



Tot plugin JDBC

Control de versiones	3
Modelo de integración	4
Extracción de metadatos	4
Muestreo de datos	5
Vendors disponibles	5
Oracle	6
Soporte	6
Configuración	6
Vendor	6
Formato de la conexión	6
Permisos	6
Extracción de metadatos	6
SQL Server	6
Soporte	6
Configuración	7
Vendor	7
Formato de la conexión	7
Permisos	7
Extracción de metadatos	7
PostgreSQL	7
Soporte	7
Configuración	7
Vendor	7
Formato de la conexión	7
Permisos	7
Snowflake	8
Soporte	8
Configuración	8
Vendor	8
Formato de la conexión	8
Permisos	8
Extracción de metadatos	8
Hive standalone	8
Soporte	8
Configuración	8
Vendor	8
Formato de la conexión	9
Permisos	9

Hive en entorno Cloudera	9
Soporte	9
Configuración	9
Vendor	9
Formato de la conexión	9
Permisos	9
DB2	9
Soporte	9
Configuración	10
Vendor	10
Formato de la conexión	10
Permisos	10
Extracción de metadatos	10
MySQL	10
Soporte	10
Configuración	10
Vendor	10
Formato de la conexión	10
Permisos	11
MariaDB	11
Soporte	11
Configuración	11
Vendor	11
Formato de la conexión	11
Permisos	11
Teradata	11
Soporte	11
Configuración	11
Vendor	11
Formato de la conexión	11
Permisos	12
Extracción de metadatos	12
Configuración	12
Ajustes específicos para Hive	13
Ajustes SSL de la JVM	13
Rutas proporcionadas	14
Añadir la CA autofirmada al contenedor de certificados de la JVM	14
Ajustes Kerberos de la JVM	14
Errores comunes	15
Nota	15

Control de versiones

Versión	Fecha de modificación	Responsable	Aprobador	Resumen de cambios
1.0	22/11/2023	Anjana Producto	Anjana Producto	Creación del documento. Compatibilidad con la v4.5 de todos los módulos de Anjana
1.1	22/12/2023	Anjana Producto	Anjana Producto	Se informa, en cada tecnología, acerca de cómo se obtiene el campo longitud
1.2	24/01/2024	Anjana Producto	Anjana Producto	Añadido Teradata
1.3	11/03/2024	Jose González	Ana Melcón	Cambiada configuración Vendors, añadida documentación de campos
1.4	22/03/2024	Jose González	Ana Melcón	Añadida url de alternativa de Teradata
1.5	04/04/2024	Jose González	Ana Melcón	Añadida explicación de url alternativa de Teradata

Modelo de integración

Extracción de metadatos

Se utilizan los métodos que ofrece el driver de JDBC mediante los cuales se accede a la definición de esquemas y tablas.

El plugin extrae los siguientes atributos que deben llamarse igual en la tabla `attribute_definition`, campo `name` para que aparezcan en la plantilla:

- **catalog** con el valor de catalog en la base de datos
- **schema** con el valor de schema en la base de datos
- **physicalName** y **name** con el mismo valor, el nombre de la tabla
- **path** con la concatenación de los valores de catalog, schema and table
- **infrastructure** con el valor seleccionado
- **technology** con el valor seleccionado
- **zone** con el valor seleccionado

También envía los siguientes atributos relativos a los campos del recurso pedido:

- **name** con el valor del campo correspondiente
- **physicalName** con el valor del campo correspondiente
- **defaultValue** con el valor por defecto definido para el campo correspondiente
- **fieldDataType** con el tipo de dato definido para el campo correspondiente
- **length** con el tamaño del campo correspondiente
- **incrementalField** indicando si es un campo incremental
- **position** posición que ocupa el campo correspondiente
- **precision** con el valor de la precisión del campo correspondiente
- **nullable** indicando si el campo correspondiente es nullable
- **pk** indicando si el campo es una pk
- **description** con el valor correspondiente para el campo

Los atributos a crear en Anjana deben de tener los siguientes tipos:

Nombre de atributo	Tipo de atributo
catalog	INPUT_TEXT
schema	INPUT_TEXT
physicalName	INPUT_TEXT
path	INPUT_TEXT
infrastructure	SELECT
technology	SELECT

zone	SELECT
name	INPUT_TEXT
defaultValue	INPUT_TEXT
fieldDataType	INPUT_TEXT
length	INPUT_NUMBER
incrementalField	INPUT_CHECKBOX
position	INPUT_NUMBER
precision	INPUT_NUMBER
nullable	INPUT_CHECKBOX
pk	INPUT_CHECKBOX
description	ENRICHED_TEXT_AREA_INTERNATIONAL

