données le composant est configuré de la manière suivante :



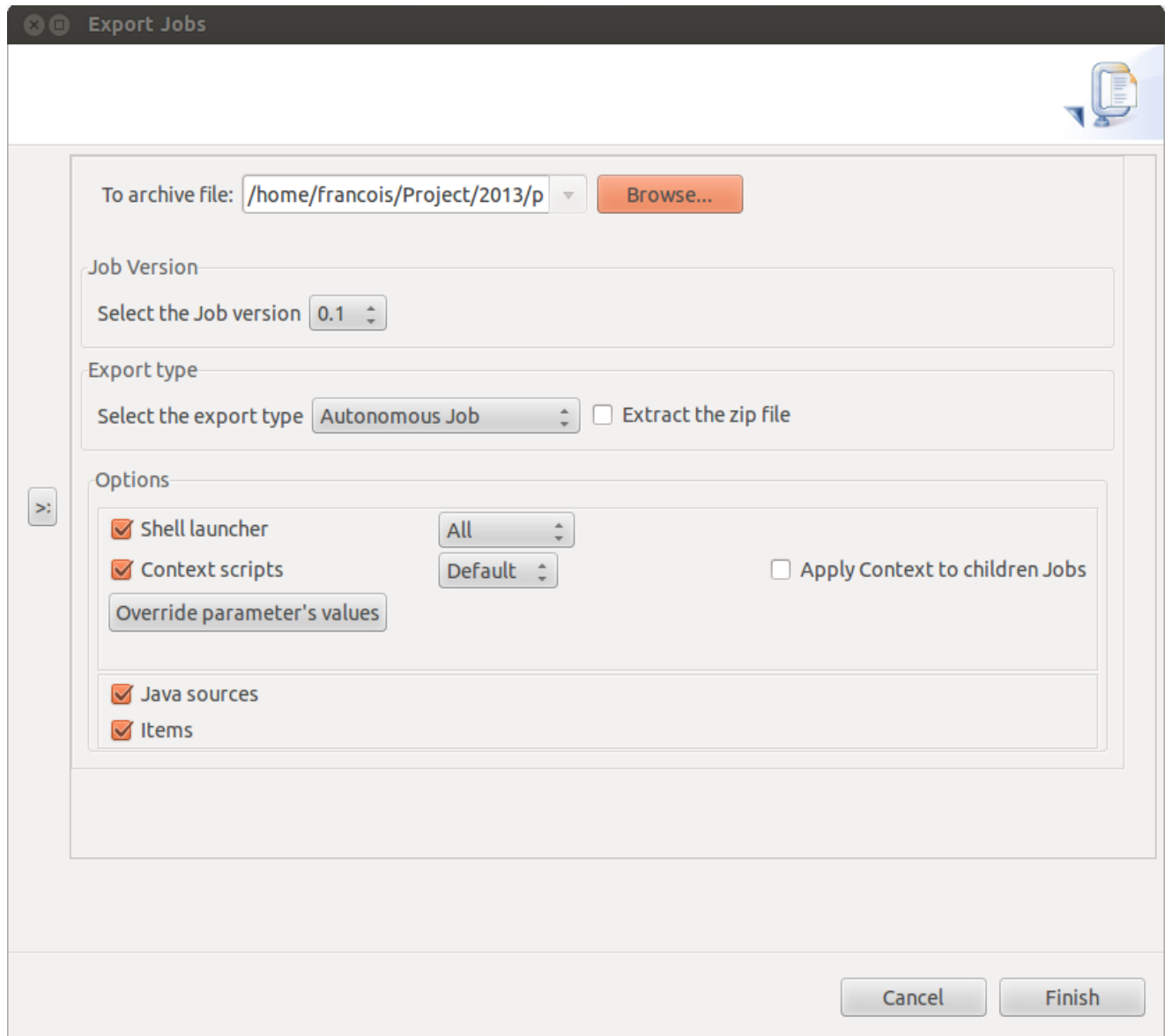
Pour la publication des imagerie, la configuration est la suivante :



Ajuster les variables pack_data_publication_dir et overview_publication_dir en fonction du répertoire dans lequel les fichiers doivent être copiés.

Export du job

Une fois le job opérationnel dans le studio, il est possible de l'exporter pour le lancer de manière autonome. Pour cela, cliquer sur le job "run" est faite "exporter".

The image shows a software window titled "Export Jobs". It contains several sections for configuring an export. At the top, there's a "To archive file:" label followed by a text field containing "/home/francois/Project/2013/p" and a "Browse..." button. Below this is a "Job Version" section with a label "Select the Job version" and a spinner box set to "0.1". The "Export type" section has a label "Select the export type" with a dropdown menu showing "Autonomous Job" and an unchecked checkbox labeled "Extract the zip file". An "Options" section is on the left, indicated by a right-pointing arrow button. It contains two checked checkboxes: "Shell launcher" and "Context scripts". Next to "Shell launcher" is a dropdown menu set to "All", and next to "Context scripts" is a dropdown menu set to "Default". To the right of these is an unchecked checkbox labeled "Apply Context to children Jobs". Below these is a button labeled "Override parameter's values". At the bottom of the "Options" section are two more checked checkboxes: "Java sources" and "Items". At the very bottom of the window are "Cancel" and "Finish" buttons.

