



Talend Open Studio for Big Data

Guide de prise en main

6.3.0

Convient à la version 6.3.0. Annule et remplace toute version antérieure de ce guide.

Date de publication : 27 octobre 2016

Copyright

Cette documentation est mise à disposition selon les termes du Contrat Public Creative Commons (CPCC).

Pour plus d'informations concernant votre utilisation de cette documentation en accord avec le Contrat CPCC, consultez : <http://creativecommons.org/licenses/by-nc-sa/2.0/>

Mentions légales

Talend est une marque déposée de Talend, Inc.

Tous les noms de marques, de produits, les noms de sociétés, les marques de commerce et de service sont la propriété de leurs détenteurs respectifs.

Licence applicable

Le logiciel décrit dans cette documentation est soumis à la Licence Apache, Version 2.0 (la "Licence"). Vous ne pouvez utiliser ce logiciel que conformément aux dispositions de la Licence. Vous pouvez obtenir une copie de la Licence sur <http://www.apache.org/licenses/LICENSE-2.0.html> (en anglais). Sauf lorsqu'explicitement prévu par la loi en vigueur ou accepté par écrit, le logiciel distribué sous la Licence est distribué "TEL QUEL", SANS GARANTIE OU CONDITION D'AUCUNE SORTE, expresse ou implicite. Consultez la Licence pour connaître la terminologie spécifique régissant les autorisations et les limites prévues par la Licence.

Ce produit comprend les logiciels développés par AOP Alliance (standards Java/J2EE AOP), ASM, Amazon, AntLR, Apache ActiveMQ, Apache Ant, Apache Avro, Apache Axiom, Apache Axis, Apache Axis 2, Apache Batik, Apache CXF, Apache Cassandra, Apache Chemistry, Apache Common Http Client, Apache Common Http Core, Apache Commons, Apache Commons Bcel, Apache Commons JXPath, Apache Commons Lang, Apache DataFu, Apache Derby Database Engine and Embedded JDBC Driver, Apache Geronimo, Apache HCatalog, Apache Hadoop, Apache Hbase, Apache Hive, Apache HttpClient, Apache HttpComponents Client, Apache JAMES, Apache Log4j, Apache Lucene Core, Apache Neethi, Apache Oozie, Apache POI, Apache Parquet, Apache Pig, Apache PiggyBank, Apache ServiceMix, Apache Sqoop, Apache Thrift, Apache Tomcat, Apache Velocity, Apache WSS4J, Apache WebServices Common Utilities, Apache Xml-RPC, Apache Zookeeper, Box Java SDK (V2), CSV Tools, Cloudera HTrace, ConcurrentLinkedHashMap for Java, Couchbase Client, DataNucleus, DataStax Java Driver for Apache Cassandra, Ehcache, Ezmorph, Ganymed SSH-2 for Java, Google APIs Client Library for Java, Google Gson, Groovy, Guava : Google Core Libraries for Java, H2 Embedded Database and JDBC Driver, Hector : client Java haut niveau pour Apache Cassandra, Hibernate BeanValidation API, Hibernate Validator, HighScale Lib, HsqlDB, Ini4j, JClouds, JDO-API, JLine, JSON, JSR 305: Annotations for Software Defect Detection in Java, JUnit, Jackson Java JSON-processor, Java API for RESTful Services, Java Agent for Memory Measurements, Jaxb, Jaxen, JetS3T, Jettison, Jetty, Joda-Time, Json Simple, LZ4 : Extremely Fast Compression algorithm, LightCouch, MetaStuff, Metrics API, Metrics Reporter Config, Microsoft Azure SDK pour Java, Mondrian, MongoDB Java Driver, Netty, Ning Compression codec for LZ4 encoding, OpenSAML, Paracel JDBC Driver, Parboiled, PostgreSQL JDBC Driver, Protocol Buffers - Google's data interchange format, Resty : client simple HTTP REST pour Java, Riak Client, Rocoto, SDSU Java Library, SL4J : Simple Logging Facade for Java, SQLite JDBC Driver, Scala Lang, Simple API for CSS, Snappy for Java a fast compressor/decompressor, SpyMemCached, SshJ, StAX API, StAXON - JSON via StAX, Super SCV, The Castor Project, The Legion of the Bouncy Castle, Twitter4J, Uuid, W3C, bibliothèques Windows Azure Storage pour Java, Woden, Woodstox : High-performance XML processor, Xalan-J, Xerces2, XmlBeans, XmlSchema Core, Xmlsec - Apache Santuario, YAML parser et emitter pour Java, Zip4J, atinject, dropbox-sdk-java : bibliothèque Java pour l'API Dropbox Core, google-guice. Fournis sous leur licence respective.

Table des matières

chapitre 1. Introduction au Talend Open Studio for Big Data	1
1.1. Architecture fonctionnelle des solutions Big Data de Talend	2
chapitre 2. Prérequis à l'utilisation des produits Talend	3
2.1. Recommandations relatives à la mémoire	4
2.2. Recommandations logicielles	4
2.3. Installation de Java	4
2.4. Configuration des variables d'environnement Java sous Windows	5
2.5. Configuration des variables d'environnement Java sous Linux	5
2.6. Installation de 7-Zip (Windows)	6
chapitre 3. Téléchargement et installation de Talend Open Studio for Big Data	7
3.1. Téléchargement de Talend Open Studio for Big Data	8
3.2. Installation de Talend Open Studio for Big Data	8
3.2.1. Extraire via 7-Zip (recommandé pour Windows)	8
3.2.2. Extraire via l'outil de dézippage Windows par défaut	8
3.2.3. Extraire via l'outil de dézippage Linux	9
chapitre 4. Configuration de votre produit Talend	11
4.1. Démarrage du Studio pour la première fois	12
4.2. Connexion au Studio	12
4.3. Installation des packages supplémentaires	12
4.4. Configuration manuelle de la connexion à Hadoop	13
4.5. Configuration de la connexion à HDFS	15
4.6. Chargement des fichiers dans HDFS	18
4.7. Préparation de la métadonnée du fichier	21
chapitre 5. Tâches d'intégration de données pour Big Data	27
5.1. Fusionner les informations des films et réalisateurs	28
5.1.1. Créer le Job	28
5.1.2. Ajouter et relier les composants	29
5.1.3. Configurer les données d'entrée	30
5.1.4. Configurer la transformation de données	33
5.1.5. Écrire la sortie dans HDFS	35
5.2. Que faire ensuite ?	36



Chapitre 1. Introduction au Talend Open Studio for Big Data

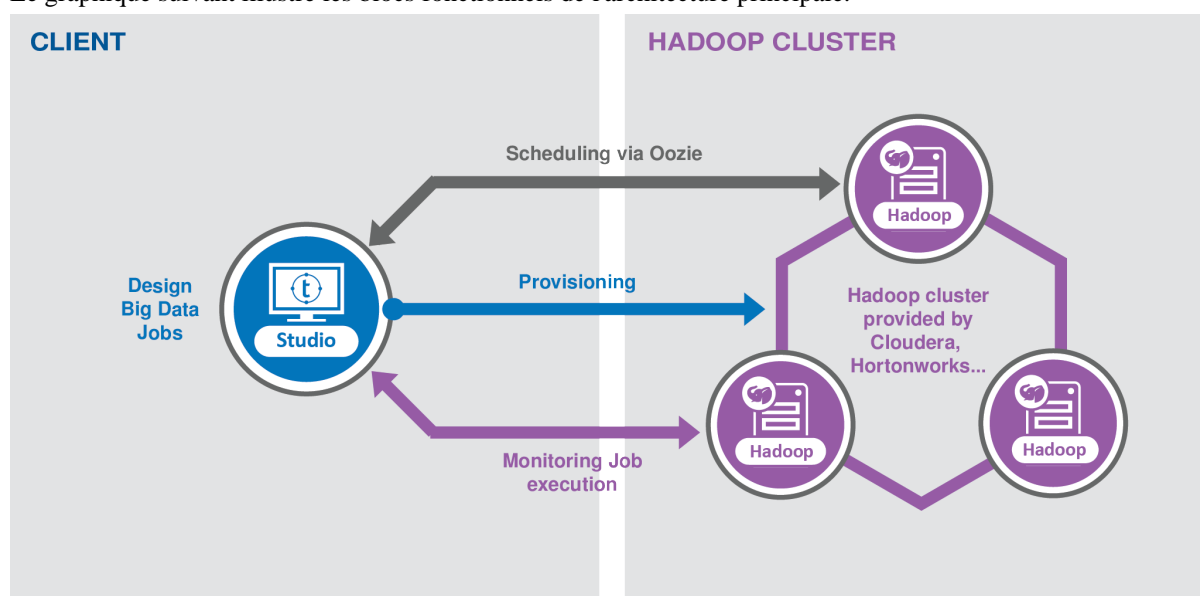
Talend Open Studio for Big Data fournit des outils de développement et de gestion unifiés pour intégrer et traiter toutes vos données dans un environnement graphique simple à utiliser.

Construit sur les solutions d'intégration de données de *Talend*, les solutions Big Data de *Talend* fournissent un outil puissant permettant aux utilisateurs d'accéder aux données volumineuses, de les transformer, de les déplacer et de les synchroniser, en tirant parti de la plateforme Apache Hadoop Big Data et en facilitant l'utilisation de cette plateforme.

1.1. Architecture fonctionnelle des solutions Big Data de Talend

L'architecture fonctionnelle de Talend Open Studio for Big Data est un modèle architectural qui identifie les fonctions, les interactions et les besoins informatiques correspondants de Talend Open Studio for Big Data. L'architecture d'ensemble a été décrite en isolant les fonctionnalités spécifiques en blocs fonctionnels.

Le graphique suivant illustre les blocs fonctionnels de l'architecture principale.



Les trois différents types de blocs fonctionnels sont définis comme suit :

- Dans le *Studio Talend*, vous créez et exécutez des Jobs Big Data tirant parti du cluster Hadoop afin de gérer de grands volumes de données. Une fois lancés, ces Jobs sont envoyés, déployés et exécutés sur ce cluster Hadoop.
- Oozie, un système d'ordonnancement de workflows, est intégré dans le studio, à travers lequel vous pouvez déployer, ordonnancer et exécuter des Jobs Big Data dans un cluster Hadoop et monitorer le statut d'exécution, ainsi que les résultats des Jobs.
- Un cluster Hadoop indépendant du système *Talend* pour gérer d'importants ensembles de données.



Chapitre 2. Prérequis à l'utilisation des produits Talend

Ce chapitre vous fournit des informations simples concernant le logiciel et le matériel requis et recommandés pour prendre en main votre produit *Talend* :

- [*Recommandations relatives à la mémoire.*](#)
- [*Recommandations logicielles.*](#)

Il vous guide également à travers les étapes d'installation et de configuration des outils tiers requis et recommandés :

- [*Installation de Java.*](#)
- [*Configuration des variables d'environnement Java sous Windows*](#) or [*Configuration des variables d'environnement Java sous Linux.*](#)
- [*Installation de 7-Zip \(Windows\).*](#)

Pour bien installer le logiciel, vous devez avoir les accès Administrateur sur votre ordinateur. Pour obtenir ces accès, contactez votre Administrateur.