El plugin es capaz de realizar la extracción de metadatos de los siguientes tipos de elementos:

- Tabla de base de datos

Documentación genérica de Java sobre los campos disponibles en JDBC si algún driver no disponibiliza su catálogo de metadata
<https://docs.oracle.com/javadb/10.10.1.2/ref/rrefcrsrgpc1.html>

Muestreo de datos

Utilizando el driver genérico de JDBC de Java se ejecuta una query simple de SELECT para acceder a un número limitado de elementos de la tabla para recuperar una muestra de los datos almacenados. Adicionalmente se sustituyen los valores de los campos sensibles por asteriscos.

NOTA:

- Si la muestra de datos se hace sobre una tabla del esquema default de Hive (o database por defecto), Anjana puede mostrar algún error debido a la discrepancia del nombre de las columnas que gestiona el driver y el nombre de las mismas en una consulta select. Esta discrepancia impide que se asocie de manera correcta el valor con la columna representada en Anjana.

Vendors disponibles

Para cada vendor se detalla qué versiones se soportan, el formato de conexión, y que permisos se requieren para las funcionalidades del plugin (extracción y muestreo de datos).

Oracle

Soporte

Soporte de la versión 19.X hasta la 23.3, con compatibilidad hacia atrás a la 12.1 (sin embargo ya no tiene soporte oficial con versiones anteriores a la 19.X, no se certifica el completo funcionamiento).

Configuración

Vendor

ORACLE

Formato de la conexión

El formato de la url de conexión por BD o SID es
jdbc:oracle:thin:@<host>:<puerto>:<nombre_bbdd o sid>.

El formato de la url de conexión service name
jdbc:oracle:thin:@<host>:<port>/<service_name>.

Permisos

Usuario o rol con permisos SELECT o READ sobre las tablas o vistas que se quieran extraer metadatos u obtener un muestreo de datos. La diferencia es que el SELECT permite bloquear los registros durante el sampleo (si se configura la query para eso).

Si se quiere dar acceso a todas las tablas del sistema para gobernar, se puede dar el rol predefinido SELECT_CATALOG_ROLE.

Extracción de metadatos

A continuación se adjunta la documentación de Oracle donde se observan los distintos campos que se extraen de las tablas *:

<https://docs.oracle.com/javase/8/docs/api/java/sql/DatabaseMetaData.html#getColumns-java.lang.String-java.lang.String-java.lang.String-java.lang.String->

*Importante → Para algunos tipos de columnas la longitud observada en la tabla puede variar con respecto a la extraída por Anjana, revisar la documentación anteriormente indicada.

SQL Server

Soporte

Soporte desde SQL Server 2016 a SQL Server 2019.

Configuración

Vendor

MSSQLSERVER

Formato de la conexión

El formato de la url de conexión es

jdbc:sqlserver://<database host><instance_name>:<port>;database=<database_name>

Permisos

Usuario o rol con permisos SELECT sobre las tablas o vistas que se quieran obtener un muestreo de datos.

Usuario o rol con permisos VIEW DEFINITION sobre las tablas o vistas que se quieran extraer el metadata.

También se puede aplicar sobre esquemas o base de datos directamente y aplicará a todo lo que contiene.

Extracción de metadatos

A continuación se adjunta la documentación de SQLServer donde se observan los distintos campos que se extraen de las tablas *:

<https://learn.microsoft.com/es-es/sql/connect/jdbc/reference/getcolumns-method-sqlserverdatabasemetadata?view=sql-server-ver16>

*Importante → Para algunos tipos de columnas la longitud observada en la tabla puede variar con respecto a la extraída por Anjana, revisar la documentación anteriormente indicada.

PostgreSQL

Soporte

Soporte de la 8.2 hasta la versión 13.

Configuración

Vendor

POSTGRESQL

Formato de la conexión

El formato de la url de conexión es

jdbc:postgresql://<database_host>:<port>/database=<database>

Permisos

Usuario o rol con permisos USAGE sobre el esquema que contiene las tablas que se gobiernan y el permiso SELECT sobre cada tabla que se quiera hacer extracción de metadata y muestreo de datos.

