talend | Connecting the Data-Driven Enterprise

Talend Real-time Big Data Platform

Installation Guide for Linux

6.4.1

Adapted for v6.4.1. Supersedes previous releases.

Publication date: June 29, 2017

Notices

End User License Agreement

Table of Contents

Preface

1. General information

1.1. Purpose

This Installation Guide explains how to install and configure the *Talend* modules and related applications. For detailed explanation on how to use and fine-tune the *Talend* applications, please refer to the appropriate Administrator or User Guides of the *Talend* solutions.

Information presented in this document applies to *Talend* products **6.4.1**.

1.2. Audience

This guide is for administrators and users of the *Talend* products.

The layout of GUI screens provided in this document may vary slightly from your actual GUI.

1.3. Typographical conventions

This guide uses the following typographical conventions:

- text in **bold:** window and dialog box buttons and fields, keyboard keys, menus, and menu options,
- text in **[bold]:** window, wizard, and dialog box titles,
- text in `courier`: system parameters typed in by the user,
- text in *italics*: file, schema, column, row, and variable names,
- The  icon indicates an item that provides additional information about an important point. It is also used to add comments related to a table or a figure,
- The  icon indicates a message that gives information about the execution requirements or recommendation type. It is also used to refer to situations or information the end-user needs to be aware of or pay special attention to.
- `Any command is highlighted with a grey background or code typeface.`

2. Feedback and Support

Your feedback is valuable. Do not hesitate to give your input, make suggestions or requests regarding this documentation or product and find support from the **Talend** team, on **Talend Community** at:

https://community.talend.com/

Chapter 1. Before installing your Talend product

These pages present and list everything you need to know before installing your *Talend* product:

- *Preparing your installation*
- *Hardware requirements*
- *Software requirements*

1.1. Preparing your installation

These pages provide information about:

- *Installation modes*
- *Files to download*
- *Community and Support*

1.1.1. Installation modes

There are two different methods to install your *Talend* product:

- the automatic mode, using *Talend Installer*. It is the recommended way of installing your *Talend* product. For more information, see *Installing your Talend product using Talend Installer (recommended)*.
- the manual installation. This method allows you to customize every step of your installation. For more information, see *Installing your Talend product manually*.

1.1.2. Files to download

Here are the files you need to download to install your *Talend* product:

- the licence file. For more information, see *License key*.
- the software packages. For more information, see *Software packages*.

1.1.2.1. License key

You should have received an email from *Talend* including your personal license key in a file with no extension.

The license key is mandatory to be able to access each module of *Talend*. **Keep this file at hand in a safe place.**

1.1.2.2. Software packages

This page details the software packages you need to download to install your *Talend* product.

In this page:

- YYYYMMDD_HHmm corresponds to the package timestamp
- A.B.C. corresponds to package version number (Major. Minor. Patch.)

The software modules must be all in the same versions/revisions. This means that both YYYYMMDD_HHmm and A.B.C must match on both client side and server side.

Table 1.1. Talend Installer software package

Zip file name	Description
Talend-Installer-Starter-YYYYMMDD_HHmm-**V**A.B.C-**installer.zip**+ *dist* file (for *Talend Platform* solutions)	Talend Installer: wizard-based application which guides you step by step through the installation and configuration of the *Talend Platform* modules. The *Talend Installer* package includes two files (a *.zip* and a *dist* file) which should be both downloaded and stored in the same place. The *dist* file is only required to install Talend products. Once the installation and configuration is complete, you can remove it.
TalendStudio-YYYYMMDD_HHmm-**V**A-B-C-**linux-x64-installer.run**	Talend Studio Installer: wizard-based application which guides you step by step through the installation of your *Talend Studio*. This package comes with an embedded Java Environment to make your installation easier.

Table 1.2. Manual installation software packages

Zip/jar file name	Description
Talend-Studio-YYYYMMDD_HHmm-**V**A.B.C**.zip**	CommandLine interface to the IDE + Studio IDE (GUI)
Talend-AdministrationCenter-YYYYMMDD_HHmm-**V**A.B.C**.zip**	*Talend Administration Center*: Web-based application used to administrate the Talend projects and users + Nexus artifact repository.
Talend-JobServer-YYYYMMDD_HHmm-**V**A.B.C**.zip**	JobServer: Standalone execution server
Talend-Runtime-VA.B.C-YYYYMMDDHHmm**.zip**	Talend Runtime: OSGi Container including JobServer. Talend Runtime is a standalone equivalent to the Talend ESB OSGi Container (*container* folder) of *Talend ESB*.
Talend-AMC_Web-YYYYMMDD_HHmm-**V**A.B.C**.zip**	*Talend Activity Monitoring Console* Web application used to monitor Talend Jobs and projects.
Talend-BRMS-YYYYMMDD_HHmm-**V**A.B.C**.zip**	Drools: Business Rules Management System based on Drools Workbench and Drools Expert.
Talend-DQPortal-YYYYMMDD_HHmm-**V**A.B.C**.zip**	*Talend Data Quality Portal*: portal which provides customizable web-based data quality monitoring and reporting.
Talend-DSC-YYYYMMDD_HHmm-**V**A.B.C**.jar**	The *Talend Data Stewardship Console* is deprecated since Talend 6.4. Consider migrating to *Talend Data Stewardship*. Talend Data Stewardship Console: legacy stewardship tool that helps to manage manually data resolution and data integration tasks that are the output of data matching and data integrity processes. Note that the availability of this tool depends on the options enabled in your license.
Talend-ESB-YYYYMMDD_HHmm-**V**A.B.C**.zip**	Talend ESB: application integration solution with an OSGi Container, Service Locator, Service Activity Monitoring and Security Token Service. It includes the Talend Runtime (in the *container* folder) and provides additional parts like examples, standalone, Tomcat deployment relevant parts and other additional parts primarily used by Java Developers.
DQDictServer-A.B.C**.zip**	*Talend Dictionary Service* stores semantic categories and propagates them to other products like *Talend Data Preparation* or *Talend Data Stewardship*. It can be enriched with personal semantic categories.
Talend-DataPreparation-Server-VA.B.C**.zip**	*Talend Data Preparation* enables information workers to cut hours out of their work day by simplifying and expediting the laborious and time-consuming process of preparing data for analysis or other data-driven tasks.

Zip/jar file name	Description
data-stewardship-dist-A.B.C**-GA-distribution.zip**	*Talend Data Stewardship*: a comprehensive tool you can use to configure and manage data assets and organize the interactions on data whenever human intervention is required.

The links to download these packages are listed in your licence email.

1.1.3. Community and Support

There are several ways to get help and support for your *Talend* installation:

- Official Talend Documentation. Here you can find everything to help you install and use your *Talend* product.
- Talend Community. This is the place where you can ask questions to the community, and get answers.
- Talend Professional Support. If you are a *Talend* subscription customer, you can open a ticket to the *Talend* Support.
- Talend Consulting Portal. If you are a *Talend* subscription customer, you can ask for a consultant to help through the installation of your *Talend* product.

1.2. Hardware requirements

Before installing your *Talend* product, make sure the machines you are using meet the following hardware requirements recommended by *Talend*.

Memory usage heavily depends on the size and nature of your *Talend* projects. However, in summary, if your Jobs include many transformation components, you should consider upgrading the total amount of memory allocated to your servers, based on the following recommendations.

Table 1.3. Memory usage

Product	Client/Server	Recommended alloc. memory
Talend Administration Center + *Talend Activity Monitoring Console* Web application	Server	4 GB minimum, 8 GB recommended
CommandLine	Server	2 GB minimum, 5 GB recommended
JobServer	Server	1 GB minimum, more recommended[1]
Studio	Client	3 GB minimum, 4 GB recommended
Talend Runtime	Server	2 GB minimum, 4 GB recommended[1]
Talend Data Quality Portal	Server	1 GB minimum, 1.5 GB recommended
Talend Data Preparation	Server	2 GB minimum, 4 GB recommended
Talend Data Stewardship	Server	1 GB minimum, 2 GB recommended
Talend Dictionary Service	Server	1 GB minimum, 2 GB recommended

1. Memory requirements depend on the executed processes.

The same requirements also apply for disk usage. It also depends on your projects but can be summarized as:

Table 1.4. Disk usage

Product	Client/Server	Required disk space for installation	Required disk space for use
Talend Administration Center with Nexus artifact repository + *Talend*	Server	800 MB + more than 50 MB for *Talend Activity*	800 MB minimum + project size = 20+ GB recommended

Product	Client/Server	Required disk space for installation	Required disk space for use
Activity Monitoring Console Web application		*Monitoring Console* Web application	
CommandLine	Server	3 GB	2 GB minimum + project size = 20+ GB recommended
JobServer	Server	20 MB	2 GB minimum + Jobs deployed = 20+ GB recommended
Studio	Client	3 GB	3+ GB
Talend Runtime	Server	400 MB	400+ MB
Talend Data Quality Portal	Server	1.5 GB	1.5+ GB
Talend Data Preparation	Server	300 MB	1 GB + datasets size These requirements do not take the MongoDB metadata size into account.
Talend Data Stewardship	Server	3 GB	100 MB for a campaign that counts 50,000 tasks, each task having 50 attributes.
Talend Dictionary Service	Server	1 GB	1+ GB

Ulimit settings on Unix systems

- To make the most out of the *Talend* server modules and improve performance on Unix systems, you should set the limit of system resources (ulimit) to unlimited.

1.3. Software requirements

These pages contain the exhaustive list of the databases and third party software that are compatible and supported with the 6.4.1 version of your *Talend* product.

- *Compatible Operating Systems*
- *Java*
- *Compatible Apache software and JMS Brokers for Talend ESB*
- *Compatible web application servers and containers*
- *Compatible Web browsers*
- *Compatible version control systems*
- *Compatible Databases*
- *Compatible Messaging Systems*
- *Compatible Nexus Artifact Repository*
- *Compatible execution servers*
- *Talend Data Preparation and Talend Administration Center compatibility matrix*
- *Port information*
- *Database privileges for Talend Administration Center*
- *Database privileges for Talend Data Quality Portal*
- *Installing the XULRunner package*

1.3.1. Compatible Operating Systems

In the following documentation:

- recommended: designates an environment recommended by *Talend* based on our experiences and customer usage;
- supported: designates a supported environment for use with the listed component or service;
- supported with limitations: designates an environment that is supported by *Talend* but with certain conditions explained in notes.

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

These tables provide a summary of the supported Operating Systems.

Table 1.5. Talend Studio

Support type	**Operating System (64-bit)**	
Recommended	Linux	Ubuntu 16.04 LTS
	Windows	Microsoft Windows Professional 7
Supported	Linux	Ubuntu 17.04
		Ubuntu 14.04 LTS
		Red Hat Enterprise Linux Server/CentOS 7.3
		Red Hat Enterprise Linux Server/CentOS 7.2
		Red Hat Enterprise Linux Server/CentOS 7.1
		Red Hat Enterprise Linux Server/CentOS 6.8
		Red Hat Enterprise Linux Server/CentOS 6.7
	Windows	Microsoft Windows 10
		Microsoft Windows 8.1
		Microsoft Windows Server 2016 RTM
		Microsoft Windows Server 2012 RTM
	Mac	OS X 10.12 Sierra
		OS X 10.11 El Capitan
		OS X 10.10 Yosemite

Table 1.6. Talend server modules

Support type	**Operating System**		**Processor**
Recommended	Linux	Red Hat Enterprise Linux Server 7.2	64-bit
	Windows	Microsoft Windows Server 2012 R2	64-bit
Supported	Linux	Ubuntu 17.04	64-bit
		Ubuntu 16.04 LTS	64-bit
		Ubuntu 14.04 LTS	64-bit
		Red Hat Enterprise Linux Server/CentOS 7.3	64-bit
		Red Hat Enterprise Linux Server/CentOS 7.1	64-bit
		Red Hat Enterprise Linux Server/CentOS 6.9	64-bit
		Red Hat Enterprise Linux Server/CentOS 6.8	64-bit
		SUSE SLES 12	64-bit
		SUSE SLES 11	64-bit
	Unix	Solaris (SunOs) 11	x86/64-bit[1]
		Solaris (SunOs) 11	Sparc/64-bit[2]

Support type	Operating System		Processor
		Solaris (SunOs) 10[3]	x86/64-bit[1]
		Solaris (SunOs) 10[3]	Sparc/64-bit[2]
	Windows	Microsoft Windows Server 2016	64-bit
		Microsoft Windows Server 2012	64-bit
Deprecated	Unix	AIX 7.1	64-bit (IBM Java 8 only)[4]

1. Only supported for Talend Administration Center, CommandLine, JobServer, Talend ESB and Talend Runtime.

2. Only supported for Talend ESB and Talend Runtime.

3. At least patch level 9 should be installed.

4. Only supported for CommandLine, JobServer, Talend ESB and Talend Runtime.

The server modules include Artifact Repository, CommandLine, JobServer, Talend Activity Monitoring Console, Talend Administration Center, Talend Data Preparation, Talend Data Quality Portal, Talend Data Stewardship, Talend ESB, Talend Log Server, Talend Repository Manager, Talend Runtime.

Note that *Talend Metadata Bridge* is only supported on Windows.

1.3.1.1. Statement regarding Virtualization and Docker deployments

In general, *Talend* supports deployment on virtual machines. For Virtualization Systems, *Talend* relies on the vendors' operating-system compatibility statements.

Talend does not deliver prepackaged Docker Images for the *Talend* Servers, and cannot maintain a standard setup for customer-based Docker environments, so standard Service Level Agreements do not apply.

For any customer issue which also can be reproduced in a non-Docker environment on a supported platform, Support Service Level can be applied as usual. For any issue which only occurs in a customer-composed Docker environment, *Talend* will only provide best effort to address any issues that arise.

1.3.2. Java

In order to use your *Talend* product, Java must be installed on your machine. If you install your *Talend Studio* using the *Talend Studio Installer*, you do not need to set up a Java Environment as it is embedded in the Installer.

These pages list:

- *Compatible Java environments*
- *Setting up JAVA_HOME*

1.3.2.1. Compatible Java environments

In the following documentation:

- recommended: designates an environment recommended by *Talend* based on our experiences and customer usage;
- supported: designates a supported environment for use with the listed component or service;
- supported with limitations: designates an environment that is supported by *Talend* but with certain conditions explained in notes.

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

These tables provide a summary of the supported Java Runtime environments.

In this table:

- ✅ **(R)** means that this combination is recommended;
- ✅ means that this combination is supported;
- ❌ means that this combination is not supported.

Note that only the 64-bit versions of the compatible Java Runtime environments are supported.

The **Compiler Compliance Level** corresponds to the Java version used for the Job code generation. This option can be changed in the project settings of the Studio. For more information, see *Talend Studio User Guide*.

Table 1.7. Studio Java environments

Support Type	JRE Version	Studio JDK Compiler Compliance Level	Notes
Recommended	Oracle 8	1.8 (default)	
Supported	Oracle 8	1.7	Needs to be switched to manually. Only supported for Big Data Distributions requiring it. Routes are not supported with JDK Compiler Compliance level 1.7.

Depending on the license you have, the available Execution Servers may differ.

Table 1.8. Server Java environments

JRE Version	JobServer	MDM Server	ESB/Talend Runtime	ESB/ Microservices	Big Data Distributions	Talend Server Applications[1]	Comment/ Limitation
Oracle 7 (running alongside Oracle 8)	❌	❌	❌	❌	Compatible with Java 1.7 only	❌	Compatible with Studio JDK Compiler Compliance Level 1.7 only
Oracle 8 (recommended)	✅ **(R)**	✅ **(R)**	✅ **(R)**	✅ **(R)**	Compatible with Java 1.8	✅ **(R)**	Compatible with Studio JDK Compiler Compliance Level 1.7 (default/ recommended) or 1.8
IBM 8 (deprecated)	✅	❌	❌	❌	❌	❌	Only for AIX and SUSE SLES

1: Talend server applications include Artifact Repository, CommandLine, Talend Activity Monitoring Console, Talend Administration Center, Talend Data Preparation, Talend Data Quality Portal, Talend Data Stewardship, Talend Log Server, Talend Repository Manager.

For example, the recommended combination is:

- Oracle 8 installed on the machine running the Studio;
- The **Compiler Compliance Level** set to 1.7 in the project settings of the Studio;
- Oracle 8 installed on the machine(s) running the Execution Server(s) and the Talend Server Application(s);
- Big Data Distributions compatible with Java 1.8 used.

For more information on Java specificities (version, Operating Systems compatibility), see Talend Help Center and Talend Community.

1.3.2.2. Setting up JAVA_HOME

In order for your *Talend* product to use the Java environment installed on your machine, you must set the JAVA_HOME environment variable.

To do so, proceed as follows:

1. Find the folder where Java is installed, usually */usr/lib/jvm/java-x-oracle*.
2. Open a terminal.
3. Use the `export` command to set the JAVA_HOME and Path variables.

   For example:

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

4. Add these lines at the end of the global profiles in the */etc/profile* file or in the user profiles in the *~/.profile* file.

   After changing one of these files you have to log on again.

1.3.3. Compatible Apache software and JMS Brokers for Talend ESB

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

Some of the ESB tools use Apache software components. Talend ESB contains the following Apache Project versions:

Table 1.9. Supported Apache software

Software	Notes	More information
Apache Karaf 4.1.1	Service release upgrade.	Release notes: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12311140&version=12339244

Software	Notes	More information
Apache CXF 3.1.11	Service release upgrade.	Release notes and Migration Guide: http://cxf.apache.org/cxf-3111-release-notes.html
Apache Camel 2.17.6	Minor release upgrade.	Release notes: http://camel.apache.org/camel-2176-release.html
Apache ActiveMQ 5.14.5	Minor release upgrade.	Release notes: http://activemq.apache.org/activemq-5145-release.html

Talend ESB supports the following JMS Brokers.

Note that ESB Java Consumer & Provider using SOAP/JMS-based messaging are supported for the following (JMS) Message-Brokers.

Table 1.10. Supported Messaging Brokers for SOAP/JMS

Software	More information
Apache ActiveMQ 5.14.5	Release notes: http://activemq.apache.org/activemq-5145-release.html
IBM WebSphere MQ 7.5	Release notes: http://www-01.ibm.com/support/docview.wss?uid=swg27043190

Table 1.11. Supported Containers

Software	Product
Jetty 9.3.14	ESB Runtime - OSGi Container
Apache Tomcat 8.0.44	ESB Microservices

1.3.4. Compatible web application servers and containers

In the following documentation:

- recommended: designates an environment recommended by *Talend* based on our experiences and customer usage;
- supported: designates a supported environment for use with the listed component or service;
- supported with limitations: designates an environment that is supported by *Talend* but with certain conditions explained in notes.

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

Please refer to the following grids for a summary of supported web application servers and runtime containers.

Table 1.12. Talend Administration Center, Talend Activity Monitoring Console and Talend Repository Manager

Support type	Web application servers
Recommended	Apache Tomcat 8.0[1]
Supported	Apache Tomcat 7[1]
	Pivotal tc Server 3.2
Deprecated	JBoss EAP 6.4
	IBM Websphere 8.5

Support type	Web application servers
	Weblogic 12c

1. TLS 1.2 is supported. For more information, see https://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html.

Table 1.13. Talend Data Quality Portal

Support type	Web application servers
Recommended	Apache Tomcat 8.0[1]
Supported	Apache Tomcat 7[1]
	Pivotal tc Server 3.2

1. TLS 1.2 is supported. For more information, see https://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html.

Table 1.14. Talend Data Stewardship

Support type	Web application servers
Recommended	Apache Tomcat 8.0[1]

1. TLS 1.2 is supported. For more information, see https://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html.

Table 1.15. Talend Dictionary Service

Support type	Web application servers
Recommended	Apache Tomcat 8.0[1]

1. TLS 1.2 is supported. For more information, see https://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html.

Table 1.16. Talend ESB

Support type	Runtime Containers
Recommended	Talend Runtime (Apache Karaf) 6.4[1]
	Apache Tomcat 8.0[25]
Supported with limitations	Apache Tomcat 7 and 8.0[35]
Deprecated	JBoss EAP 6.4[4]
	Weblogic 12c[4]
	IBM Websphere 8.5[4]

1. Except for Talend Identity Management Service, where Tomcat Apache 8 is recommended.

2. Only for Talend Identity Management

3. Only for CXF Services, Camel Routes, Service Activity Monitoring, Talend Identity Management and Security Token Service.

4. Only for CXF Services and Camel Routes.

5. TLS 1.2 is supported. For more information, see https://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html.

1.3.5. Compatible Web browsers

In the following documentation:

- recommended: designates an environment recommended by *Talend* based on our experiences and customer usage;
- supported: designates a supported environment for use with the listed component or service;
- supported with limitations: designates an environment that is supported by *Talend* but with certain conditions explained in notes.

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

Please refer to the table below for a summary of supported Web browsers.

Talend Administration Center, Drools, Talend Data Preparation, Talend Data Stewardship	**Web browser**
recommended	Mozilla Firefox 51 and 53[1]
supported	Firefox ESR 45 and 52
	Microsoft Internet Explorer 11
	Microsoft Edge
	Apple Safari 10
	Google Chrome 53 to 58[1]

1. Talend maintenance releases will support the most recent browser version at the time of the release.

1.3.6. Compatible version control systems

In the following documentation:

- recommended: designates an environment recommended by *Talend* based on our experiences and customer usage;
- supported: designates a supported environment for use with the listed component or service;
- supported with limitations: designates an environment that is supported by *Talend* but with certain conditions explained in notes.

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

The following version control clients are embedded in the *Talend* products:

- Apache Subversion (version 1.8)
- Git (JGit/EGit 3.4.2)

The tables below provide a summary of the supported version control system servers that you can use to store your projects.

Table 1.17. Apache Subversion version control servers

Support type	**Version control servers**
recommended	VisualSVN Server 3.3 (compatible with Apache Subversion 1.8)
supported	VisualSVN Server 3.5 (compatible with Apache Subversion 1.9)
	Bitnami Subversion Stack (compatible with Apache Subversion 1.8)

Support type	Version control servers
	Svnserve (compatible with Apache Subversion 1.8)
	SVNEdge 5.1

Table 1.18. Git version control servers

Support type	Version control servers
recommended	GitHub (SaaS)
supported	BitBucket (SaaS)
	AWS CodeCommit (SaaS)
	GitLab 9.1
	GitLab 8.13
	Gitblit 1.7.1

1.3.7. Compatible Databases

In the following documentation:

- recommended: designates an environment recommended by *Talend* based on our experiences and customer usage;
- supported: designates a supported environment for use with the listed component or service;
- supported with limitations: designates an environment that is supported by *Talend* but with certain conditions explained in notes.

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

Please refer to the following grids for a summary of supported databases.

Table 1.19. Talend Administration Center

Support type	Databases
Recommended	MySQL 5.7[1]
	Oracle 12c[1]
Supported	Azure SQL
	H2 1.4
	MariaDB 10.1
	MS SQL Server 2016[1]
	MS SQL Server 2014
	MS SQL Server 2012 (SP2)
	MySQL 5.6
	Oracle 11g
	PostgreSQL 9.6[12]
	PostgreSQL 9.4 and 9.5[3]

1. The corresponding Amazon Relational Database Service (Amazon RDS) version is supported.

2. The corresponding Google Cloud SQL version is supported.

3. PostgreSQL JDBC driver 9.4 is used.

Table 1.20. Talend Identity and Access Management

Support type	Databases
Recommended	MySQL 5.7[1]
	Oracle 12c[1]
Supported	Azure SQL
	H2 1.4
	Derby DB > 10.8
	MS SQL Server 2016[1]
	MS SQL Server 2014
	MS SQL Server 2012 (SP2)
	MySQL 5.6
	Oracle 11g
	PostgreSQL 9.6[12]
	PostgreSQL 9.4 and 9.5[3]

1. The corresponding Amazon Relational Database Service (Amazon RDS) version is supported.

2. The corresponding Google Cloud SQL version is supported.

3. PostgreSQL JDBC driver 9.4 is used.

Table 1.21. Talend Activity Monitoring Console

Support type	Databases
Recommended	MySQL 5.7[1]
	Oracle 12c[1]
Supported	IBM DB2 10.1
	MS SQL Server 2016[1]
	MS SQL Server 2014
	MS SQL Server 2012 (SP2)
	MySQL 5.6
	Oracle 11g
	PostgreSQL 9.6[12]
	PostgreSQL 9.4 and 9.5[3]

1. The corresponding Amazon Relational Database Service (Amazon RDS) version is supported.

2. The corresponding Google Cloud SQL version is supported.

3. PostgreSQL JDBC driver 9.4 is used.

Table 1.22. Service Activity Monitoring (SAM)

Support type	Databases
Recommended	MySQL 5.7[1]
	Oracle 12c[1]
Supported	Derby DB > 10.8
	IBM DB2 10.1
	MS SQL Server 2016[1]
	MS SQL Server 2014
	MS SQL Server 2012 (SP2)

Support type	Databases
	MySQL 5.6
	Oracle 11g
	PostgreSQL 9.6[12]
	PostgreSQL 9.4 and 9.5[3]

1. The corresponding Amazon Relational Database Service (Amazon RDS) version is supported.

2. The corresponding Google Cloud SQL version is supported.

3. PostgreSQL JDBC driver 9.4 is used.

Table 1.23. ESB Service Registry/Authorization/Talend Identity Management/Event Logging

Support type	Databases
Recommended	MySQL 5.7[1]
	Oracle 12c[1]
Supported	Derby DB > 10.8
	MS SQL Server 2016[1]
	MS SQL Server 2014
	MS SQL Server 2012 (SP2)
	MySQL 5.6
	Oracle 11g
	PostgreSQL 9.6[12]
	PostgreSQL 9.4 and 9.5[3]

1. The corresponding Amazon Relational Database Service (Amazon RDS) version is supported.

2. The corresponding Google Cloud SQL version is supported.

3. PostgreSQL JDBC driver 9.4 is used.

Table 1.24. Talend Data Quality Portal

Support type	Databases used to store the analyses
Recommended	MySQL 5.7[1]
	Oracle 12c[1]
Supported	MS SQL Server 2016[1]
	MS SQL Server 2014
	MS SQL Server 2012 (SP2)
	MySQL 5.6
	Oracle 11g
	PostgreSQL 9.6[12]
	PostgreSQL 9.4 and 9.5[3]

1. The corresponding Amazon Relational Database Service (Amazon RDS) version is supported.

2. The corresponding Google Cloud SQL version is supported.

3. PostgreSQL JDBC driver 9.4 is used.

Support type	Database used to store the portal configuration
Supported (embedded in the product)	HSQL 2.3.2

If you are using MySQL, the my.ini file of your MySQL server installation directory must be edited in order to add the following line:

```
max_allowed_packet = 64M
```

The `64M` value is only given as example. It represents the maximum size of a query packet the server can handle as well as the maximum query size the server can process. You can replace it with any value superior to `16M`.