2.1. Recommandations relatives à la mémoire

Pour optimiser l'utilisation des produits *Talend*, référez-vous aux recommandations de mémoire et espace disque ci-dessous :

Utilisation de la mémoire	3 Go minimum, 4 Go recommandés
Utilisation du disque	3 Go

2.2. Recommandations logicielles

Pour optimiser l'utilisation des produits *Talend*, référez-vous aux recommandations système et logicielles ci-dessous :

Logiciels requis

- Systèmes d'exploitation pour le *Studio Talend* :

Type de support	Système d'exploitation	Version	Processeur
Recommandé	Microsoft Windows Professional	7	64 bits
Recommandé	Linux Ubuntu	14.04	64 bits
Supporté	Apple OS X	El Capitan/10.11	64 bits
		Yosemite/10.10	64 bits
		Mavericks/10.9	64 bits

- Java 8 JRE Oracle. Consultez [Installation de Java](#).
- Un cluster Hadoop installé et configuré.

Vous devez avoir vérifié que la machine cliente sur laquelle est installé le *Studio Talend* peut reconnaître les noms d'hôtes des nœuds du cluster Hadoop à utiliser. Dans cet objectif, ajoutez les mappings des entrées adresse IP/nom d'hôte pour les services de ce cluster Hadoop dans le fichier *hosts* de la machine cliente.

Par exemple, si le nom d'hôte du serveur du NameNode Hadoop est *talend-cdh550.weave.local* et son adresse IP est *192.168.x.x*, l'entrée du mapping est la suivante *192.168.x.x talend-cdh550.weave.local*.

Logiciel facultatif

- 7-Zip. consultez [Installation de 7-Zip \(Windows\)](#).

2.3. Installation de Java

Pour utiliser votre produit *Talend*, vous avez besoin d'une JRE Oracle (Oracle Java Runtime Environment) installée sur votre ordinateur.

- Dans la page [Java SE Downloads](#) (en anglais), sous **Java Platform, Standard Edition**, cliquez sur **JRE Download**.
- Dans la page **Java SE Runtime Environment 8 Downloads**, sélectionnez le bouton radio **Accept License Agreement**.

3. Sélectionnez le téléchargement correspondant à votre système d'exploitation.
4. Suivez les étapes d'installation de Java proposées par l'assistant Oracle.

Lorsque Java est installé sur votre ordinateur, vous devez configurer la variable d'environnement `JAVA_HOME`. Pour plus d'informations, consultez :

- [Configuration des variables d'environnement Java sous Windows](#).
- [Configuration des variables d'environnement Java sous Linux](#).

2.4. Configuration des variables d'environnement Java sous Windows

Avant d'installer votre produit *Talend*, vous devez configurer les variables d'environnement `JAVA_HOME` et `Path` :

1. Dans le menu **Démarrer** de votre ordinateur, cliquez-droit sur **Ordinateur** et sélectionnez **Properties**.
2. Dans la fenêtre **[Control Panel Home]**, cliquez sur **Advanced system settings**.
3. Dans la fenêtre **[System Properties]**, cliquez sur **Environment Variables...**
4. Sous **System Variables**, cliquez sur **New...** pour créer une variable. Nommez la variable `JAVA_HOME`, saisissez le chemin d'accès à votre JRE 8 Java, puis cliquez sur **OK**.

Exemple de chemin vers la JRE par défaut : `C:\Program Files\Java\jre1.8.0_77`.

5. Sous **System Variables**, sélectionnez la variable **Path** et cliquez sur **Edit...** pour ajouter la variable `JAVA_HOME` précédemment définie à la fin de la variable d'environnement `Path`, en les séparant par un point-virgule.

Exemple : `<PathVariable>;%JAVA_HOME%\bin`.

2.5. Configuration des variables d'environnement Java sous Linux

Avant d'installer votre produit *Talend*, vous devez configurer les variables d'environnement `JAVA_HOME` et `Path` :

1. Trouvez le répertoire d'installation de la JRE.
Exemple : `/usr/lib/jvm/jre1.8.0_65`
2. Spécifiez-le dans la variable d'environnement `JAVA_HOME`.

Exemple :

```
export JAVA_HOME=/usr/lib/jvm/jre1.8.0_65
export PATH=$JAVA_HOME/bin:$PATH
```

3. Ajoutez ces lignes à la fin des profils utilisateurs dans le fichier `~/.profile` ou, en tant que super-utilisateur, à la fin des profils globaux dans le fichier `/etc/profile`.
4. Connectez-vous à nouveau.

2.6. Installation de 7-Zip (Windows)

Talend recommande d'installer 7-Zip et de l'utiliser pour extraire les fichiers d'installation : <http://www.spiroo.be/7zip/>.

1. Téléchargez l'installeur de 7-Zip correspondant à votre système d'exploitation.
2. Naviguez dans vos dossiers locaux, trouvez le fichier .exe de 7-Zip et double-cliquez dessus pour l'installer.

Le téléchargement démarre automatiquement.



Chapitre 3. Téléchargement et installation de Talend Open Studio for Big Data

Talend Open Studio for Big Data est simple à installer. Après l'avoir téléchargé depuis le site Web de *Talend*, un simple dézippage permet de l'installer sur votre ordinateur.

Ce chapitre vous fournit les informations de base relatives au téléchargement et à l'installation.

3.1. Téléchargement de Talend Open Studio for Big Data

Talend Open Studio for Big Data est un produit open source libre que vous pouvez télécharger directement depuis le site Web de *Talend* :

1. Allez à la [page de téléchargement](#) de *Talend Open Studio for Big Data* .
2. Cliquez sur **TÉLÉCHARGER L'OUTIL LIBRE**.

Le téléchargement démarre automatiquement.

3.2. Installation de Talend Open Studio for Big Data

L'installation s'effectue en dézipant le fichier .zip **TOS_BD** précédemment téléchargé.

Vous pouvez faire ceci en utilisant :

- 7-Zip (recommandé sous Windows) : [Extraire via 7-Zip \(recommandé pour Windows\)](#).
- le dézippeur par défaut de Windows : [Extraire via l'outil de dézippage Windows par défaut](#).
- le dézippeur par défaut de Linux (pour un système d'exploitation basé Linux) : [Extraire via l'outil de dézippage Linux](#).

3.2.1. Extraire via 7-Zip (recommandé pour Windows)

Sous Windows, *Talend* vous recommande d'installer 7-Zip et de l'utiliser pour extraire des fichiers. Pour plus d'informations, consultez [Installation de 7-Zip \(Windows\)](#).

Pour installer le Studio, suivez les étapes suivantes :

1. Naviguez dans vos dossiers locaux, trouvez le fichier .zip **TOS** et déplacez-le à un autre emplacement, avec un chemin d'accès aussi court que possible et sans caractère d'espace.

Exemple : *C:/Talend/*

2. Dézippez-le en cliquant-droit sur le fichier compressé et sélectionnez **7-Zip > Extract Here**.

3.2.2. Extraire via l'outil de dézippage Windows par défaut

Si vous ne souhaitez pas utiliser 7-Zip, vous pouvez utiliser l'outil de dézippage par défaut de Windows :

1. Dézippez-le en cliquant-droit sur le fichier compressé et sélectionnez **Extract All**.
2. Cliquez sur **Browse** et naviguez jusqu'au disque **C:**.

3. Sélectionnez **Make new folder** et nommez le dossier *Talend*. Cliquez sur **OK**.
4. Cliquez sur **Extract** pour commencer l'installation.

3.2.3. Extraire via l'outil de dézippage Linux

Pour installer le Studio, suivez les étapes ci-dessous :

1. Naviguez dans vos dossiers locaux, trouvez le fichier .zip **TOS** et déplacez-le à un autre emplacement, avec un chemin d'accès aussi court que possible, sans caractère d'espace.

Exemple : *home/user/talend/*

2. Dézippez-le en cliquant-droit sur le fichier compressé et sélectionnez **Extract Here**.



Chapitre 4. Configuration de votre produit Talend

Ce chapitre vous fournit les informations simples nécessaires à la configuration de votre produit *Talend*, notamment concernant le lancement du produit, la connexion au logiciel et la création d'un projet :

- *Démarrage du Studio pour la première fois.*
- *Connexion au Studio.*
- *Installation des packages supplémentaires.*
- *Configuration manuelle de la connexion à Hadoop*
- *Configuration de la connexion à HDFS*
- *Chargement des fichiers dans HDFS*
- *Préparation de la métadonnée du fichier*

4.1. Démarrage du Studio pour la première fois

Le répertoire d'installation du Studio contient des fichiers binaires pour différentes plateformes, notamment Mac OS X et Linux/Unix.

Pour ouvrir le *Studio Talend* pour la première fois, procédez comme suit :

1. Double-cliquez sur le fichier exécutable correspondant à votre système d'exploitation, par exemple :
 - TOS_*-win-x86_64.exe, sous Windows.
 - TOS_*-linux-gtk-x86_64, sous Linux.
 - TOS_*-macosx-cocoa.app, sous Mac.
2. Dans la fenêtre **[User License Agreement]** qui s'ouvre, lisez et acceptez les termes de la licence pour procéder aux étapes suivantes.

4.2. Connexion au Studio

Pour vous connecter au *Studio Talend* pour la première fois, procédez comme suit :

1. Dans la fenêtre de login du *Studio Talend*, sélectionnez l'option **Create a new project**, spécifiez le nom du projet : *getting_started* et cliquez sur **Finish** pour créer un nouveau projet local.
2. Selon le produit que vous utilisez, vous voyez s'ouvrir :
 - la présentation rapide (Quick Tour). Jouez-la pour obtenir plus d'informations relatives à l'interface du Studio, puis cliquez sur **Stop** pour la terminer.
 - la page de bienvenue (**Welcome**). Suivez les liens pour obtenir plus d'informations concernant le Studio et cliquez sur **Start Now!** pour fermer la page et continuer l'ouverture du Studio.

Vous êtes connecté au *Studio Talend*. Vous devez installer les packages supplémentaires requis pour que le *Studio Talend* fonctionne correctement.

4.3. Installation des packages supplémentaires

Talend vous recommande d'installer des packages supplémentaires, y compris des bibliothèques tierces et les pilotes de bases de données, dès que vous vous connectez à votre *Studio Talend*, afin de tirer pleinement parti de toutes les fonctionnalités du Studio.

1. Lorsque l'assistant **[Additional Talend Packages]** s'ouvre, installez les packages supplémentaires, en cochant les cases **Required** et **Optional third-party libraries**. Cliquez sur **Finish**.

Ces packages vous permettent de bénéficier pleinement des fonctionnalités du studio.

Cet assistant s'affiche à chaque fois que vous lancez le studio si des packages supplémentaires sont disponibles à l'installation à moins que vous ne cochiez la case **Do not show this again**. Vous pouvez également afficher cet assistant en sélectionnant **Help > Install Additional Packages** dans la barre de menu.

Pour plus d'informations, consultez la section concernant l'installation de packages supplémentaires dans le *Guide d'installation et de migration Talend*.

2. Dans la fenêtre [**Download external modules**], cliquez sur le bouton **Accept all** au bas de l'assistant pour accepter toutes les licences des modules externes dans le studio.

Attendez que toutes les bibliothèques soient installées avant de commencer à utiliser le studio.

3. Si nécessaire, redémarrez votre *Studio Talend* pour que certains packages supplémentaires soient pris en compte.

4.4. Configuration manuelle de la connexion à Hadoop

Configurez la connexion à une distribution Hadoop donnée dans le **Repository** vous permettant d'éviter de configurer cette connexion à chaque fois que vous devez utiliser la même distribution Hadoop.