Export Jobs

To archive file:

Job Version

Select the Job version

Export type

Select the export type ☐ Extract the zip file

Options

☒ Shell launcher

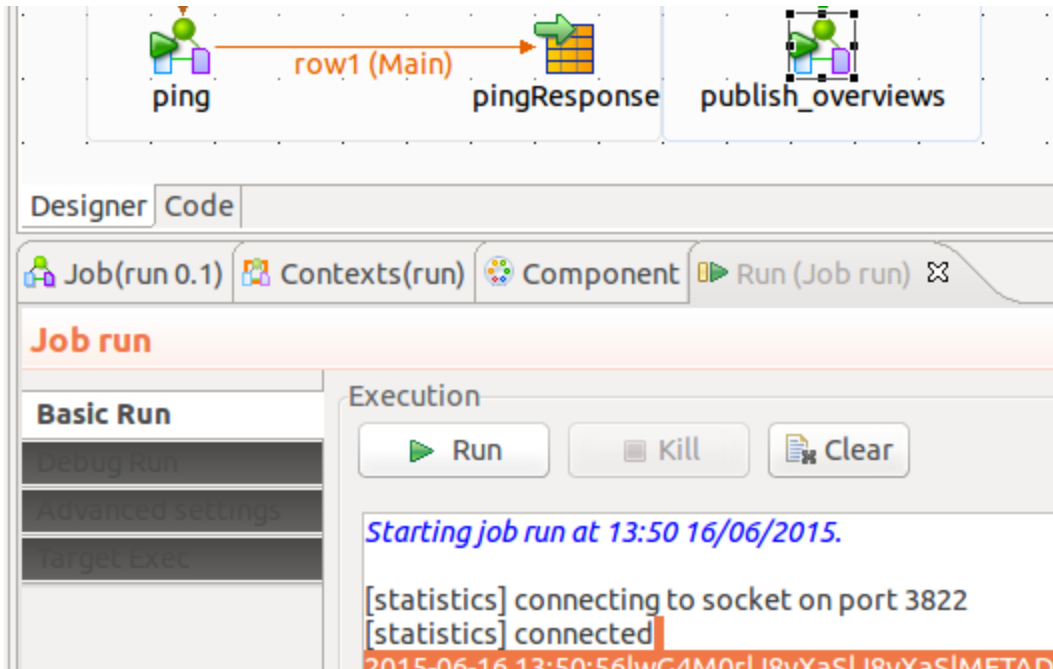
☒ Context scripts ☐ Apply Context to children Jobs

☒ Java sources

☒ Items

Lancer le job

Lancer le job à partir de l'onglet "Run" ou cliquer sur F6.