Table 1.25. Talend Repository Manager

Support type	Databases
Recommended	MySQL 5.7[1]
	Oracle 12c[1]
Supported	Azure SQL
	H2 1.4
	MariaDB 10.1
	MS SQL Server 2016[1]
	MS SQL Server 2014
	MS SQL Server 2012 (SP2)
	MySQL 5.6
	Oracle 11g
	PostgreSQL 9.6[12]
	PostgreSQL 9.4 and 9.5[3]

1. The corresponding Amazon Relational Database Service (Amazon RDS) version is supported.

2. The corresponding Google Cloud SQL version is supported.

3. PostgreSQL JDBC driver 9.4 is used.

Table 1.26. Talend Data Preparation

Support type	Database
Recommended (embedded in the product)	MongoDB 3.4

Table 1.27. Talend Data Stewardship

Support type	Database
Recommended (embedded in the product)	MongoDB 3.4

Table 1.28. Talend Dictionary Service

Support type	Database
Recommended (embedded in the product)	MongoDB 3.4

1.3.8. Compatible Messaging Systems

In the following documentation:

- recommended: designates an environment recommended by *Talend* based on our experiences and customer usage;
- supported: designates a supported environment for use with the listed component or service;
- supported with limitations: designates an environment that is supported by *Talend* but with certain conditions explained in notes.

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

Please refer to the following grid for a summary of supported messaging systems.

Talend Data Preparation, Talend Data Stewardship,Talend Dictionary Service	**Messaging System**
recommended	Apache Kafka 0.10.1

1.3.9. Compatible Nexus Artifact Repository

In the following documentation:

- recommended: designates an environment recommended by *Talend* based on our experiences and customer usage;
- supported: designates a supported environment for use with the listed component or service;
- supported with limitations: designates an environment that is supported by *Talend* but with certain conditions explained in notes.

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

Please refer to the table below for the recommended Nexus Artifact Repository version that you can use to publish and store your artifacts.

Support type	**Artifact repository**
recommended[1]	Sonatype Nexus 2.14

1. Embedded in the Talend Administration Center archive file.

1.3.10. Compatible execution servers

Make sure the execution server version is compatible with the *Talend Administration Center*, CommandLine and Studio versions.

Note that the information contained in this section is valid at the date of publication, but may be subject to change at a later date.

Table 1.29. Job servers (JobServer and job server in Talend Runtime)

		Job Server	Job Server	Job Server	Job Server
	Version	6.1.x	6.2.x	6.3.x	6.4.x
Talend Administration Center, Studio, CommandLine	6.1.x	✔			
Talend Administration Center, Studio, CommandLine	6.2.x	✔	✔		
Talend Administration	6.3.x	✔	✔	✔	

Center, Studio, CommandLine					
Talend Administration Center, Studio, CommandLine	6.4.x	✔	✔	✔	✔

Table 1.30. ESB OSGI servers (Talend Runtime)

		Talend Runtime (ESB OSGI Server)	Talend Runtime (ESB OSGI Server)	Talend Runtime (ESB OSGI Server)	Talend Runtime (ESB OSGI Server)
	Version	6.1.x	6.2.x	6.3.x	6.4.x
Talend Administration Center, Talend ESB Infrastructure Services	6.1.x	✔			
Talend Administration Center, Talend ESB Infrastructure Services	6.2.x	✔	✔		
Talend Administration Center, Talend ESB Infrastructure Services	6.3.x	✔	✔	✔	
Talend Administration Center, Talend ESB Infrastructure Services	6.4.x	✔	✔	✔	✔

1.3.11. Talend Data Preparation and Talend Administration Center compatibility matrix

Make sure that your *Talend Administration Center* version is compatible with the *Talend Data Preparation* version that you are using.

Table 1.31. Talend Data Preparation and Talend Administration Center compatibility

	Talend Data Preparation 1.2	Talend Data Preparation 1.3	Talend Data Preparation 2.0	Talend Data Preparation 2.1
Talend Administration Center 6.2	✔	✔	✘	✘
Talend Administration Center 6.3	✘	✘	✔	✘
Talend Administration Center 6.4	✘	✘	✘	✔

1.3.12. Port information

This section describes the most important TCP/IP ports the *Talend* products use. Make sure that your firewall configuration is compatible with these ports or change the default ports where needed.

Additionally, add the web site http://talendforge.org/ and the port 80 to the whitelist. To be able to download patches or external libraries, also add the web site http://talend-update.talend.com and the port 443 to the whitelist.

In this table:

Port: a TCP/IP port or a range of ports.

Active: Active for a standard installation of the product (Standard Installation is defined here as **Server** or **Client** installation using Talend Installer with the default values provided in the Installer User Interface)

Direction: In (Inbound); Out (Outbound) - related to the communication direction (for example a HTTP Port for a CXF Route or Service we listen on request) will be an 'Inbound' port. For example, a browser which sends a request to port 7080 will have this port as 'Outbound' port in this list.

Usage: which part of the Product component uses this port (for example 1099 is used by the JMX Monitoring component of Talend Runtime).

Config: the file or location where the value can be changed.

Remark: anything which is important to mention additionally.

Table 1.32. Talend Studio and CommandLine Ports

Port	Direction	Usage	Config
8090 **Active**: N	IN	**tESBProviderRequest** (SOAP Data Server) and **tRESTRequest** (REST Data Service default port) components	REST: Preferences / Talend / ESB SOAP: **tESBProviderRequest** component details
Port	**Direction**	**Usage**	**Config**
8002 **Active**: Y	IN	CommandLine port	*commandline-linux.sh*
3334 to *4333* **Active**: Y	IN	*Talend Studio* live statistics	The values can be configured in *Talend Studio*.
4334 to *5333* **Active**: Y	IN	*Talend Studio* trace mode	The values can be configured in *Talend Studio*.

Table 1.33. Talend Administration Center Ports

Port	Direction	Usage	Config	Remark
8080 **Active**: Y	IN	Talend Administration Center Server - Apache Tomcat HTTP Port	*/conf/server.xml*	
8009 **Active**: Y	IN	Talend Administration Center Server - Apache Tomcat AJP Connector Port	*/conf/server.xml*	
(none) **Active**: Y*	OUT	Talend Administration Center Server - Database	Configuration Page in TAC Web-UI	* By default an embedded H2 Database is used (not network accessible). If another database should be used the port is related to the type and configuration of this database.

Table 1.34. Talend Data Preparation Ports

Port	Direction	Usage	Config
9999 **Active**: Y	IN	Talend Data Preparation User Interface port	*config/application.properties*
8989	IN OUT	Talend Data Preparation backend port	*config/application.properties*

Port	Direction	Usage	Config
Active: Y			
27017 **Active**: Y	IN OUT	MongoDB port	*<MongoDB>/mongod.cfg*

Table 1.35. Talend Data Stewardship Ports

Port	Direction	Usage	Config
19999 **Active**: Y	IN	Apache Tomcat HTTP Port	*tomcat/conf/server.xml*
19924 **Active**: Y	IN	Apache Tomcat Shutdown Port	*tomcat/conf/server.xml*
19928 **Active**: Y	IN	Apache Tomcat AJP Connector Port	*tomcat/conf/server.xml*
27017 **Active**: Y	IN OUT	MongoDB port	*<MongoDB>/mongod.cfg*
2181 **Active**: Y	IN OUT	Apache Zookeeper port	*<Kafka>/config/zookeeper.properties*
9092 **Active**: Y	IN OUT	Apache Kafka port	*<Kafka>/config/server.properties*

Table 1.36. Talend Log Server Ports

Port	Direction	Usage	Config
5601 **Active**: Y	IN	Talend logging module - Logstash port	*<TalendLogServerPath>/logstash-talend.conf*
8050 **Active**: Y	IN	Talend logging module - Talend Administration Center log4j port	*logstash-talend.conf*
8052 **Active**: Y	IN	Talend logging module - Talend Components log4j port	*logstash-talend.conf*
8053 and *8054* **Active**: Y	IN	Talend logging module - MDM log4j ports	*logstash-talend.conf*
8055 **Active**: Y	IN	Talend logging module - JobServer log4j port	*logstash-talend.conf*
9200 **Active**: Y	IN	Talend logging module - Elasticsearch port	*<TalendLogServerPath>/elasticsearch-X.X.X/config/elasticsearch.yml*

Table 1.37. Talend ESB including Talend Runtime Ports

Port	Direction	Usage	Config (./etc/)	Remark
8040 **Active**: Y	IN	Standard HTTP port	*org.ops4j.pax.web.cfg*	See the *Talend ESB Container Administration Guide* for config scripts and also the *admin:* command which allows you to set ports to different values.
9001 **Active**: Y	IN	Standard HTTS port	*org.ops4j.pax.web.cfg*	

Port	Direction	Usage	Config (./etc/)	Remark
1099 **Active**: Y	IN	JMX - RMI Registry Port	*org.apache.karaf.management.cfg*	
44444 **Active**: Y	IN	JMX - RMI Registry Port	*org.apache.karaf.management.cfg*	
8101 **Active**: Y	IN	Apache Karaf - SSH Port	*org.apache.karaf.shell.cfg*	
8000 **Active**: Y	IN	Talend JobServer - Command Port	*org.talend.remote.jobserver.server. cfg*	
8001 **Active**: Y	IN	Talend JobServer - File Transfer Port	*org.talend.remote.jobserver.server. cfg*	
8888 **Active**: Y	IN	Talend JobServer - Monitoring Port	*org.talend.remote.jobserver.server. cfg*	
61616 **Active**: N	IN	Messaging - ActiveMQ Broker Port	system.properties	
2181 **Active**: N	IN OUT	ESB Locator - Apache Zookeeper Port	Server: *org.talend.esb.locator.server.cfg* Client: *org.talend.esb.locator.cfg*	
1527 **Active**: N	IN	ESB SAM Database - Apache Derby Port	The port value of the embedded Derby database depends on the Talend Runtime Container configuration, as the database is shipped with the container.	The embedded Apache Derby DB is only supported for Development and Testing purpose in production system environments. The Container, which hosts the SAM Server, needs access to the related Database port. The port depends on the Database and Database configuration.
8082 **Active**: Y	OUT	ESB Runtime Features Installer - Artifact Repository access	*org.ops4j.pax.url.mvn.cfg*	
(*)	IN OUT	Customer Services, Routes etc.		Any of the Data Services, Routes and other components additionally deployed to the container might require additional port to be accessible.

Table 1.38. JobServer Ports

Port	Direction	Usage	Config
8000 **Active**: Y	IN	Talend JobServer - Command Port	*org.talend.remote.jobserver.server. cfg*
8001 **Active**: Y	IN	Talend JobServer - File Transfer Port	*org.talend.remote.jobserver.server. cfg*
8888 **Active**: Y	IN	Talend JobServer - Monitoring Port	*org.talend.remote.jobserver.server. cfg*

Table 1.39. Nexus Artifact Repository Ports

Port	Direction	Usage	Config
8081 **Active**: Y	IN	Nexus Artifact Repository - Jetty HTTP Port	*/conf/jetty.xml*

Table 1.40. Talend Data Quality Portal Ports

Port	Direction	Usage	Config
8580 **Active**: Y	IN	Standard port	*/tdqp/apache-tomcat/conf/server.xml*
7890 **Active**: Y	IN	Port of the Talend Data Quality Portal configuration embedded database	*/tdqp/apache-tomcat/conf/server.xml* and */tdqp/apache-tomcat/database/startup.[sh|bat]*

Table 1.41. Talend Dictionary Service Port

Port	Direction	Usage	Config
8187 **Active**: Y	IN OUT	Standard port	*/conf/server.xml*
27017 **Active**: Y	IN OUT	MongoDB port	*<MongoDB>/mongod.cfg*
2181 **Active**: Y	IN OUT	Apache Zookeeper port	*<Kafka>/config/zookeeper.properties*
9092 **Active**: Y	IN OUT	Apache Kafka port	*<Kafka>/config/server.properties*

Table 1.42. Streams Runner and Spark Job Server Ports

Port	Direction	Usage	Config
9060 **Active**: Y	IN	Streams Runner default port	*<Streams_Runner_Path>/conf/application.conf*
8098 **Active**: Y	IN	Spark Job Server default port	*<Spark_Job_Server_Path>/settings.sh*
8099 **Active**: Y	IN	Spark Job Server H2 port	*<Spark_Job_Server_Path>/settings.sh*
9998 **Active**: Y	IN	Spark Job Server JMX port	*<Spark_Job_Server_Path>/settings.sh*

1.3.13. Database privileges for Talend Administration Center

To be able to manage the Talend Administration Center database (create, edit or drop tables for example), the administrator user must have the following system privileges:

- *Create*
- *Read*
- *Update*
- *Delete*

In order to perform database backup operations in the web application, this user also needs to be able to execute the `<database> dump` command into the target database schema.

1.3.14. Database privileges for Talend Data Quality Portal

When you use *Talend Installer* to install *Talend Data Quality Portal*, you can select a specific database to store analysis results. For further information about supported databases, see *Compatible Databases*.

Prerequisites

If you select to use the MySQL, PostgreSQL or SQL server database to store the analysis results, the database is created automatically by the installer. However, when you select to use the Oracle database you must create the database and the schema before installing the Portal via *Talend Installer*.

When defining the database to store analysis results, you must:

- make sure that users of the Portal are granted the right privileges on different databases as described in *Privileges in databases*.
- make sure, for the PostgreSQL database, to add the IP of the computer which will install the Portal on the PostgreSQL server.

1.3.14.1. Privileges in databases

To enable users to write analysis results to any of the supported databases, you must grant them certain system privileges.

What privileges are necessary for Mysql

The user you define when you configure the report datamart connection must have certain privileges. Use grant commands to give at least the following system privileges:

- *Select*.
- *Insert*.
- *Update*.
- *Create*.
- *Drop*.
- *Index*.
- *Alter*.
- *Create View*.

SQL statement: *GRANT <privilege> ON`<database_name>`.* TO <user_name>@'%'* identified by *<user_password>*.

What privileges are necessary for Oracle

The user you define when you configure the report datamart connection must have either the *DBA* role, or the *CONNECT* and *RESOURCE* roles. Use grant commands to give the user system privileges as the following:

The privileges required when creating a datamart within the user's own schema are:

- *create a sequence*.
- *create session*.
- *create alter session*.
- *create table*.
- *create view*.

An example of an SQL statement could be: *GRANT create session to <user>*.

The privileges required when creating a datamart within a schema which is not owned by the current user are:

- *alter any table*.
- *comment any table*.
- *create any index*.
- *create any sequence*.
- *create any table*.
- *create any view*.
- *create session*.
- *insert any table*.
- *select any sequence*.
- *select any table*.
- *update any table*.

An example of an SQL statement could be: *GRANT alter any table to <user_name>*.

What privileges are necessary for PostgreSQL

The user you define when you configure the report datamart connection must either have the *DBA* role (like the *postgres* user), or at least has necessary privileges as the following:

- *CREATE USER <user_name> WITH PASSWORD <user_password>*.
- *CREATE DATABASE <database_name>*.
- *GRANT ALL PRIVILEGES ON DATABASE <database_name> to <user_name>*.

If you want to use the PostgreSQL database with the Portal, you must add the IP of the computer which will install the Portal on the PostgreSQL server.

Only the default schema, *public*, is used for now.

What privileges are necessary for SQL server

The user you define when you configure the report datamart connection must either have the *DBA* role (like the *sa* user), or you can use SQL commands to create a user and a database as the following:

Create database <database_name>.

Create login:

- *CREATE LOGIN <user_name> WITH PASSWORD = <user_password>*.

Grant <*user_name*> the db_owner role in <database_name>.

- *USE <database_name>*.
- *ALTER LOGIN <user_name> with default_database = <database_name>*.
- *CREATE USER <user_name> FROM LOGIN <user_name>*.
- *EXEC sp_addrolemember db_owner, <user_name>*.

Only the default schema, *dbo* is used for now.

1.3.15. Installing the XULRunner package

On Linux, the XULRunner package is required to run the Studio.

The XULRunner package version that is recommended is XULRunner v1.9.2.28.

The supported versions are v1.8.x - 1.9.x and v3.6.x.

1. Download XULRunner v1.9.2.28 from this location.
2. Unpack the archive file in the same directory where you unpacked the studio archive, but do not unpack it within the Studio folder.
3. Add the following line at the end of the Studio *.ini* file that corresponds to your Linux architecture:

   ```
   -Dorg.eclipse.swt.browser.XULRunnerPath=</usr/lib/xulrunner>
   ```

   where `</usr/lib/xulrunner>` is the *xulrunner* installation path.

   For example, if you have unpacked the Studio in a directory under your user home directory */home/<user>/Talend/*, you need to add the following to the *.ini* file:

   ```
   -Dorg.eclipse.swt.browser.XULRunnerPath=/home/<user>/Talend/xulrunner/
   ```

Chapter 2. Installing your Talend product using Talend Installer (recommended)

The following pages contain procedures allowing you to automatically install your *Talend* product using *Talend Installer*.

This method is the recommended way of installing your *Talend* product. However, if you want to fully customize every step of your installation, you can perform a manual installation. For more information, see *Installing your Talend product manually*.

2.1. Introducing Talend Installers

Talend provides different installers to install your product:

- *Talend Studio Installer*: This installer allows you to automatically install your *Talend Studio* without any prerequisites thanks to its embedded Java Environment. For more information see *Installing Talend Studio with the Talend Studio Installer*.
- *Talend Installer*: This installer allows you to automatically install your *Talend Studio* and all *Talend* Server modules. For more information see *Talend Installer specific prerequisites*.

2.2. Installation modes of Talend Installer and Talend Studio Installer

This section provides information about the different installation modes that *Talend Installer* and *Talend Studio Installer* can run in.

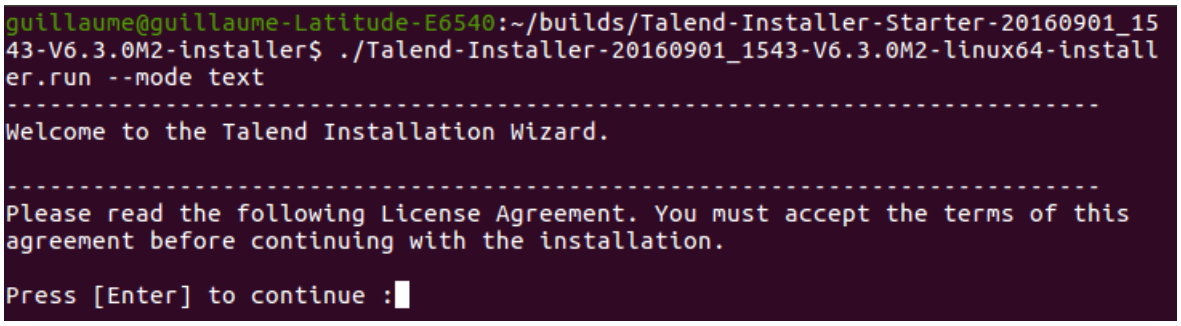
Note that the log files generated during the installation can be found in */tmp/*.

Note also that, once *Talend Installer* has completed the installation of the products, a directory (called *Talend* by default) is created with sub-folders for each *Talend* product.

The following installation modes are available:

- Graphical mode: allows full interactivity through a graphical user interface.
- Text mode: provides full interactivity with users in the command line. It is equivalent to any GUI mode but the pages are displayed in text mode in a console.

  The image below shows an example of text mode where the user enters the `--mode text` option from the command line:



  This installation mode is only available on Unix platforms. It is automatically used if no graphical mode is available but it also can be forced using the `--mode text` command.

- Unattended mode: is especially useful for automating the installation processes. This silent mode will perform an unattended installation that will not prompt the user for any information.

To perform this Unattended installation:

1. Write a simple *.txt* script in which you will define the options values. For a complete list of values, use the `help` command or see the unattended mode available options documentation Talend Help Center (https://help.talend.com).

   ```
   mode=unattended
   ```

```
debugtrace=/home/user/Talend_install_files/debugInstall.txt
licenseFile=/home/user/Talend_install_files/license
installdir=/home/user/Talend
installType=full
installStyle=easy
```

In this example, the script details the silent installation of the Server type (full type). The installation directory that will be created is called *Talend* and the license file used is located in the */home/user/Talend_install_files* directory.

You can also create a script for a Custom type installation for example: in this case, specify in your script the products and modules to install as well as the configuration information of these products. For example, the `enable-components` parameter allows you to do a comma-separated list of these products, while the `tacPort` parameter allows you to specify the port to use for Talend Administration Center. For more information about the available parameters and their values, enter `help` in the console.

2. Launch the silent installation using the `--optionfile <filename>` command, where `<filename>` is the name of the script which contains the list of pairs `<key>=<value>`.

```
guillaume@guillaume-Latitude-E6540:~$ cd /home/guillaume/builds/Talend-Installer
-Starter-20160901_1543-V6.3.0M2-installer
guillaume@guillaume-Latitude-E6540:~/builds/Talend-Installer-Starter-20160901_15
43-V6.3.0M2-installer$ ./Talend-Installer-20160901_1543-V6.3.0M2-linux64-install
er.run --optionfile ../script.txt
guillaume@guillaume-Latitude-E6540:~/builds/Talend-Installer-Starter-20160901_15
43-V6.3.0M2-installer$
```

An unattended installation is performed.

To install *Talend* products as services via the Installer, you are required to run the application as Administrator OR to disable User Account Control.

For more information on these installation modes, please refer to the online Bitrock documentation.

2.3. Installing Talend Studio with the Talend Studio Installer

Talend Studio Installer is a convenient way of installing your *Talend Studio*.

As it comes with an embedded Java Environment, you can install it without any prerequisite.

To install you *Talend Studio* with the *Talend Studio Installer*, proceed as follows:

1. Download the *TalendStudio-A-B-C-linux-x64-installer.run* file.
2. Make the *TalendStudio-A-B-C-linux-x64-installer.run* file executable with the following command:

   ```
   chmod +x TalendStudio-A-B-C-linux-x64-installer.run
   ```

3. Launch the *Talend Studio Installer* with the following command:

   ```
   ./TalendStudio-A-B-C-linux-x64-installer.run
   ```

4. Accept the License Agreement.
5. Choose the directory where you want your *Talend* product to be installed.
6. Add your license file.

7. Choose where you want the workspace directory to be located.
8. Launch the installation.

2.4. Talend Installer specific prerequisites

Prior to launching the *Talend Installer*, check that:

- you have downloaded two files, a *.zip* and a *dist* file, which are both stored in the same directory.

  In the .zip file that you will extract in this directory, you will find executable files for every supported operating system.

  For Linux, use *Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run*.

  In the file name, *YYYYYYYY_YYYY* is the timestamp and *A.B.C* is the revision level (Major.Minor.Patch).

  The *dist* file is only required to install *Talend* products. Once the installation and configuration is complete, you can remove it.

- the following software and modules are properly set up on the station where you are to install the *Talend* modules:
  - JRE 1.8.0 or higher must be installed on your machine;
  - (optional) a mail server (to send notifications by email).

Note that *Talend Installer* does not support the *sdshell* utility.

IMPORTANT:

Talend Installer allows you to get out-of-the-box *Talend* solutions that do not require any manual installation. However, these solutions are not provided in a production-ready environment as they may require additional configurations or optimizations according to your specific needs.

For example, you may want to change the H2 database that is embedded by default in *Talend Administration Center* with your own database (MySQL or Oracle for example).

Talend Installer is used only for first installations of *Talend* solutions. Therefore, if you want to know more about the migration and upgrade processes, please refer to *Talend Migration Guide*.

2.5. Using Talend Installer graphical installation mode

When using Talend Installer graphical installation mode, three installation types are available:

- The Server type: allows you to install all Talend server components with default configuration. For more information see *Installing Talend server modules using Talend Installer*.
- The Client type: allows you to install the Talend Studio. For more information, see *Installing Talend Studio using Talend Installer*.
- The Custom type: allows you to select and configure the *Talend* modules you want to install. For more information, see *Performing a Custom installation with Talend Installer*.

2.5.1. Installing Talend server modules using Talend Installer

The Server installation type allows you to install *Talend Studio* and all Talend server components and configure them if required.

There are two different styles of Server installation:

- *Performing an Easy Server installation with Talend Installer*
- *Performing an Advanced Server installation with Talend Installer*

The following table shows the configuration options that are available in the different Server installation styles.

For example, the Advanced Server installation lets you choose the Tomcat server and port you want to use for *Talend Administration Center* whereas the Easy Server installation installs *Talend Administration Center* on a new Tomcat server on the 8080 port.

For the following module...	You can configure...	Advanced Server installation	Easy Server installation
Talend Administration Center	Tomcat instance to use	✅	
	Administrator user name and password	✅	
	Enable external Single-Sign On (SSO)	✅	
	Database	✅	
	Port	✅	
	Web application directory	✅	
Nexus Artifact Repository	Host and port	✅	
Talend Log Server	Cluster name	✅	
Talend Data Stewardship	Tomcat instance to use	✅	
	MongoDB database[1]	✅	
	Kafka connection parameters	✅	
	Zookeeper connection parameters	✅	
	Talend Administration Center connection parameters	✅	
Talend Runtime	Port configuration	✅	
Talend Data Quality Portal	Tomcat instance to use	✅	✅
	Database	✅	✅
	Port	✅	✅
	Memory	✅	✅
	Server address	✅	✅
Talend Data Preparation	Big Data Support	✅	
	Kerberos cluster	✅	
	MongoDB database[1]	✅	
	Talend Administration Center connection parameters	✅	
	Server IP and ports	✅	

For the following module...	You can configure...	Advanced Server installation	Easy Server installation
Talend Dictionary Service	Tomcat port	✅	
	MongoDB database[1]	✅	
	Talend Administration Center connection parameters	✅	
Talend Kafka and Zookeeper	Zookeeper data directory	✅	
Talend SAP RFC Server	SAP configuration	✅	
	JMS Broker URL	✅	
	Library	✅	
Talend Identity and Access Management	Tomcat instance to use	✅	
	Talend Administration Center connection parameters	✅	
Talend Studio	Workspace directory location	✅	
Talend ESB	Tomcat instance to use	✅	

[1]: If you want to secure connections with MongoDB using SSL, MongoDB Enterprise Server has to be manually installed on your machine. For more information, see https://docs.mongodb.com/v3.2/security/.

2.5.1.1. Performing an Easy Server installation with Talend Installer

The Easy Server installation is a convenient way of installing *Talend Studio* and all the *Talend* server modules included in your licence with their default configuration. It also installs these modules as services on your machine.

Talend Installer installs the *Talend* Server modules with their default configuration:

Modules installed	Details
Talend Administration Center	• Access URL: http://localhost:8080/org.talend.administrator • Default administrator username: security@company.com • Default administrator password: admin
Talend Log Server	N/A
Talend Data Stewardship	Access URL: http://localhost:19999
Talend CommandLine	N/A
Talend Runtime	N/A
Talend Data Quality Portal	• Access URL: http://localhost:8580/tdqportal • Default administrator username: tdq_admin • Default administrator password: tdq
Talend Studio	N/A
Talend ESB	N/A
Talend Data Preparation	Access URL: http://localhost:9999
Talend Identity and Access Management	N/A
Talend Server Services	N/A

Prerequisites:

- All the required files are downloaded. For more information, see *Talend Installer specific prerequisites*.