Prérequis :

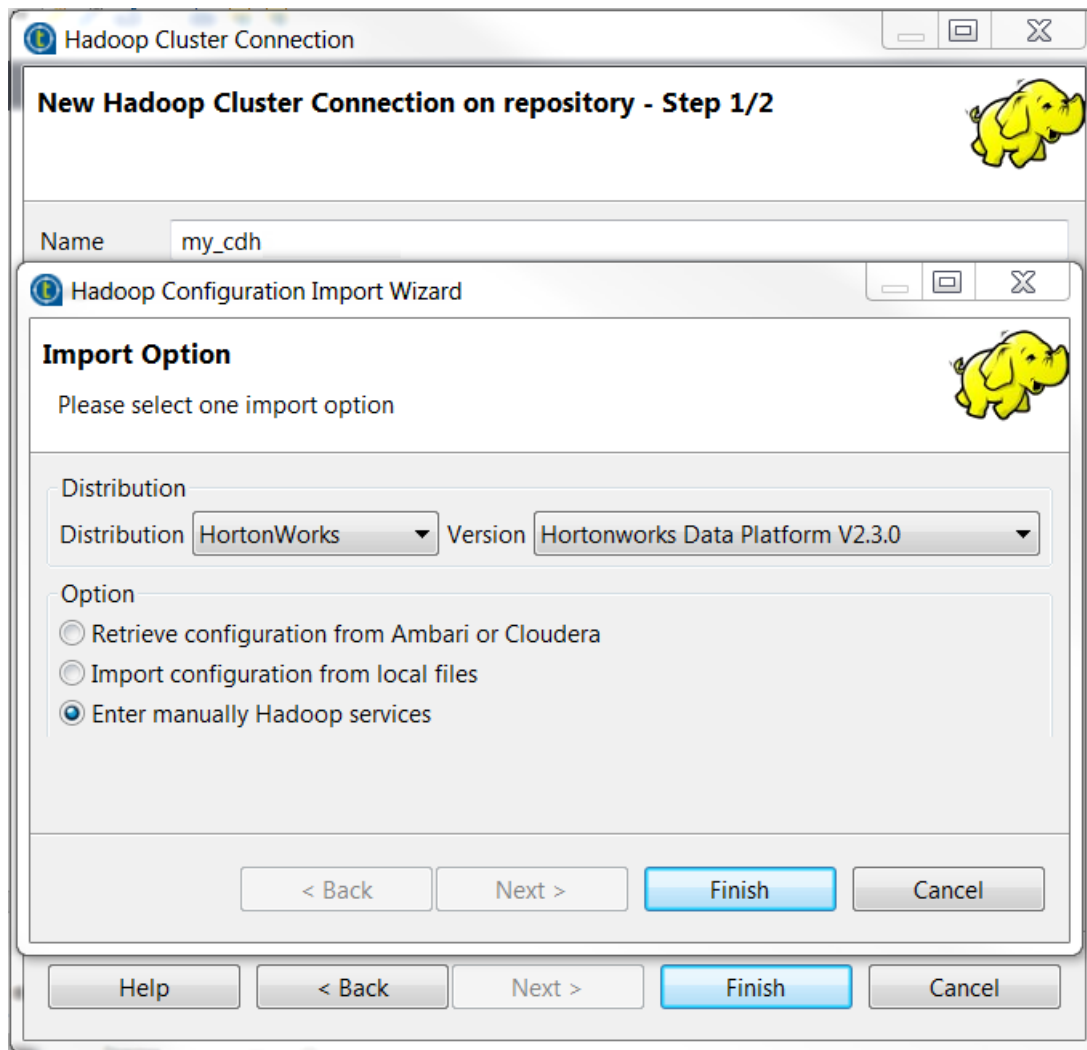
- Vous devez avoir vérifié que la machine cliente sur laquelle est installé le *Studio Talend* peut reconnaître les noms d'hôtes des nœuds du cluster Hadoop à utiliser. Dans cet objectif, ajoutez les mappings des entrées adresse IP/nom d'hôte pour les services de ce cluster Hadoop dans le fichier *hosts* de la machine cliente.

Par exemple, si le nom d'hôte du serveur du NameNode Hadoop est *talend-cdh550.weave.local* et son adresse IP est *192.168.x.x*, l'entrée du mapping est la suivante *192.168.x.x talend-cdh550.weave.local*.

- Le cluster Hadoop à utiliser doit avoir été correctement configuré et être en cours d'exécution.

Le cluster Hadoop Cloudera à utiliser dans cet exemple est CDH V5.5 en mode YARN et applique la configuration par défaut de la distribution sans activer la sécurité Kerberos. Pour plus d'informations concernant la configuration par défaut de la distribution CDH V5.5, consultez [Deploy CDH 5 on a cluster](#) et [Default ports used in CDH5](#) (liens en anglais).

1. Dans la vue **Repository** de votre studio, développez **Metadata** et cliquez-droit sur **Hadoop cluster**.
2. Dans le menu contextuel, sélectionnez **Create Hadoop cluster** pour ouvrir l'assistant [**Hadoop cluster connection**].
3. Renseignez les informations génériques relatives à cette connexion, comme les champs **Name** et **Description**, puis cliquez sur **Next** pour ouvrir l'assistant [**Hadoop configuration import wizard**] vous permettant d'importer une configuration prête à l'emploi, s'il y en a.
4. Cochez la case **Enter manually Hadoop services** afin de saisir manuellement les informations de configuration pour la connexion Hadoop en cours de création.



5. Cliquez sur **Finish** pour fermer l'assistant d'import.
6. Dans la liste **Distribution**, sélectionnez **Cloudera** et, dans la liste **Version**, sélectionnez **Cloudera CDH5.5 (YARN mode)**.
7. Dans le champ **Namenode URI**, saisissez une URI pointant vers la machine utilisée en tant que service du NameNode du cluster Hadoop Cloudera à utiliser.

Le NameNode est le nœud maître d'un système Hadoop. Par exemple, si vous avez choisi une machine nommée *machine1* en tant que NameNode, l'emplacement à saisir est *hdfs://machine1:portnumber*.

Du côté cluster, la propriété associée est spécifiée dans le fichier de configuration associé nommé *core-site.xml*. Si vous ne connaissez pas l'URI à saisir, vérifiez la propriété *fs.defaultFS* dans le fichier *core-site.xml* de votre cluster.

8. Dans les champs **Resource manager** et **Resource manager scheduler**, saisissez les URI pointant vers ces deux services, respectivement.

Du côté cluster, ces deux services partagent la même machine hôte mais utilisent différents numéros de port par défaut. Par exemple, si la machine les hébergeant est *resourcemanager.company.com*, l'emplacement du Resource Manager est *resourcemanager.company.com:8032* et l'emplacement de l'ordonnanceur du gestionnaire de ressources est *resourcemanager.company.com:8030*.

Si vous ne connaissez pas le nom de la machine hébergeant ces services, vérifiez la propriété *yarn.resourcemanager.hostname* dans le champ de configuration nommé *yarn-site.xml* de votre cluster.

9. Dans le champ **Job history**, saisissez l'emplacement du service du JobHistory. Ce service permet aux informations de métriques du Job courant d'être stockées sur le serveur du JobHistory.

La propriété associée est spécifiée dans le fichier de configuration nommé *mapred-site.xml* de votre cluster. Pour la valeur saisie dans ce champ, vérifiez la propriété *mapreduce.jobhistory.address* dans le fichier *mapred-site.xml*.

10. Dans le champ **Staging directory**, saisissez le chemin d'accès au répertoire défini dans votre cluster Hadoop pour les fichiers temporaires créés par les programmes d'exécutant.

La propriété associée est spécifiée dans le fichier *mapred-site.xml* de votre cluster. Pour plus d'informations, vérifiez la propriété *yarn.app.mapreduce.am.staging-dir* dans le fichier *mapred-site.xml*.

11. Cochez la case **Use datanode hostname** pour permettre au Studio d'accéder à chaque Datanode de votre cluster via leurs noms d'hôtes.

Cela configure la propriété *dfs.client.use.datanode.hostname* de votre cluster à *true*.

12. Dans le champ **User name**, saisissez le nom d'authentification que vous souhaitez que le Studio utilise pour se connecter au cluster Hadoop.

13. Puisque le cluster Hadoop auquel se connecter utilise la configuration par défaut, laissez les autres champs et cases dans l'assistant tels qu'ils sont, car ils sont utilisés pour définir les configurations Hadoop personnalisées.

14. Cliquez sur le bouton **Check services** afin de vérifier que le Studio peut se connecter aux services du NameNode et du ResourceManager spécifiés.

Une boîte de dialogue s'ouvre pour indiquer le statut du processus de vérification et de la connexion.

Si la connexion échoue, vous pouvez cliquer sur **Error log** à la fin de chaque barre de progression afin de diagnostiquer les problèmes de connexion.

15. Une fois que la vérification indique que la connexion est établie, cliquez sur **Finish** pour valider vos modifications et fermer l'assistant.

La nouvelle connexion, nommée *my_cdh* dans cet exemple, est affichée dans le dossier **Hadoop cluster** de la vue **Repository**.

Vous pouvez continuer à créer les connexions filles aux différents éléments Hadoop, comme HDFS ou Hive, à partir de cette connexion.

4.5. Configuration de la connexion à HDFS

Une connexion à HDFS dans le **Repository** vous permet de réutiliser cette connexion dans différents Jobs associés.

Prérequis :

- La connexion au cluster Hadoop hébergeant le système HDFS à utiliser doit avoir été configurée depuis le nœud **Hadoop cluster** dans le **Repository**.

Pour plus d'informations concernant la création de cette connexion, consultez [Configuration manuelle de la connexion à Hadoop](#).

- Le cluster Hadoop à utiliser doit avoir été correctement configuré et être en cours d'exécution. Vous devez avoir les droits d'accès à cette distribution et à HDFS.
- Vous devez avoir vérifié que la machine cliente sur laquelle est installé le *Studio Talend* peut reconnaître les noms d'hôtes des nœuds du cluster Hadoop à utiliser. Dans cet objectif, ajoutez les mappings des entrées adresse IP/nom d'hôte pour les services de ce cluster Hadoop dans le fichier *hosts* de la machine cliente.

Par exemple, si le nom d'hôte du serveur du NameNode Hadoop est *talend-cdh550.weave.local* et son adresse IP est *192.168.x.x*, l'entrée du mapping est la suivante *192.168.x.x talend-cdh550.weave.local*.

1. Développez le nœud **Hadoop cluster** sous **Metadata** dans le **Repository**, cliquez-droit sur la connexion Hadoop à utiliser et sélectionnez **Create HDFS** dans le menu contextuel.
2. Dans l'assistant de connexion qui s'ouvre, renseignez les propriétés génériques de la connexion que vous devez créer, dans les champs **Name**, **Purpose** et **Description**.

HDFS Connection

New HDFS Connection on repository - Step 1/2

Define the properties

Name

Purpose

Description

Author

Locker

Version

Status

Path

< Back Next > Finish Cancel

3. Cliquez sur **Next** lorsque vous avez terminé. L'étape suivante nécessite de renseigner les informations de connexion à HDFS.

La propriété **User name** est automatiquement renseignée par la valeur héritée de la connexion Hadoop sélectionnée dans les étapes précédentes.

Les champs **Row separator** et **Field separator** utilisent les valeurs par défaut.

HDFS Connection

Update HDFS Connection - Step 2/2

You must press the Check Button to check the HDFS Setting

Connection Settings

User name

Separator Settings

Row Separator Field Separator

Rows To Skip

If any rows must be ignored, specify the following parameters

Header ☒ ☒ Set heading row as column names

Hadoop Properties (Empty)

4. Cochez la case **Set heading row as column names** pour utiliser les lignes d'en-tête du fichier HDFS comme noms de colonnes dans ce fichier.