Lire le fichier journal

Afin de mieux comprendre le déroulement du job, des informations sont redirigées vers le fichier journal (ou log) :

```
2015-06-16
13:50:56|wG4M0r|J8yXaS|J8yXaS|METADATA_CRAWLER|scan_vector|Default|3|tWarn|tWarn_2|#
Analyzing datasource : /home/francois/data/test/toscan/CCM.aa.shp|0
### Une source de données est identifiée

2015-06-16
13:50:56|wG4M0r|J8yXaS|J8yXaS|METADATA_CRAWLER|scan_vector|Default|3|tWarn|tWarn_3|##
Analyzing layer: CCM.aa|0
### Une couche est identifiée dans la source de données

2015-06-16
13:50:57|wG4M0r|J8yXaS|J8yXaS|METADATA_CRAWLER|scan_vector|Default|3|tWarn|tWarn_1|Dataset
metadata record and feature catalog created for layer : CCM.aa from files :
/home/francois/data/test/toscan/CCM.aa.shp (driver ESRI Shapefile)|0
### Les fiches sont créées

2015-06-16
13:50:57|MjQHMo|J8yXaS|wG4M0r|METADATA_CRAWLER|generateVectorOverview|Default|3|tWarn|tWarn_
3|## Creating overview for : /home/francois/data/test/toscan/CCM.aa.shp|0
### Début de la création de l'imagette

Creating overview for /home/francois/data/test/toscan/CCM.aa.shp (layer -1 CCM.aa) ...
0...10...20...30...40...50...60...70...80...90...100 - done.
Input file size is 800, 800
```

```

0...10...20...30...40...50...60...70...80...90...100 - done.
Done.
2015-06-16
13:50:57|MjQHMo|J8yXaS|wG4M0r|METADATA_CRAWLER|generateVectorOverview|Default|3|tWarn|tWarn_4|## Overview created : /tmp/talend/metadata/CCM.aa.shp_overview.png|0
### L'imagette est créée.

2015-06-16
13:50:57|x9az4P|J8yXaS|wG4M0r|METADATA_CRAWLER|pack_data|Default|3|tWarn|tWarn_1|Packing
data: /home/francois/data/test/toscan/CCM.aa.shp layers: CCM.aa...|0
### Début de la création du ZIP contenant les données

2015-06-16
13:50:57|x9az4P|J8yXaS|wG4M0r|METADATA_CRAWLER|pack_data|Default|3|tWarn|tWarn_2|Creating
archive: CCM.aa.shp.zip in /tmp/talend/download/ (all files matching: 'CCM.aa.*' in
'/home/francois/data/test/toscan' will be included in the archive) ...|0
2015-06-16
### ZIP créé.

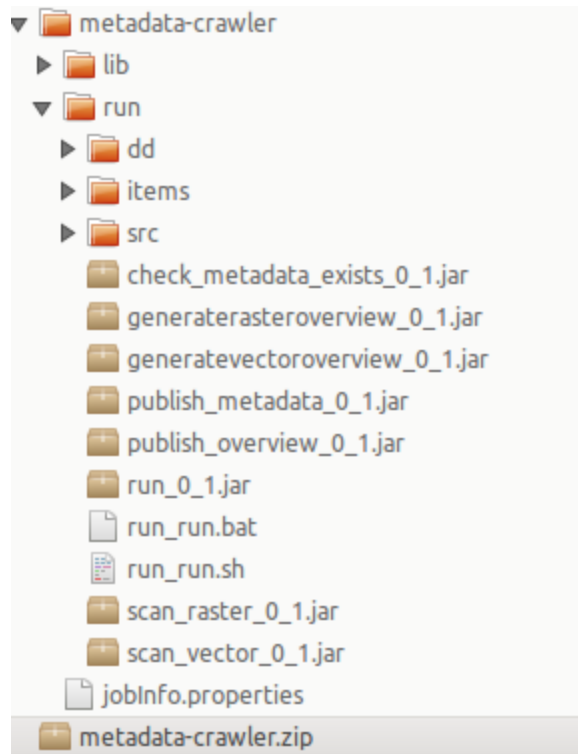
13:50:58|wG4M0r|J8yXaS|J8yXaS|METADATA_CRAWLER|scan_vector|Default|3|tWarn|tWarn_4|###
Metadata record /tmp/talend/metadata/CCM.aa.shp_md_iso19139-.xml created for : CCM.aa|0
2015-06-16
13:50:58|wG4M0r|J8yXaS|J8yXaS|METADATA_CRAWLER|scan_vector|Default|3|tWarn|tWarn_5|###
Metadata record for feature catalog /tmp/talend/metadata/CCM.aa.shp_md_iso19110.xml created
for : CCM.aa|0
### Fiche créée.

2015-06-16
13:50:58|ICbvq9|J8yXaS|J8yXaS|METADATA_CRAWLER|scan_raster|Default|3|tWarn|tWarn_4|#
Analyzing raster datasource : /home/francois/data/test/toscan/ebk1km1.tif|0
### ... suivant ...

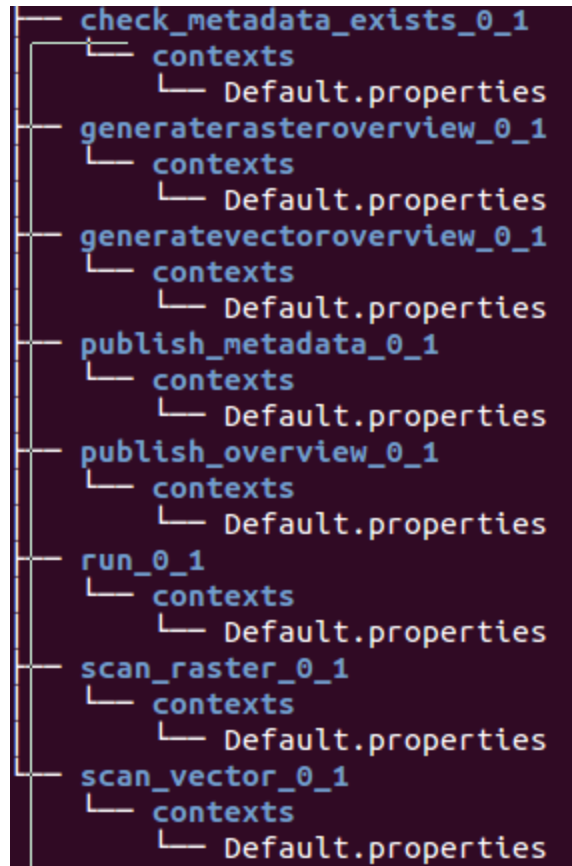
```

Lancement des jobs en mode autonome

Une fois exporté, le job se présente sous la forme d'un fichier ZIP. Une fois dézippé, il est possible de l'exécuter avec le fichier run_run.bat sous Windows ou run_run.sh sous Linux.



Il est également possible de modifier les variables pour le job autonome. Pour cela, aller dans le répertoire portant le nom du workspace et modifier les fichiers properties.