- All the default ports are opened. For more informations, see *Port information*.
- If you want to use the embedded MongoDB database, make sure that there are no other instance of MongoDB installed on your machine.

To perform an Easy Server installation, proceed as follows:

1. Make sure the *dist* file is in the same folder as the *Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run* file.
2. Make the *Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run* file executable with the following command:

   ```
   chmod +x Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run
   ```

3. Launch *Talend Installer* with the following command:

   ```
   ./Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run
   ```

   If you want to install *Talend* server modules as services, execute this command with the super-user rights.
4. Accept the License Agreement.
5. Choose the directory where you want your *Talend* product to be installed.
6. Choose **Easy Install** in the installation style list and **Server** in the installation type list.
7. Add your license file.
8. Configure *Talend Data Quality Portal* according to the database you want to use.
9. Launch the installation.
10. Once the installation is complete, you can remove the *dist* file to save some space on your disk.

Talend Installer creates a *usedports.txt* file where all the ports used by *Talend* Server modules are listed.

A user with *tds-user* as username and *duser* as password is automatically created in MongoDB for *Talend Data Stewardship*.

A user with *dataprep_user* as username and *duser* as password is automatically created in MongoDB for *Talend Data Preparation*.

2.5.1.2. Performing an Advanced Server installation with Talend Installer

The Advanced Server installation is a convenient way of installing *Talend Studio* and all the *Talend* server modules included in your licence with custom configuration. It also installs these modules as services on your machine.

For more information on the options you can customize using the Advanced Server installation, see *Installing Talend server modules using Talend Installer*.

Prerequisites:

- All the required files are downloaded. For more information, see *Talend Installer specific prerequisites*.
- All the default ports are opened. For more informations, see *Port information*.
- There are no other instance of MongoDB installed on your machine.

To perform an Advanced Server installation, proceed as follows:

1. Make sure the *dist* file is in the same folder as the *Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run* file.
2. Make the *Talend Installer* file executable with the following command:

   ```
   chmod +x Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run
   ```

3. Launch *Talend Installer* with the following command:

   ```
   ./Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run
   ```

   If you want to install *Talend* server modules as services, execute this command with the super-user rights.
4. Accept the License Agreement.
5. Choose the directory where you want your *Talend* product to be installed.
6. Choose **Advanced Install** in the installation style list and **Server** in the installation type list.
7. Add your license file.
8. Follow the configuration steps.
9. Launch the installation.
10. Once the installation is complete, you can remove the *dist* file to save some space on your disk.

Talend Installer creates a *usedports.txt* file where all the ports used by *Talend* Server modules are listed.

If you choose to install *Talend Data Stewardship*, a user with *tds-user* as username and *duser* as password is automatically created in MongoDB.

If you choose to install *Talend Data Preparation*, a user with *dataprep_user* as username and *duser* as password is automatically created in MongoDB.

2.5.2. Installing Talend Studio using Talend Installer

The Client installation type allows you to install *Talend Studio*, *Talend Data Stewardship* and *Talend Runtime* and configure them if required.

There are two different styles of Client installation:

- *Performing an Easy Client installation with Talend Installer*
- *Performing an Advanced Client installation with Talend Installer*

The following table shows the configuration options that are available in the different Client installation styles.

For example, the Advanced Client installation lets you choose the workspace directory location whereas the Easy Client installation puts it in a *workspace* folder in the user directory.

For the following module...	You can configure...	Advanced Client installation	Easy Client installation
Talend Runtime	Port configuration	✅	
Talend Studio	Workspace directory location	✅	

2.5.2.1. Performing an Easy Client installation with Talend Installer

The Easy Client installation is a convenient way of installing your *Talend Studio*, *Talend Data Stewardship* and, according to your license, *Talend Runtime* with its default configuration.

Prerequisites:

- All the required files are downloaded. For more information, see *Talend Installer specific prerequisites*.
- All the default ports are opened. For more informations, see *Port information*.
- There are no other instance of MongoDB installed on your machine.

To perform an Easy Client installation, proceed as follows:

1. Make sure the *dist* file is in the same folder as the *Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run* file.
2. Make the *Talend Installer* file executable with the following command:

   ```
   chmod +x Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run
   ```

3. Launch *Talend Installer* with the following command:

   ```
   ./Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run
   ```

   If you want to install *Talend* server modules as services, execute this command with the super-user rights.
4. Accept the License Agreement.
5. Choose the directory where you want your *Talend* product to be installed.
6. Choose **Easy Install** in the installation style list and **Client** in the installation type list.
7. Add your license file.
8. Launch the installation.
9. Once the installation is complete, you can remove the *dist* file to save some space on your disk.

Talend Installer creates a *usedports.txt* file where all the ports used by *Talend* Server modules are listed.

A user with *tds-user* as username and *duser* as password is automatically created in MongoDB for *Talend Data Stewardship*.

2.5.2.2. Performing an Advanced Client installation with Talend Installer

The Advanced Client installation is a convenient way of installing *Talend Studio*, *Talend Data Stewardship* and, according to your license, *Talend Runtime* with custom configuration. It also installs these modules as services on your machine and configures Talend SAP RFC Server.

For more information on the options you can customize using the Advanced Client installation, see *Installing Talend Studio using Talend Installer*.

Prerequisites:

- All the required files are downloaded. For more information, see *Talend Installer specific prerequisites*.
- All the default ports are opened. For more informations, see *Port information*.
- There are no other instance of MongoDB installed on your machine.

To perform an Advanced Client installation, proceed as follows:

1. Make sure the *dist* file is in the same folder as the *Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run* file.
2. Make the *Talend Installer* file executable with the following command:

   ```
   chmod +x Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run
   ```

3. Launch *Talend Installer* with the following command:

   ```
   ./Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run
   ```

   If you want to install *Talend* server modules as services, execute this command with the super-user rights.
4. Accept the License Agreement.
5. Choose the directory where you want your *Talend* product to be installed.
6. Choose **Advanced Install** in the installation style list and **Client** in the installation type list.
7. Add your license file.
8. Follow the configuration steps.
9. Launch the installation.
10. Once the installation is complete, you can remove the *dist* file to save some space on your disk.

Talend Installer creates a *usedports.txt* file where all the ports used by *Talend* Server modules are listed.

If you choose to install *Talend Data Stewardship*, a user with *tds-user* as username and *duser* as password is automatically created in MongoDB.

2.5.3. Performing a Custom installation with Talend Installer

The Custom installation is the more customizable installation method with Talend Installer. It allows you to choose what to install, where and how. This way, you can fully customize your installation and choose, for example, to install *Talend Administration Center* on a machine and *Talend Studio* on another.

Here are the modules you can install with Talend Installer Custom Installation:

- *Talend Administration Center*
- *Talend Log Server*
- *Talend Data Stewardship*
- *Talend CommandLine*
- *Talend Runtime*

- *Talend JobServer*
- *Talend Data Quality Portal*
- *Talend Data Preparation*
- *Talend SAP RFC Server*
- *Talend IAM Service*
- *Talend Studio*
- *Talend ESB*
- *Talend Server Services*

The following table sums up all the details you can configure for each chosen module.

For the following module...	**You can configure...**
Talend Administration Center	Tomcat instance to use
	Administrator user name and password
	Enable external Single-Sign On (SSO)
	Database
	Port
	Web application directory
	Email notifications
Nexus Artifact Repository	Host and port
Talend Log Server	Cluster name
Talend Data Stewardship	Tomcat instance to use
	MongoDB database[1]
	Kafka connection parameters
	Zookeeper connection parameters
	Talend Administration Center connection parameters
Talend CommandLine	Port
Talend Runtime	Port configuration
Talend JobServer	Ports
	Cache duration
Talend Data Quality Portal	Tomcat instance to use
	Database
	Port
	Memory
	Server address
Talend Data Preparation	Big Data Support
	Kerberos cluster
	MongoDB database[1]
	Kafka connection parameters
	Talend Administration Center connection parameters
	Server IP and ports
Talend Dictionary Service	Tomcat port
	MongoDB database[1]
	Talend Administration Center connection parameters
Talend Kafka and Zookeeper	Zookeeper data directory

For the following module...	You can configure...
Talend SAP RFC Server	SAP configuration
	JMS Broker URL
	Library
Talend Identity and Access Management	Tomcat instance to use
	Talend Administration Center connection parameters
Talend Studio	Workspace directory location
Talend Server Services	Services to install

[1]: If you want to secure connections with MongoDB using SSL, MongoDB Enterprise Server has to be manually installed on your machine. For more information, see https://docs.mongodb.com/v3.2/security/.

Prerequisites:

- All the required files are downloaded. For more information, see *Talend Installer specific prerequisites*.
- All the default ports are opened. For more informations, see *Port information*.
- If you want to use the embedded MongoDB database, make sure that there are no other instance of MongoDB installed on your machine.

To perform a Custom installation, proceed as follows:

1. Make sure the *dist* file is in the same folder as the *Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run* file.
2. Make the *Talend Installer* file executable with the following command:

   ```
   chmod +x Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run
   ```

3. Launch *Talend Installer* with the following command:

   ```
   ./Talend-Installer-YYYYYYYY_YYYY-VA.B.C-linux64-installer.run
   ```

   If you want to install *Talend* server modules as services, execute this command with the super-user rights.
4. Accept the License Agreement.
5. Choose the directory where you want your *Talend* product to be installed.
6. Choose **Advanced Install** in the installation style list and **Custom** in the installation type list.
7. Add your license file.
8. Follow the configuration steps.
9. Launch the installation.
10. Once the installation is complete, you can remove the *dist* file to save some space on your disk.

Talend Installer creates a *usedports.txt* file where all the ports used by *Talend* Server modules are listed.

If you choose to install *Talend Data Stewardship*, a user with *tds-user* as username and *duser* as password is automatically created in MongoDB.

If you choose to install *Talend Data Preparation*, a user with *dataprep_user* as username and *duser* as password is automatically created in MongoDB.

Chapter 3. Installing your Talend product manually

The following pages contain procedures allowing you to manually install your *Talend* product.

The recommended installation method is the automatic installation with *Talend Installer*. The manual installation should only be performed if you want to customize every step of your installation. For more information on the automatic installation, see *Installing your Talend product using Talend Installer (recommended)*.

The manual installation procedures must be executed in the following order:

1. *Setting up your version control system*
2. *Installing and configuring Talend Administration Center*
3. *Installing and configuring Talend Identity and Access Management*
4. *Talend logging modules*
5. *Installing your Talend Studio*
6. *Installing and configuring CommandLine*
7. *Installing and configuring Talend ESB*
8. *Installing and configuring your Talend Data Quality Portal*
9. *Installing and configuring Talend SAP RFC Server*
10. *Installing and configuring Talend Dictionary Service*
11. *Installing and configuring Talend Data Preparation*
12. *Installing and configuring Talend Data Stewardship*

For more information on the functional architecture of your *Talend* product, see *Architecture of the Talend products*.

3.1. Setting up your version control system

The following pages contain procedures on how to install your version control system:

- *Installing and configuring an Apache Subversion (SVN) server*
- *Installing and configuring Git*

For more information on version control systems, see *Version control system*.

3.1.1. Installing and configuring an Apache Subversion (SVN) server

This procedure describes how to install and configure an Apache Subversion (SVN) server in order to store all your project data (Jobs, Database connections, Routines, Joblets, etc.) in the shared Repository of the Talend Studio.

In the following instructions, lines starting with "#" mean that they must be executed as root. A command line starting with "$" means that it must be executed as standard user.

There are three different methods to install Apache Subversion on Linux:

- Installing and configuring Apache Subversion on Linux Ubuntu
- Installing and configuring Apache Subversion on Linux Redhat

Installing and configuring Apache Subversion on Linux Ubuntu

1. Install the *subversion*, *apache2* and *libapache2-svn* packages.

```
# apt-get install subversion apache2 libapache2-svn
```

2. Create an *svn* directory, at the root of your system for example.

```
# mkdir /svn
```

3. Change the owner of the directory to the webserver user, *www-data* by default.

```
# chown www-data:www-data /svn
```

4. Switch to *www-data* user.

```
# su - www-data
```

5. Create a new SVN repository, *my_repo* for example.

```
$ svnadmin create /svn/my_repo
```

6. Create a user and a password.

```
$ htpasswd -cmb /svn/passwd talend_admin secretpassword
```

7. Open the SVN configuration file.

```
# vi /etc/apache2/mods-enabled/dav_svn.conf
```

8. Update it as follows:

```
<Location /my_repo>
DAV svn
```

```
SVNPath /svn/my_repo
AuthUserFile /svn/passwd
Require valid-user
AuthType basic
AuthName "Subversion"
</Location>
```

9. Restart Apache Subversion.

```
# /etc/init.d/apache2 restart
```

10. If you have SELinux (Security-Enhanced Linux) enabled on your system, see Talend Help Center (https://help.talend.com).

Installing and configuring Apache Subversion on Linux Redhat

1. As user *apache* or as root, install the *httpd*, *mod_dav_svn* and *subversion* packages.

```
# yum install httpd mod_dav_svn subversion
```

2. Create an *svn* directory.

```
# mkdir /var/www/svn
```

3. Create an SVN repository.

```
# svnadmin create /var/www/svn/repository
```

4. Create a username and a password.

```
# htpasswd -cmb /var/www/svn/passwd admin@company.com admin
```

5. Change the owner of the directory to *apache*.

```
# chown -R apache.apache /var/www/svn
```

6. Open the SVN configuration file.

```
# vim /etc/httpd/conf.d/subversion.conf
```

7. Update is as follows:

```
<Location /svn>
DAV svn
SVNPath /var/www/svn/repository
Require valid-user
AuthType basic
AuthName "Subversion"
AuthUserFile /var/www/svn/passwd
</Location>
```

8. Restart httpd.

```
# /etc/init.d/httpd start
```

9. If you have SELinux (Security-Enhanced Linux) enabled on your system, see Talend Help Center (https://help.talend.com).

3.1.2. Installing and configuring Git

This procedure describes how to install and configure Git in order to store all your project data (Jobs, Database connections, Routines, Joblets, etc.) in the shared Repository of the Talend Studio.

For more information on the supported Git servers, see *Compatible version control systems*.

1. Download the Git version corresponding to your system at https://git-scm.com/downloads and follow the installation instructions.
2. Open a terminal instance.
3. Create an SSH key using the following command:

   ```
   ssh-keygen
   ```

4. Put the generated key files in the */home/User_Name/.ssh* folder.
5. Add the generated public key to settings of your Git server.
6. Use the following command to create a *known-hosts* file:

   ```
   ssh-keyscan -H git_server_hostname >> known_hosts
   ```

7. Create a *config* file in your *.ssh* folder.
8. Add the following content and adapt it to your configuration:

   ```
   Hostname  git_server_hostname
   IdentityFIle  /home/User_Name/.ssh/id_rsa
   ```

9. Add the connection information to the *Talend Administration Center* configuration. For more information, see the *Talend Administration Center User Guide*.

3.2. Installing and configuring Talend Administration Center

Talend Administration Center is a Web-based administration application that allows *Talend Studio* project managers to administrate users and projects and manage access to the remote repository.

For more detailed information regarding *Talend Administration Center* and Tomcat, see *Apache Tomcat Server*.

To install and configure *Talend Administration Center*, follow these procedures:

- *Deploying Talend Administration Center on an application server*
- *Talend Administration Center basic configuration*
- *Installing and Configuring Talend server modules*
- *Talend Administration Center advanced configuration*

For more information on the scheduling management strategy in the *Talend Administration Center* application, see the article about the *Talend Administration Center* recommendations about environment and configuration on Talend Help Center.

3.2.1. Deploying Talend Administration Center on an application server

The following sections show how to deploy the *Talend Administration Center* Web application on an application server.

In the rest of this documentation, <*TomcatPath*> points out the path where Tomcat has been installed.

3.2.1.1. Deploying Talend Administration Center on Tomcat

To install and deploy *Talend Administration Center* on Tomcat, proceed as follows:

1. Install the Apache Tomcat application server and stop the Tomcat service if it is automatically started.
2. Edit the following file:

   etc/default/tomcat8
3. Uncomment the tomcat security setting and change the default setting, in order to read:

   ```
   TOMCAT8_SECURITY=no
   ```
4. Unzip the package delivered by *Talend*: *Talend-AdministrationCenter-YYYYYYYY_YYYY-VA.B.C.zip*.

   This will give you access to the different components needed to benefit from all the *Talend Administration Center* functionalities:

   - *org.talend.administrator.war*, the archive containing the actual *Talend Administration Center* Web application.
   - *Artifact-Repository-Nexus-VA.B.C.D.zip*, the archive containing an artifact repository software, based on Sonatype Nexus, that will be used to handle software updates, DI artifacts, and also ESB artifacts, only if you subscribed to one of our ESB oriented solutions. For more information, see *Introduction to the Talend products*.
   - an *endorsed* folder containing a library necessary to use the ESB Conductor of *Talend Administration Center*, only if you subscribed to one of our ESB oriented solutions.
5. Copy the Web application, *org.talend.administrator.war*, into the following directory of Tomcat:

   <TomcatPath>/webapps/

   Once you have copied this war file, you can either unzip it manually under the same directory, or let Tomcat unzip the web application at startup.
6. If you subscribed to one of our ESB oriented solutions, to be able to use the Conductor module of *Talend Administration Center*, copy the *endorsed* folder provided into the *root* directory of Tomcat.
7. Start Tomcat using the following command:

   ```
   sh <TomcatPath>/bin/startup.sh
   ```

The storage of log outputs is managed by Tomcat application server, by default, but you are also able to define your own path for storing the logs. From 4.0, you can configure the path directly from Talend Administration Center. For more information on manual configuration in prior versions, refer to Configuring the log storage mode.

If you wish to implement high availability to the task execution scheduling using *Talend* clustering feature, then read *Setting up High Availability* before starting Tomcat.

For reasons of right management, make sure you launch Tomcat using the same administrator account as for the CommandLine. For example: Create an account *talend_admin* for both Tomcat and CommandLine.

If you deploy a large number of applications on Tomcat, you should increase its memory to improve its performance. For more information on this process, see *Increasing the memory of Tomcat*.

If you encounter performance issues during Tomcat startup, it may be due to the use of symbolic links during SVN checkout. For more information on how to solve these issues, see Talend Help Center.

3.2.1.2. Deploying Talend Administration Center on JBoss

To install and deploy *Talend Administration Center* on JBoss, proceed as follows:

1. Install the JBoss application server and stop the JBoss service if it is automatically started.
2. Unzip the archive delivered by *Talend*.
3. Move *bcprov-jdk15on-1.51.jar* from *org.talend.administrator.war/WEB-INF/lib* to *<JBossPath>/modules/system/layers/base/org/bouncycastle/main*.
4. In *<JBossPath>/modules/system/layers/base/org/bouncycastle/main/*, create a *module.xml* file.
5. Paste the following content:

```
<?xml version="1.0" encoding="UTF-8"?>
<module xmlns="urn:jboss:module:1.1" name="org.bouncycastle">
<resources>
<resource-root path="bcprov-jdk15on-1.51.jar"/>
</resources>
<dependencies>
<module name="javax.api" slot="main" export="true"/>
</dependencies>
</module>
```

6. In *org.talend.administrator.war/WEB-INF*, create a *jboss-deployment-structure.xml* file.
7. Paste the following content:

```
<jboss-deployment-structure>
<deployment>
<dependencies>
<module name="org.bouncycastle" slot="main" export="true" />
</dependencies>
</deployment>
</jboss-deployment-structure>
```

8. Move the *org.talend.administrator.war* file to *<JBossPath>/standalone/deployments*.
9. Start JBoss using the following command:

```
sh <JBossPath>/bin/run.sh
```

The storage of log outputs is managed by Tomcat application server, by default, but you are also able to define your own path for storing the logs. You can configure the path directly from Talend Administration Center. For more information on manual configuration in prior versions, refer to Configuring the log storage mode.

For reasons of right management, make sure you launch JBoss using the same administrator account as for the Commandline. For example: Create an account *talend_admin* for both JBoss and CommandLine.

3.2.1.3. Deploying Talend Administration Center on Pivotal tc Server

To install and deploy Talend Administration Center on Pivotal tc Server, proceed as follows:

1. Install Pivotal tc Server as explained in Pivotal documentation: http://tcserver.docs.pivotal.io/docs-tcserver/topics/install-getting-started.html.
2. Create a Pivotal tc Server instance as explained in Pivotal documentation: http://tcserver.docs.pivotal.io/docs-tcserver/topics/postinstall-getting-started.html.

3. Stop your Pivotal tc Server instance.
4. Unzip the package delivered by *Talend*: *Talend-AdministrationCenter-YYYYYYYY_YYYY-VA.B.C.zip*.
5. Copy the Web application, *org.talend.administrator.war*, into the *webapps* folder of your Pivotal tc Server instance, for example:

   /home/tcserver/pivotal-tc-server/myserver/webapps/
6. Copy the *.jar* files contained in the *endorsed* folder to the *lib* folder of your Pivotal tc Server instance, for example:

   /home/tcserver/pivotal-tc-server/myserver/lib/
7. Start your Pivotal tc Server instance to automatically deploy *Talend Administration Center*.

3.2.1.4. Increasing the memory of Pivotal tc Server

To increase the memory heap size of the server in order to make all the applications work properly, proceed as follows:

1. Change directory to <*PivotalPath*>*/bin* where <*PivotalPath*> is the Pivotal tc Server installation directory, and edit the file *setenv.sh*.
2. Add the following line:

   ```
   set JAVA_OPTS=%JAVA_OPTS% -XX:MaxMetaspaceSize=512m -Xmx1024m -Xms256m
   ```

   The Pivotal tc Server memory size is now increased and the server can hold several web applications.

3.2.1.5. Deploying Talend Administration Center on WebLogic

To deploy Talend Administration Center on WebLogic 12c, proceed as follows:

1. Create the *endorsed* folder, for instance */home/user1/weblogic/endorsed*.
2. Copy *org.talend.administrator/WEB-INF/lib/joda-time-2.1.jar* to */home/user1/weblogic/endorsed*.
3. Edit <*WebLogicPath*>*/user_projects/domains/base_domain/bin/setDomainEnv.sh* to reference the *endorsed* folder, as follows:

   ```
   JAVA_OPTIONS="${JAVA_OPTIONS} ${JAVA_PROPERTIES} -Djava.endorsed.dirs=/home/user1/
   weblogic/endorsed"
   ```
4. For a WebLogic 12.1.x server:

   1. Extract the war file *org.talend.administrator.war* to the location of your choice.

   For a WebLogic 12.2.1 server:

   1. Deploy the file *jsf-1.2.war* following the steps described at https://docs.oracle.com/cd/E24329_01/web.1211/e21049/configurejsfandjtsl.htm#WBAPP206 (*Deploying JSF 1.2 and JSTL Libraries*).
   2. Extract the war file *org.talend.administrator.war* to the location of your choice.
   3. Copy the file *weblogic.xml* to <*ExtractedWarPath*>*/WEB-INF/*.
5. Deploy the extracted *org.talend.administrator.war* file to WebLogic.

3.2.2. Talend Administration Center basic configuration

The following pages detail basic configuration procedures you should follow in order to install *Talend Administration Center*:

- *Increasing the memory of Tomcat*
- *Installing database drivers in your Web application server*
- *Configuring Tomcat to use a proxy server*
- *Synchronizing Web application and server time zones*
- *Launching Talend Administration Center*
- *Configuring Talend Administration Center to run on a different database than H2*
- *Link Talend Administration Center to your version control system*
- *Configuring the log storage mode*
- *Reduce the number of unauthenticated calls to your Git server*

3.2.2.1. Increasing the memory of Tomcat

To increase the memory heap size of the server in order to make all the applications work properly, proceed as follows:

1. If you are using Ubuntu, change directory to <*TomcatPath*>*/bin* where <TomcatPath> is the Tomcat installation directory, and edit the file *catalina.sh*.

   If you are using another Linux distribution, edit the file */usr/share/tomcat/conf*.

2. Add the following line:

   ```
   set JAVA_OPTS=%JAVA_OPTS% -XX:MaxMetaspaceSize=512m -Xmx1024m -Xms256m
   ```

   The Tomcat memory size is now increased and the server can hold several web applications.

   For Oracle users only:

   Add the following line to the *.sh* file in order to specify the *catalog* and *schema* database parameters, and to avoid errors during *Talend Administration Center* startup:

   ```
   Xmx<1G> -Dtalend.catalog=<catalogName> -Dtalend.schema=<schemaName>
   ```

3.2.2.2. Installing database drivers in your Web application server

Installing the database drivers

If you are not using the embedded H2 database with *Talend Administration Center* or *Talend Activity Monitoring Console*, you must install the driver for the database to use in your Web application server.

For more information regarding the databases compatible with *Talend Administration Center* and *Talend Activity Monitoring Console*, see *Compatible Databases*.

To install database drivers in your Web application server, proceed as follows:

1. Stop your Web application server.
2. In case you use Tomcat, clean the *<apache-tomcat>/work/Catalina/localhost* folder, where *<apache-tomcat>* corresponds to Tomcat installation directory.
3. Make sure that the driver for the database you want to use does not exist in any of these folders:

Web application Server used	Folders to check
Tomcat	• *<apache-tomcat>/webapps/org.talend.administrator/WEB-INF/lib* • *<apache-tomcat>/webapps/amc/WEB-INF/lib* • *<apache-tomcat>/lib*
JBoss	• *<JBoss_installation_folder>/standalone/lib/ext* • *<JBoss_installation_folder>/standalone/lib/endorsed* • *<JBoss_installation_folder>/standalone/tmp/work/jboss.web/default-host/amc/eclipse/plugins/org.talend.amc.libraries_X.X.X.XXXXXXXX_XXXX/lib/ext*

   If the driver already exists in one of these folders, skip the next step.
4. Download the correct database driver(s) from the official provider website, according to the version of the JVM you use to run your Web application server and the version of the database you want to use.

   In case you use Oracle, use a copy of the *ojdbcX.jar* file from your Oracle installation.

   Note that those drivers are specific and that you should only download the one(s) that you need.

Database used	Driver to download
MySQL	http://dev.mysql.com/downloads/connector/j/
Oracle	http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html
MS SQL	http://sourceforge.net/projects/jtds/files/jtds/
PostgreSQL	http://jdbc.postgresql.org/download.html
Microsoft JDBC Drivers 6.0, 4.2, 4.1, and 4.0 for SQL Server	http://www.microsoft.com/en-us/download/details.aspx?id=11774
MariaDB	https://downloads.mariadb.org/connector-java/

5. For Tomcat, if it does not exist, create a new *endorsed* folder under *<apache-tomcat>*.

   For JBoss, if it does not exist, create a new *ext* folder under *<JBoss_installation_folder>/standalone/tmp/work/jboss.web/default-host/amc/eclipse/plugins/org.talend.amc.libraries_X.X.X.XXXXXXXX_XXXX/lib/*.
6. Place the driver(s) you need in the folder you created.
7. Restart your Web application server.