Automatiquement, la case **Header** est cochée et la valeur dans le champ **Header** est *1*. Cela signifie que la première ligne du fichier sera ignorée en tant que corps des données mais sera utilisée comme noms de colonnes dans le fichier.

5. Cliquez sur **Check** afin de vérifier votre connexion.

Un message s'ouvre pour indiquer que la connexion est établie.

6. Cliquez sur **Finish** afin de valider vos modifications.

La nouvelle connexion à HDFS est disponible sous le nœud **Hadoop cluster**, dans le **Repository**. Vous pouvez l'utiliser pour définir et centraliser les schémas des fichiers stockés dans le système HDFS connecté afin de réutiliser ces schéma dans des Jobs *Talend*.

4.6. Chargement des fichiers dans HDFS

Charger un fichier dans HDFS permet à des Jobs Big Data de lire et traiter ce fichier.

Prérequis :

- La connexion au cluster Hadoop à utiliser et la connexion au système HDFS de ce cluster doivent avoir été configurées dans le nœud **Hadoop cluster** du **Repository**.

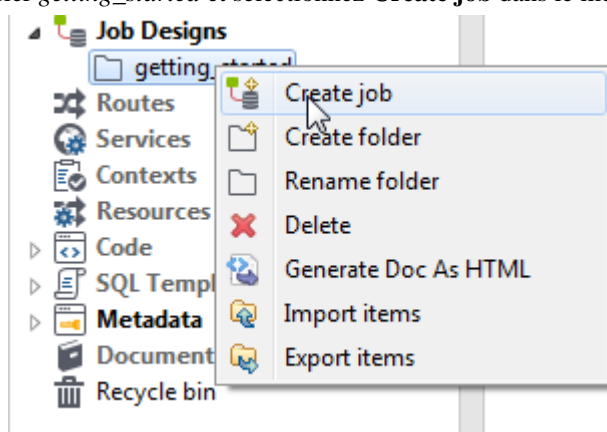
Si ce n'est pas le cas, consultez [Configuration manuelle de la connexion à Hadoop](#) et [Configuration de la connexion à HDFS](#) afin de créer ces connexions.

- Le cluster Hadoop à utiliser doit avoir été configuré correctement et être en cours d'exécution. Vous devez avoir les droits d'accès à cette distribution et au dossier HDFS à utiliser.
- Vous devez avoir vérifié que la machine cliente sur laquelle les Jobs *Talend* sont exécutés peut reconnaître les noms d'hôtes du cluster Hadoop à utiliser. Dans cet objectif, ajoutez les mappings des entrées adresse IP/nom d'hôte pour les services de ce cluster Hadoop dans le fichier *hosts* de la machine cliente.

Par exemple, si le nom d'hôte du serveur du NameNode Hadoop est *talend-cdh550.weave.local* et son adresse IP est *192.168.x.x*, l'entrée du mapping est la suivante *192.168.x.x talend-cdh550.weave.local*.

Au cours de cette procédure, vous allez créer un Job écrivant des données dans le système HDFS du cluster Cloudera Hadoop auquel la connexion a été configurée dans le **Repository**, comme expliqué dans [Configuration manuelle de la connexion à Hadoop](#). Ces données sont nécessaires pour le scénario décrit dans . Les fichiers nécessaires à ce scénario peuvent être téléchargés [ici](#).

- Dans la vue **Repository**, cliquez-droit sur le nœud **Job Designs** et sélectionnez **Create folder** dans le menu contextuel.
- Dans l'assistant **[New Folder]**, nommez votre dossier de Jobs *getting_started* puis cliquez sur **Finish** pour créer votre dossier.
- Cliquez-droit sur le dossier *getting_started* et sélectionnez **Create job** dans le menu contextuel.



- Dans l'assistant **[New Job]**, saisissez un nom pour le Job à créer, ainsi que d'autres informations utiles.

Par exemple, saisissez *write_to_hdfs* dans le champ **Name**.

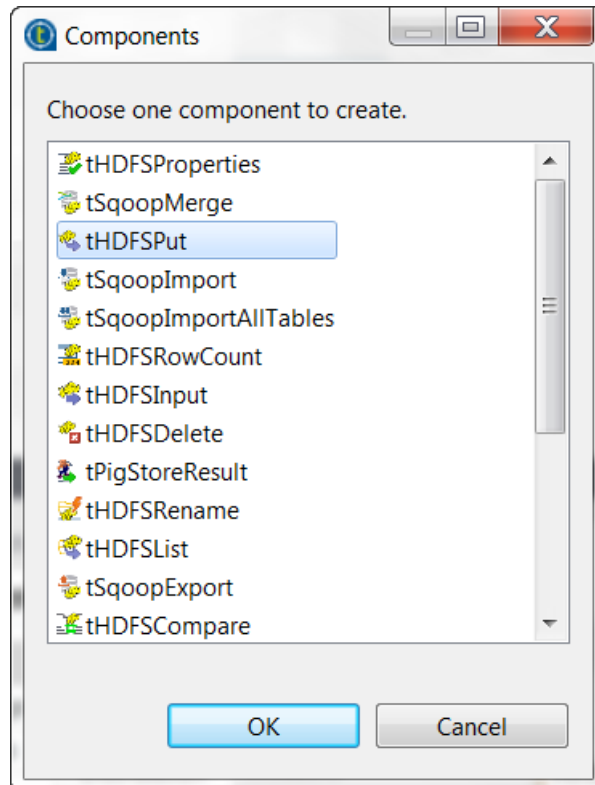
Dans cette étape de l'assistant, **Name** est le seul champ obligatoire. Les informations que vous fournissez dans le champ **Description** s'affichent en tant qu'info-bulle lorsque vous passez votre curseur sur le Job dans la vue **Repository**.

- Cliquez sur **Finish** pour créer votre Job.

Un Job vide s'ouvre dans le Studio.

6. Développez le nœud **Hadoop cluster** sous **Metadata** dans le **Repository**.
7. Développez la connexion Hadoop précédemment créée et le dossier **HDFS** en-dessous. Dans cet exemple, développez la connexion *my_cdh*.
8. Déposez la connexion HDFS du dossier **HDFS** dans l'espace de modélisation graphique du Job que vous créez. Cette connexion est, dans cet exemple, *cdh_hdfs*.

La fenêtre [**Components**] s'ouvre et affiche tous les composants pouvant directement utiliser cette connexion HDFS dans un Job.



9. Sélectionnez le **tHDFSPut** et cliquez sur **OK** afin de valider votre choix.

La fenêtre [**Components**] se ferme et un composant **tHDFSPut** est automatiquement ajouté dans l'espace de modélisation graphique du Job, composant nommé d'après la connexion HDFS mentionnée dans l'étape précédente.

10. Double-cliquez sur le **tHDFSPut** pour ouvrir sa vue **Component**.

The screenshot shows the Talend Open Studio interface with the 'cdh_hdfs(tHDFSPut_1)' component selected. The 'Basic settings' tab is active, showing the following configuration:

- Property Type:** Repository
- HDFS:** cdh_hdfs
- Use an existing connection:** ☐
- Version:**
 - Distribution:** Cloudera
 - Version:** Cloudera CDH5.5(YARN mode)
- Connection:**
 - NameNode URI:** "hdfs://talend-cdh550.weave.local:8020"
 - Use Datanode Hostname:** ☒
- Authentication:**
 - User kerberos authentication:** ☐
 - Username:** "ychen"
- Local directory:** "C:/gettingstarted/input_data"
- HDFS directory:** "/user/ychen/input_data"
- Overwrite file:** always
- Use Perl5 Regex Expressions as Filemask (Unchecked means Glob Expressions):** ☐
- Files:**

Filemask	New name
"*"	"
- Die on error:** ☒

La connexion au système HDFS à utiliser a été automatiquement configurée via la connexion HDFS configurée et stockée dans le **Repository**. Les paramètres dans cet onglet passent en lecture seule. Ces paramètres sont : **Distribution**, **Version**, **NameNode URI**, **Use Datanode Hostname**, **User kerberos authentication** et **Username**.

11. Dans le champ **Local directory**, saisissez le chemin d'accès ou parcourez votre système jusqu'au dossier dans lequel stocker les fichiers à copier dans HDFS.

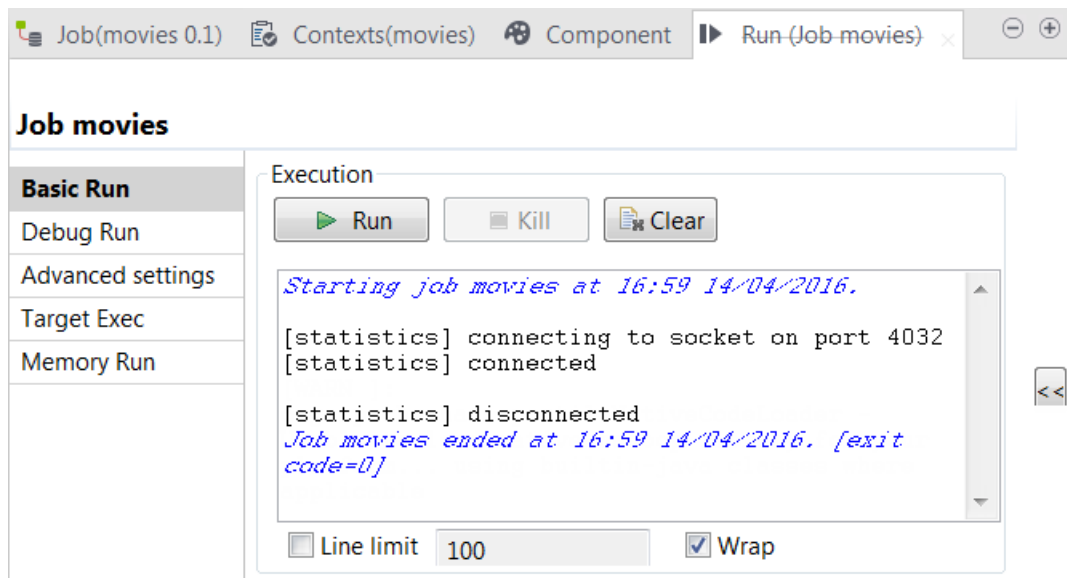
Les fichiers concernant les films et leurs réalisateurs sont stockés dans ce répertoire.

12. Dans le champ **HDFS directory**, saisissez le chemin d'accès ou parcourez votre système jusqu'au répertoire cible HDFS dans lequel stocker les fichiers.