Sinon, il est également possible de modifier les paramètres du contexte via la commande de lancement du job :

```
# Pour scanner le répertoire /tmp
./run_run.sh --context_param data_dir=/tmp

# Pour scanner le répertoire
/home/francois/Desktop/foss4g-talend-spatial et publier dans le
catalogue http://localhost:8080/geonetwork/srv/eng/csw-publication
./run_run.sh --context_param
data_dir=/home/francois/Desktop/foss4g-talend-spatial
--context_param
csw_server_host=http://localhost:8080/geonetwork/srv/eng/csw-public
ation --context_param csw_username=admin --context_param
csw_password=admin
```


Video: <http://www.youtube.com/watch?v=RXq4gwVRE9Q>

Lancer le job en tant que service web

Exporter le job au format WAR. Déployer le WAR dans un container Java (eg. Tomcat).

Une fois déployé, l'URL suivant permet de lancer le job :

http://localhost:8080/run_0.1/services/run?method=runJob

Pour vérifier la configuration, il est possible d'utiliser le paramètre ping :

http://localhost:8080/scan_0.1/services/run?method=runJob&arg1=--context_param%20ping=true

Le service retourne alors la configuration actuelle :

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<soapenv:Body>
<runJobReturn xmlns="http://talend.org" xmlns:ns1="http://talend.org"
xsi:type="ns1:runJobReturn">
<ns1:item xsi:type="ns1:ArrayOf_xsd_string">
<ns1:item xsi:type="xsd:string">
{"metadata_dir": "/tmp/talend/metadata/", "data_dir":
"/foss4g-talend-spatial/dataren/uk0071_edinburgh/", "resource_dir":
"/foss4g-talend-spatial/resources/", "log_dir": "/tmp/talend/logs/", "tmp_dir":
"/tmp/talend/tmp/", "gdal_path": "", "scan_raster": "true", "raster_file_extension":
".*\\.tif|.*\\.tiff|.*\\.ecw|.*\\.asc", "scan_vector":
".*\\.kml|.*\\.shp|.*\\.mif|.*\\.tab|.*\\.gml|.*\\.dxf", "db_list": "PG:dbname=talend
host=localhost port=5432 user=www-data password=www-data", "service_list":
"<http://services.sandre.eaufrance.fr/geo/ouvrage?SERVICE=WFS>", "csw_server_host":
"http://localhost:8080/geonetwork/srv/fre/csw-publication", "csw_username":
"admin", "csw_password": "admin"}
</ns1:item>
</ns1:item>
</runJobReturn>
</soapenv:Body>
</soapenv:Envelope>
```

Si la configuration n'est pas bonne, il est possible de modifier le job dans le studio puis

d'exporter le job ou d'utiliser les paramètres de l'URL pour définir le Context dynamiquement :

http://localhost:8080/run_0.1/services/run?method=runJob&arg1=--context_param%20data_dir%3D%2Fdata%2Ffolder (%20 correspond à " ", %3D correspond à "=" et %2F correspond à "/")



This XML file does not appear to have any style information associated with it. The d

```
▼<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  ▼<soapenv:Body>
    ▼<runJobReturn xmlns="http://talend.org" xmlns:ns1="http://talend.org/talend-core-1.0"
      ▼<ns1:item xsi:type="ns1:ArrayOf_xsd_string">
        <ns1:item xsi:type="xsd:string">0</ns1:item>
      </ns1:item>
    </runJobReturn>
  </soapenv:Body>
</soapenv:Envelope>
```

FAQ

- **Comment modifier le catalogue à alimenter ?**

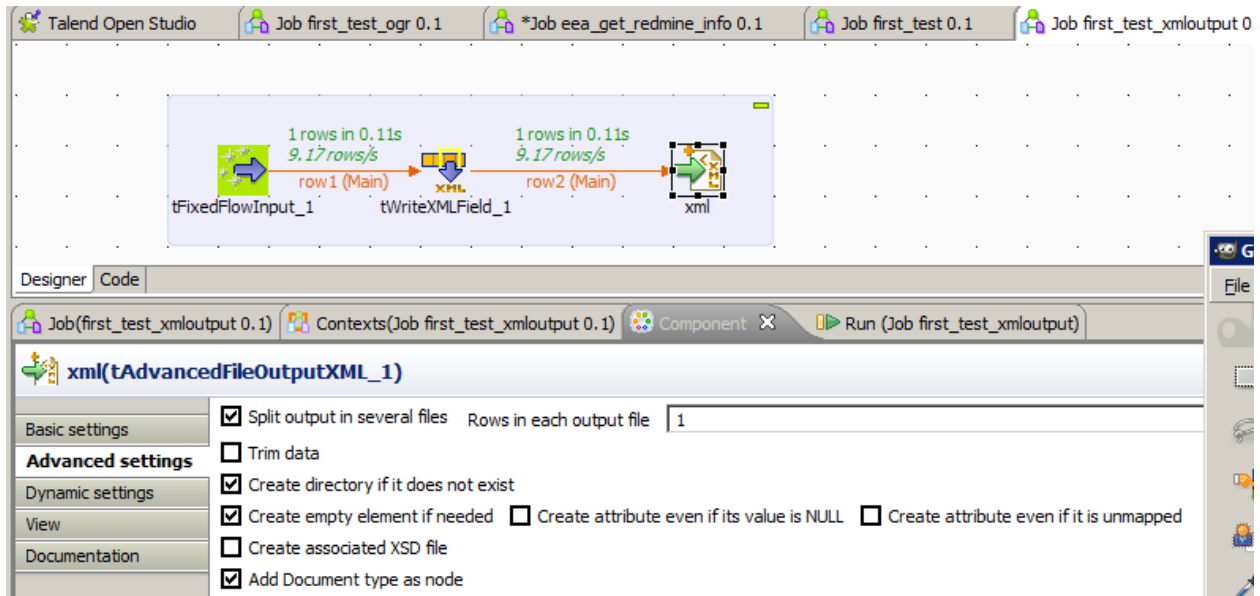
Dans le contexte, éditer les variables globales et modifier la variable csw_server_url.

- Les fiches générées font 0Ko ?

TODO

- **Les fichiers générés pour les requetes CSW sont invalides ? Les fiches sont bien générées mais le catalogue ne semble pas importer les fiches ?**

Dans certain cas les fichiers contenant les requetes CSW sont incomplets. Lancer le job first_test_xmloutput. Verifier le parametrage du composant tAdvancedFileOutputXML. Dans les parametres avances, l'option "Add Document type as node" doit etre disponible et cocher.



Ensuite, verifier la sortie dans le repertoire tmp_dir. Le fichier doit avoir la structure suivante:

```
<?xml version="1.0" encoding="UTF-8"?>
- <test>
  - <xml>
    - <rootTag>
      <value><test>value</test></value>
    </rootTag>
  </xml>
</test>
```

Si le contenu du fichier est le suivant :

```
<?xml version="1.0" encoding="UTF-8"?>
- <test>
  <xml><rootTag> <value>&lt;test&gt;value&lt;/test&gt;</value> </rootTag></xml>
</test>
```

Le composant tAdvancedFileOutputXML a été mal installé.

- Lors de l'ouverture des jobs "Some components are not loaded tAdvancedFileOutputXML" ?

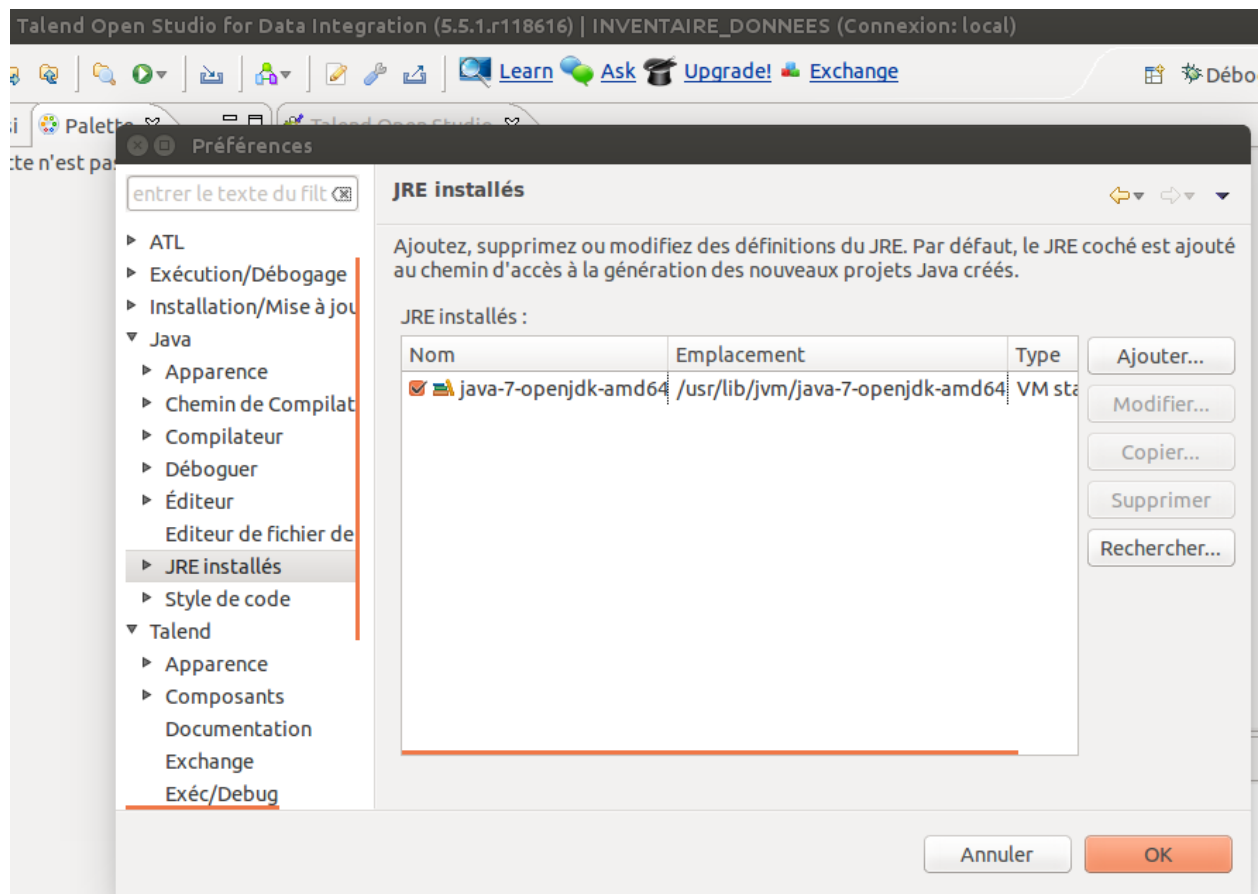
Vérifier dans le répertoire

plugins/org.talend.components_localprovider_xyz/tAdvancedFileOutputXML vous avez bien dézippé les fichiers du composant (eg. tAdavencedOutpuXML_begin.javajet). Ils ne doivent pas être dans un répertoire à l'intérieur du composant (eg. tAdvancedFileOutputXML/tAdvancedFileOutputXML/*).

- **Comment résoudre Can't load AMD 64-bit .dll on a IA 32-bit platform ?**