(Best Practice) Using VACUUM with PostgreSQL for Talend Administration Center users

When using *Talend Administration Center* to retrieve, schedule and/or execute Jobs, many update/delete database operations are performed, which may result in performance slowdown if you are using PostgreSQL.

Indeed, it is recommended to execute the `VACUUM` command with PostgreSQL, as items that are deleted or obsoleted by an update are not physically removed from their table.

For more information on the `VACUUM` command, see the PostgreSQL documentation.

For more information on how to set up automatic vacuuming (which is a process launched at regular intervals by the PostgreSQL server to execute `VACUUM` only on the tables that have been updated), see the PostgreSQL documentation.

3.2.2.3. Configuring Tomcat to use a proxy server

To configure Tomcat to connect to a proxy server, proceed as follows:

1. Stop your Tomcat server.
2. If you are using Ubuntu, change directory to <*TomcatPath*>*/bin* where <TomcatPath> is the Tomcat installation directory, and edit the file *setenv.sh*.

   If the file does not exist, create it.

   If you are using another Linux distribution, edit the file */usr/share/tomcat/conf*.
3. Add the following lines, changing the parameters to match with your configuration:

```
[Tomcat Proxy Property]
-Dhttp.proxySet=true
-Dhttp.proxyHost=http(s)://proxy.server.com # Specify the host name or IP address of
 the proxy. You can use this parameter for http and https host names.
-Dhttp.proxyPort=3128 # Specify the port number of the proxy server.
-Dhttp.nonProxyHost=localhost|host.mydomain.com|192.168.0 # Specify a list of hosts
 separated by "|" that do not require access through the proxy server.
```

4. Restart your Tomcat server.

3.2.2.4. Synchronizing Web application and server time zones

To make sure that the DST change and the time zones are correctly taken into account, check that your OS includes an environment variable set as follows:

```
On Windows: TZ=Europe/Paris

On Linux: Export TZ="Europe/Paris"
```

If you wish to implement high availability to the task execution scheduling using *Talend* clustering feature, then read *Talend High Availability*.

3.2.2.5. Launching Talend Administration Center

The recommended way to configure the connection to the database and to the shared repository (Git or SVN) is through the Web interface of *Talend Administration Center*.

1. Start the application server on which *Talend Administration Center* is installed.
2. Open a Web browser and type in the following URL:

   http://localhost:8080/<ApplicationPath>

   Replace <*localhost*> with the IP address or the hostname of the Web server if the Web browser IP is different from the machine you are on, and <*ApplicationPath*> with the *Talend Administration Center* Web application path. For example, *http://localhost:8080/org.talend.administrator*.

   Choose a port according to your environment. The default port 8080 may clash with another application.

When connecting for the first time, a dialog box is displayed, asking you the password to access the **Database Configuration** page.

3. Type in the default *admin* password. H2 database connection parameters are displayed and some automatic checks are performed on driver, url, connection, version information.

   The administration database (storing users, rights, etc.) being an H2 embedded database, its access information is automatically set. However:

   - for security reasons and to improve performances, the `;MV_STORE=FALSE;MVCC=TRUE` additional parameters must be added to the H2 default URL. The H2 database URL should thus looks like the following:

     ```
     jdbc:h2:~/
     talend_administrator;MV_STORE=FALSE;MVCC=TRUE;AUTO_SERVER=TRUE;LOCK_TIMEOUT=15000
     ```

   - if you do not want to use the embedded H2 database, you can set up a different database server (MySQL, MSSQL or Oracle) and set the corresponding connection parameters. For more information, see *Configuring Talend Administration Center to run on a different database than H2*.

4. Click **Set new license**, then browse your system to the License file you received from *Talend* and click **Upload**. A final License check is performed.

5. Click **Go to Login**.

6. On the **Login** page, type in the default connection login for your first access (login: *security@company.com*, password: *admin*).

   Those credentials correspond to the default user of the Web application.

   After the first connection, it is strongly recommended not to use the default user account to access the application for security reasons. You can either change the default credentials of this account (*security@company.com*/*admin*) or create another administrator user and remove the default account.

If your Web access is restricted, you may need to click **Validate your license manually** to perform the validation of your license key. Follow the instructions on screen.

Once the license is validated, the navigation bar of *Talend Administration Center* opens with all the pages accessible for the default administrator user account.

For more information on which pages of *Talend Administration Center* an administrator user can access, see the *Talend Administration Center User Guide*.

3.2.2.6. Configuring Talend Administration Center to run on a different database than H2

By default, the Talend Administration Center Web application is configured to run with the default H2 embedded database.

For more information on *Talend Administration Center* database, see *Database*.

Prerequisite:

- The external database must have been created with a *utf8* collation.

To run Talend Administration Center on another database:

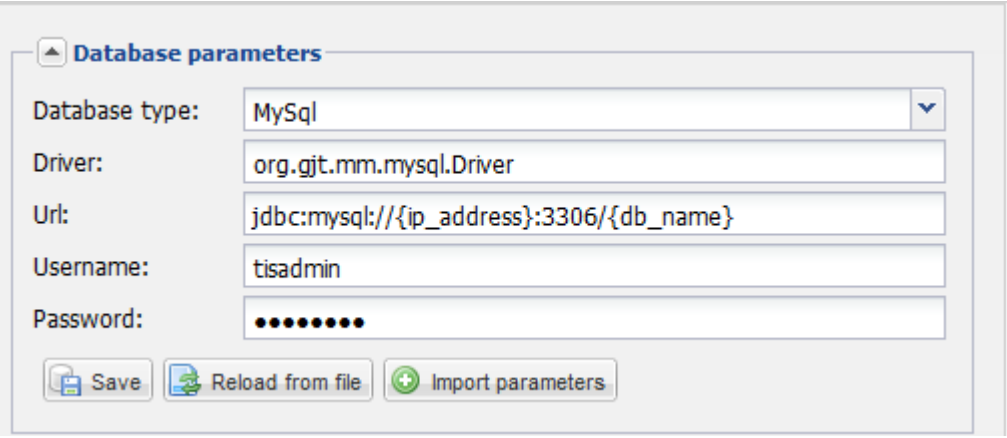
1. If you want to use a MySQL, Oracle or MS SQL database for *Talend Administration Center*, install the right database driver in the application server as described in *Installing database drivers in your Web application server*.

For MySQL users: to prevent further transaction issues when resuming a trigger on the **Job Conductor** page of *Talend Administration Center*, it is recommended to configure MySQL as described in the documentation on Talend Help Center.

2. Start the application server, then open a Web browser and type the URL of the *Talend Administration Center* Web application.

3. On the Login page, click **Go to db config page**, then enter the administrator password (by default, it is *admin*).

   Note that if you are starting Talend Administration Center for the first time, you already are on the database configuration page.



4. In the **Database type** list, select your database. As a result, the **Driver** and **Url** fields are automatically updated with the template corresponding to this database.

5. In the **Url** field, replace the parameters in brackets with your database details.

   Note that you can click the **Reload from file** button to reload your previous database as changes are not saved until you click **Save**.

6. Click **Save** to take your changes into account.

3.2.2.7. Link Talend Administration Center to your version control system

1. Click **Configuration** to access the setting page of *Talend Administration Center*.

2. Change the following parameters for the Git or SVN module using the parameters you have set during the installation process of the Git or SVN server.

   - **Server Location URL**: your Git or SVN repository URL.

   - **Username**: your Git or SVN repository user.

   - **Password**: your Git or SVN repository password.

For examples of Git or SVN URLs, and more details, see *Setting up your version control system*.

If you use several Git or SVN repositories to store your projects, refer to the User Guide of *Talend Administration Center* and check the *Advanced settings* procedure.

The link to Git or SVN is now established, you can thus create a new project in order for the *Talend* clients to have at least one project in their workspace. Next steps:

- Create one or more users from the **Users** page.
- Create a new, remote, collaborative project from the **Projects** page.
- Associate the user(s) with the project from the **Project authorizations** page.

For more details, see the *Talend Administration Center User Guide*.

(Git only) Configuring Talend Administration Center to handle multiple repositories

You might need to connect Talend Administration Center to another Git repository than the one you entered on the **Configuration** page of the web application, for example when you are creating new projects. If so, you need to update a configuration file.

Prerequisite: You need to shut down Tomcat.

1. Open the following file to edit it:

   <tomcat_path>WEB-INF\classes\configuration.properties

2. Add the following:

   ```
   git.conf.enableHashRepositoryUrl=true
   ```

   Note that this configuration may increase disk space usage if you use different protocols (http / https / ssh, etc.) to access the same repository.

3. Restart Tomcat.

   Now a separate local folder will be created for each Git repository URL entered in Talend Administration Center.

(Git only) Configuring Talend Administration Center to improve performance when creating execution tasks

When creating execution tasks on the **Job Conductor** page of Talend Administration Center, you might notice some slowness when retrieving the Jobs stored on the Git repository. If so, you have the possibility to update a configuration file in order to prevent Git from refreshing the repository and thus to improve speed of item selection.

Prerequisite: You need to shut down Tomcat.

1. Open the following file to edit it:

   <tomcat_path>WEB-INF\classes\configuration.properties

2. Add the following:

   ```
   git.conf.refreshDataProjectWhenSelectJob.enable=false
   ```

   Note that the recommended parameter value is *true*, and that you should only disable to prevent refresh of local Git repositories.

3. Restart Tomcat.

   Now the selection of Git items from the **Job Conductor** page of Talend Administration Center will be quicker.

3.2.2.8. Configuring the log storage mode

The log outputs are stored by default in the server application standard log file (STDOUT) as defined in the *Log4j.xml* file located in the *<ApplicationPath>/WEB-INF/classes* folder. However you can store the log in a different file by setting the path to this file in the *Log4j.xml* file.

- To do so, simply set the path in the **Configuration** page in *Talend Administration Center*. For more information, refer to your *Talend Administration Center* User Guide. If you leave the path field blank in the **Configuration** page, then you can also customize the *Log4j.xml* to address your custom needs.

3.2.2.9. Reduce the number of unauthenticated calls to your Git server

When using the Git HTTP protocol, you can force the use of username/password authentication for all *pull*, *push*, *fetch* and *ls-remote* operations.

1. Stop your Tomcat server.
2. Open the following file to edit it:

   <tomcat_path>/WEB-INF/classes/configuration.properties
3. Add the following line:

   git.conf.http.onlyUsernamePasswordAuth=true
4. Restart your Apache Tomcat server.

3.2.3. Installing and Configuring Talend server modules

The following pages provide installation and configuration procedures for your *Talend* server modules.

- *Installing and configuring the Nexus artifact repository*
- *Installing and configuring your JobServers*
- *Installing Talend Runtime*
- *Installing the Talend Activity Monitoring Console web application*
- *Installing and configuring the Drools Business Rules Management System (BRMS)*

3.2.3.1. Installing and configuring the Nexus artifact repository

Talend Administration Center is provided together with Nexus artifact repository. It is used to store software updates, ESB and Data Integration Job artifacts.

This tool is used for the **Software Update** feature and its instance holds the *talend-updates* repository where the updates are retrieved by the user.

It can also be used as a catalog for the Jobs, Services, Routes, and Generic OSGi Features created from *Talend Studio* or any other Java IDE. For this, two repositories are available: *repo-snapshot* for development purposes and *repo-release* for production purposes.

This instance is embedded in the *.zip* file of *Talend Administration Center* Web application and it allows you to store artifacts designed from *Talend Studio* or any other Java IDE and ready to be deployed and executed in an execution server. For more information, see the *Talend Administration Center User Guide*.

So when unzipping *Talend Administration Center* zip file, you will find an archive file called *Artifact-Repository-Nexus-VA.B.C.D.E* containing a ready-to-be-used Nexus artifact repository.

Nexus is based on Sonatype Nexus. For more information on how to use it, see *Artifact Repository* and Sonatype Nexus's documentation on http://www.sonatype.org/nexus.

To install and configure Nexus artifact repository, see the following procedures:

- *Installing the Nexus artifact repository*
- *Configuring the Software Update repository in Talend Administration Center*
- *Configuring Nexus in Talend Administration Center*

For more information on how to configure Nexus in **Talend Runtime**, see *Configuring Nexus in Talend Runtime*

Installing the Nexus artifact repository

To install the Nexus artifact repository, proceed as follow:

1. Unzip the *Artifact-Repository-Nexus-VA.B.C.D.E* archive file in a dedicated folder.
2. Add the execution rights to the relevant files using the following commands:

```
$ chmod 755 bin/nexus
$ chmod 755 bin/jsw/linux-ppc-64/*
$ chmod 755 bin/jsw/linux-x86-32/*
$ chmod 755 bin/jsw/linux-x86-64/*
$ chmod 755 bin/jsw/macosx-universal-32/*
$ chmod 755 bin/jsw/macosx-universal-64/*
```

3. Launch the Nexus instance using the command `nexus.sh console`. If you installed Nexus as a service, run the `start` command to launch it.
4. Log in the Nexus Web application (default login information: *admin*/*Talend123*). Make sure you have the relevant rights to access the *releases* and *snapshots* repositories.

Configuring the Software Update repository in Talend Administration Center

Once you installed the Nexus artifact repository and started it, you can configure it to use *Talend* Software Update.

Once you have launched and configured the **Software Update** repository, go to the **Configuration** page of *Talend Administration Center* and fill in the following information in the **Software Update** group:

- **Talend update url** : Location URL to the *Talend* remote repository from which software updates are retrieved, this field is filled by default.
- **Talend update username** et **Talend update password** : Type in the credentials of the software update repository user that you received from *Talend*.
- **Local repository url** : Type in the location URL to the repository where software updates are stored. By default, it is *http://localhost:8081/nexus/*.
- **Local deployment username** and **Local deployment password** : Type in the credentials of the user with deployment rights to the local repository. By default, it is *talend-updates-admin/talend-updates-admin*.

- **Local reader username** and **Local reader password** : Type in the credentials of the user with read rights to the local repository. By default, no credentials are required but you are free to define them if you want to disable public access to the repository.
- **Local repository ID** : Type in the ID of the repository in which software updates are published. By default, it is *talend-updates*.

In the **Software Update** page of *Talend Administration Center*, you can now see the versions and patches available and download them according to your needs.

Configuring Nexus in Talend Administration Center

Once you have launched the Nexus artifact repository, go to the **Configuration** page of Talend Administration Center and do the following:

- Fill in the following information in the **Artifact Repository** node:
  - **Artifact repository type**: select **NEXUS**.
  - **Nexus url**: Type in the location URL to your Nexus Artifact repository, *http://localhost:8081/nexus/* for example.

    "*http://localhost:8081/nexus*" is only given as example. Depending on your configuration, you may have to replace <*localhost*> with the IP address of the Web server and <*8081*> with the port that is used for your repository instance.
  - **Nexus username**: Type in the name of the repository user with Manager role. By default, it is *admin*.
  - **Nexus password**: Type in the password of the repository user with Manager role. By default, it is *Talend123*.
  - **Nexus Default Release Repo**: Type in the Nexus artifact repository's Release repository name. By default, it is *releases*.
  - **Nexus Default Snapshot Repo**: Type in the Nexus artifact repository's Snapshot repository name. By default, it is *snapshots*.
  - **Nexus Default Group ID**: Type in the name of the group in which to publish your Jobs, Service and Route artifacts. By default, it is *org.example*.

From the **Job Conductor** and **ESB Conductor** pages of *Talend Administration Center*, you can retrieve all the artifacts published in the two repositories to configure their execution in your execution server. For more information, see the *Talend Administration Center User Guide*.

3.2.3.2. Installing and configuring your JobServers

The execution servers allow you to execute the Jobs (processes) developed with *Talend Studio* from the *Talend Administration Center* web application.

Note that you can also use Talend Runtime servers to deploy and execute Jobs tasks if these Jobs are linked with Services or Routes, but these servers are especially used to deploy and execute Services, Routes, or even generic OSGi features when you are using our service-oriented products. For more information about the installation of Talend Runtime, see *Installing Talend Runtime*.

To install and configure your JobServers, see the following procedures:

- *Installing your JobServers*

- *Configuring the JVM for your JobServer (optional)*
- *Configuring the SSL Keystore (optional)*
- *Configuring user impersonation for JobServer*
- *Disabling some SSL ciphers (optional)*

Installing your JobServers

In order to install your JobServers, proceed as follows:

Unzip the archive file

1. First select the servers that will be used to execute the Jobs developed with *Talend Studio*.
2. Then, on each server, uncompress the archive file containing the JobServer application matching your version of *Talend Studio*.

   The archive file name for example reads: *Talend-JobServer-YYYYMMDD_HHmm-VA.B.C.zip*
3. In the uncompressed file you need to configure the file *TalendJobServer.properties* that you can find in the directory <*root*>*/conf/* where <root> is the JobServer path.

   For example, if you want to change the directory where the JobServer stores its data, change the *org.talend.remote.jobserver.commons.config.JobServerConfiguration.ROOT_PATH* parameter.
4. Modify the installation directory of JobServer and check that the *8000*, *8001* and *8888* ports are available.

Enable user authentication

1. To enable user authentication on JobServer, you need to define one or more lines of username and password pairs in the file *users.csv* that you can find in the directory <*root*>*/conf/* where <root> is the JobServer path.
2. In the directory you have unzipped, you will find the files *start_rs.sh* and *stop_rs.sh* that will let you respectively start and stop the JobServer.

You may need to change the *java.library* path in order to load the correct native library for your system. In this case, adapt the variable `MY_JSYSMON_LIB_DIR` in the script *start_rs.sh*.

JobServer is an application that allows a system installed on the same network as the Web application to declare itself as an execution server. These systems must obviously have a working JVM. For more information about the prerequisites of JobServer, see *Software requirements*.

Information about JobServer resources and load balancing :

Once you have declared these execution servers in the **Servers** page of the *Talend Administration Center* Web application, their resources (CPU, RAM, etc.) are displayed. For more information on how to do this, see your *Talend Administration Center User Guide.*

For some operating systems, the CPU information may not be available. You can test your system by setting up the following variable as `true`:

`org.talend.monitoring.jmx.api.OsInfoRetriever.FORCE_LOAD` in the file *TalendJobServer.properties.*

For users working in cluster mode, note that the ranking of servers to be used for load balancing is based on indicators, whose bounds (such as free disk space limits) and weight are defined in the file: `monitoring_client.properties` which is located in <*ApplicationPath*>*\WEB-INF\lib \org.talend.monitoring.client-A.B.C.jar*. These values can be edited according to your needs. For more information, see *Configuring the indicators which determine which server to be used for load balancing*.

Configuring the JVM for your JobServer (optional)

The JobServer application provided by *Talend* allows you to choose another JVM than the one used by default to launch your Jobs.

To change the Job launcher path, proceed as follows:

1. Go to the directory *<root>/conf/*, where <root> is the JobServer path, and open the *TalendJobServer.properties* file to edit it.
2. In the line dedicated to the Job launcher path, add the path to your java executable after the equal sign.

```
# Set the executable path of the binary which will run the job, for example: /usr/
bin/java/java or "c:\\Program Files\\Java\\bin\\java.exe"
org.talend.remote.jobserver.commons.config.JobServerConfiguration.JOB_LAUNCHER_PATH=C:
\Program Files\Java\jre1.8.0_65\bin\java.exe
```

The use of quotes is only necessary when your path contains spaces, as shown in the capture. Otherwise, type in the path without quotes.

3. Save your changes and close the file.

The next time you launch JobServer, the java executable used will be the one you have previously set in the *TalendJobServer.properties* file.

Configuring the SSL Keystore (optional)

You are also able to choose another Keystore if needed.

To override the existing Keystore file, you have to:

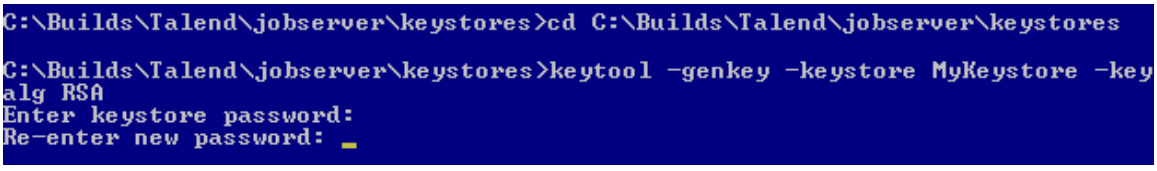
- generate a new Keystore with the utility tool called Keytool (Key and Certificate Management Tool);
- set the new Keystore location;
- enable the SSL Keystore at server side.

To generate a Keystore

1. Open a command prompt and change directory to *<root>/keystores* where <root> is the JobServer path.
2. Type in the following:

   `keytool -genkey -keystore <myKeystoreName> -keyalg RSA`

   where `<myKeystoreName>` refers to the name of the Keystore you are creating.



3. Enter the password for your Keystore twice, then enter the other optional information, such as your name, the name of your organization, your state etc., if needed.
4. Type in `yes` to confirm your information.
5. Type in the password you have previously defined. The new Keystore file has been created in *<root>/keystores*.

To set the location of the new Keystore

To set the new Keystore location, you can either edit the JAVA_OPTS environment variable or edit the launching script (*start_rs.sh*) of the Jobserver.

1. To edit the JAVA_OPTS environment variable, add:

   ```
   -Djavax.net.ssl.keyStore=/<myDirectory>/<myKeystore>
   ```

   ```
   -Djavax.net.ssl.keyStorePassword=<myPassword>
   ```

   to your JAVA_OPTS environment variable, where `<myDirectory>` is the installation directory of your Keystore, `<myKeystore>` is the name of your Keystore and `<myPassword>` is the password you have previously defined for your Keystore.

   If you have not created the JAVA_OPTS environment variable yet, you have to create it before completing this procedure.

   OR

2. To edit the launching script, open the *start_rs.sh* to edit it.

   ```
   rem set the JVM arguments here
   set MY_JMV_ARGS=-Dcom.sun.management.jmxremote -Djava.library.path="%MY_ROOT_PATH%sigar-bin\lib>"
   -Djavax.net.ssl.keyStore=/C:/Builds/Talend/jobserver/keystores/MyKeystore
   -Djavax.net.ssl.keyStorePassword=
   ```

3. As shown in the capture, add

   ```
   -Djavax.net.ssl.keyStore=/<myDirectory>/<myKeystore>
   ```

   ```
   -Djavax.net.ssl.keyStorePassword=<myPassword>
   ```

   to the JVM arguments location, where `<myDirectory>` is the installation directory of your Keystore, `<myKeystore>` is the name of your Keystore and `<myPassword>` is the password you have previously defined for your Keystore.

To configure the service

- Edit an init script with `start` and `stop` commands as described in *Installing JobServer as a service*.

Now you just have to enable Secure Sockets Layer as described in *Enabling the SSL encryption in Talend Runtime*.

Configuring user impersonation for JobServer

The *Talend Administration Center* web application allows you to run tasks as different UNIX system users, through the **Run As** option. To avoid errors when starting the task on the server, you need first to:

- give specific permissions to some server directories.
- give necessary authorizations to the directories and files created by the JobServer by configuring the umask.
- define the Operating System users allowed to run tasks from the server.

For more information on this feature, see the *Talend Administration Center User Guide*.

Setting the server directory permissions

Prerequisite: If you have already started Jobs from this server, it is recommended to remove the directory *<jobserver_path>/TalendJobServerFiles* to avoid unexpected authorizations on already deployed Jobs or cached files.

1. Add each user allowed to run tasks (for example, user called *subuser*) to the 'root' group as well as to the group of the user who owns the parent directories of JobServer (for example, group of the user called *myuser*), such as:

```
> sudo usermod -a -G myuser_group subuser
> sudo usermod -a -G root subuser
```

2. Give the permissions **execute** to *myuser_group* in the following directories by executing the command `chmod g+rx /<directory_path>`:

   /DIRECTORY_1

   /DIRECTORY_1/DIRECTORY_2

   /DIRECTORY_1/DIRECTORY_2/Talend-JobServer

   /DIRECTORY_1/DIRECTORY_2/Talend-JobServer/cache

   /DIRECTORY_1/DIRECTORY_2/Talend-JobServer/cache/lib

   /DIRECTORY_1/DIRECTORY_2/Talend-JobServer/repository

   Note that the read authorization for the group is only required for deployed files.

Configuring the umask of the user which launches the JobServer

- Set the user profile with the following umask: `umask u=rwx,g=rx,o=`

  which is the same as `umask 0027`

  This configuration will create:

  - directories with *group* authorization equal to `r-x`
  - files with *group* authorization equal to `r--`
  - no authorizations for *others*

Defining the list of users allowed to run tasks as different users

1. Open the following file: *<jobserver_path>/conf/TalendJobServer.properties*
2. Edit the `org.talend.remote.jobserver.server.TalendJobServer.RUN_AS_WHITELIST` value and add all the users you need.

   Note that spaces as well as commas are valid separators for user name values in this file.

Starting the JobServer

- Start the JobServer using this command: `sudo sh start_rs.sh`

  Note that if you do not use `sudo`, the Jobs will hang because a password will be required at JobServer side.

Disabling some SSL ciphers (optional)

SSL ciphers are encryption algorithms that are used to establish a secure communication. Some cipher suites offer a lower level of security than others, and you may want to disable these ciphers. To do so:

1. Go to the directory *<root>/conf/* and open the *TalendJobServer.properties* file.
2. Add to the following parameter the list of ciphers that you want to disable:

   `org.talend.remote.jobserver.server.TalendJobServer.DISABLED_CIPHER_SUITES`

   Here is the list of the ciphers supported by JobServer:

```
TLS_KRB5_WITH_3DES_EDE_CBC_MD5
TLS_KRB5_WITH_RC4_128_SHA
SSL_DH_anon_WITH_DES_CBC_SHA
TLS_DH_anon_WITH_AES_128_CBC_SHA
TLS_DHE_RSA_WITH_AES_128_CBC_SHA
SSL_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA
SSL_RSA_EXPORT_WITH_RC4_40_MD5
SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA
TLS_KRB5_WITH_3DES_EDE_CBC_SHA
SSL_RSA_WITH_RC4_128_SHA
TLS_KRB5_WITH_DES_CBC_MD5
TLS_KRB5_EXPORT_WITH_RC4_40_MD5
TLS_KRB5_EXPORT_WITH_DES_CBC_40_MD5
SSL_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA
TLS_KRB5_EXPORT_WITH_RC4_40_SHA
SSL_DH_anon_EXPORT_WITH_RC4_40_MD5
SSL_DHE_DSS_WITH_DES_CBC_SHA
TLS_KRB5_WITH_DES_CBC_SHA
SSL_RSA_WITH_NULL_MD5
SSL_DH_anon_WITH_3DES_EDE_CBC_SHA
TLS_RSA_WITH_AES_128_CBC_SHA
SSL_DHE_RSA_WITH_DES_CBC_SHA
TLS_KRB5_EXPORT_WITH_DES_CBC_40_SHA
SSL_DH_anon_EXPORT_WITH_DES40_CBC_SHA
SSL_RSA_WITH_NULL_SHA
TLS_KRB5_WITH_RC4_128_MD5
SSL_RSA_WITH_DES_CBC_SHA
TLS_EMPTY_RENEGOTIATION_INFO_SCSV
SSL_RSA_EXPORT_WITH_DES40_CBC_SHA
SSL_DH_anon_WITH_RC4_128_MD5
SSL_RSA_WITH_RC4_128_MD5
TLS_DHE_DSS_WITH_AES_128_CBC_SHA
SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA
SSL_RSA_WITH_3DES_EDE_CBC_SHA
```

3.2.3.3. Installing Talend Runtime

According to the solution you have subscribed to, *Talend* recommend you to use:

- JobServer, if you have subscribed to a Data Integration solution;
- *Talend Runtime*, if you have subscribed to both a Data Integration and a ESB solution.