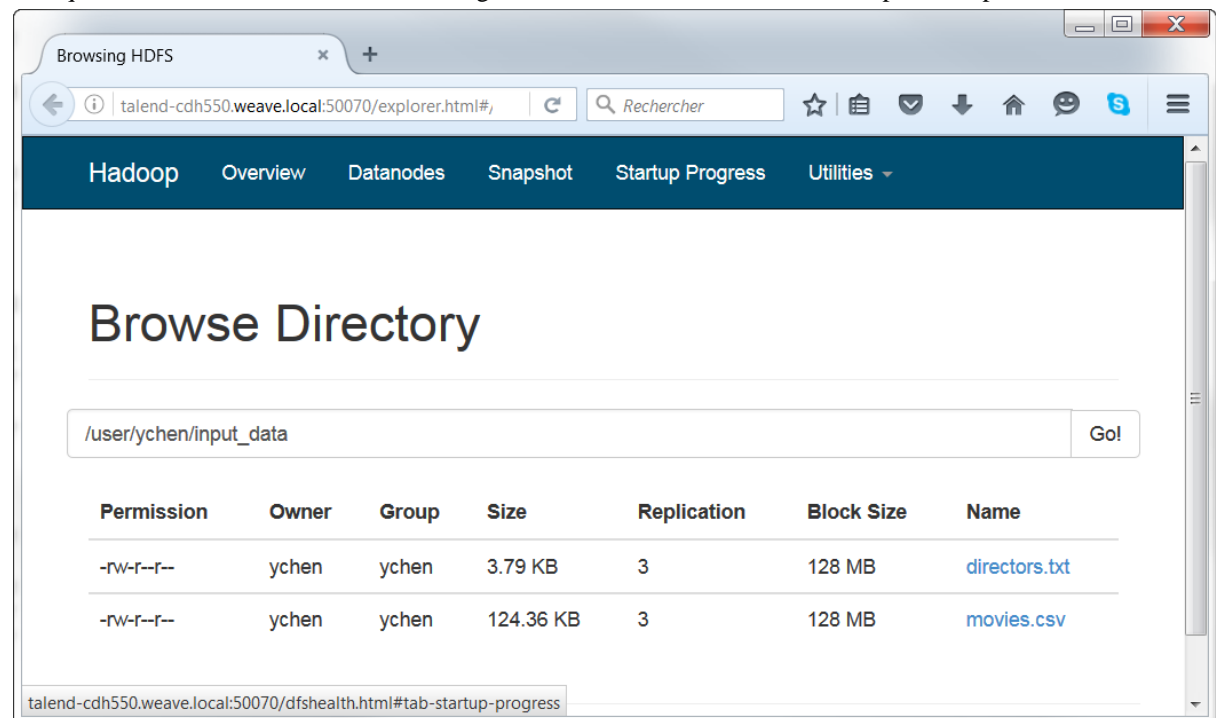
Ce répertoire est créé à la volée s'il n'existe pas.

13. Dans la liste **Overwrite file**, sélectionnez **always** pour écraser les fichiers s'ils existent déjà dans le répertoire cible, dans HDFS.
14. Dans la table **Files**, ajoutez une ligne en cliquant sur le bouton [+] afin de définir les critères de sélection des fichiers à copier.
15. Dans la colonne **Filemask**, saisissez un astérisque (*) entre guillemets doubles, pour que le **tHDFSPut** sélectionne tous les fichiers stockés dans le dossier spécifié dans le champ **Local directory**.
16. Laissez la colonne **New name** vide, c'est-à-dire, laissez les guillemets doubles par défaut pour ne pas modifier le nom des fichiers après chargement.
17. Appuyez sur **F6** pour exécuter le Job.

La vue **Run** s'ouvre automatiquement et affiche l'avancement de l'exécution du Job.



Lorsque le Job est terminé, les fichiers chargés se trouvent dans HDFS, dans le répertoire spécifié.



4.7. Préparation de la métadonnée du fichier

Dans le **Repository**, configurer la métadonnée d'un fichier stocké dans HDFS vous permet de réutiliser directement son schéma dans un composant Big Data associé, sans avoir à configurer manuellement chaque paramètre.

Prérequis :

- Vous devez avoir démarré votre *Studio Talend* et ouvert la perspective **Integration**.
- Les fichiers source, *movies.csv* et *directors.txt* doivent avoir été chargés dans HDFS comme expliqué dans [Chargement des fichiers dans HDFS](#).

- La connexion au cluster Hadoop à utiliser et la connexion au système HDFS de ce cluster doivent avoir été configurées dans le nœud **Hadoop cluster** du **Repository**.

Si ce n'est pas le cas, consultez [Configuration manuelle de la connexion à Hadoop](#) et [Configuration de la connexion à HDFS](#) pour créer ces connexions.

- Le cluster Hadoop à utiliser doit avoir été correctement configuré et être en cours d'exécution. Vous devez avoir les droits d'accès à cette distribution et au dossier HDFS à utiliser.
- Vous devez avoir vérifié que la machine cliente sur laquelle est installé le *Studio Talend* peut reconnaître les noms d'hôtes des nœuds du cluster Hadoop à utiliser. Dans cet objectif, ajoutez les mappings des entrées adresse IP/nom d'hôte pour les services de ce cluster Hadoop dans le fichier *hosts* de la machine cliente.

Par exemple, si le nom d'hôte du serveur du NameNode Hadoop est *talend-cdh550.weave.local* et son adresse IP est *192.168.x.x*, l'entrée du mapping est la suivante *192.168.x.x talend-cdh550.weave.local*.

Comme le fichier *movies.csv* que vous devez traiter a été stocké dans le système HDFS, vous pouvez récupérer son schéma afin de configurer ses métadonnées dans le **Repository**.

Le schéma du fichier *directors.txt* peut également être récupéré mais est délibérément ignoré lors de la procédure de récupération expliquée ci-dessous, car, dans ce scénario, le fichier *directors.txt* est utilisé pour démontrer comment définir manuellement un schéma dans un Job.

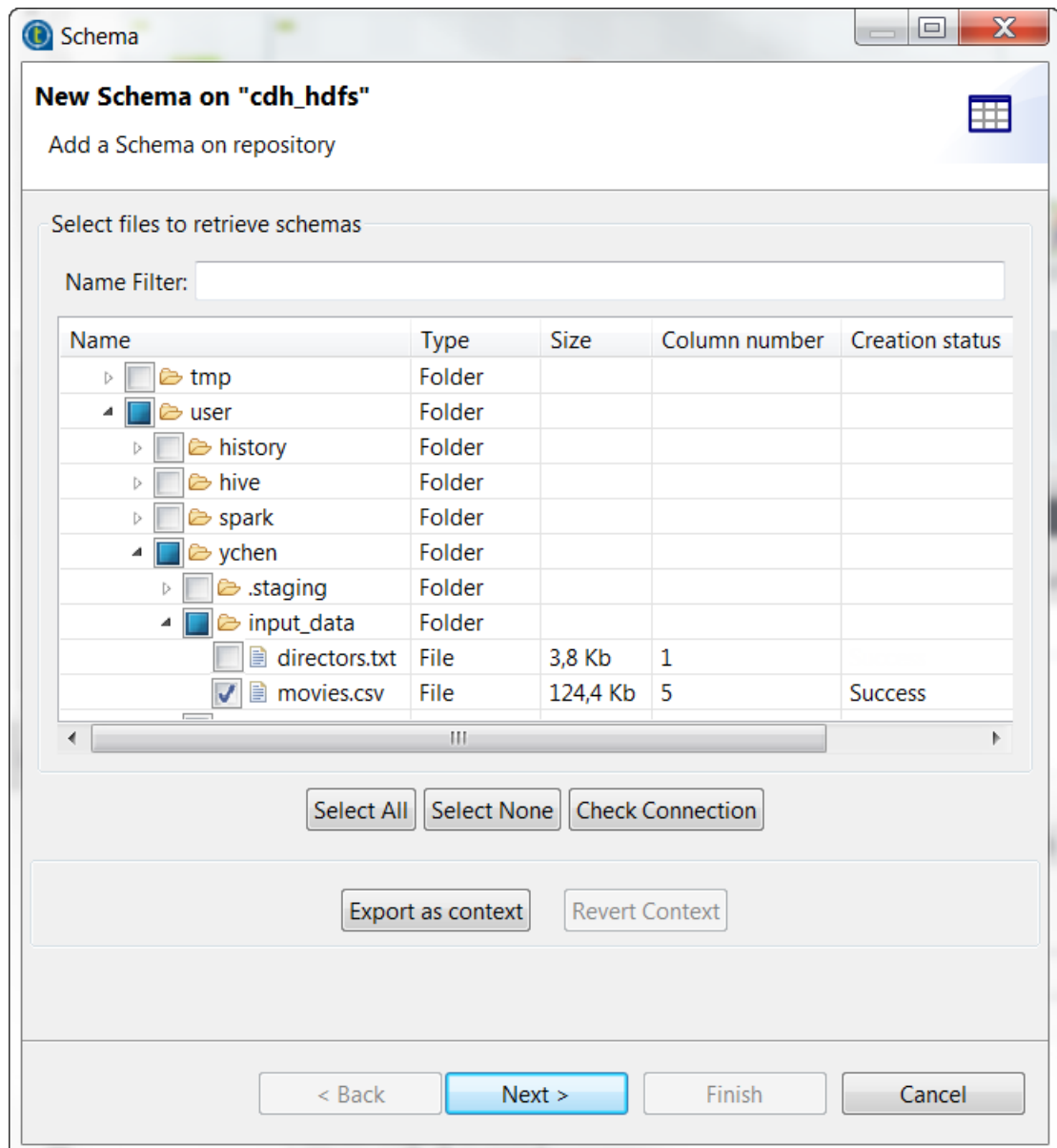
1. Développez le nœud **Hadoop cluster** sous **Metadata** dans la vue **Repository**.
2. Développez la connexion Hadoop créée et le dossier **HDFS** en-dessous.

Dans cet exemple, développez la connexion *my_cdh*.

3. Cliquez-droit sur la connexion HDFS dans le dossier **HDFS** et, dans le menu contextuel, sélectionnez **Retrieve schema**.

Dans ce scénario, cette connexion à HDFS se nomme *cdh_hdfs*.

L'assistant [**Schema**] s'ouvre et vous permet de parcourir vos fichiers dans HDFS.



4. Développez l'arborescence jusqu'à afficher le fichier *movies.csv*, duquel vous devez récupérer le schéma et sélectionnez-le.

Dans ce scénario, le fichier *movies.csv* est stocké dans le répertoire suivant : */user/ychen/input_data*.

5. Cliquez sur **Next** pour afficher le schéma récupéré dans l'assistant.

Le schéma des données de films est affiché dans l'assistant et la première ligne des données est automatiquement utilisée comme noms de colonnes.

Update Schema "cdh_hdfs"
Update an existing Schema on repository

Schema: movies

Name: movies

Comment:

Base on file: /user/ychen/input_data/movies.csv

Guess Schema

Schema

Column	Key	Type	✓	N..	Date Patter...	Length	Precision
movieID	<input type="checkbox"/>	Integer	<input checked="" type="checkbox"/>			4	0
title	<input type="checkbox"/>	String	<input checked="" type="checkbox"/>			29	0
releaseYear	<input type="checkbox"/>	Integer	<input checked="" type="checkbox"/>			4	0
url	<input type="checkbox"/>	String	<input checked="" type="checkbox"/>			66	0
directorID	<input type="checkbox"/>	String	<input checked="" type="checkbox"/>			3	0

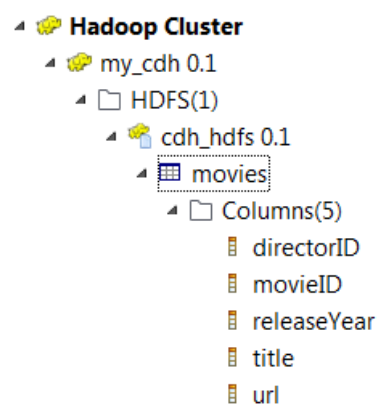
Export as context Revert Context

Finish Cancel

Si la première ligne des données que vous utilisez n'est pas utilisée pour les noms de colonnes, vous devez vérifier la configuration de l'option **Header** définie lors de la création de la connexion à HDFS, comme expliqué dans [Configuration de la connexion à HDFS](#).

6. Cliquez sur **Finish** afin de valider ces modifications.

Vous pouvez voir la métadonnée sous la connexion à HDFS que vous utilisez, dans la vue **Repository**.