Snowflake

Soporte

Al ser tecnología cloud, se soporta hasta la versión más reciente a la hora de la redacción de este documento, que es la versión de Julio de 2023.

Configuración

Vendor

SNOWFLAKE

Formato de la conexión

El formato de la url de conexión es

*jdbc:snowflake://<account>.snowflakecomputing.com/?db=<database>&warehouse=<wareh
use>*

Permisos

Usuario o rol con permisos USAGE sobre la base de datos y el esquema donde están las tablas/vistas que se quieren gobernar.

Usuario o rol con permisos REFERENCE sobre las tablas o vistas que se quieran extraer el metadata.

Usuario o rol con permisos SELECT sobre las tablas o vistas que se quieran obtener un muestreo de datos.

Extracción de metadatos

A continuación se adjunta la documentación de Snowflake donde se observan los distintos campos que se extraen de las tablas *:

<https://docs.snowflake.com/en/developer-guide/jdbc/jdbc-api#object-databasemetadata>

*Importante → Para algunos tipos de columnas la longitud observada en la tabla puede variar con respecto a la extraída por Anjana, revisar la documentación anteriormente indicada.

Hive standalone

Soporte

Soporte con hive 2.3.9.

Configuración

Vendor

HIVE_APACHE

Formato de la conexión

El formato de la url de conexión directo a hive server es

`jdbc:hive2://<database_host>:<database_port>/<database>;principal=<principal>;ssl=true.`

El formato de la url de conexión por HTTP es

`jdbc:hive2://<database_host>:<database_port>/<database>;principal=<principal>;ssl=true;transportMode=http;httpPath=cliservice.`

Permisos

Usuario o rol con permisos SELECT sobre las tablas o vistas que se quieran obtener un muestreo de datos o extracción de metadata.

Hive en entorno Cloudera

Soporte

Soporte en entornos cdh desde 6.0 a 6.3 o cdp 7.0 y 7.1 con hive 1.0 hasta 3.1.

Configuración

Vendor

HIVE_CLOUDERA

Formato de la conexión

El formato de la url de conexión directo a hive server es

`jdbc:hive2://<database_host>:<database_port>/<database>;principal=<principal>;ssl=true.`

El formato de la url de conexión por HTTP es

`jdbc:hive2://<database_host>:<database_port>/<database>;principal=<principal>;ssl=true;transportMode=http;httpPath=cliservice.`

El formato de la url de conexión por zookeeper es

`jdbc:hive2://<database_host>:<database_port>/<database>;principal=<principal>;ssl=true;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=hiveserver2.`

Permisos

Usuario o rol con permisos SELECT sobre las tablas o vistas que se quieran obtener un muestreo de datos o extracción de metadata.

DB2

Soporte

Soporte desde versiones 10.1 hasta 11.5.

Configuración

Vendor

DB2

Formato de la conexión

El formato de la url de conexión es
jdbc:db2://<database_host>:<port>/<database>.

Permisos

En el caso de que no se quiera dar acceso al esquema completo el usuario o rol que se utilice necesita el permiso SELECT sobre las tablas sobre las que se quiera extraer metadata o hacer muestreo de datos.

A partir de la versión 11.5 se pueden dar permisos sobre un esquema completo para poder extraer metadata o muestreo de datos, se necesita un usuario o rol con el permiso SELECTIN sobre el esquema.

Extracción de metadatos

A continuación se adjunta la documentación de IBM para db2 donde se observan los distintos campos que se extraen de las tablas *:

<https://www.ibm.com/docs/en/db2-for-zos/12?topic=functions-sqlcolumns-get-column-information>

*Importante → Para algunos tipos de columnas la longitud observada en la tabla puede variar con respecto a la extraída por Anjana, revisar la documentación anteriormente indicada.

MySQL

Soporte

Soporte de MySQL 5.7 y 8.0.

Configuración

Vendor

MYSQL

Formato de la conexión

El formato de la url de conexión es
jdbc:mysql://<database_host>:<port>/<database>.

Para el driver de mysql los schemas son catalogs y viceversa, es decir, para crear schemas hay que configurar la propiedad de using-catalogs a true y la de using-schemas a false.

Permisos

Usuario o rol con permisos SELECT sobre las tablas o vistas que se quieran extraer metadatos u obtener un muestreo de datos.

MariaDB

Soporte

Soporte desde 5.5.3 a 11.0.