  However, if you are willing to use both *Talend Runtime* and JobServer on the same machine, you are required to change the port numbers because, by default, both servers are using the same ports.

Talend Runtime is an OSGi container, based on Apache Karaf, allowing you to deploy and execute various components and applications inside its *deploy* folder. It can be used as an execution server to deploy and execute all the Services, Routes and Generic OSGi Features created from *Talend Studio* or any other Java IDE.

You have the possibility to adapt the configuration of the *Talend Runtime* container to deploy several containers on the same machine. For more information, see the *Talend ESB Container Administration Guide*.

To install and configure your *Talend Runtime*, see the following procedures:

- *Installing the Talend Runtime containers*

- *Enabling the SSL encryption in Talend Runtime*
- *Configuring Nexus in Talend Runtime*

Installing the Talend Runtime containers

You need now to define on which server(s) you will install *Talend Runtime*.

1. First select the servers that will be used for the execution.
2. Then, on each server, unzip the archive file containing the *Talend Runtime* application matching your release version of *Talend*.

   The archive file name for example reads: *Talend-Runtime-V6.2.0.zip*
3. In the unzipped file you might need to configure the files *org.ops4j.pax.web.cfg* to change the HTTP listening port and *org.apache.karaf.management.cfg* to manage RMI connection to connect to Talend ESB Container via JMX, that you can find in the directory *Talend-Runtime-VA.B.C/etc*. Note that this file also allows you to define the artifact repository URL.
4. To launch *Talend Runtime*, browse to the *bin* directory and run the *trun* file.

Now we simply have to declare these runtime instances in the Web application and their resources (CPU, RAM, etc.) should become available. To do this:

1. Go to the **Servers** page of *Talend Administration Center*.

   Only users that have Operation Manager role and rights can have a read-write access to this page. For more information on access rights, see your Talend Administration Center User Guide. So, you have to connect to Talend Administration Center as an Operation Manager to be able to configure your servers.
2. And define the server as follows:

Label	*TestingServer*
Description	Type in the description of server.
Host	*localhost*
Command port	*8000*
File transfer port	*8001*
Monitoring port	*8888*
Timeout on unknown status(s)	*120*
Username	Type in the username for user authentication to access a Job server.
Password	Type in the password for user authentication to access a Job server.
Active	Select/clear the check box to activate/deactivate this server
Use SSL	Select/clear the check box to use or not your own SSL Keystore to encrypt the data prior to transmission. For more information about how to enable SSL, see *Enabling the SSL encryption in Talend Runtime*.
Talend Runtime	By default, servers created are Job servers. To deploy and execute your Jobs, Services, Routes or Generic tasks into *Talend Runtime*, select the **Talend Runtime** check box. The following fields will display: **Mgmt-Server port**, **Mgmt-Reg port**, **Admin Console port** and **Instance**.
Mgmt-Server port	RMI Server Port (44444 by default). This field is mandatory.
Mgmt-Reg port	RMI Registry Port (1099 by default). This field is mandatory.

Admin Console port	Port of the Administration Web Console (8040 by default). This field is mandatory and allows to activate the **Admin server** button allowing you to access the Administration Web console.
Instance	Type in the name of the container instance in which you will deploy and execute your Jobs, Services, Routes or Generic tasks, *trun* by default.

This corresponds to the configuration of a Talend Runtime on the system that hosts the Web application. For any other system, the **Host** field should contain the IP address of the system. Check also that the ports *8000*, *8001* and *8888* are available. These ports must be the same as defined in the *TalendJobServer.properties* defined above. Note that if no username and password pairs are defined in the file *users.csv* in the directory *<root>/conf/* where <root> is the JobServer path, then you do not have to set the **Username** and the **Password**.

3. Click the **Servers** page again so that the *Talend Runtime* servers appear with their properties.

Enabling the SSL encryption in Talend Runtime

The execution servers provided by *Talend* allows you to encrypt data prior to transmission via an existing SSL Keystore. To enable Secure Sockets Layer (SSL) at server side in order to establish an encrypted link between the Jobserver and its clients, proceed as follows:

1. If you want to configure *Talend Runtime*, go to the *etc* directory and open the *org.talend.remote.jobserver.server.cfg* file to edit it.

   If you want to configure the JobServer, go to the *<root>/conf/* directory and open the *TalendJobServer.properties* file to edit it.

2. Edit the following line

   ```
   org.talend.remote.jobserver.server.TalendJobServer.USE_SSL=false
   ```

   and replace `false` with `true`.

   The next time you launch your execution server, the SSL protocol will be used to secure the communication between servers and clients.

From *Talend Administration Center*, you have to select the **Use SSL** check box to enable the encryption.

Configuring Nexus in Talend Runtime

The default Nexus artifact repository URL is described in the *etc/org.ops4j.pax.url.mvn.cfg* file.

If your artifact repository has been installed on another URL, edit the *org.ops4j.pax.url.mvn.repositories* part of the file.

3.2.3.4. Installing the Talend Activity Monitoring Console web application

The *Talend Activity Monitoring Console* Web application we are talking about in this section is available in *Talend Administration Center*. However, the **AMC** is also a perspective available in Talend Studio. For more information about this perspective, see the *Talend Activity Monitoring Console User Guide*.

For more information on *Talend Activity Monitoring Console*, see *Talend Activity Monitoring Console log database*.

To install the *Talend Activity Monitoring Console*, follow these procedures:

- *Deploying the Talend Activity Monitoring Console Web application*
- *Configuring the Talend Activity Monitoring Console in Talend Administration Center*

Deploying the Talend Activity Monitoring Console Web application

1. Unzip the *Talend-AMC_Web-YYYYYYYY_YYYY-VA.B.C.zip* archive file containing the *amc.war* file on the same machine as *Talend Administration Center* Web application.
2. Paste the *amc.war* file in the same *webapps* folder as the one where *Talend Administration Center* is located, for example *<TomcatPath>/webapps/*.
3. Restart your Web application server.

If you want to install *Talend Activity Monitoring Console* on other Web application server than the one where *Talend Administration Center* is installed, follow the same procedure as *Deploying Talend Administration Center on an application server*.

If you do not want *Talend Activity Monitoring Console* to use the default H2 database, see *Installing database drivers in your Web application server*.

Configuring the Talend Activity Monitoring Console in Talend Administration Center

In the *Talend Administration Center* web application, you have to set up the link to the *Talend Activity Monitoring Console*.

- To do so, specify the following information on the **Monitoring** group of the **Configuration** page:
  - **AMC url**: type in the URL address of the *Talend Activity Monitoring Console* application, `http://localhost:8080/amc` for example.

    "http://localhost:8080/amc" is only given as example. Depending on your configuration, you may have to replace <localhost> with the IP address of the Web server and <8080> with the actual port of the <AMC>.

    You also have to configure three database tables that will store statistics and log information.

    For more information about this application, see the *Talend Activity Monitoring Console User Guide*.

3.2.3.5. Installing and configuring the Drools Business Rules Management System (BRMS)

To install and configure the Drools Business Rules Management System (BRMS) on your machine, follow these procedures:

- *Installing manually the Drools Business Rules Management System (BRMS)*

- *Configuring the Drools Business Rules Management System (BRMS) in Talend Administration Center*

Installing manually the Drools Business Rules Management System (BRMS)

To manually install the Drools Business Rules Management System (BRMS), proceed as follows:

1. Download the *Talend-BRMS-YYYYMMDD_HHmm-VA.B.C.zip* file and unzip it.
2. Stop your Tomcat server.
3. Go to the *Talend-BRMS-Webapp-A.B.C* folder.
4. Copy the *talend-brms-A.B.C.war* file to *<TomcatPath>/webapps*.
5. Unzip the *WorkbenchInstallationResources.zip* file.
6. Copy the content of the *WorkbenchInstallationResources/lib* folder to *<TomcatPath>/lib*.
7. Got to the *WorkbenchInstallationResources/conf* folder.
8. Copy the *btm-config.properties* and *resources.properties* files to *<TomcatPath>/conf*.
9. Create a backup copy of your *<TomcatPath>/bin/setenv.sh* file.
10. Copy the *WorkbenchInstallationResources/bin/setenv.sh* file to *<TomcatPath>/bin*.
11. Restart Tomcat to deploy the Drools Business Rules Management System (BRMS).

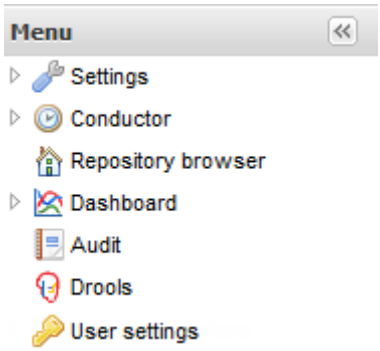
Configuring the Drools Business Rules Management System (BRMS) in Talend Administration Center

Before being able to use Drools BRMS in *Talend Administration Center* web application, you will need to configure its URL in the **Configuration** page. For more information about configuring the Web application, first read *Installing and configuring Talend Administration Center*.

Then after you access *Talend Administration Center*'s **Configuration** page, set Drools URL, *http://10.42.10.84:8080/kie-drools-wb/* for example.

For more information about the actual data contained on the **Configuration** page of *Talend Administration Center*, check out the *Talend Administration Center User Guide*.

When Drools Guvnor URL is correctly set up, the application can be accessed through the menu tree view of *Talend Administration Center*:



If you have used Drools with older *Talend* product releases and you do not want to use the latest Drools version installed with *Talend Administration Center*, you must upgrade your Drools repository before being able to use it with the current

release. For more information, see the article about migrating Drools repository from version 5.x to version 6.0 on Talend Help Center.

3.2.4. Talend Administration Center advanced configuration

Most of the configuration parameters are stored in the Talend Administration Center database, like backup-related settings, port information, timeout duration, security settings, login delay and so on.

Some parameters can be updated, activated or deactivated from the **Configuration** page of the Web application or directly in the *configuration.properties* file, but you might need to edit some of them manually in the *configuration* table of the Talend Administration Center database. To access this database, open the database web console. To edit this database, open its web console which is accessible from the **Database** node of the **Configuration** page of *Talend Administration Center*.

The following pages detail advanced configuration procedures for *Talend Administration Center*:

- *Setting up Talend Administration Center Single Sign-On (SSO)*
- *Setting up High Availability*
- *Migrating database X to database Y*
- *Disabling SSL3 in Tomcat*
- *Managing the database parameters*
- *Managing the connection pool via Tomcat*
- *Configuring the indicators which determine which server to be used for load balancing*
- *Customizing the Talend Administration Center Menu tree view*
- *Configuring Talend Administration Center login delay*
- *Configuring LDAP(S) for Talend Administration Center*

3.2.4.1. Setting up Talend Administration Center Single Sign-On (SSO)

You have the possibility to implement a unified sign-on and authentication to access Talend Administration Center through different Identity provider systems (IdP) and to manage the roles and project types of the application users:

1. You need first to enable SSO for Talend Administration Center during installation, either via *Talend Installer* or from a configuration file, see *Enabling Single-Sign On for Talend Administration Center*.
2. Then set up SSO and user roles and project types from your Identity Provider system:
   - Okta, see *Setting up a Talend Administration Center SSO from Okta*
   - SiteMinder, see *Setting up a Talend Administration Center SSO from SiteMinder*.
3. (Optional) You can create an "emergency user" in Talend Administration Center in case your Identity Provider is temporarily unavailable, see *Defining an emergency user for Talend Administration Center*.

Setting up SSO in your Identity Provider system allows users to access all their applications, including Talend Administration Center, by signing in one time for all services. If a user tries to sign in to Talend Administration Center when SSO is set up, he or she is redirected to the SSO sign-in page.

Enabling Single-Sign On for Talend Administration Center

To activate SSO for *Talend Administration Center* during installation, you can:

- activate SSO via *Talend Installer* (recommended)
- activate SSO by editing a configuration file

Note that, if you do not activate SSO during installation, you still have the possibility to do so on the **Configuration** page once you are logged in the web application. For more information, see the *Talend Administration Center User Guide*.

Via Talend Installer :

Prerequisite: You have chosen to perform an **Advanced** installation and a **Server** or a **Custom** installation type, that allows you to customize settings during installation. See *Installation modes of Talend Installer and Talend Studio Installer* and *Using Talend Installer graphical installation mode* for more information.

- In the **Talend Administration Center Configuration** step of the Installer, select the **Enable SSO** check box to activate SSO during installation and continue the installation process..

  SSO is activated, which means the first time the administrator logs in Talend Administration Center, he or she will be able to configure the link between the application and his or her Identity provider system directly from the **Talend Administration Center Database Configuration** page.

  For more information, see *Talend Administration Center User Guide*.

In the configuration file:

You also have the possibility to enable SSO directly from a configuration file.

1. Open the following file to edit it:

   <tomcat_path>WEB-INF\classes\configuration.properties

2. Set the `sso.field.useSSOLogin` parameter value to `true` and save your changes.

   SSO is activated, which means the first time the administrator logs in Talend Administration Center, he or she will be able to configure the link between the application and his or her Identity provider system directly from the **Talend Administration Center Database Configuration** page.

   For more information, see *Talend Administration Center User Guide*.

Setting up a Talend Administration Center SSO from Okta

Prerequisite: You have an administrator Okta account in your organization.

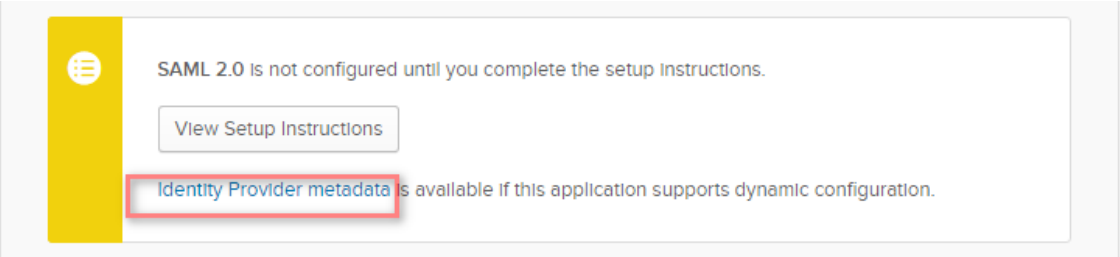
Add the Talend Administration Center application in Okta

1. Log in to your Okta organization.
2. Click the **Admin** button.

3. Click **Add Applications**, then click the **Create New App** button.
4. Select SAML 2.0, then click **Create**.
5. In the **General Settings** step, enter a name and description for your application, for example *Talend Administration Center*, then click **Next**.
6. Fill in the **SAML Settings** :

Field	Value
Single sign on URL	*http://<host>:<port>/<application_name>/ssologin* Ex: *http://localhost:8080/org.talend.administrator/ssologin*
Audience URI(SP Entity ID)	*http://<host>:<port>/<application_name>/ssologin* Ex: *http://localhost:8080/org.talend.administrator/ssologin*
Name ID format	Select **Email Address** in the list.
Application username	Select **Email** in the list.

7. Once you have created your application, download the **Identity Provider metadata** from the **Sign On** tab of your application.



8. Click **Next** and **Finish**.

Define the user attributes of your application

Single-Sign On is only available for Talend Administration Center, but user information of the related applications can be centralized in Okta: Talend allows you to manage your application user roles and user project types, including roles of Talend Administration Center, Talend Data Preparation and Talend Data Stewardship users, outside of Talend Administration Center from Okta.

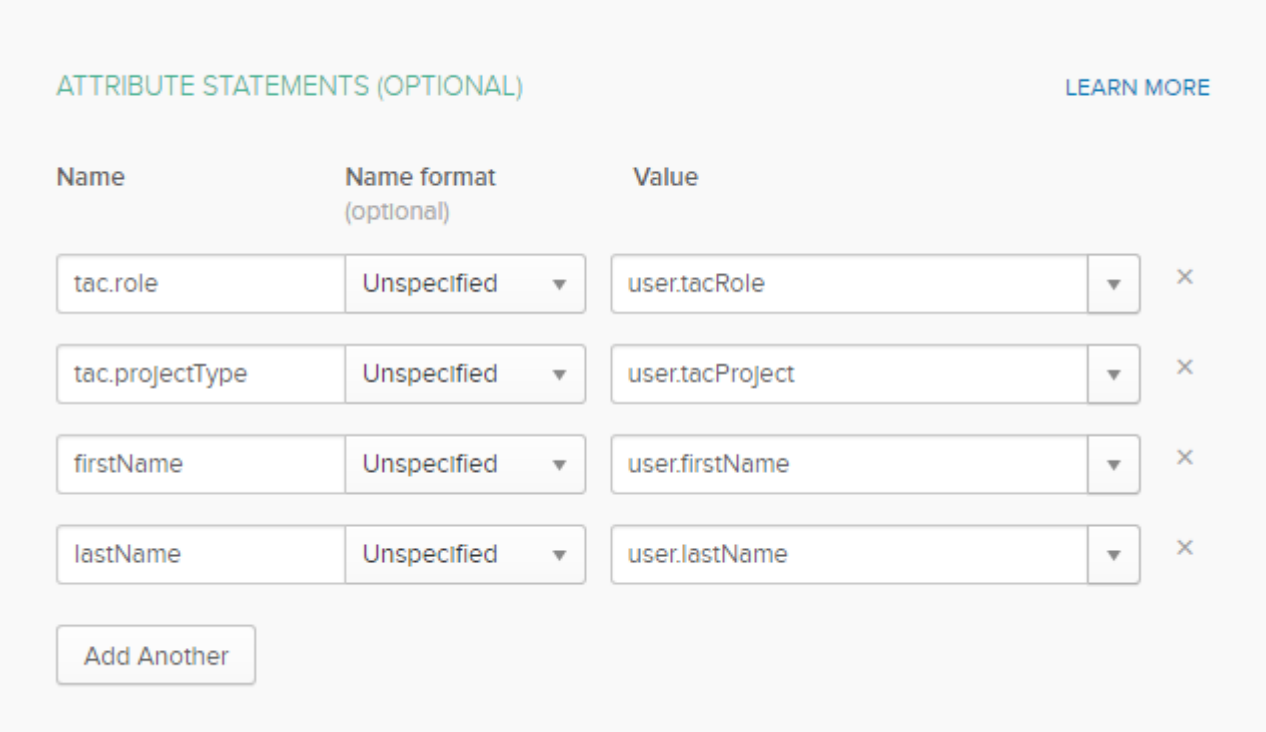
Note that once Single-Sign On is enabled, you will not be able to manage from *Talend Administration Center* all the user settings handled by the Identity Provider, such as user passwords, project types on which users are assigned or user roles.

1. Select **Directory** > **Profile Editor** from the top menu.
2. Open the user **Profile** corresponding to the Talend Administration Center application you have just created in Okta.
3. In the **Custom** tab, click **Add Attribute**.
4. Create the role attribute: In the **Add Attribute** window, enter the Display Name Attribute (*TACRole* for example), variable name (*tacRole* for example), and select **string array** in the **Data type** list, then click **Add Attribute**.
5. Create the project type attribute: In the **Add Attribute** window, enter the Display Name Attribute (*TACProjectType* for example), variable name (*tacProject* for example), select **string** in the **Data type** list, define a field length (between 1 and 10 characters for example) then click **Add Attribute**.

Add the user attributes to your application

1. Select your existing application and click **Edit** in the **SAML Settings** of the **General** tab.
2. In the **Attribute Statements** area, add four attributes **tac.role**, **tac.projectType**, **firstName** and **lastName**:

Talend Administration Center attribute name	SAML attribute name (Okta)	Value	Attribute value in user profile
Talend Administration Center Role attribute	**tac.role**	*user.tacRole*	Any string of your choice that will map the value entered in Talend Administration Center SSO Configuration Example: *tac_admin* (for a Talend Administration Center Administrator user) *tac_om* (for a Talend Administration Center Operation Manager user) *dp_dm* (for a Talend Data Preparation Dataset Manageer user)
Talend Administration Center Project attribute	**tac.projectType**	*user.tacProject*	Either, *DI* (Data Integration), *DQ* (Data Management), *MDM* (Master Data Management) or *NPA* (No Project Access)
Optional (if not set, the email address login will be used) - First Name	**firstName**	*user.firstName*	User first name
Optional (if not set, the email address login will be used) - Last Name	**lastName**	*user.lastName*	User last name

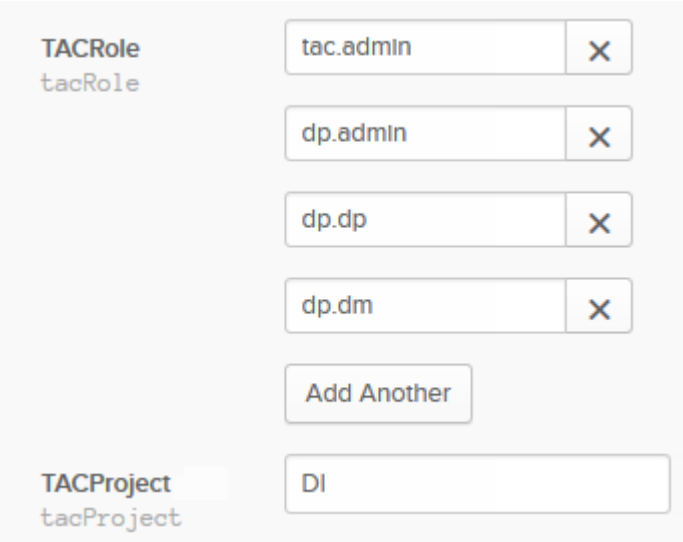


Define the user information and assign the user to the application

1. Select **Directory** > **People** from the top menu.
2. Select the user you want to edit then go to the **Profile** tab to edit this user.

3. Set the desired roles values (the same role and project type values will have to be used in the Talend Administration Center SSO configuration), and click **Add Another** to add several user roles.

   Do the same for the project type value ((Either, *DI* (Data Integration), *DQ* (Data Management), *MDM* (Master Data Management) or *NPA* (No Project Access)).



4. Open the **People** tab in a new browser tab and click **Assign to People**.

5. Enter the username(s) and email address(es) of the person(/people) you want to assign to the application.

   Once your application and users are set in Okta, you need to link the Identity Provider to Talend Administration Center in order to retrieve the user information you have defined.

In Talend Administration Center :

1. From the configuration page, expand the **SSO** node.

2. If SSO has not been enabled yet, select **true** in the **Use SSO Login** field.

3. Click **Launch Upload** in the **IDP metadata** field and upload the Identity Provider metadata file you have previously downloaded from the Identity Provider system.

4. In the **Service Provider Entity ID** field, enter the Entity ID of your Service Provider (available in the configuration of the IdP), *ssologin* for example.

5. Select your **Identity Provider System** in the corresponding list.

6. - If your provider is Okta: enter the corresponding Okta administrator **Organization URL**, as well as the **Okta App Embed link** which is the link used to sign into Talend Administration Center from a portal outside of Okta (can be found in Okta configuration).

   - If your provider is SiteMinder: enter the corresponding SiteMinder administrator **SiteMinder SSO Service URL**, *http://<host>/affwebservices/public/saml2sso?SPID=<SPEntityName>* for example.

7. Set the **Use Role Mapping** field to *true* to map the application project types and the user roles with those defined in the Identity Provider system.

   Once you have defined project types/roles at the Identity Provider side, you will not be able to edit them from Talend Administration Center.

   Fill in the role/project type fields with the corresponding SAML attributes previously set in the Identity Provider system.

Project type examples:

- MDM=*MDM*; DI = *DI*; DM=*DQ*; NPA=*NPA*

Role examples:

- **Talend Administration Center Roles** > Administrator = *tac_admin*; Operation Manager = *tac_om*
- **Talend Data Preparation Roles** > Administrator = *dp_admin*; Data Preparator = *dp_dp*
- **Talend Data Stewardship Roles** > Data Steward = *tds_ds*

The project types and roles set in the Identity Provider will override the roles set in *Talend Administration Center*.

Setting up a Talend Administration Center SSO from SiteMinder

Prerequisite: You have a SiteMinder administrator account and have installed and configured Web Agent and Web Agent OptionPack.

See below the main configuration steps to set up Single-Sign On for Talend Administration Center in SiteMinder.

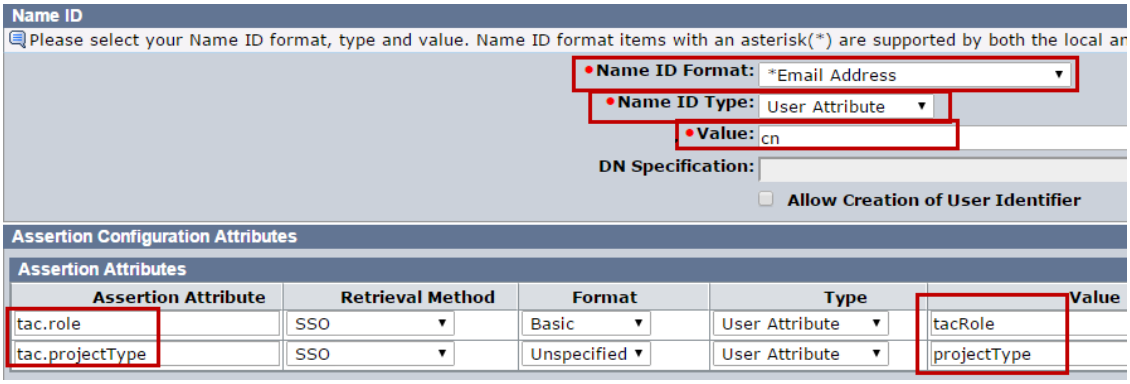
For more detailed information, see the article about SiteMinder configuration on Talend Help Center.

Configure the SAML2 Identity Provider in SiteMinder

1. Create a User Directory from the SiteMinder Administrative UI.

   In the **LDAP Settings** area, set the email address attribute as user search in the **LDAP user DN lookup** setting.