Chapitre 5. Tâches d'intégration de données pour Big Data

Ce chapitre prend l'exemple d'une entreprise fournissant des services de locations de films et de streaming de vidéos. Il vous explique comment une telle entreprise peut tirer parti de Talend Open Studio for Big Data.

Vous allez utiliser des données relatives à des films et des réalisateurs, ainsi que des données concernant vos clients, tout en apprenant à

- charger les données stockées dans un système de fichiers local vers le système de fichiers HDFS du cluster Hadoop de votre entreprise
- fusionner les données des réalisateurs et celles des films, afin de produire un nouveau jeu de données et le stocker dans le système HDFS également

5.1. Fusionner les informations des films et réalisateurs

Dans ce scénario, un Job Big Data est utilisé pour lire, transformer et écrire des données concernant des films et réalisateurs dans un environnement Hadoop.

Ce scénario vous apprend à :

1. Créer un Job *Talend*. Consultez [Créer le Job](#) pour plus de détails.
2. Ajouter et relier les composants à utiliser dans un Job. Consultez [Ajouter et relier les composants](#) pour plus de détails.
3. Configurer les composants d'entrée à l'aide de la métadonnées du **Repository**. Consultez [Configurer les données d'entrée](#) pour plus de détails.
4. Configurer la transformation pour effectuer une jointure sur les données d'entrée. Consultez [Configurer la transformation de données](#) pour plus de détails.
5. Écrire les données transformées dans HDFS. Consultez [Écrire la sortie dans HDFS](#) pour plus de détails.

5.1.1. Créer le Job

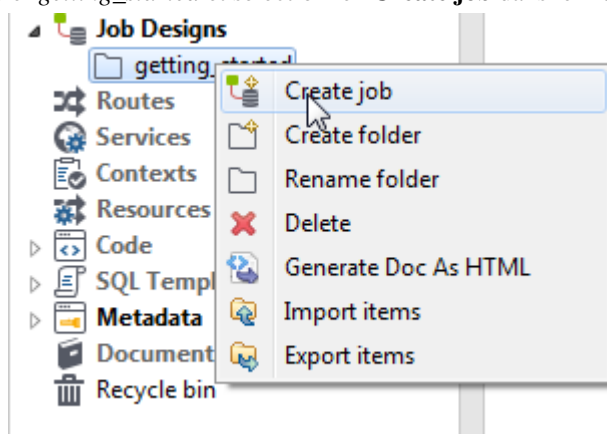
Un Job *Talend* vous permet d'accéder aux composants *Talend* et de les utiliser pour créer des processus techniques permettant de lire, transformer ou écrire des données.

Prérequis :

- Vous devez avoir démarré votre *Studio Talend* et ouvert la perspective **Integration**.

Procédez comme suit afin de créer le Job :

1. Dans la vue **Repository**, cliquez-droit sur le nœud **Job Designs** et sélectionnez **Create folder** dans le menu contextuel.
2. Dans l'assistant **[New Folder]**, nommez votre dossier de Jobs *getting_started* puis cliquez sur **Finish** pour créer votre dossier.
3. Cliquez-droit sur le dossier *getting_started* et sélectionnez **Create job** dans le menu contextuel.



4. Dans l'assistant **[New Job]**, saisissez un nom pour le Job à créer, ainsi que d'autres informations utiles.

Par exemple, saisissez *aggregate_movie_director* dans le champ **Name**.

Dans cette étape de l'assistant, **Name** est le seul champ obligatoire. Les informations que vous fournissez dans le champ **Description** s'affichent en tant qu'info-bulle lorsque vous passez votre curseur sur le Job dans la vue **Repository**.

5. Cliquez sur **Finish** pour créer votre Job.

Un Job vide s'ouvre dans le Studio.

La **Palette** des composants est disponible dans le Studio. Vous pouvez commencer à concevoir le Job en utilisant cette **Palette** et le nœud **Metadata** dans le **Repository**.

5.1.2. Ajouter et relier les composants

Les composants Pig à utiliser dans l'espace de modélisation graphique permettent de créer un processus Pig de transformation de données.

Prérequis :

- Vous devez avoir démarré votre *Studio Talend* et ouvert la perspective **Integration**.
- Un Job vide a été créé comme décrit dans [Créer le Job](#) et s'ouvre dans l'espace de modélisation graphique.

Procédez comme suit pour ajouter et relier les composants :

1. Dans l'espace de modélisation graphique du Job, saisissez le nom du composant à utiliser et sélectionnez ce composant dans la liste qui s'affiche. Dans ce scénario, les composants sont deux **tPigLoad**, un **tPigMap** et deux **tPigStoreResult**.

- Les deux **tPigLoad** sont utilisés pour charger les données des films et réalisateurs, respectivement, de HDFS dans le flux de données du Job.
- Le **tPigMap** est utilisé pour transformer les données d'entrée.
- Les **tPigStoreResult** écrivent les résultats dans des répertoires données dans HDFS.

2. Double-cliquez sur le libellé d'un composant **tPigLoad** pour modifier ce libellé et saisissez *movie* comme nouveau nom.
3. Répétez l'opération pour nommer le second **tPigLoad** *director*.
4. Cliquez-droit sur le composant **tPigLoad** nommé *movie* et, dans le menu contextuel, sélectionnez **Row > Pig combine** et cliquez sur le **tPigMap** pour relier ce **tPigLoad** au **tPigMap**.

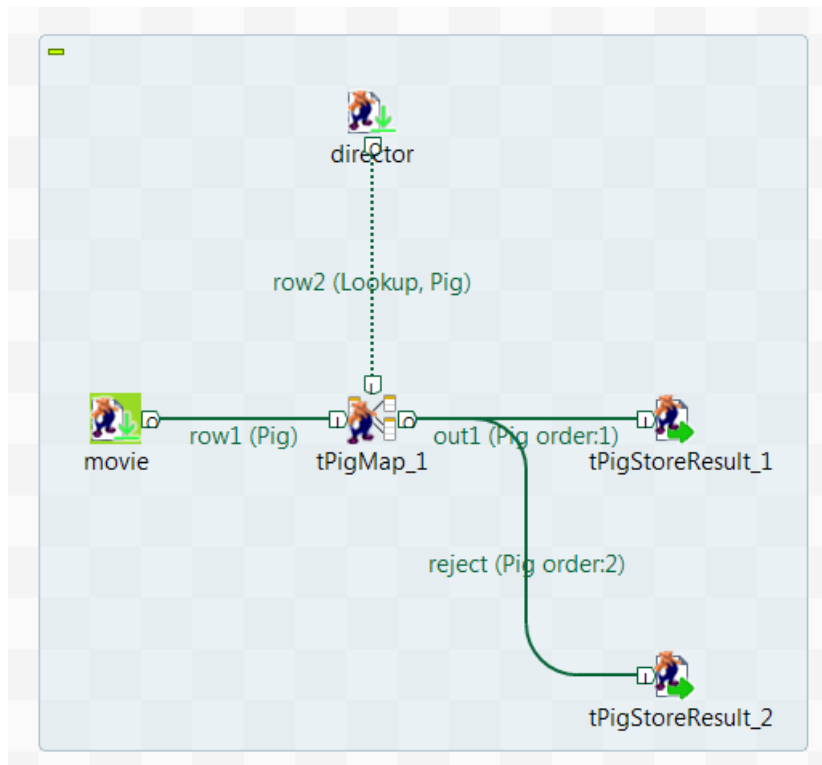
Ceci forme le lien principal via lequel les données de films sont envoyées au **tPigMap**.

5. Répétez l'opération pour relier le composant *director* au **tPigMap**, à l'aide d'un lien **Row > Pig combine**.

Ceci est le lien **Lookup** via lequel les données des réalisateurs sont envoyées au **tPigMap** en tant que données de référence.

6. Répétez l'opération pour connecter le **tPigMap** au premier **tPigStoreResult**, à l'aide d'un lien **Row > Pig combine** et, dans la boîte de dialogue qui s'ouvre, nommez ce lien *out1*, puis cliquez sur **OK** afin de valider ces modifications.
7. Répétez ces opérations pour connecter le **tPigMap** à l'autre **tPigStoreResult**, à l'aide d'un lien **Row > Pig combine** que vous nommez *reject*.

Le Job complet doit ressembler à ceci :



5.1.3. Configurer les données d'entrée

Deux composants **tPigLoad** sont configurés pour charger des données de HDFS dans le Job.

Prérequis :

- Les fichiers source, **movies.csv** et **directors.txt** ont été chargés dans HDFS, comme expliqué dans [Chargement des fichiers dans HDFS](#).
- La métadonnée du fichier **movie.csv** a été configurée dans le dossier HDFS sous le nœud **Hadoop cluster** dans le **Repository**.

Si ce n'est pas le cas, consultez [Préparation de la métadonnée du fichier](#) pour créer la métadonnée.

Une fois le Job créé et tous les composants Pig à utiliser inclus dans le Job et reliés, vous devez configurer les composants **tPigLoad** pour lire les données de HDFS.

1. Développez le nœud **Hadoop cluster** sous **Metadata** dans le **Repository** puis le nœud de connexion Hadoop *my_cdh* et son nœud fils pour afficher le nœud du schéma de la métadonnée *movies* configuré, dans le dossier **HDFS**, comme expliqué dans [Préparation de la métadonnée du fichier](#).
2. Déposez ce schéma sur le composant **tPigLoad** nommé **movie**, dans l'espace de modélisation graphique du Job.
3. Double-cliquez sur le composant **tPigLoad** nommé **movie** pour ouvrir sa vue **Component**.

Le **tPigLoad** a réutilisé automatiquement la configuration HDFS et la métadonnée relative aux films pour définir ses paramètres dans la vue **Basic settings**.

movie(tPigLoad_1)

Basic settings

Property Type: Repository | HDFS:cdh_hdfs

Schema: Repository | HDFS:cdh_hdfs - movies

Configuration

☐ Local

Distribution: Cloudera | Version: Cloudera CDH5.5(YARN mode)

Load function: PigStorage

☐ Inspect the classpath for configurations

NameNode URI: "hdfs://talend-cdh550.weave.local:8020"

Resource Manager: "talend-cdh550.weave.local:8032"

☒ Set jobhistory address: "talend-cdh550.weave.local:10020"

☒ Set resource manager scheduler address: "talend-cdh550.weave.local:8030"

☒ Set staging directory: "/user"

☒ Use datanode hostname

Authentication

☐ Use kerberos authentication

User name: "ychen"

☐ Use S3 endpoint

Input file URI: "/user/ychen/input_data/movies.csv"

Field separator: ";"

Compression

☐ Force to compress the output data

☐ Die on subjob error

4. Dans la liste **Load function**, sélectionnez **PigStorage** pour utiliser la fonction **PigStorage**, une fonction built-in de Pig, pour charger les données des films en tant que fichier texte structuré.