Configuración

Vendor

MARIADB

Formato de la conexión

El formato de la url de conexión es

jdbc:mariadb://<database_host>:<port>/<database>.

Para el driver de mariadb los schemas son catalogs y viceversa, es decir, para crear schemas hay que configurar la propiedad de using-catalogs a true y la de using-chemas a false.

Permisos

Usuario o rol con permisos SELECT sobre las tablas o vistas que se quieran extraer metadatos u obtener un muestreo de datos.

También se puede aplicar sobre la base de datos directamente o incluso el servidor entero y aplicará a todo lo que contiene.

Teradata

Soporte

Soporte desde 16.20 hasta 20.0

Configuración

Vendor

TERADATA

Formato de la conexión

El formato de la url de conexión es

jdbc:teradata://<database_host>:<port>/<params>

En ocasiones el puerto o la base de datos puede ser un problema y se puede usar la url en este formato *jdbc:teradata://<database_host>/DATABASE=<database>,DBS_PORT=<port>*

Permisos

Usuario o rol con permisos SELECT sobre las base de datos, tablas o vistas que se quieran extraer metadatos u obtener un muestreo de datos.

En el caso que la tabla contenga UDTs será necesario que el usuario tenga permisos también sobre los procedimientos correspondiente (SYSUDTLIB.X, siendo X el nombre del procedimiento)

Extracción de metadatos

A continuación se adjunta la documentación de Teradata donde se observan los distintos campos que se extraen de las tablas *:

<https://teradata-docs.s3.amazonaws.com/doc/connectivity/jdbc/reference/current/frameset.html>

*Importante → Para algunos tipos de columnas la longitud observada en la tabla puede variar con respecto a la extraída por Anjana, revisar la documentación anteriormente indicada.

Configuración

Aquí se incluye el detalle de la configuración específica del plugin.

En la Guía de Configuración técnica se explica la configuración común.

```
server:
  port: 15001

totplugin:
  server:
    urls:
      - http://totserver:15000/tot/
  connection:
    vendor: POSTGRESQL
    url: jdbc:postgresql://rdbservice:5432/anjana
    user: anjana
    password: *****
    path-separator: "/"
    using-catalogs: false
    using-schemas: true

  aris:
    - ari: "anja:totplugin:sample:/jdbc/Postgresql/postgresql/"
    - ari: "anja:totplugin:extract:/jdbc/Postgresql/postgresql/"
```

```
groupPrefix: Dsa_  
  
eureka:  
  client:  
    serviceUrl:  
      defaultZone: http://totserver:15000/tot/eureka
```

- Si “using-catalogs” y “using-schemas” están a false, solo muestran el schema por defecto o el elegido en la url de conexión.
- Using-schemas lista todas las tablas de todos los esquemas que tenga acceso.
- Using-catalogs lista todas las tablas de todos los schemas de todas las bases de datos que vea.
- Path-separator va a indicar el separador utilizado por parte de Anjana para el path.
- Query-separator sirve para concatenar el nombre del datasource, con el schema y con la tabla cuando se hace la query de sample data.
Por ejemplo, si en un dataset de Anjana se define el catálogo como “catalog”, el schema como “schema”, el nombre de la tabla como “table” y la propiedad *query-separator* es “.”, Anjana genera la siguiente query para obtener el sampleo de los datos (ejemplo para la base de datos Postgresql):
SELECT * FROM catalog.schema.table FETCH FIRST {1} ROWS ONLY
- El parámetro groupPrefix indica el prefijo que tiene el nombre del grupo.

Ajustes específicos para Hive

Para la ejecución de consultas JDBC contra Hive se tienen que realizar una serie de ajustes en la JVM

- Uso de autorización Kerberos
- SSL

Ajustes SSL de la JVM

Cloudera necesita securizar las conexiones con SSL para poder utilizar Kerberos. La solución que utiliza por defecto es desplegar su propia CA autofirmada en un contenedor JKS utilizando unas propiedades análogas a estas:

```
-Djavax.net.ssl.keyStore=/...../...../cm-auto-host_keystore.jks  
-Djavax.net.ssl.keyStoreType=jks  
-Djavax.net.ssl.trustStore=/...../...../cm-auto-global_truststore.  
jks  
-Djavax.net.ssl.trustStoreType=jks
```

Hay dos opciones

- Que las rutas sean proporcionadas al integrador por el Cliente
- Añadir la CA autofirmada de Cloudera al contenedor de certificados por defecto de la JVM