2. Protect the authentication URL to establish the user sessions as described in the SiteMinder documentation:
   - Select your Web agent (create and configure it as described in the SiteMinder documentation)
   - Select the User Directory created previously.
   - Select a **Basic** Authentication Scheme (see the SiteMinder documentation for more information)
   - Clear the **Persistent** check box in the **Session** section in order not to store session information.
3. Create a Signing certificate by importing a key/certificate pair (**Infrastructure** > **X509 Certificate Management** > **Trusted Certificates and Private Keys**)
4. Create a local Identity Provider Entity (**Federation** > **Partnership Federation** > **Entities**):
   - Select **Local** and **SAML2 IDP** in the **Entity Type** step.
   - Select the **Unspecified** and **Email Address** check boxes in the **Entity configuration** step.
5. Create a Partnership (**Federation** > **Partnership Federation** > **Partnerships**):
   - Select **SAML2 IDP** and **tac** in the **Configure Partnership** step.
   - Select **All Users in Directory** in the **Federation Users** step.
   - In the **Assertion Configuration** step, enter required information and add *tac.role* and *tac.projectType* that will match the custom LDAP User attributes (*tacRole* and *projectType* in this example). The values of these attributes will later be retrieved when configuring SSO in Talend Administration Center.



- In the **SSO and SLO** step, enter the URL of the web service to *redirect.jsp* in **Authentication URL**, select *urn:oasis:names:tc:SAML:2.0:classes:Password* in **Authentication Class**, select **HTTP-Redirect** and **HTTP-POST** bindings, enter the URL to the Talend Administration Center SSO Servlet (*http://<TACapplicationURL>/<TACapplicationName>/ssologin*) in the **Remote Assertion Consumer Service URLs** area. Then leave the other parameters as is and finish the creation process.

6. Activate the Partnership you created and export its metadata. You will need to upload the metadata later on the Talend Administration Center SSO configuration page.

7. On your LDAP server, test the SSO login to the Talend Administration Center application:

   - Create a LDAP user with the custom role and project type attributes you want (**tacRole**=*tac_admin,tac_viewer* and **projectType**=*DI* for example) and check that the user credentials bind is successful.

     Note that:

     - project type values can only be: *DI* (Data Integration), *DQ* (Data Quality), *MDM* (Master Data Management) or *NPA* (No Project Access).

     - if you want to add several roles for a user, roles should be separated with a comma.

   - Go to the Authentication URL previously defined (*http://<host>/affwebservices/public/saml2sso?SPID=<SPEntityName>*) and enter the *uid/userPassword* values to log in Talend Administration Center.

Once your application and users are set in SiteMinder and LDAP, you need to link the Identity Provider to Talend Administration Center in order to retrieve the user information you have defined.

Note that Single-Sign On is only available for Talend Administration Center, but user information of the related applications can be centralized in SiteMinder: Talend allows you to manage your application user roles and user project types, including roles of Talend Administration Center, Talend Data Preparation and Talend Data Stewardship users, outside of Talend Administration Center from the Identity Provider.

In Talend Administration Center :

1. From the configuration page, expand the **SSO** node.

2. If SSO has not been enabled yet, select **true** in the **Use SSO Login** field.

3. Click **Launch Upload** in the **IDP metadata** field and upload the Identity Provider metadata file you have previously downloaded from the Identity Provider system.

4. In the **Service Provider Entity ID** field, enter the Entity ID of your Service Provider (available in the configuration of the IdP), *ssologin* for example.

5. Select your **Identity Provider System** in the corresponding list.

6. • If your provider is Okta: enter the corresponding Okta administrator **Organization URL**, as well as the **Okta App Embed link** which is the link used to sign into Talend Administration Center from a portal outside of Okta (can be found in Okta configuration).

   • If your provider is SiteMinder: enter the corresponding SiteMinder administrator **SiteMinder SSO Service URL**, *http://<host>/affwebservices/public/saml2sso?SPID=<SPEntityName>* for example.

7. Set the **Use Role Mapping** field to *true* to map the application project types and the user roles with those defined in the Identity Provider system.

   Once you have defined project types/roles at the Identity Provider side, you will not be able to edit them from Talend Administration Center.

   Fill in the role/project type fields with the corresponding SAML attributes previously set in the Identity Provider system.

   Project type examples:

   • MDM=*MDM*; DI = *DI*; DM=*DQ*; NPA=*NPA*

   Role examples:

   • **Talend Administration Center Roles** > Administrator = *tac_admin*; Operation Manager = *tac_om*

   • **Talend Data Preparation Roles** > Administrator = *dp_admin*; Data Preparator = *dp_dp*

   • **Talend Data Stewardship Roles** > Data Steward = *tds_ds*

   The project types and roles set in the Identity Provider will override the roles set in *Talend Administration Center*.

Defining an emergency user for Talend Administration Center

In case your Identity Provider is temporarily unavailable and you need to connect to *Talend Administration Center*, you have the possibility to create a temporary emergency user.

1. Open the following file to edit it:

   <tomcat_path>WEB-INF\classes\configuration.properties

2. Uncomment the parameters `sso.emergency.username` and `sso.emergency.password`, edit the credentials of the emergency user if needed then save your changes.

3. Restart Tomcat.

4. Log into *Talend Administration Center* using the previously defined credentials. After logging out from the current session, this user account will be removed.

3.2.4.2. Setting up High Availability

To implement this High Availability architecture, you need to follow these procedures:

• *Installing Tomcat in cluster mode*

• *Duplicating Tomcat and the TAC web application*

Installing Tomcat in cluster mode

1. Install one Tomcat server as described in *Deploying Talend Administration Center on an application server*.
2. Before starting Tomcat and deploying *Talend Administration Center*, set it into *cluster mode*. To do so:
   - Edit the following file:

     /<ApplicationPath>/WEB-INF/classes/quartz.properties
   - Uncomment the following lines by removing the hash character preceding the command :

```
#org.quartz.scheduler.instanceName = MyClusteredScheduler
#org.quartz.scheduler.instanceId = AUTO
#org.quartz.jobStore.isClustered = true
#org.quartz.jobStore.clusterCheckinInterval = 20000
```

Duplicating Tomcat and the TAC web application

1. Duplicate this Tomcat instance as many times as needed. Make sure that all Tomcat instances use different port numbers.

   Make sure that all system clocks are synchronized (the clocks must be within a second of each other). For more information on time-sync services, please refer to the appropriate Microsoft documentation about SNTP, Windows Time Service tools and Network Clocks.
2. Duplicate the *org.talend.administrator* Web application to all Tomcat instances. Make sure that all Web application configurations are identical.
3. Launch one Tomcat instance following the commands given at the end of *Deploying Talend Administration Center on an application server*.
4. Launch the other instances of Tomcat following the same procedure.

Fail-over will occur when one of the multiple execution servers fails while in the midst of executing one or more tasks. When a server fails, the other servers of the cluster detect the condition and identify the tasks in the database that were in progress within the failed server. Any tasks marked for recovery will be taken over by another server.

Note that the ranking of servers to be used for load balancing is based on indicators, whose bounds (such as free disk space limits) and weight are defined in the file: `monitoring_client.properties` which is located in *<ApplicationPath>\WEB-INF\lib\org.talend.monitoring.client-A.B.C.jar*. These values can be edited according to your needs. For more information, see *Configuring the indicators which determine which server to be used for load balancing*.

You can also deploy *Talend Administration Center* on a JBoss application server (instead of a Tomcat). So, you can follow the same above instructions for Jboss. For more information on how to deploy the Web application on JBoss, see *Deploying Talend Administration Center on JBoss*.

One known minor issue related to the DST change might prevent the failover to operate properly. However as a simple workaround, simply restart Tomcat after the time change. This should have no impact on executions.

3.2.4.3. Migrating database X to database Y

If you want to migrate from one database to another, for example from H2 to MySQL, you need to use the MetaServlet command called `migrateDatabase`.

The MetaServlet application is located in *<TomcatPath>/webapps/<TalendAdministrationCenter>/WEB-INF/classes* folder.

Note that, to display the help of this command (with related parameters), you need to enter the following in the MetaServlet application:

```
./MetaServletCaller.sh --tac-url=<yourApplicationURL> -h migrateDatabase
```

For more information on the MetaServlet application, see the *Talend Administration Center User Guide*.

See below an example of migration between H2 and PostgreSQL databases.

Please note that to be able to use this command, you need to put it on one single line first.

```
./MetaServletCaller.sh --tac-url http://localhost:8080/org.talend.administrator --
json-params={"actionName": "migrateDatabase", "dbConfigPassword": "admin", "mode":
 "synchronous", "sourceUrl": "jdbc:h2:C:/Talend/5.6.1/tac/apache-tomcat/webapps/
org.talend.administrator/WEB-INF/database/talend_administrator", "sourceUser": "admin",
 "sourcePasswd": "admin", "targetPasswd": "root", "targetUrl": "jdbc:postgresql://
localhost:5432/postgres", "targetUser": "postgres"}
```

3.2.4.4. Disabling SSL3 in Tomcat

In order to avoid POODLE vulnerability which allows attackers to downgrade SSL/TLS protocol to version SSL v3, and then break the cryptographic security, you might want to disable SSL v3 on the Tomcat server. For more information on how to do this, read the procedure on the Apache website.

3.2.4.5. Managing the database parameters

The configuration parameters are stored in the database, except for the parameters related to the Talend Administration Center database that are stored in the following file:

<ApplicationPath>/WEB-INF/classes/configuration.properties

The database-related passwords are encrypted at start up, when this file is parsed and loaded in the database.

Change the encrypted default account password

1. Open the *configuration.properties* file to edit it.

```
#################################################################
# DEFAULT ADMIN ACCOUNT PARAMETERS
#################################################################
install.defaultaccount.login=admin@company.com
install.defaultaccount.password=oV0/gYJwoWqvorcePQ+Mrg==,Encrypt
install.defaultaccount.firstname=admin
install.defaultaccount.lastname=admin
install.defaultaccount.type=DI
```

2. Note that the encrypted password is followed by: *",Encrypt"*

   Remove all that is after the = sign, including *",Encrypt"*, and type in the new password of the default account.

3. Save your changes and close the file. At next startup, the password will be encrypted in the database and the file will be updated with this encrypted password.

Change the default password used to configure the database

After the first connection, it is strongly recommended not to use the default user account to access the application for security reasons. You can either change the default credentials of this account (*security@company.com*/*admin*) or create another administrator user and remove the default account.

If you want to change the admin default password that allows you to change the database configuration, do the following:

1. Scroll down the *configuration.properties* file until you find the `database.config.password` parameter.
2. Change the *admin* default password to a more individual and secure password.

3.2.4.6. Managing the connection pool via Tomcat

By default, a third-party application (c3p0) has been embedded into the configuration file of *Talend Administration Center*, to manage the connection pool.

However if you want Tomcat to manage directly the connection pool, perform the following operations:

In the Web application installation directory, proceed as follows:

1. In the *<ApplicationPath>/WEB-INF/classes* folder, change the default setting of the *configuration.properties* file to:

   ```
   database.useContext=True
   ```

2. In the *WEB-INF* folder, edit the *web.xml* file and add the following piece of code before the closing tag `</web-app>`:

   ```
   <resource-ref>

        <description>Our Datasource</description>
        <res-ref-name>jdbc/ADMINISTRATOR_CONNECTION</res-ref-name>
        <res-type>javax.sql.DataSource</res-type>
        <res-auth>Container</res-auth>

   </resource-ref>
   ```

3. In the file *<ApplicationPath>/META-INF/context.xml*, configure the parameters of connection to the database by modifying the following elements:

Element name	Value	Note
url	`jdbc:mysql://{ip_address}:3306/{db_name}`	For MySQL, where *ip_address* corresponds to the database IP address and *db_name* corresponds to its name.
	`jdbc:oracle:thin:@{ip_address}:1521:{db_name}`	For Oracle, where *ip_address* corresponds to the database IP address and *db_name* corresponds to its name.
	`jdbc:jtds:sqlserver://{ip_address}:1433/{db_name}`	For SQL Server, where *ip_address* corresponds to the database IP address and *db_name* corresponds to its name.
	`jdbc:h2:file:{dir_path/}<db_name>;MVCC=TRUE;AUTO_SERVER=TRUE;LOCK_TIMEOUT=15000`	For H2, where *dir_path* corresponds to the database path and *db_name* corresponds to its name.

Element name	Value	Note
username	The username used to log in your database, talend_admin by default.	
password	The password used to log in your database, talend_admin by default.	
driverClassName	`org.gjt.mm.mysql.Driver`	For MySQL.
	`oracle.jdbc.driver.OracleDriver`	For Oracle.
	`net.sourceforge.jtds.jdbc.Driver`	For SQL Server.
	`org.h2.Driver`	For H2.

4. Copy the relevant *.jar* file corresponding to the database in which your data is stored in *<TomcatPath>/lib/*.

You can also deploy *Talend Administration Center* on a JBoss application server (instead of a Tomcat). So, you can follow the same above instructions for JBoss. For more information on how to deploy the Web application on JBoss, see *Deploying Talend Administration Center on JBoss*.

3.2.4.7. Configuring the indicators which determine which server to be used for load balancing

If you want to edit and overwrite the default configuration used to determine which server to be used for load balancing in cluster mode, do the following.

1. Open the *monitoring_client.properties* file which is located in the following *.jar* file:

   <ApplicationPath>/WEB-INF/lib/org.talend.monitoring.client-x.y.z.rabcd.jar

2. The weight values defined in this file will impact the server to be used to process data. Edit the values according to your needs and save your modifications.

3. Copy the edited file in the following directory to overwrite the one located in the *.jar* file:

   <ApplicationPath>/WEB-INF/classes

3.2.4.8. Customizing the Talend Administration Center Menu tree view

You also have the possibility to customize the **Menu** tree view of the *Talend Administration Center* Web application by adding dynamic links to the website of your choice.

To set up dynamic links:

1. Open the following file:

   <ApplicationPath>/WEB-INF/classes/configuration.properties

2. At the end of the file, enter the dynamic link of interest using the given syntax:

   `dynamiclink.<key>=<label>#<url>#<order>`.

   For example, you can create the link to http://www.talend.com by entering

   `dynamiclink.talendcom=Talend#http://www.talend.com#8`

   or the link to http://www.talendforge.org by entering

`dynamiclink.talendforge=Talendforge#http://www.talendforge.org#9`.

In this syntax, `<key>` indicates the technical key of this link configured, `<label>` is the link name displayed on the **Menu** tree view, `<url>` is the website address you need to link to and `<order>` specifies the position of this link on the **Menu** tree view.

```
dynamiclink.talendcom=Talend#http://www.talend.com#8
dynamiclink.talendforge=Talendforge#http://www.talendforge.org#9
```

For further information about the order numbers used by *Talend Administration Center* to arrange the **Menu** items, check the *menuentries.properties* file provided in the same *classes* folder.

3. Save the *configuration.properties* file edited.

For more information on how these links are displayed in the Menu tree view of the *Talend Administration Center* Web application, see the *Talend Administration Center User Guide*.

3.2.4.9. Configuring Talend Administration Center login delay

Setting up a login delay allow you to improve the security of your Web application by slowing brute force attacks.

- In the *configuration* table of the *Talend Administration Center* database, change the value of the `useLoginDelay` parameter to *true*.

Failed login attempts will now generate a time delay which increases exponentially with each failed attempt.

3.2.4.10. Configuring LDAP(S) for Talend Administration Center

To configure LDAP(S) for *Talend Administration Center*, proceed as follows:

Generate a key

1. Create a folder where you want to store your Keystore.
2. Open a command prompt.
3. Using the `cd` command, go to the folder you created.
4. Enter the following command:

   ```
   <JAVA_HOME>/bin/keytool -genkey -keystore <myKeystoreName> -keyalg RSA
   ```

   Replace <*JAVA_HOME*> with the path to the folder where Java is installed and <*myKeystoreName*> with the name of your Keystore.
5. Enter the password you want to create for your Keystore twice. Then, if needed, enter other optional information, such as your name or the name of your organization.
6. Enter `yes` to confirm the information you provided.
7. Enter the password you have previously defined.

Configure LDAP(S) for Talend Administration Center

To set the new Keystore location, edit the JAVA_OPTS environment variable.

- To edit the JAVA_OPTS environment variable, add the following lines

```
-Djavax.net.ssl.keyStore=/<myDirectory>/<myKeystore>
-Djavax.net.ssl.keyStorePassword=<myPassword>
```

to your JAVA_OPTS environment variable, where <*myDirectory*> is the installation directory of your Keystore, <*myKeystore*> is the name of your Keystore and <*myPassword*> is the password you have previously defined for your Keystore.

For more information on how to enable LDAP(S) in Talend Administration Center, see the *Talend Administration Center User Guide*.

3.3. Installing and configuring Talend Identity and Access Management

This section describes the installation and configuration of *Talend Identity and Access Management* that allow you to manage the user access to *Talend Data Preparation* and *Talend Data Stewardship*.

The recommended installation method for *Talend Identity and Access Management* is the automatic installation with *Talend Installer*.

To manually install *Talend Identity and Access Management*, proceed as follows:

- *Installing Talend Identity and Access Management*
- *Changing Talend Identity and Access Management database*
- *Linking Talend Identity and Access Management with Talend Data Preparation*
- *Linking Talend Identity and Access Management with Talend Data Stewardship*

3.3.1. Installing Talend Identity and Access Management

If you want to install *Talend Identity and Access Management* manually, proceed as follows:

1. Copy and extract the *iam-A.B.C-distribution.zip* archive file in the directory of your choice.
2. Go to *iam-A.B.C/apache-tomcat-x.x.xx/bin*.
3. Add the execution rights to the executable files by typing `chmod 755 ./bin/*.sh`.
4. Start *Talend Identity and Access Management* by executing the *startup.sh* file.

Now that *Talend Identity and Access Management* is installed, it is strongly recommended not to use the default Apache Syncope user account to access the application for security reasons. You can change the default credentials of this account (*admin*/*password*) by editing the *adminPassword* parameter in the iam-A.B.C/apache-tomcat-x.x.xx/webapps/syncope/WEB-INF/classes/security.properties file. For more information, see https://syncope.apache.org/docs/reference-guide.html#set-admin-credentials.

You can now access the *Talend Identity and Access Management* Apache Syncope Console with the following URL: http://localhost:9080/syncope-console/.

You can now access the list of registered OIDC clients with the following URL: http://localhost:9080/oidc/console/clients.

3.3.2. Changing Talend Identity and Access Management database

As the embedded H2 database is not recommended for production environments, it is advised to change *Talend Identity and Access Management* database.

1. Stop *Talend Identity and Access Management* if it has been already started.
2. Place the JDBC driver jar file corresponding to the database you want to use in the *iam-A.B.C/apache-tomcat-x.x.xx/lib* folder and make sure that it has the same permissions as the other jar files.

   For more information on the supported databases, see *Compatible Databases*.
3. Update the *provisioning.properties* and *domains/Master.properties* files as described in Apache Syncope documentation.
4. Edit the *iam-A.B.C/apache-tomcat-x.x.xx/conf/iam.properties* file and update the following parameters:

Parameter	Description
`idp.db.url`	Database JDBC URL.
`idp.db.driverClassName`	Fully qualified driver class name, *com.mysql.jdbc.Driver* for example.
`idp.db.username`	User name used to connect to the database.
`idp.db.password`	Password used to connect to the database. The password is encrypted at first launch.
`idp.db.platform`	OpenJPA 2.4.2 platform name without the package name, *MariaDBDictionary* for example. For more information, see https://openjpa.apache.org/builds/2.4.2/apidocs/org/apache/openjpa/jdbc/sql/DBDictionary.html.
`oidc.db.url`	Database JDBC URL.
`oidc.db.driverClassName`	Fully qualified driver class name, *com.mysql.jdbc.Driver* for example.
`oidc.db.username`	User name used to connect to the database.
`oidc.db.password`	Password used to connect to the database. The password is encrypted at first launch.
`oidc.db.databasePlatform`	Hibernate 5 platform name. For more information, see https://docs.jboss.org/hibernate/orm/5.2/javadocs/.
`oidc.db.dialect`	Hibernate 5 dialect for the database. For more information, see https://docs.jboss.org/hibernate/orm/5.2/javadocs/.

5. Start *Talend Identity and Access Management* by executing the *startup.sh* file.

3.3.3. Linking Talend Identity and Access Management with Talend Data Preparation

If you have installed *Talend Identity and Access Management* manually, you need to create an OIDC client in order to link *Talend Identity and Access Management* with *Talend Data Preparation*. Note that this operation is automatically done if you install *Talend Identity and Access Management* using *Talend Installer*.

1. Stop *Talend Identity and Access Management* and *Talend Data Preparation* if they have been already started.
2. Go to *iam-A.B.C/apache-tomcat-x.x.xx/clients*.
3. Create a *tdp-client.json* file.
4. Paste the following content:

```
{
"post_logout_redirect_uris" : [ "http://my-machine:9999", "http://localhost:9999",
 "http://127.0.0.1:9999" ],
"grant_types" : [ "authorization_code", "refresh_token", "password" ],
"scope" : "openid refreshToken",
"client_secret" : "+1/7vegEOVHeQD9JKmtz8I9s4tgVuRMqC2ja7efFHro=",
"redirect_uris" : [ "http://my-machine:9999/signIn", "http://localhost:9999/signIn",
 "http://127.0.0.1:9999/signIn" ],
"client_name" : "TDP DataPrep",
"client_id" : "64xIVPxviKWSog"
}
```

   Adapt the parameters to your needs:

Parameter	Description
`post_logout_redirect_uris`	URI to which the user is redirected after logging out. If *Talend Identity and Access Management* and *Talend Data Preparation* are located on the same machine, be sure to put the name of the machine in addition to *localhost* and *127.0.01* as shown in the example.
`grant_types`	The OAuth specification has different grant types. These authorizations allow the client application to obtain an access token. This token represents the client permission to access user data. Set the `grant_types` to the values shown in the example.
`scope`	OpenID defined scopes. Set it to the value shown in the example.
`client_secret`	Client password. The client password is encrypted at first launch.
`redirect_uris`	URI to which the user is redirected after logging in. The */signIn* part of the URI is mandatory. If *Talend Identity and Access Management* and *Talend Data Preparation* are located on the same machine, be sure to put the name of the machine in addition to *localhost* and *127.0.01* as shown in the example.
`client_name`	Name of the OIDC client. The *TDP* part of the client name (with the trailing space) is mandatory.
`client_id`	Identifier of the OIDC client.

5. Start *Talend Identity and Access Management* and *Talend Data Preparation*.

3.3.4. Linking Talend Identity and Access Management with Talend Data Stewardship

If you have installed *Talend Identity and Access Management* manually, you need to create an OIDC client in order to link *Talend Identity and Access Management* with *Talend Data Stewardship*. Note that this operation is automatically done if you install *Talend Identity and Access Management* using *Talend Installer*.

1. Stop *Talend Identity and Access Management* and *Talend Data Stewardship* if they have been already started.

2. Go to *iam-A.B.C/apache-tomcat-x.x.xx/clients*.
3. Create a *tds-client.json* file.
4. Paste the following content:

```
{
"post_logout_redirect_uris" : [ "http://my-machine:19999/", "http://
localhost:19999/", "http://127.0.0.1:19999/" ],
"grant_types" : [ "password", "authorization_code", "refresh_token" ],
"scope" : "openid refreshToken",
"client_secret" : "cB/gNxe2SXR3SPDbhshZXzErZoxVy8yUcs/f6K39rsg=",
"redirect_uris" : [ "http://my-machine:19999/login", "http://localhost:19999/login",
 "http://127.0.0.1:19999/login" ],
"client_name" : "TDS OIDC Gateway",
"client_id" : "tl6K6ac7tSE-LQ"
}
```

Adapt the parameters to your needs:

Parameter	Description
`post_logout_redirect_uris`	URI to which the user is redirected after logging out. If *Talend Identity and Access Management* and *Talend Data Stewardship* are located on the same machine, be sure to put the name of the machine in addition to *localhost* and *127.0.01* as shown in the example.
`grant_types`	The OAuth specification has different grant types. These authorizations allow the client application to obtain an access token. This token represents the client permission to access user data. Set the `grant_types` to the values shown in the example.
`scope`	OpenID defined scopes. Set it to the value shown in the example.
`client_secret`	Client password. The client password is encrypted at first launch.
`redirect_uris`	URI to which the user is redirected after logging in. The */login* part of the URI is mandatory. If *Talend Identity and Access Management* and *Talend Data Stewardship* are located on the same machine, be sure to put the name of the machine in addition to *localhost* and *127.0.01* as shown in the example.
`client_name`	Name of the OIDC client. The *TDS* part of the client name (with the trailing space) is mandatory.
`client_id`	Identifier of the OIDC client.

5. Start *Talend Identity and Access Management* and *Talend Data Stewardship*.

3.4. Talend logging modules

This section describes the installation and configuration of the logging modules (*Talend* Log Server based on Elasticsearch and Kibana) that allow you to display in Talend Administration Center the **Logging** page which groups and display the output logs filtered by categories and event types (Data Integration, ESB or MDM events). For more information on how to display the logs in Talend Administration Center, see the *Talend Administration Center User Guide*.

The recommended installation method for the *Talend* logging modules is the automatic installation with *Talend Installer*.

To manually install the *Talend* logging modules, proceed as follows:

- *Installing the Talend logging modules*
- *Defining the name of the log cluster*
- *Configuring Talend logging modules with an external Elastic stack*

3.4.1. Installing the Talend logging modules

If you want to install the logging modules manually, you need to install the Talend Log Server which includes Kibana.

1. Copy and extract the *Talend-LogServer-VA.B.C.zip* archive file in the directory of your choice.
2. To start Talend Log Server, launch the *start_logserver.sh* executable file.

You can now access Talend Log Server with the following URL: http://localhost:5601/app/kibana#/dashboard/Talend-Dashboard

The ESB event logs that are displayed in the **Logging** page of *Talend Administration Center* are collected thanks to the Event Logging features that can be started from the Runtime container. For more information, see the *Talend ESB Container Administration Guide*.

3.4.2. Defining the name of the log cluster

By default, all logs are centralized in a cluster group called *talend-log-central*. However, if you want to use a different cluster name, for example if you do not want to mix logs from different environments such as Quality Assurance logs and production logs, do the following.

1. Open the *logstash-talend.conf* file located in the Talend Log server installation directory.
2. Edit the name of the cluster in the `output` part:

```
output {
  elasticsearch {
    cluster => "talend-log-central"
  }
```

3.4.3. Configuring Talend logging modules with an external Elastic stack

The following sections explain how to configure your external Elastic for Data Integration, ESB or MDM.

3.4.3.1. Configuring Logstash and Elasticsearch in an external Elastic stack

If you want to use your external ELK stack to collect and classify logs from Data Integration, ESB or MDM, you need to perform the following configuration on your Logstash and Elasticsearch:

1. Create/edit your Logstash configuration file as follows:
   - In the `input` section, you add the configuration ports for **Talend** components.
   - In the `output` section, you configure Elasticsearch as output.

   The options `user` and `password` should match your Elasticsearch credentials.