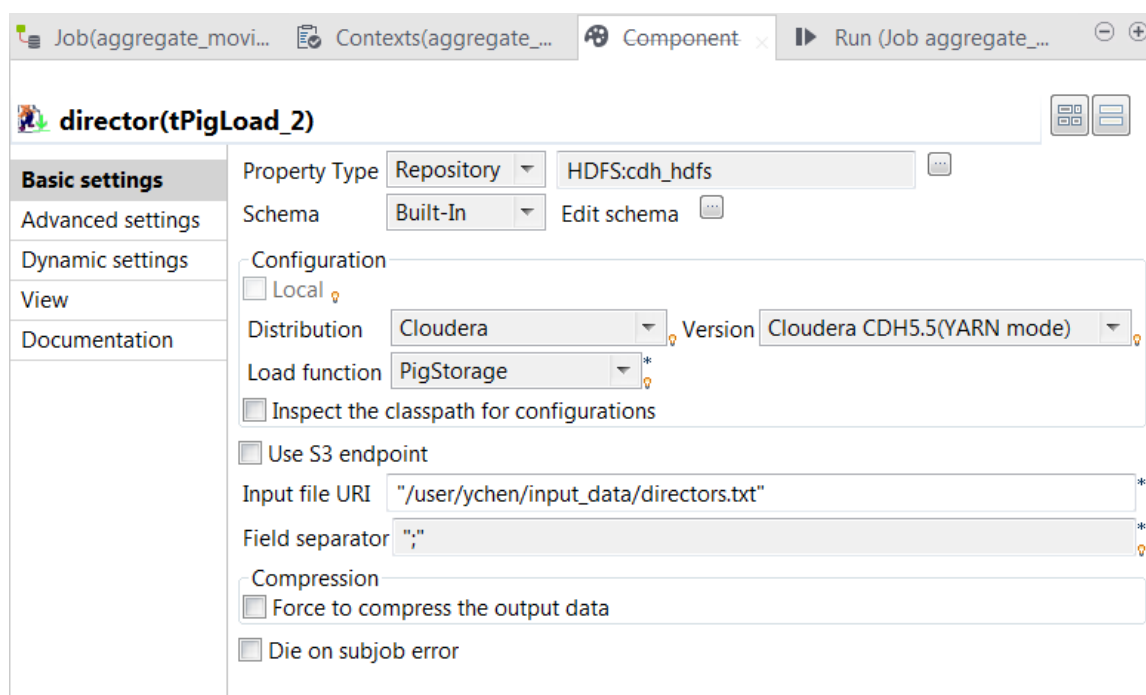
Pour plus d'informations concernant la fonction PigStorage de Pig, consultez [PigStorage](#) (en anglais).

5. À partir du nœud de connexion Hadoop nommé *my_cdh* dans le **Repository**, déposez la connexion HDFS **cdh_hdfs** du dossier **HDFS** sur le composant **tPigLoad** nommé **director** dans l'espace de modélisation graphique du Job.

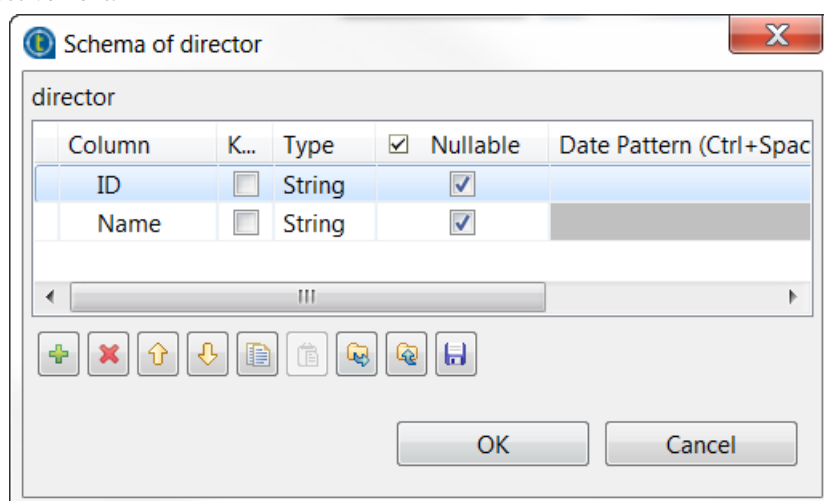
Cela permet d'appliquer la configuration de la connexion HDFS précédemment créée dans le **Repository** sur les paramètres relatifs à HDFS dans le composant **tPigLoad** courant.

6. Double-cliquez sur le composant *director tPigLoad* pour ouvrir sa vue **Component**.

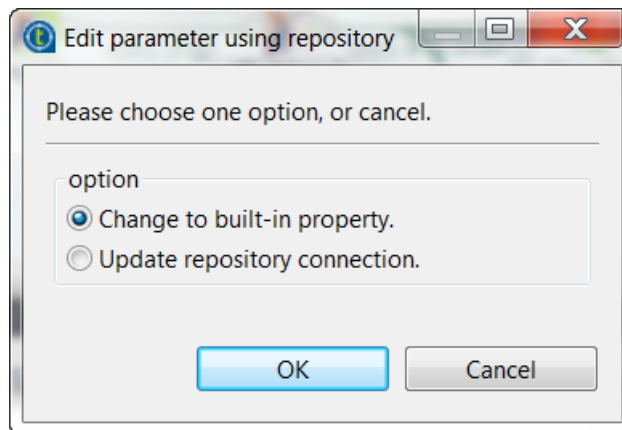
Ce composant **tPigLoad** a automatiquement réutilisé la configuration HDFS du **Repository** pour définir les paramètres associés dans la vue **Basic settings**.



7. Cliquez sur le bouton [...] à côté du champ **Edit schema** pour ouvrir l'éditeur du schéma.
8. Cliquez deux fois sur le bouton [+] pour ajouter deux lignes et, dans la colonne **Column**, renommez-les *ID* et *Name*, respectivement.



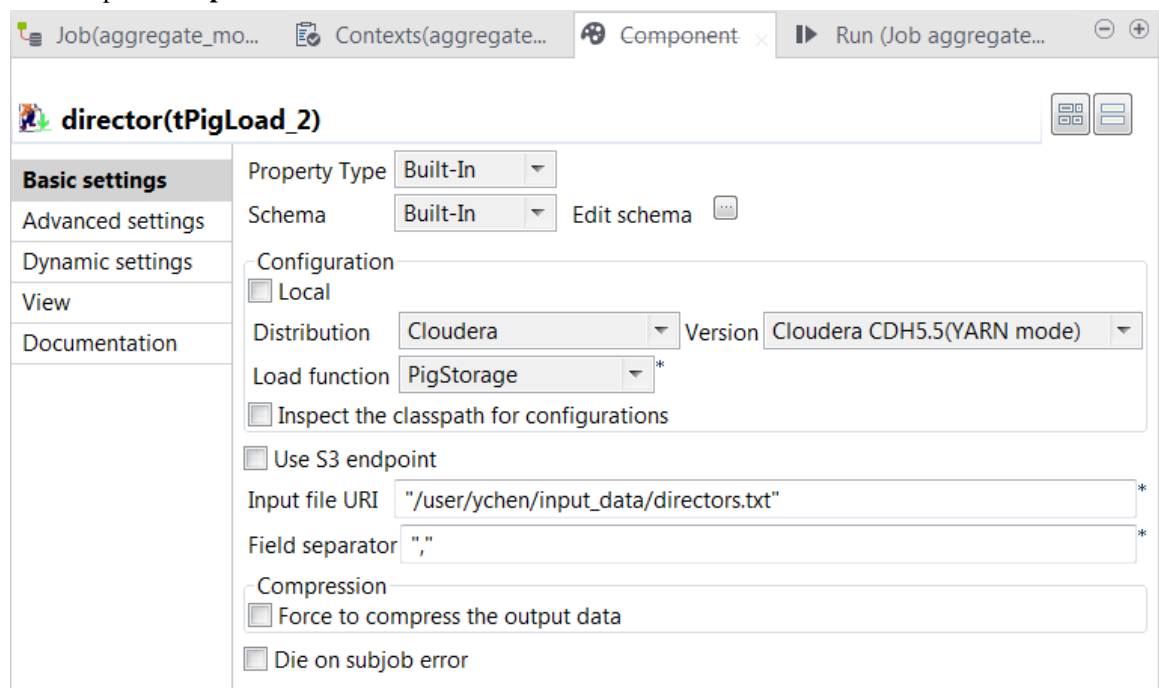
9. Cliquez sur **OK** pour valider ces modifications et acceptez la propagation proposée par la boîte de dialogue.
10. Dans la liste **Load function**, sélectionnez **PigStorage** pour utiliser la fonction PigStorage.
11. Dans le champ **Input file URI**, saisissez le chemin d'accès au répertoire où sont stockées les données relatives aux réalisateurs. Comme expliqué dans [Chargement des fichiers dans HDFS](#), ces données ont été écrites dans */user/ychen/input_data/directors.txt*.
12. Cliquez dans le champ **Field separator** pour ouvrir la boîte de dialogue **[Edit parameter using repository]** pour mettre à jour le séparateur de champs.



Vous devez modifier ce séparateur de champs car ce **tPigLoad** utilise le séparateur par défaut, un point-virgule (;), défini pour la métadonnée HDFS, alors que les données contiennent une virgule (,) comme séparateur.

13. Sélectionnez **Change to built-in property** puis cliquez sur **OK** pour valider votre choix.

Le champ **Field separator** devient modifiable.



14. Saisissez une virgule entre guillemets doubles.

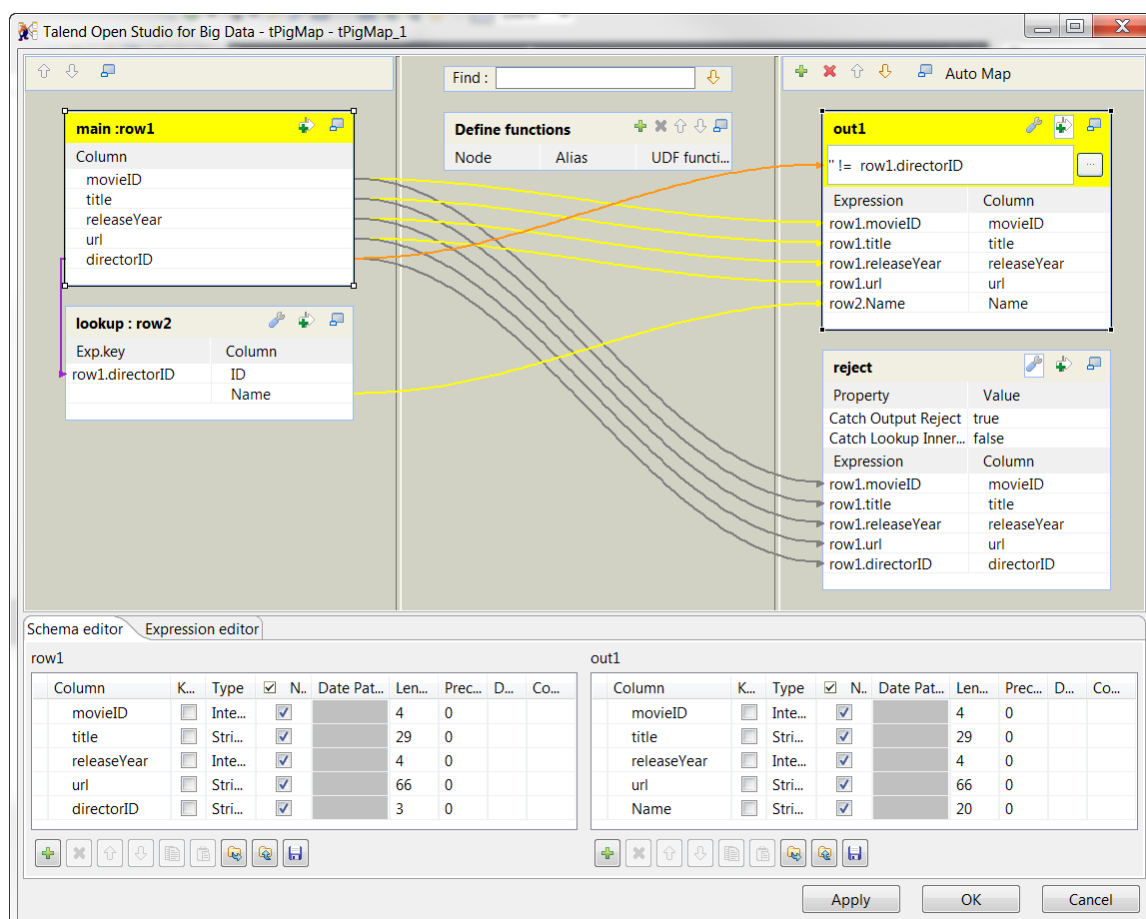
Les composants **tPigLoad** sont configurés pour charger les données des films et réalisateurs dans le Job.