```
[statistics] connected
Native library load failed.
java.lang.UnsatisfiedLinkError: C:\Program Files\GDAL\ogrjni.dll: Can't load AMD 64-bit .dll on a IA
32-bit platform
```

Ce problème est lié au fait d'utiliser un Java 32b avec GDAL/OGR en 64. Remplacer le Java utilisé sur le système ou celui utilisé par Talend dans Fenêtre > Préférence > Java > JRE installé : pointer vers un JDK 64b et le sélectionner :



- **Comment résoudre les erreurs du type org.gdal.gdal.gdalJNI.Dataset_SWIGUpcast ?**

Parfois, et plus généralement sur les données RASTER, l'erreur suivante est retournée :

```
Exception in thread "main" java.lang.UnsatisfiedLinkError:
org.gdal.gdal.gdalJNI.Dataset_SWIGUpcast(J)J
  at org.gdal.gdal.gdalJNI.Dataset_SWIGUpcast(Native Method)
  at org.gdal.gdal.Dataset.<init>(Dataset.java:15)
  at org.gdal.gdal.gdal.Open(gdal.java:583)
  at
```

```

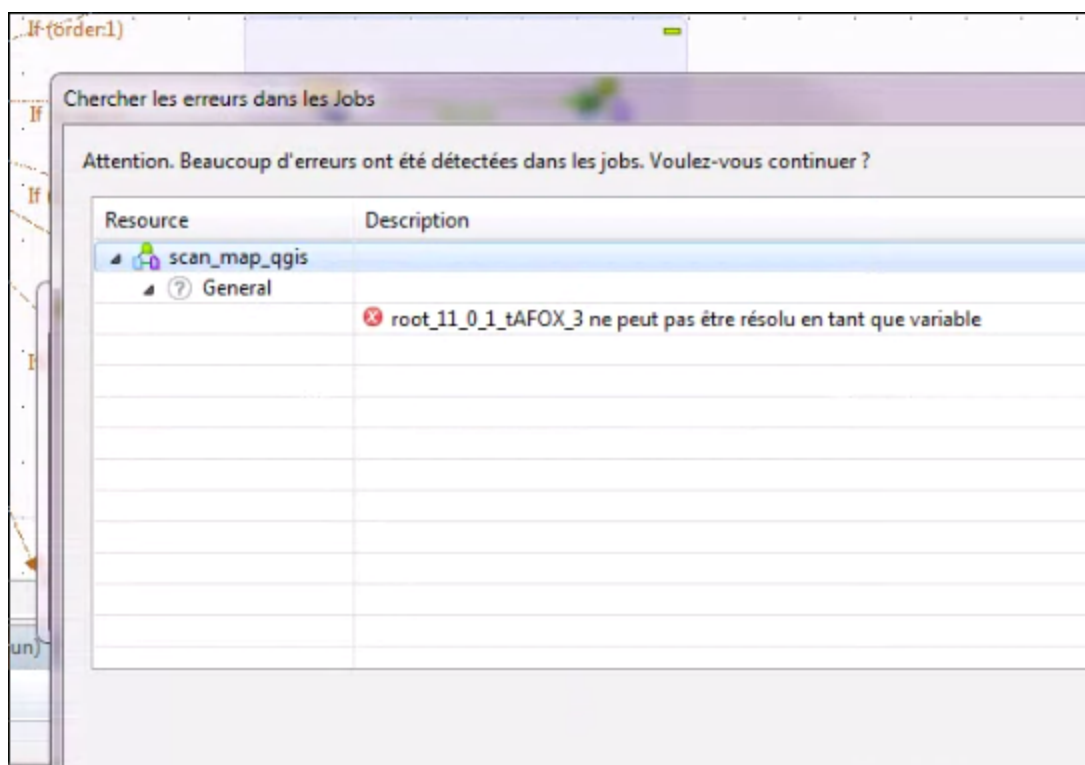
inventaire_pnf.scan_raster_0_1.scan_raster.sGdalInfoInput_1Process(scan_raster.java:4943)
  at inventaire_pnf.scan_raster_0_1.scan_raster.runJobInTOS(scan_raster.java:11994)
  at inventaire_pnf.scan_raster_0_1.scan_raster.runJob(scan_raster.java:11791)
  at inventaire_pnf.run_0_1.run.tFileList_1Process(run.java:973)
  at inventaire_pnf.run_0_1.run.tFileList_2Process(run.java:744)
  at inventaire_pnf.run_0_1.run.runJobInTOS(run.java:2010)
  at inventaire_pnf.run_0_1.run.main(run.java:1828)

```

La version de GDAL installée et la version utilisée par les composants n'est pas la même. Dans ce cas rechercher gdal.jar dans le répertoire d'installation de GDAL et le copier à la place des fichiers gdal.jar (c:/programme/gdal/java) présent dans l'installation de Talend (normalement 4 fichiers sont à remplacer).

- ./lib/java/gdal.jar
- ./workspace/.Java/lib/gdal.jar
- ./plugins/org.talend.sdi.designer.components_X.Y.Z/components/sGdalInfoInput/gdal.jar