   If you are not using a true Certificate Authority (CA), set the option `ssl_certificate_verification` to *false*.

```
input {
  log4j {
    port => 8050
    type => "TAC"
  }
  log4j {
    port => 8052
    type => "Components"
  }
  log4j {
    port => 8053
    type => "MDM-Server"
  }
  log4j {
    port => 8054
    type => "MDM-Match"
  }
  log4j {
    port => 8055
    type => "JobServer"
  }
}

output {
  elasticsearch {
      hosts => ["https://ELK_HOST:9200"]
      ssl => true
      ssl_certificate_verification => true
      user => "<es_user>"
      password => "<es_password>"
  }
}
```

2. Before starting Elasticsearch, open a shell/prompt and configure the `ES_JAVA_OPTS` environment variable as follows:
   - `export ES_JAVA_OPTS="-Dmapper.allow_dots_in_name=true"`.
3. Start Elasticsearch.
4. Restart Logstash with the `.conf` configuration file:
   - `${LOGSTASH_HOME}/bin/logstash -f ${LOGSTASH_CONFIG_FILE}`.

3.4.3.2. Importing Talend Kibana dashboard templates: prerequisites

If you wish to import Talend dashboards in Kibana, you must have:

- Imported the Logstash templates in Elasticsearch using the following commands:

In the commands below, you must replace *ELK_HOST* by the Elasticsearch host. The option `es_user` corresponds to the Elasticsearch user.

- If you are using a true Certificate Authority (CA):

```
curl -u es_user -s -XPUT https://ELK_HOST:9200/_template/template_logstash -d
 @template_logstash.json
curl -u es_user -s -XPUT https://ELK_HOST:9200/.kibana/index-pattern/logstash-*  -d
 @dashboards/index-pattern/logstash.json
```

```
curl -u es_user -s -XPUT https://ELK_HOST:9200/_template/template_esb -d
 @template_talendesb.json
curl -u es_user -s -XPUT https://ELK_HOST:9200/.kibana/index-pattern/talendesb-*  -d
 @dashboards/index-pattern/talendesb.json
```

- If you are using self-signed certificates add --insecure option:

```
curl --insecure -u es_user -s -XPUT https://ELK_HOST:9200/_template/
template_logstash -d @template_logstash.json
curl --insecure -u es_user -s -XPUT https://ELK_HOST:9200/.kibana/index-pattern/
logstash-*  -d @dashboards/index-pattern/logstash.json
```

```
curl --insecure -u es_user -s -XPUT https://ELK_HOST:9200/_template/template_esb -d
 @template_talendesb.json
curl --insecure -u es_user -s -XPUT https://ELK_HOST:9200/.kibana/index-pattern/
talendesb-*  -d @dashboards/index-pattern/talendesb.json
```

- Created a *logstash-** index pattern:

```
curl -u es_user -s -XPUT https://ELK_HOST:9200/.kibana/config/4.6.1 -d
 "{\"defaultIndex\" : \"logstash-*\"}"
```

- Created a *talendesb-** index pattern:

```
curl -u es_user -s -XPUT https://ELK_HOST:9200/.kibana/config/4.6.1 -d
 "{\"defaultIndex\" : \"talendesb-*\"}"
```

3.4.3.3. Importing Talend Kibana dashboard templates in an external Elastic stack

Talend provides the following Kibana dashboard templates as part of the open-source Elastic stack shipped with the Talend Log Server. You can also import these dashboard templates in Kibana.

1. Start Kibana.
2. Go to **Objects**, under the **Settings** tab.
3. Within each sub-tab, import the templates using the **Import/Export** buttons:

Kibana UI tab	Location of file(s) to upload
Dashboards	Under *Talend-LogServer/dashboards/dashboard*: • `default.json`
Searches	Under *Talend-LogServer/dashboards/search*: • `TABLE.json`
Visualizations	Under *Talend-LogServer/dashboards/visualization*: • `FILTER-BY-SEVERITY.json` • `FILTER-BY-SOURCE.json` • `GROUP-BY-SEVERITY.json`

Kibana UI tab	Location of file(s) to upload
	• GROUP-BY-SOURCE.json • Middleware-Navigation.json • TIMELINE.json

Kibana UI tab	Location of file(s) to upload
Dashboards	Under *Talend-LogServer/dashboards/dashboard*: • ESB-Events.json • ESB-Locator-Endpoints.json • ESB-SAM.json
Searches	Under *Talend-LogServer/dashboards/search*: • All-Events-ESB-Locator-Endpoints.json • All-Evetns-ESB-SAM.json • DOCUMENTS-ESB-Events.json
Visualizations	Under *Talend-LogServer/dashboards/visualization*: • DOCUMENT-TYPES-CHART-ESB-Events.json • DOCUMENT-TYPES-TABLE-ESB-Events.json • ENDPOINT-AVAILABILITY-ESB-Locator-Endpoints.json • ENDPOINTS-PER-SERVICES-ESB-Locator-Endpoints.json • HISTOGRAM-ESB-Events.json • OFFLINE-ENDPOINTS-ESB-Locator-Endpoints.json • OFFLINE-SERVICES-ESB-Locator-Endpoints.json • REMOTE-CONSUMER-ESB-SAM.json • SERVICE-AVAILABILITY-ESB-Locator-Endpoints.json • SERVICE-REQUESTS-ESB-SAM.json • SERVICE-RESPONSES-ESB-SAM.json • TOP-10-ENDPOINTS-ESB-Locator-Endpoints.json • TOP-SERVICES-ESB-SAM.json

3.5. Installing your Talend Studio

To install and configure your *Talend Studio*, follow these procedures:

- *Installing and launching your Talend Studio*
- *Setting up a Talend Administration Center connection in Talend Studio*
- *Configuring Nexus in Talend Studio*
- *Installing external modules*

For more information on *Talend Studio*, see *Talend Studio* and the *Talend Studio User Guide*.

3.5.1. Installing and launching your Talend Studio

To install and launch your *Talend Studio*, follow these procedures:

- *Installing your Talend Studio*
- *Editing the memory and JVM settings*
- *Launching your Talend Studio*

For more information on how to use the Talend Studio, see the *Talend Studio User Guide*.

3.5.1.1. Installing your Talend Studio

Unzip the archive

1. Copy the archive *Talend-Studio-YYYYYYYY_YYYY-VA.B.C.zip* to a directory of your choice.
2. Unzip it.
3. Create a file (without extension) named *license* containing your license key (found in your email), and paste the file at the root of the extracted directory.

3.5.1.2. Editing the memory and JVM settings

To gain in performance at runtime and when launching the Studio, proceed as follows: you can edit the memory settings in the *.ini*.

1. Edit the *Talend-Studio-linux-gtk-x86_64.ini* file.
2. Edit the memory attributes. For example:

   ```
   -vmargs -Xms40m -Xmx500m -XX:MaxMetaspaceSize=256m
   ```

   For more details, see http://www.oracle.com/technetwork/java/hotspotfaq-138619.html.

3.5.1.3. Launching your Talend Studio

1. Double-click the executable to launch your *Talend Studio*.

   Execute the following file corresponding to your operating system:

   - *Talend-Studio-linux-gtk-x86_64* if you are on 64-bit Linux;
   - *Talend-Studio-linux-gtk-ppc* if you are on an operating system that works on the PowerPC architecture;
   - *Talend-Studio-linux-gtk-x86.sh*, which is a script that runs the Linux command and adds a parameter to avoid some graphical issues.

   Note that to use the script file, you should add the execution rights to it using the following command:

```
$ chmod +x Talend-Studio-linux-gtk-x86.sh
```

2. In the **[License setup]** dialog box that appears:

   - paste your license key in the blank area, or
   - click **Browse License...** to browse and select your license file and then click **OK**, or
   - click **Import license...** if you have already set your license and project in Talend Administration Center and want to retrieve this license.

   For more information on how to launch the Studio and set up a connection to Talend Administration Center, refer to the *Getting started* chapter in the relevant *Studio User Guide*.

3.5.2. Setting up a Talend Administration Center connection in Talend Studio

To set up a connection to *Talend Administration Center* from *Talend Studio*, do the following:

1. Launch *Talend Studio*.
2. In the *Talend Studio* login window, click the **[...]** button to define a new connection.
3. In the **[Connections]** window that opens, click the **[+]** button to create a new connection.
4. Set the **Repository** type as *Remote* and enter a **Name** and **Description** for the connection, the **E-mail** and **Password** for the user you created in *Talend Administration Center*, and the URL for *Talend Administration Center* (for example, *http://localhost:8080/org.talend.administrator* but, depending on your configuration, you may have to replace <localhost> with the server IP address) in the **Web-app Url** field.

   Be careful not to use an existing local workspace. If needed, you can create another folder in the Talend Studio alongside the default workspace folder.

5. Click **OK**.

   You can now select the newly created connection in the *Talend Studio* login window to connect to a collaborative project.

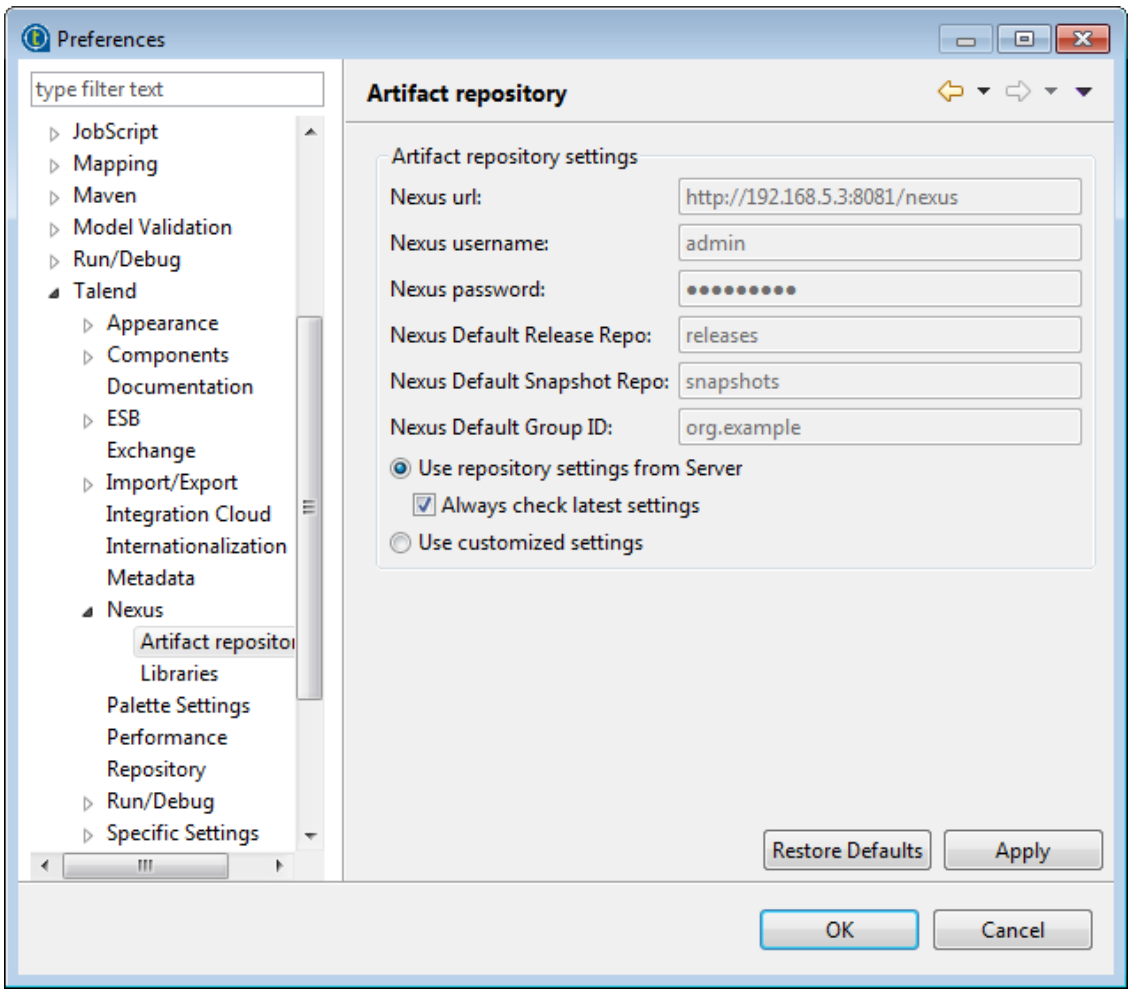
3.5.3. Configuring Nexus in Talend Studio

In the Studio, you can now configure the access to your Nexus Artifact Repository in its **Preferences** and publish your Services, Routes and Jobs into the two repositories according to your needs. For more information, see the *Talend Studio User Guide*.

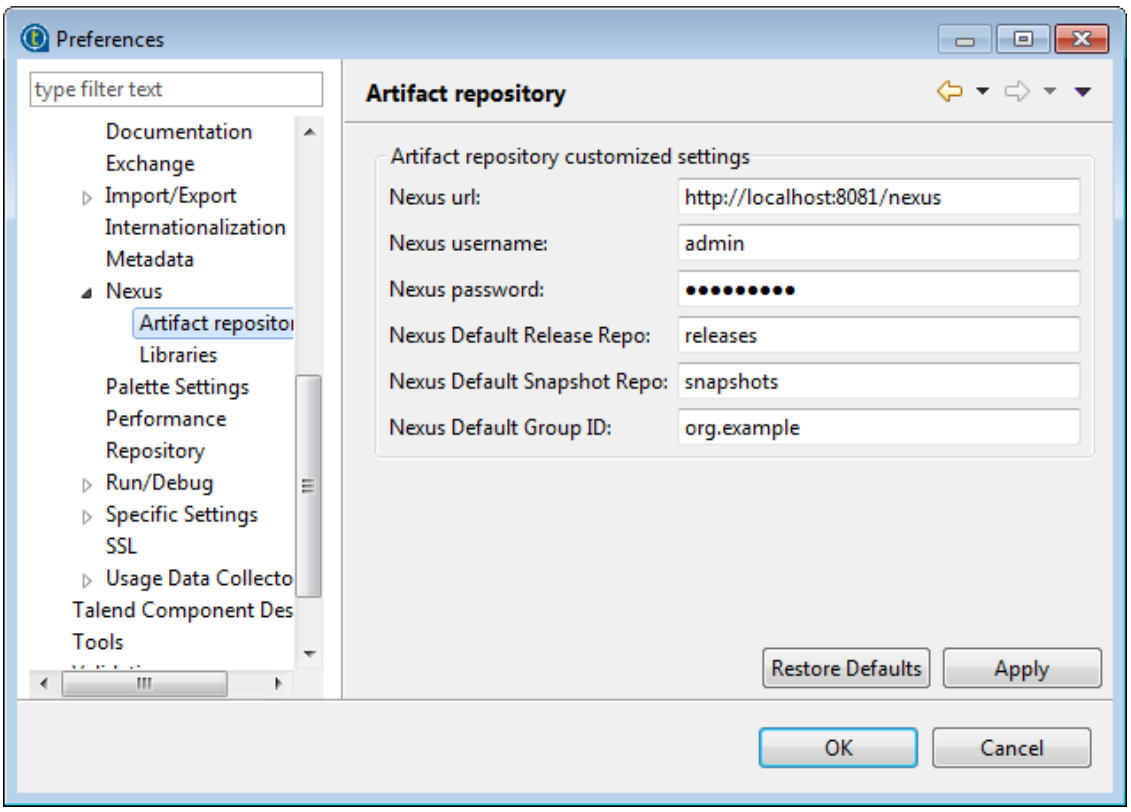
Once you have installed and launched the Nexus Artifact Repository, open your Studio and do the following:

1. Click **Window** > **Preferences** from the menu bar to open the **[Preferences]** dialog box.
2. In the tree view, expand the **Talend** > **Nexus** nodes and select **Artifact repository**.

   - If your *Talend Studio* is connected with the *Talend Administration Center*, all the Nexus artifact repository settings are automatically retrieved from the *Talend Administration Center*. You can choose to use the retrieved settings to publish your Services, Routes and Jobs, or configure your own artifact repositories.



- If your Studio is working on a local connection, all the fields are pre-filled with the locally-stored default settings.



3. When connected with the *Talend Administration Center*, by default, the Studio checks the latest artifact repository settings each time it interacts with the artifact server. To disable this, if the artifact repository settings are not subject to frequent changes or if you have a poor internet connection, for example, clear the **Always check latest settings** check box.

4. When connected with the *Talend Administration Center*, if you want to configure your own artifact repositories, select the **Use customized settings** option.

5. Modify the artifact repository settings according to your needs.

   - **Nexus url**: Type in the location URL of your repository.
   - **Nexus username**: Type in the username to connect to your repository.
   - **Nexus password**: Type in the password to connect to your repository.
   - **Nexus Default Release Repo**: Type in the name of the repository into which to publish the Release version of your artifact items by default.
   - **Nexus Default Snapshot Repo**: Type in the name of the repository into which to publish the Snapshot version of your artifact items by default.
   - **Nexus Default Group ID**: Type in the name of the group in which to publish your artifact items by default.

6. Click **Apply** to apply your changes and **OK** to close the wizard.

Now, you will be able to publish your Services, Routes and Jobs onto your Artifact repository. For more information on how to publish Services, Routes and Jobs, see the *Talend Studio User Guide*.

3.5.4. Installing external modules

Talend Studio requires specific third-party Java libraries or database drivers (*.jar* files) to be installed to connect to sources and targets. Those libraries or drivers, known as external modules, can be required by some of *Talend* components or by some connection wizards or by both. Due to license restrictions, *Talend* may not be able to ship certain external modules within *Talend Studio*.

3.5.4.1. Identify required external modules

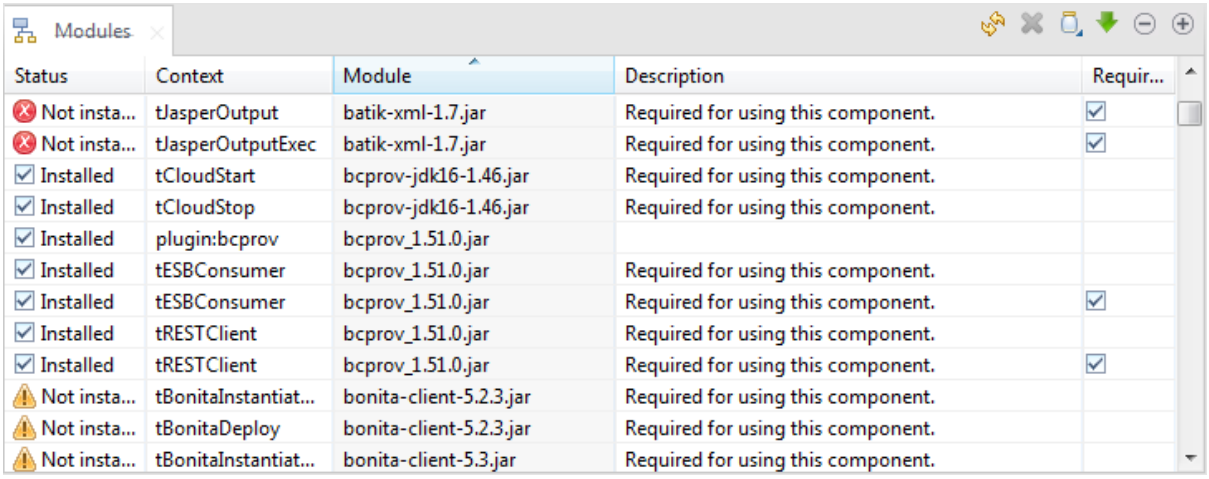
When you launch *Talend Studio* or select **Help** > **Install Additional Packages** in the *Talend Studio* menu, if any external modules are found missing for any features, the **[Additional Talend packages]** wizard opens, showing the **Optional** and **Required third-party libraries** check boxes. Make sure these check boxes are selected and click **Finish** to open the **[Download external modules]** dialog box, which lists all the available external modules, displays the license terms under which the external modules are provided, and lets you install all the modules at a single click. For more information, see *Install external modules*.

On your design workspace, if a component requires the installation of external modules before it can work properly, a red error indicator appears on the component. With your mouse pointer over the error indicator, you can see a tooltip message showing which external modules are required for that component to work.

When you open the **Basic settings** or **Advanced settings** view of a component for which one or more external modules are required, you will see a piece of highlighted information about external modules, followed by an **Install** button. Clicking the **Install** button opens a wizard that will show you the external modules to be installed.

The **Modules** view lists all the modules required to use the components embedded in the Studio, including those Java libraries and drivers that you must install to get the relevant components or Metadata connection working.

If the **Modules** view is not shown under your design workspace, go to **Window** > **Show View...** > **Talend** and then select **Modules** from the list.



The table below describes the information presented in the **Modules** view.

Column	Description
Status	points out if a module is installed or not installed on your system. The icon indicates that the module is not necessarily required for the corresponding component or Metadata connection listed in the **Context** column. The icon indicates that the module is absolutely required for the corresponding componentor Metadata connection.
Context	lists the name of **Talend** componentor Metadata connection using the module. If this column is empty, the module is then required for the general use of *Talend Studio*.

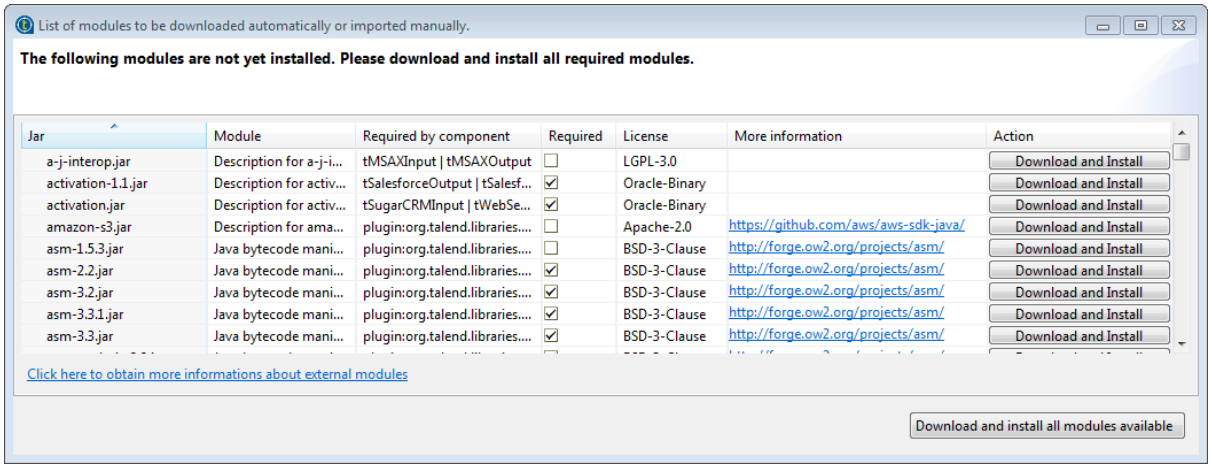
Column	Description
	This column lists any external libraries added to the routines you create and save in the Studio library folder. For more information, see the *Talend Studio User Guide*.
Module	lists the module exact name.
Description	explains why the module/library is required.
Required	the selected check box indicates that the module is required.

In addition to the **Modules** view, the Studio provides a mechanism that enables you to easily identify, download and install most of the required third-party modules from the **Talend** website and directs you to valid websites for the rest.

A Jar installation wizard appears whenever any required external module is found missing for any feature in the Studio, including when you:

- drop a component from the **Palette** if one or more external modules required for that component to work are missing in the Studio, or
- click the **Check** button in a Metadata connection setup wizard in *Talend Studio* if one or more external modules required for the connection are missing in the Studio, or
- click the **Guess schema** button in the **Component** view of a component if one or more external modules required for that component to work are missing in the Studio,
- click **Install** on the top of the **Basic settings** or **Advanced settings** view of a component for which one or more required external modules are missing,
- run a Job that involves components or Metadata connections for which one or more required external modules are missing, or
- click the button in the **Modules** view.

When you click this button, the wizard that appears will list all the required external modules that are not integrated in the Studio.



The table below describes the information presented in the wizard.

Item	Description
Jar	The file name of the external module.
Module	A short description about the nature of the module.
Required by component	Lists the components that require the external module.
Required	The selected check box indicates that the module is required.
License	The license under which the module is provided.
More information	Provides the URL of the valid website where you can find more information about this module and download the module manually.

Item	Description
Action	Download and Install: Click to open the **[Download external modules]** dialog box to download and install the module, which is available on the **Talend** website; Open in browser: Click the link to open the valid website to download the module, which is not available on the **Talend** website, and then click the jar button to import the downloaded module into your studio; : You need to find and download the module yourself and click the jar button to import it into the your studio.
Download and install all modules available	Click to open the **[Download external modules]** dialog box to download and install all the required modules that are available on the **Talend** website.
Do not show again	Select to prevent the wizard from appearing again unless you click the button in the **Modules** tab view. This check box shows only when you drop a component, set up a connection, or guess the schema of a database, that requires an external module, or click the **Install** button on the **Component** tab of a component that requires an external module.
Click here to obtain more information about external modules	Click to go to **Talend** online documentation on installing third-party modules.

This wizard lists the external modules to be installed, the licenses under which they are provided, and the URLs of the valid websites where they are downloadable, and allows you to download and install automatically all the modules available on the Talend website and download those not available on the **Talend** website by following the links provided in the **Action** column and then install them into your Studio manually.

When you drop a component, set up a connection, or guess the schema of a database, that requires an external module for which neither the Jar file nor its download URL information is available on the **Talend** website, the Jar installation wizard does not appear, but the **Error Log** view will present an error message informing you that the download URL for that module is not available. You can try to find and download it by yourself, and then install it manually into the Studio.

To show the **Error Log** view on the tab system, go to **Window** > **Show views**, then expand the **General** node and select **Error Log**.