5.1.4. Configurer la transformation de données

Dans ce scénario, le composant **tPigMap** est configuré pour effectuer une jointure sur les données des films et réalisateurs.

Une fois les données des films et réalisateurs chargées dans le Job, vous devez configurer le composant **tPigMap** pour qu'il effectue une jointure et produise la sortie attendue.

1. Double-cliquez sur le **tPigMap** pour ouvrir son éditeur **Map Editor**.



2. Déposez les colonnes *movieID*, *title*, *releaseYear* et *url* de gauche aux tables de sortie.

À gauche, dans le **Map Editor**, chacune de ces tables représente un flux d'entrée : celle du haut représente le flux principal et celle en-dessous le flux de référence.

Du côté droit, les deux tables représentent les flux de sortie nommés *out1* et *reject* lorsque vous avez relié le **tPigMap** au **tPigStoreResult** dans *Ajouter et relier les composants*.

3. Du côté de l'entrée, déposez la colonne *directorID* de la table du flux principal à la colonne **Expr.key** de la ligne *ID*, dans la table du flux de référence.

Ainsi, la clé de jointure entre le flux principal et le flux de référence est définie.

4. Déposez la colonne *directorID* de la table du flux principal à la table *reject* de sortie et déposez la colonne *Name* de la table de référence à la table de sortie *out1*.


La configuration des deux étapes précédentes décrit comment les colonnes des données d'entrée sont mappées aux colonnes du flux de sortie.

Dans l'onglet **Schema editor**, dans la partie inférieure de l'éditeur, vous pouvez voir que les schéma des deux côtés ont été automatiquement renseignés.

5. Dans la table de sortie *out1*, cliquez sur le bouton  pour afficher le champ d'expression de filtre.
6. Saisissez

```
'!= row1.directorID
```

Cela permet au **tPigMap** d'écrire en sortie uniquement les enregistrements de films dans lesquels le champ *directorID* n'est pas vide. Un enregistrement ayant un champ *directorID* vide ne sera pas écrit en sortie.

7. Dans la table de sortie *reject*, cliquez sur le bouton  pour ouvrir le panneau des paramètres.
8. Pour l'option **Catch Output Reject**, sélectionnez **true** pour écrire en sortie les enregistrements ayant un champ *directorID* vide dans le flux *reject*.
9. Cliquez sur **Apply**, puis sur **OK** afin de valider ces modifications et acceptez la propagation proposée par la boîte de dialogue.

La transformation est à présent configurée pour compléter les données des films en ajoutant le nom des réalisateurs et écrire les enregistrements relatifs aux films ne contenant pas de nom de réalisateur dans un flux de données séparé.

5.1.5. Écrire la sortie dans HDFS

Dans ce scénario, deux composants **tPigStoreResult** sont configurés pour écrire les données de films attendues et les données de films rejetées dans deux répertoires différents dans HDFS.

Prérequis :

- Vous devez avoir vérifié que la machine cliente sur laquelle les Jobs *Talend* sont exécutés peut reconnaître les noms d'hôtes du cluster Hadoop à utiliser. Dans cet objectif, ajoutez les mappings des entrées adresse IP/nom d'hôte pour les services de ce cluster Hadoop dans le fichier *hosts* de la machine cliente.

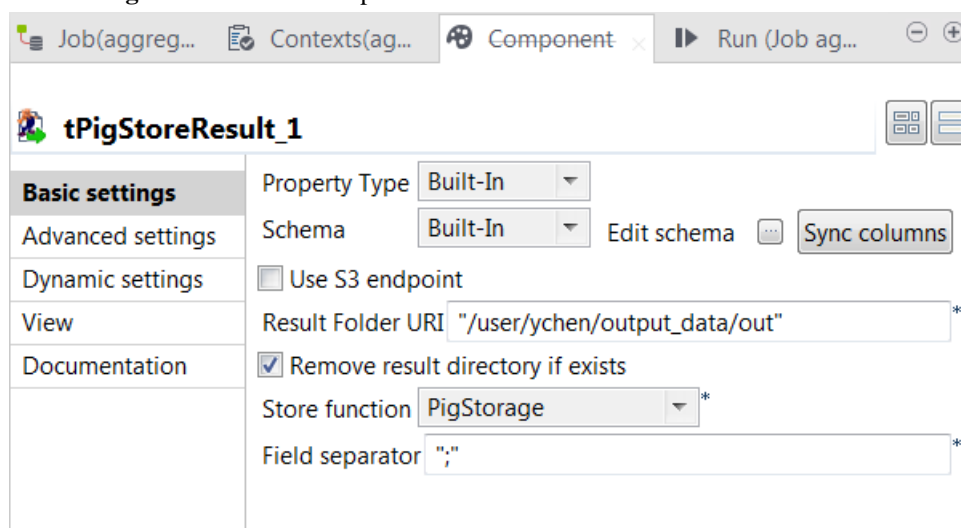
Par exemple, si le nom d'hôte du serveur du NameNode Hadoop est *talend-cdh550.weave.local* et son adresse IP est *192.168.x.x*, l'entrée du mapping est la suivante *192.168.x.x talend-cdh550.weave.local*.

- Le cluster Hadoop à utiliser doit avoir été configuré correctement et être en cours d'exécution.

Une fois les données des films et réalisateurs transformées par le **tPigMap**, vous devez configurer les deux composants **tPigStoreResult** pour écrire la sortie dans HDFS.

1. Double-cliquez sur le **tPigStoreResult** relié à l'aide du lien *out1*.

Sa vue **Basic settings** est ouverte dans la partie inférieure du Studio.

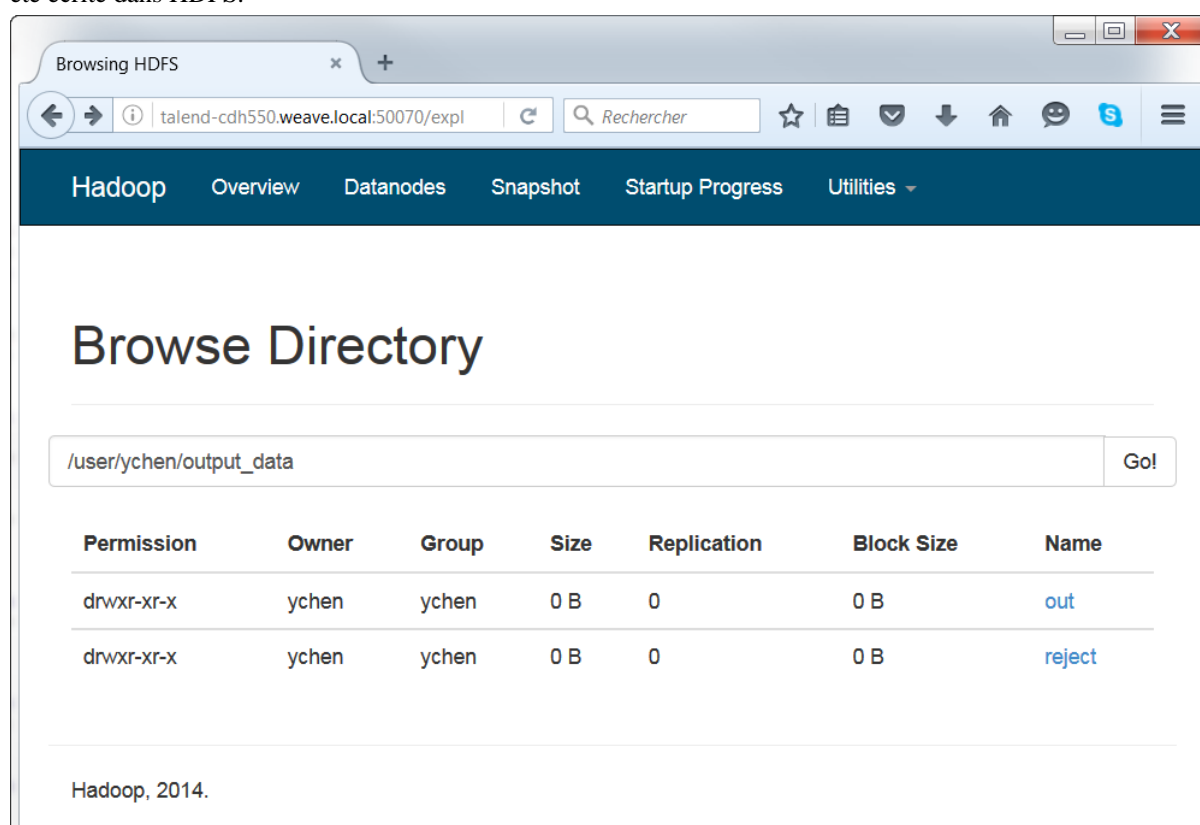


2. Dans le champ **Result file**, saisissez le chemin d'accès au répertoire dans lequel écrire les résultats. Dans ce scénario, saisissez */user/ychen/output_data/out*, le dossier recevant les enregistrements contenant les noms des réalisateurs.
3. Cochez la case **Remove result directory if exists**.

4. Dans la liste **Store function**, sélectionnez **PigStorage** pour écrire les enregistrements dans un format UTF-8 lisible par l'homme.
5. Dans le champ **Field separator**, saisissez ; entre guillemets doubles.
6. Répétez l'opération afin de configurer le **tPigStoreResult** relié à l'aide du lien *reject*, mais configurez le répertoire, dans le champ **Result file** à */user/ychen/output_data/reject*.
7. Appuyez sur **F6** pour exécuter le Job.

La vue **Run** s'ouvre automatiquement dans la partie inférieure du Studio et affiche l'avancement de l'exécution du Job.

Cela fait, vous pouvez vérifier, par exemple, dans la console Web de votre système HDFS, que la sortie a bien été écrite dans HDFS.



5.2. Que faire ensuite ?

Vous avez découvert que le *Studio Talend* vous permet de gérer vos Big Data à l'aide de Jobs *Talend*. Vous avez appris à accéder à vos données et à les déplacer dans un cluster Hadoop via le *Studio Talend*, les filtrer et les transformer, puis à stocker les données filtrées et transformées dans le système HDFS du cluster Hadoop. Vous avez également appris à centraliser les connexions à Hadoop fréquemment utilisées dans le **Repository** et les réutiliser facilement dans vos Jobs.

Pour en savoir plus au sujet du *Studio Talend*, consultez :

- le *Guide utilisateur du Studio Talend*
- le *Guide de référence des Composants Talend Open Studio for Big Data*

Pour vous assurer de la propreté de vos données, vous pouvez utiliser *Talend Open Studio for Data Quality* et *Talend Data Preparation*.

Pour en savoir plus au sujet des produits et solutions *Talend*, visitez fr.talend.com.