Rutas proporcionadas

La JVM del plugin se deberá ejecutar con los siguientes parámetros:

Parámetro	Valor
-Djavax.net.ssl.keyStore	Ruta donde se encuentra el almacén de claves privadas.
-Djavax.net.ssl.keyStoreType	jks
-Djavax.net.ssl.trustStore	Ruta donde se encuentra el almacén de certificados.
-Djavax.net.ssl.trustStoreType	jks

Añadir la CA autofirmada al contenedor de certificados de la JVM

Lo que se persigue es añadir en cacerts de Java el certificado autofirmado de Cloudera

```
keytool -importkeystore -srckeystore
/var/lib/cloudera-scm-agent/agent-cert/cm-auto-global_truststore.jks
-s -destkeystore /etc/ssl/certs/java/cacerts -srcstorepass
-deststorepass changeit -v
```

Este sistema tiene la ventaja de que no precisería añadir parámetros adicionales a la JVM.

Ajustes Kerberos de la JVM

Se necesita proporcionar a la JVM el mecanismo para la utilización automática de los tickets de Kerberos.

Para ello debemos lanzar la JVM con los siguientes parámetros.

- -Djava.security.auth.login.config=security.config
- -Djavax.security.auth.useSubjectCredsOnly=false

Teniendo el fichero security.config la siguiente estructura:

```
com.sun.security.jgss.initiate {
    com.sun.security.auth.module.Krb5LoginModule required
    useTicketCache=true
    StoreKey=true
    useKeyTab=true
    keyTab="/home/ubuntu/usr_anjana.keytab"
    principal="usr_anjana@CDP.LOCAL";
};
```

Los parámetros keytab y principal se nos han de suministrar

Errores comunes

- Si las tablas de Hive están encriptadas devuelve este log de error
No Route to Host from ip-10-202-21-147.eu-west-1.compute.internal/10.202.21.147 to ip-10-202-21-122.eu-west-1.compute.internal:8020 failed on socket timeout exception: java.net.NoRouteToHostException

Nota

- Si, al ejecutar la extracción de metadato conectando con una base de datos de Oracle sale el siguiente error:

```
0:0:0:0::[localhost]:[1024]:[1] : Initializing Spring DispatcherServlet 'dispatcherServlet'
-1] oracle.simplefan.FanManager : attempt to configure ONS in FanManager failed with oracle.ons.NoServersAvailable: Server time out
-1] c.a.t.p.js.ExtractMetadataService : Exception on extractMetadata
java.lang.IllegalArgumentException: ONS configuration failed
    at oracle.simplefan.impl.FanManager.configure(FanManager.java:108) ~[simplefan-21.3.0.0.jar!/:21.3.0.0.0]
    at oracle.jdbc.driver.HAManager.configure(HAManager.java:146) ~[ojdbc8-12.2.0.1.jar!/:12.2.0.1.0]
    at oracle.jdbc.driver.HAManager.<init>(HAManager.java:122) ~[ojdbc8-12.2.0.1.jar!/:12.2.0.1.0]
    at oracle.jdbc.driver.HAManager.getInstance(HAManager.java:126) ~[ojdbc8-12.2.0.1.jar!/:12.2.0.1.0]
    at oracle.jdbc.driver.HAManager.enableHAIfNecessary(HAManager.java:725) ~[ojdbc8-12.2.0.1.jar!/:12.2.0.1.0]
    at oracle.jdbc.driver.PhysicalConnection.connect(PhysicalConnection.java:713) ~[ojdbc8-12.2.0.1.jar!/:12.2.0.1.0]
    at oracle.jdbc.driver.T4CDriverExtension.getConnection(T4CDriverExtension.java:39) ~[ojdbc8-12.2.0.1.jar!/:12.2.0.1.0]
    at oracle.jdbc.driver.OracleDriver.connect(OracleDriver.java:691) ~[ojdbc8-12.2.0.1.jar!/:12.2.0.1.0]
    at java.sql.DriverManager.getConnection(DriverManager.java:664) ~[na:1.8.0_352]
```

Es necesario incluir en el comando de ejecución del descriptor de servicio del plugin “-Doracle.jdbc.fanEnabled=false” (info del problema: https://support.oracle.com/cloud/faces/DocumentDisplay?_afLoop=190836230347481&_afWindowMode=0&id=2616175.1&_adf.ctrl-state=qk122q2vr_4)