- ./plugins/org.talend.sdi.repository.ui.actions.metadata.ogr_X.Y.Z/lib/gdal.jar

- **Comment résoudre *root_11_0_1_tAFOX_3* ?**



Le composant tAdvancedOutputXML n'a pas été déployé correctement dans TDI. Cf. [Installer TDI](#).

- Quelle URL CSW utiliser ?

Lors de l'ajout des métadonnées des transactions CSW sont effectuées. Pour récupérer l'URL du CSW pour les transactions, consulter le GetCapabilities du service.

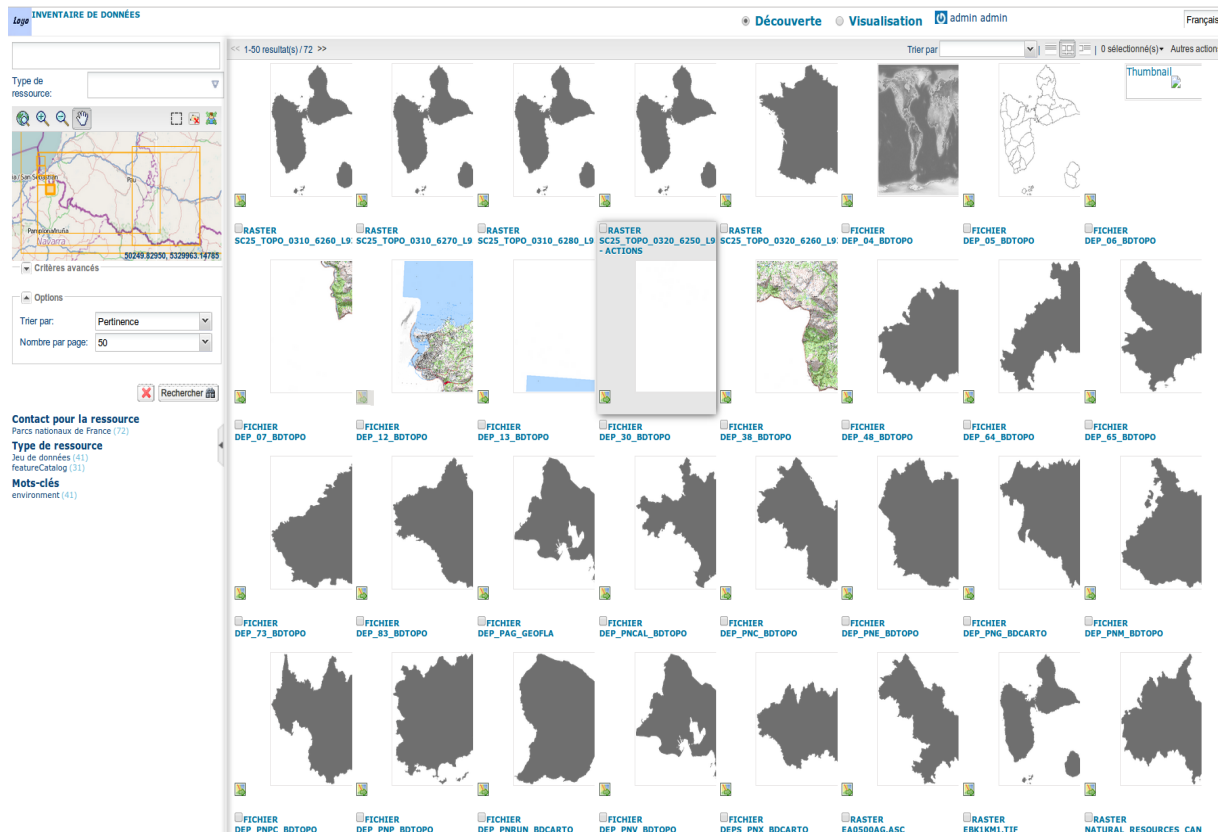
 apps.titellus.net/geonetwork-pnf/srv/eng/csw?SERVICE=CSW&REQUEST=GetCapabilities

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<csw:Capabilities xmlns:csw="http://www.opengis.net/cat/csw/2.0.2" xmlns:gml="http://www.opengis.r
xmlns:ogc="http://www.opengis.net/ogc" xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi="http:
http://schemas.opengis.net/csw/2.0.2/CSW-discovery.xsd">
  ▶<ows:ServiceIdentification>...</ows:ServiceIdentification>
  ▶<ows:ServiceProvider>...</ows:ServiceProvider>
  ▼<ows:OperationsMetadata>
    ▶<ows:Operation name="GetCapabilities">...</ows:Operation>
    ▶<ows:Operation name="DescribeRecord">...</ows:Operation>
    ▶<ows:Operation name="GetDomain">...</ows:Operation>
    ▶<ows:Operation name="GetRecords">...</ows:Operation>
    ▶<ows:Operation name="GetRecordById">...</ows:Operation>
    ▼<ows:Operation name="Transaction">
      ▼<ows:DCP>
        ▼<ows:HTTP>
          <ows:Get xlink:href="http://apps.titellus.net/geonetwork-pnf/srv/eng/csw-publication"/>
          <ows:Post xlink:href="http://apps.titellus.net/geonetwork-pnf/srv/eng/csw-publication"/>
        </ows:HTTP>
      </ows:DCP>
    </ows:Operation>
  ▼<ows:Operation name="Harvest">
    ▼<ows:DCP>
      ▼<ows:HTTP>
        <ows:Get xlink:href="http://apps.titellus.net/geonetwork-pnf/srv/eng/csw-publication"/>
        <ows:Post xlink:href="http://apps.titellus.net/geonetwork-pnf/srv/eng/csw-publication"/>
```

Conclusions

Ces jobs permettent ainsi de réaliser un inventaire rapide de son patrimoine géographique. Il assure le recensement des informations disponibles aussi bien au format vecteur fichier ou base de données que RASTER. Ils pré-remplissent les métadonnées qui nécessitent cependant d'être revue, complétée et publiée à partir de l'outil de catalogage.



Ces travaux ont été l'occasion d'améliorer les composants du module Spatial⁷ réalisant l'analyse des données SIG. Une nouvelle version 5.3.1⁸ a été publiée à cette occasion. Les jobs sont publiquement accessibles via le dépôt github suivant :

<https://github.com/talend-spatial/workspace-metadata-crawler>

Par la suite, il serait intéressant de tester les différents formats supportés par GDAL/OGR. La publication vers un serveur cartographique a été testée avec GeoServer pour les Shapefile mais mériterait des tests complémentaires pour les différents formats de données et le support de Mapserver

⁷ <http://talend-spatial.github.io/>

⁸ <https://sourceforge.net/projects/sdispatialet/files/sdispatialet/TOS.spatial.5.3.1/>

Annexes

Autres travaux similaires & références

- <https://github.com/talend-spatial/talend-spatial/wiki/Metadata-crawler-urban-atlas-use-case>
- <https://github.com/talend-spatial/talend-spatial/wiki/Generate-a-map-library-based-on-PDF-map-analysis>
- <https://github.com/talend-spatial/talend-spatial/blob/master/doc/include/en/installation.rst>

Corrections & améliorations apportées à TDI

Au cours de la réalisation de ces travaux, les corrections et améliorations suivantes ont été réalisées en coopération avec les équipes de Talend :

- XML mapper - amélioration
 - tAdvancedFileOutputXml / add option to add Document type as text (default) or node <https://jira.talendforge.org/browse/TDI-26630> - proposé
- XML mapper - bug
 - Metadata / XML File / Lost connection when schema is updated <https://jira.talendforge.org/browse/TDI-26616> - corrigé dans les prochaines versions de TDI
 - tAdvancedOutputXML : Wrong arrow in XML tree mapper <https://jira.talendforge.org/browse/TDI-26615> - non corrigé / cosmétique

Corrections & améliorations apportées au module Spatial

Les améliorations suivantes ont été apportées au module Spatial :

- sCSWRequest / Add URL parameter
- sOGRInfo / Loop on all layers available
- sGDALInfo / Do not stop flow on error
- sOGRInfo / Improve projection detection
- sOGRInfoInput / fail creating extent
- Routines / Add routine for reprojection in GeometryOperation

Cf. <https://github.com/talend-spatial/talend-spatial/issues?milestone=5&page=1&state=closed>