3.5.4.2. Install external modules

To download and install modules in the Studio

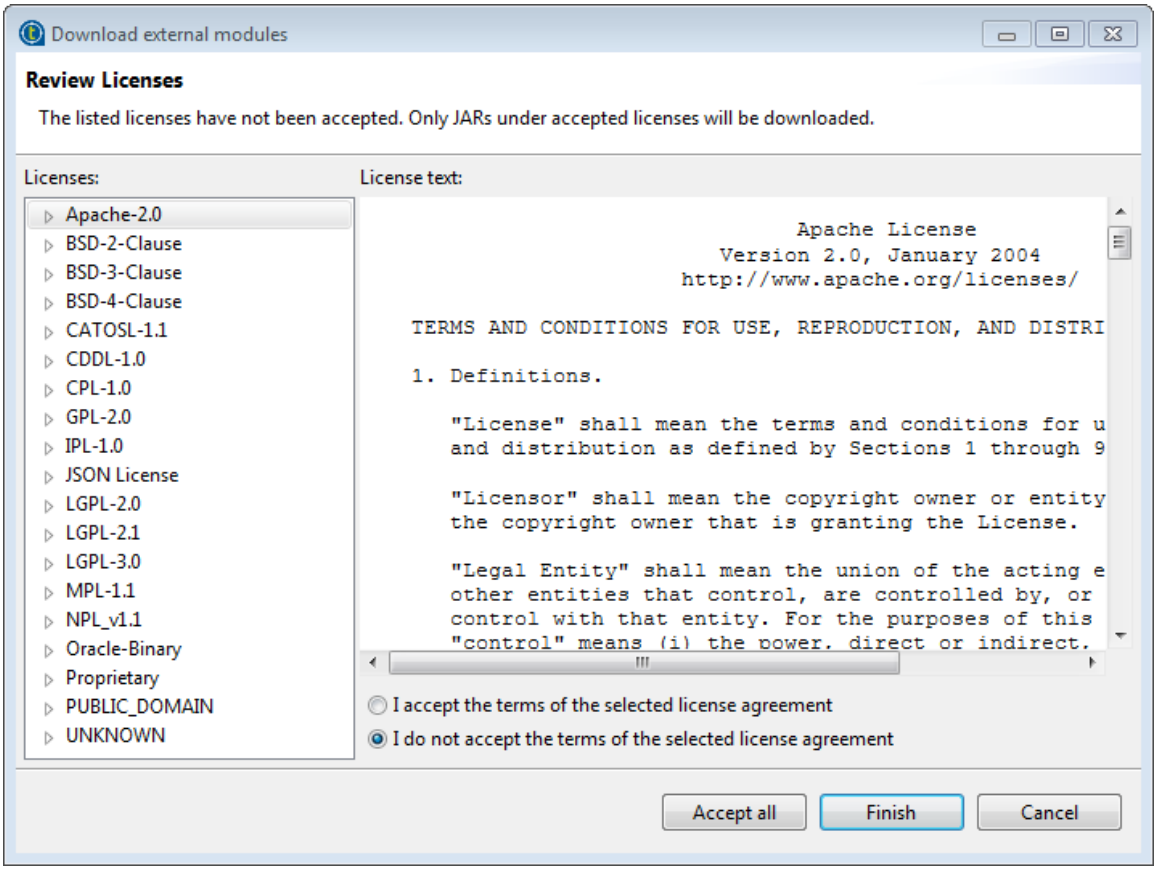
To download and install external modules automatically, do the following:

*If you are working behind a network proxy, make sure you have correctly set up your proxy before you can download and install external modules in your Studio. To access the proxy settings, select **Window** > **Preferences** from the menu to open the **[Preferences]** window, then expand the **General** node and click **Network Connections**.*

*Add the web site **http://talend-update.talend.com** and the port 443 to the whitelist.*

1. In the Jar installation wizard, click the **Download and Install** button to install a particular module, or click the **Download and install all modules available** button to install all the available modules, or select **Help** > **Install Additional Packages** from the menu to open the **[Additional Talend Packages]** wizard. From this wizard, make sure the Optional and Required third-party libraries check boxes are selected and click **Finish**. The **[Download external modules]** dialog box opens.

   This **[Additional Talend Packages]** wizard appears automatically when you launch *Talend Studio* if any additional packages, including external modules, need to be installed for any features to function in the Studio.



2. To download and install the external module(s) provided under a particular license, select that license from the **Licenses** pane, review the license terms, select the **I accept the terms of the license agreement** option, and click **Finish** to start the download and installation process.

   To download and install all external modules provided under all the listed licenses, click the **Accept all** button to start the download and installation process.

   Upon installation of the chosen external module or modules, a dialog box appears to notify you about the number of modules successfully installed and/or about the modules failed to install, if any.

   To install manually an external module you already have in your local file system, do the following:

   Talend Open Studio for Big Data does not come with the JDBC drivers for Oracle databases due to Apache license restrictions. For Oracle9*i*, the required JDBC driver downloadable from Oracle website is named *ojdbc14.jar*, the same as that for Oracle 10g. To enable the JDBC driver for Oracle9*i* you have downloaded to work in **Talend Open Studio for Big Data**, you have to change the file name to *ojdbc14-9i.jar* before installing it into the Studio.

   1. Click the button in the upper right corner of the **Modules** view or in Jar installation wizard to browse your local file system.

      If the **Modules** view is not shown under your design workspace, go to **Window** > **Show View...** > **Talend** and then select **Modules** from the list.

   2. In the **[Open]** dialog box of your file system, browse to the module you want to install, double-click the *.jar* file, or select it and then click **Open** to install it.

      The dialog box closes and the selected module is installed in the library folder of the current Studio.

      You can now use the component or Metadata connection dependent on this module in any of your Job designs.

In case of collaborative work, once a required module is installed in one user's studio, the other users can simply refresh their Modules view to add this module to their own studio(s).

To install modules in CommandLine

If you use the Studio and CommandLine on different machines, you need to retrieve the downloaded *.jar* files and add them in CommandLine.

1. Make sure CommandLine is not started, then download the external modules from the **Modules** view as explained in the previous procedure.
2. Copy the downloaded *.jar* files from *<StudioPath>/configuration/.m2* and paste them into *<CommandLinePath>/configuration/.m2*, where *<StudioPath>* and *<CommandLinePath>* are the installation directories of the Studio and CommandLine respectively.

   Since these folders are hidden, make sure your system is configured to show hidden files and folders.

   The *<CommandLinePath>/configuration/.m2* folder is not created by default. It is created the first time you start the CommandLine application.
3. Restart CommandLine.

   You can now use the component or Metadata connection dependent on these modules.

To install modules downloaded from external websites

Some modules are not available on the **Talend** website but can be downloaded directly from external websites. Once downloaded, these modules must be placed in specific folders.

- For the studio, the downloaded modules must be placed in the following folder:

  <StudioPath>/configuration/.m2

  For Talend Administration Center, the downloaded modules must be placed in the following folder:

  <TomcatPath>/webapps/org.talend.administrator/WEB-INF/lib

3.6. Installing and configuring CommandLine

CommandLine is required for generating and deploying the processes (Job) developed with *Talend Studio* onto the Job servers. This application allows the JAVA code to be generated for the Jobs on the basis of the XML files contained in the database, and must be started in order to allow Jobs to be started and deployed remotely.

To install and configure your CommandLine, proceed as follows:

- *Installing CommandLine*
- *Editing the memory and JVM settings for CommandLine*
- *Accessing user-defined components from the CommandLine*

3.6.1. Installing CommandLine

1. Copy your *Talend Studio* archive file onto the machine where you want to install CommandLine. This machine can be the same as your Web application, *Talend Administration Center*, but not necessarily.

2. Unzip it under a folder the name of which does NOT contain any space character.
3. If needed, rename the decompressed folder to *CmdLine* for more clarity.

   Renaming your CommandLine "CommandLine" is causing problems, so it is recommended to rename it differently or not to rename it at all.

   In this directory, the *commandline.sh* file lets you launch the *CommandLine* program.
4. By default, after the first connection to the Studio, the license file is automatically put at the root of the CommandLine installation directory. If not, put the license file provided by *Talend* at the root of this directory.

   Note that, during the first connection to the remote repository via *Talend Administration Center*, the external libraries required by some components and connection metadata are retrieved from the Git or SVN directory defined on the **Configuration** page of *Talend Administration Center*. Therefore, it is important that you first install *Talend Administration Center*, you configure it and you connect the Studio to the remote repository before you launch CommandLine in order to avoid errors during code generation of Jobs related to these required external libraries. For more information about how to set up the Git or SVN directory where the external libraries are stored, see the *Talend Administration Center User Guide*.
5. Run the *commandline.sh* file.

   For reasons of rights management, make sure you launch CommandLine using the same administrator account as for Tomcat. For example, you can create an account *DIAdmin* for both Tomcat and CommandLine.

   If you need to change the port number (by default 8002), simply edit the CommandLine *.sh* file and change the port number.
6. You can stop the CommandLine execution by press **Ctrl**+**C**.
7. If you chose to install CommandLine on a different machine than *Talend Administration Center*, then you need to configure the CommandLine access parameters in the **Configuration** page of *Talend Administration Center* Web application. For more information, see the *Talend Administration Center User Guide*.

For more information about the CommandLine usage, see Appendix A of the *Talend Administration Center User Guide*.

To install CommandLine as a service, see *Installing CommandLine as a service*.

3.6.2. Editing the memory and JVM settings for CommandLine

To gain in performance at runtime and when launching the CommandLine, you can edit the memory settings in the corresponding *.ini* file.

1. Edit the *Talend-Studio-linux-gtk-x86_64.ini* file.
2. Edit the memory attributes. For example:

   ```
   -vmargs -Xms40m -Xmx500m -XX:MaxMetaspaceSize=256m
   ```

   For more details, see http://www.oracle.com/technetwork/java/hotspotfaq-138619.html.
3. Edit the *commandline-linux_x86_64.sh* file.
4. Change the following information:

   ```
   ./Talend-Studio-linux-gtk-x86_64 -nosplash -application
   org.talend.commandline.CommandLine -consoleLog -data
   ```

```
commandline-workspace startServer -p 8002
```

to

```
/My_Jvm_Path/java -jar
plugins/org.eclipse.equinox.launcher_1.1.0.v20100507.jar
-nosplash -application org.talend.commandline.CommandLine
-consoleLog -data commandline-workspace startServer -p 8002
-vmargs -Xmx500m -XX:MaxMetaspaceSize=256m
```

3.6.3. Accessing user-defined components from the CommandLine

If you need to install user-defined components (that you developed locally or downloaded from Talend Exchange for example), then you need to notify the CommandLine with the user component folder.

To configure the path to these components, simply use the following command:

```
setUserComponentPath -up <UserComponentPath>
```

To clear this path, type in the command:

```
setUserComponentPath -c
```

You can also configure the user component path directly from the **Configuration** page of *Talend Administration Center*. For more information, see the *Talend Administration Center User Guide*.

3.7. Installing and configuring Talend ESB

Talend ESB is provided to you through a zip file named *Talend-ESB-VA.B.C*. To install Talend ESB on your server machines, unzip the *Talend-ESB-VA.B.C.zip* file.

The following procedures detail the installation and configuration of the ready-to-use tools contained in the Talend ESB package:

- *Accessing Talend ESB Container*
- *Installing Apache ActiveMQ*
- *Accessing Service Locator*
- *Installing Service Activity Monitoring*
- *Installing Security Token Services*
- *Installing Talend Identity Management Service*

Note about the start commands: Instead of the individual start commands that you can find in the following sections, you can also use: `tesb:start-all` in the container, which starts all the Infrastructure Services, except the Event Logging features which have to be started individually with the `tesb:start-el-default` command.

For more information about the Infrastructure Services, see the *Talend ESB Infrastructure Services Configuration Guide*.

For more information about the logging modules and the advanced configuration of those Services, see *Talend logging modules* and *Talend ESB Container Administration Guide*.

3.7.1. Accessing Talend ESB Container

Once Talend ESB installed, you can access Talend ESB Container in the *Talend-ESB-VA.B.C/container* directory.

Talend ESB Container is an OSGI container, based on Apache Karaf, allowing you to deploy and execute various components and applications inside its *Talend-ESB-VA.B.C/container/deploy* folder.

To launch Talend ESB Container:

1. Browse to the *Talend-ESB-VA.B.C/container/bin* directory.
2. Run the *trun* file.

After starting Talend ESB Container, you need to wait a few seconds for initialization to complete before entering the commands. Karaf, on which the Talend ESB Container is built, starts the non core bundles in background. So even if the console is already available, the commands may not.

For more information on Talend ESB Container usage and configuration, see the *Talend ESB Container Administration Guide* and *Talend ESB Infrastructure Services Configuration Guide*.

Once Talend ESB Container installed and launched, you will be able to install all the other components available in the Talend ESB package as features directly in the container. Thus, when launching Talend ESB Container, all the other components will be launched at the same time.

You also have the possibility to install these components as standalone. Both installation modes are described in the following sub-sections.

When installing Talend ESB components as features in the container, you might encounter memory problems. For more information on how to increase the memory allocation of the container, see Talend ESB Container Administrator's Guide.

This implementation can ease the management of Talend ESB but if you want to create a cluster environment, you will need to replicate the container to have several containers with the right components installed as features in it, whereas if you are using the different components as standalone you will only have to duplicate the corresponding instance.

If you only want to use several basic containers, you can also use Talend Runtime, as Talend Runtime is the exact equivalent of the *container* folder provided in Talend ESB. For more information about the installation of Talend Runtime, see *Installing Talend Runtime*.

3.7.1.1. Configuring the container parameters

Customize the Talend ESB Container

1. Go to the following directory: *Talend-ESB-VA.B.C/container/etc*.
2. Edit the following files for example:
   - *org.ops4j.pax.web.cfg*, to change the HTTP listening port.
   - *org.apache.karaf.management.cfg*, to manage RMI connection to connect to Talend ESB Container via JMX in order to manage and supervise each of its components and their activity from a JConsole, for example.

Configure the proxy settings

1. Open the following file to edit it:

   Talend-ESB-VA.B.C/container/etc/org.ops4j.pax.web.cfg

2. Uncomment the line:

org.ops4j.pax.url.mvn.proxySupport=true

so that the settings in the *settings.xml.sample* file are taken into account.

3. Update the *etc/settings.xml.sample* file according to your proxy configuration.

3.7.2. Installing Apache ActiveMQ

ActiveMQ is a message broker enabling to support different messaging options. It will provide you high availability, performance, scalability, reliability and security for enterprise messaging. And it allows you to mediate events between distributed applications, guaranteeing that they reach their intended recipients.

Once Talend ESB installed, you can either access a standalone instance of ActiveMQ in the *Talend-ESB-VA.B.C/activemq* directory or install it as a Feature directly within the Talend ESB Container.

Once installed, ActiveMQ can be used in Talend's Mediation routes, for example.

3.7.2.1. As standalone

To launch ActiveMQ in standalone:

1. Browse to the *Talend-ESB-VA.B.C/activemq/bin* directory.
2. Run the following command:

```
./activemq console
```

3.7.2.2. Configuring Apache ActiveMQ

There are a number of configuration options, and these are listed by entering `activemq -h`.

You can configure the ActiveMQ broker by using either a configuration file or via broker URI. For more information about the broker URI syntax, see the online Apache ActiveMQ documentation.

The default location for configuration files is in *activemq/conf*.

For more information on how to configure Apache ActiveMQ, see the *Talend ESB Infrastructure Services Configuration Guide*.

3.7.2.3. As an OSGi Feature

ActiveMQ can also be installed as a Feature in Talend ESB Container, this way, it will be automatically launched when launching Talend ESB Container.

Install Apache ActiveMQ and create a broker

1. In the Talend Runtime container, use the following command to start ActiveMQ:

```
karaf@trun> feature:install activemq
```

2. By default, no broker is created in the Container. To start a broker within the Talend Runtime container, use the following command:

```
karaf@trun> feature:install activemq-broker
```

It creates a default broker named **amq-broker** and its configuration file: `<TalendRuntimePath>/container/etc/org.apache.activemq.server-default.cfg`. You can modify the broker's default configuration by editing this file. For more information on how to create multiple brokers, to remove or to query a broker, see the *Talend ESB Infrastructure Services Configuration Guide*.

This command also installs the ActiveMQ Web console, available at: `http://localhost:8040/activemqweb/`. For more information on ActiveMQ Web console advanced configuration, see the *Talend ESB Infrastructure Services Configuration Guide*.

3.7.3. Accessing Service Locator

Service Locator provides automatic and transparent failover and load balancing between service Consumers and Providers and allows for dynamic endpoint registration and lookup.

Once Talend ESB installed, you can access the Service Locator in the *Talend-ESB-VA.B.C/zookeeper* directory or install it as a feature directly within the Talend ESB Container.

3.7.3.1. As an OSGi Feature (Recommended)

Service Locator can also be installed as a Feature in Talend ESB Container, this way, it will be automatically launched when launching Talend ESB Container.

To launch Service Locator as a Feature in Talend ESB Container:

1. Run the Container.
2. Type in the following command to start the feature corresponding to Service Locator: `tesb:start-locator`.
3. To stop the Service Locator, type in the following command: `tesb:stop-locator`.

3.7.3.2. As standalone (Alternative)

To launch Service Locator in standalone:

1. Open a command window.
2. Browse to the *Talend-ESB-VA.B.C/zookeeper/bin* directory.
3. Run the following command:

```
zkServer.sh start
```

Ensure execution rights for the locator startup scripts:

```
chmod a+x zookeeper/bin/*.sh
```

To customize the configuration of the Service Locator standalone, edit the following file: *Talend-ESB-VA.B.C/zookeeper/conf/zoo.cfg* and change the parameters according to your needs:

Field name	Description
tickTime	the basic time unit in milliseconds used by the Service Locator. It is used to do heartbeats, and the minimum session timeout will be twice the tickTime
dataDir	the location to store the in-memory database snapshots and, unless specified otherwise, the transaction log of updates to the database
clientPort	the port to listen for client connections

3.7.4. Installing Service Activity Monitoring

Service Activity Monitoring (SAM) facilitates the capture of analysis of service activity, including service response times, traffic patterns, auditing and more, by capturing events and storing information. This component consists of two parts:

- Agents (sam-agent) which gather and send monitoring data
- A monitoring Server (sam-server) which processes and stores the data

The sequence of how these are used is as follows:

1. The Agent creates events out of requests and replies from both the service consumer and provider side.
2. The events are first collected locally and then sent to the Monitoring Server periodically (so as not to disturb the normal message flow).
3. When the Monitoring Server receives events from the Agent, it optionally uses filters and/or handlers on those events and stores them into a database.

The Agent and Monitoring Server are made available as follows:

- The agent is by default installed as a feature in Talend ESB Container.
- The Monitoring Server needs to be installed into a Servlet Container (Tomcat) or an OSGi Container (Talend ESB Container) and needs access to a database.

Once Talend ESB installed, you can access the Service Activity Monitoring server in the *Talend-ESB-VA.B.C/add-ons/sam* directory to install it or directly install it as a feature within the Talend ESB Container.

3.7.4.1. Prerequisite to the Monitoring Server

The Monitoring Server requires a database engine to store Events data. The supported databases are listed in *Compatible Databases*.

The following are the script files corresponding to the databases, run them to configure the database properly. You can find the SQL scripts in the *Talend-ESB-VA.B.C/add-ons/sam/db* directory.

SQL script filename	**Database**
create.sql	Apache Derby
create_mysql.sql	MySQL
create_oracle.sql	Oracle
create_sqlserver.sql	SQL Server
create_h2.sql	H2 Database Engine
create_db2.sql	IBM DB2

1. Make sure your chosen database is installed properly and is accessible.
2. Login with a user which has CREATE permissions.
3. Run the init SQL script for the corresponding database from the table above.

If the value of *db.recreate* property in the *logserver.properties* is set to true, the init SQL script will be executed automatically when starting the Monitoring Server. But this is not recommended for any database except Apache Derby running in embedded mode.

You will then find the EVENTS and EVENTS_CUSTOMINFO table have been created in your database.

Now, you can install the Monitoring server either in standalone or as a Feature in the Talend ESB Container.

3.7.4.2. As an OSGi Feature (Recommended)

Service Activity Monitoring server can be installed as a feature in Talend ESB Container, this way, it will be automatically launched when launching Talend ESB Container.

To launch Service Activity Monitoring as a Feature in Talend ESB Container:

1. Run the Container.
2. Type in the following command to start the SAM server Feature: `tesb:start-sam`.
3. To stop the SAM server, type in the following command: `tesb:stop-sam`.

3.7.4.3. As Web application (Alternative)

To install the Service Activity Monitoring (SAM) server as Web application, you need to:

- deploy it in an Servlet Container.
- configure the database connection information,
- configure the Monitoring endpoint in the Talend ESB Container.

For more information, see the procedures below.

Deploy SAM into Apache Tomcat

1. Copy the *sam-server-war.war* file of the *Talend-ESB-VA.B.C/add-ons/sam* directory.
2. Paste it in the *<TomcatPath>/webapps* directory.

   The next time you will start Tomcat, the SAM Server application will automatically be deployed on the server.

   To do it in command line, you can use the following command:

   cp Talend-ESB-VA.B.C/add-ons/sam/sam-server-war.war <TomcatPath>/webapps
3. You can check whether the SAM Server has been successfully installed and is running by going to the following URL: `http://localhost:8080/sam-server-war/services/sam`

   "http://localhost:8080/sam-server-war/services/sam" is only given as example. Depending on your configuration, you may have to replace <localhost> with the IP address of the Web server and <8080> with the actual port used for the application.

Configure the database connection information

1. Open the *<TomcatPath>/conf/context.xml* file and add the following lines, according to your database server:

   For H2:

```
<Resource name="jdbc/datasource" auth="Container"
type="javax.sql.DataSource" username="sa" password=""
driverClassName="org.h2.Driver"
url="jdbc:h2:tcp://localhost/~/test"
maxActive="8" maxIdle="30" maxWait="10000"/>
```

   For Derby:

```
<Resource name="jdbc/datasource" auth="Container"
type="javax.sql.DataSource" username="test" password="test"
```

```
driverClassName="org.apache.derby.jdbc.ClientDriver"
url="jdbc:derby://localhost:1527/db;create=true"
maxActive="8" maxIdle="30" maxWait="10000"/>
```

For MySql:

```
<Resource name="jdbc/datasource" auth="Container"
type="javax.sql.DataSource" username="test" password="test"
driverClassName="com.mysql.jdbc.Driver"
url="jdbc:mysql://localhost:3306/test"
maxActive="8" maxIdle="30" maxWait="10000"/>
```

For DB2:

```
<Resource name="jdbc/datasource" auth="Container"
type="javax.sql.DataSource" username="db2admin" password="qwaszx"
driverClassName="com.ibm.db2.jcc.DB2Driver"
url="jdbc:db2://localhost:50000/TEST"
maxActive="8" maxIdle="30" maxWait="10000"/>
```

For SQLServer:

```
<Resource name="jdbc/datasource" auth="Container"
type="javax.sql.DataSource" username="test" password="test"
driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver"
url="jdbc:sqlserver://localhost:1029;instanceName=sqlexpress;databaseName=Test"
maxActive="8" maxIdle="30" maxWait="10000"/>
```

For Oracle:

```
<Resource name="jdbc/datasource" auth="Container"
type="javax.sql.DataSource" username="xxx" password="xxx"
driverClassName="oracle.jdbc.pool.OracleDataSource"
url="jdbc:oracle:thin:@localhost:1521:XE"
maxActive="8" maxIdle="30" maxWait="10000"/>
```

2. Check the database connection information specified in the following file, and edit them if needed:

   <TomcatPath>/webapps/sam-server-war/WEB-INF/logserver.properties

Configure the Monitoring endpoint in the Talend ESB Container

1. In the Service Activity Monitoring Server page available at *http://localhost:8080/sam-server-war/services/sam*, click the **services** link.
2. In the services page, copy the Endpoint address, for example: *http://localhost:8080/sam-server-war/services/MonitoringServiceSOAP*
3. Go to Talend ESB Container configuration directory: *Talend-ESB-VA.B.C/container/etc* to configure its SAM agent with the right Monitoring endpoint.
4. Edit the *org.talend.esb.sam.agent.cfg* file.
5. Replace the **service.url** field with the new Endpoint address.

3.7.5. Installing Security Token Services

An informal description of a Security Token Service is that it is a web service that offers some or all of the following services (among others):

- It can issue a Security Token of some sort based on presented or configured credentials.

- It can say whether a given Security Token is valid or not.
- It can renew (extend the validity of) a given Security Token.
- It can cancel (remove the validity of) a given Security Token.
- It can transform a given Security Token into a Security Token of a different sort.

Offloading this functionality to another service greatly simplifies client and service provider functionality, as they can simply call the STS appropriately rather than have to handle the security processing logic themselves. For example, the WSDL of a service provider might state that a particular type of security token is required to access the service. Then:

1. A client of the service can ask an STS for a Security Token of that particular type, which is then sent to the service provider.
2. The service provider could choose to validate the received token locally, or dispatch the token to an STS for validation.

These are the two most common use cases of an STS.

3.7.5.1. Running STS server as feature in container (Recommended)

To enable the STS server Feature in the Karaf container, execute the following command:

tesb:start-sts

The STS service will start automatically. To make sure that it is running, execute the following command in the console:

list

and find two additional bundles: *Apache CXF STS Core* and *Talend :: ESB :: STS :: CONFIG* which enable the STS functionality.

It is normal that the status of this (fragment) bundle is only Resolved and not Active, as the other one.

Sample keys distributed with the RentACar demo should not be used in production. For more information on how to replace the keys used, see the chapter "Using STS with the Talend Runtime" from the *Talend ESB Infrastructure Services Configuration Guide*.

For additional information about the usage of STS, please read the *Talend ESB STS User Guide* and the chapter "Using STS with the Talend Runtime" from the *Talend ESB Infrastructure Services Configuration Guide*.

3.7.5.2. Running STS server as Web application (Alternative)

The STS war file is located at *add-ons/sts/SecurityTokenService.war* of the distributive directory and ready for deployment on Tomcat.

For the STS war file deployment, please use standard deployment instructions for your J2EE container (Deployment guide for Tomcat 8: http://tomcat.apache.org/tomcat-8.0-doc/deployer-howto.html) and the chapter "Using STS with the Talend Runtime" from the *Talend ESB Infrastructure Services Configuration Guide*.

Sample keys distributed with the RentACar demo should not be used in production. For more information on how to replace the keys used, see the chapter "Using STS with the Talend Runtime" from the *Talend ESB Infrastructure Services Configuration Guide*.

3.7.6. Installing Talend Identity Management Service

Talend Identity Management Service, based on Apache Syncope, is a system that allows you to manage digital identities in enterprise environments.

The recommended application server for the Syncope Web application is Apache Tomcat 8, however Apache Tomcat 7 is also supported and requires the same installation procedure. In the following sections, <*TomcatPath*> designates the Tomcat installation path.

3.7.6.1. Configuring the Tomcat application server

Set the environment

- Create the following file: <*TomcatPath*>*/bin/setenv.sh*, edit it and add the following in a single line:

```
JAVA_OPTS="-Djava.awt.headless=true -Dfile.encoding=UTF-8 -server \ -Xms1536m -
Xmx1536m
-XX:NewSize=256m -XX:MaxNewSize=256m -XX:PermSize=256m \ -XX:MaxMetaspaceSize=256m -
XX:+DisableExplicitGC"
```

Update the context.xml file

1. Open the following file:<*TomcatPath*>*/conf/context.xml* to edit it.
2. Uncomment the line: `<Manager pathname="" />`
3. For production, it is highly recommended to define a datasource as internal storage to be used with Talend Identity Management Service:

```
<Resource name="jdbc/syncopeDataSource" auth="Container"
    type="javax.sql.DataSource"
    factory="org.apache.tomcat.jdbc.pool.DataSourceFactory"
    testWhileIdle="true" testOnBorrow="true" testOnReturn="true"
    validationQuery="SELECT 1" validationInterval="30000"
    maxActive="100" minIdle="2" maxWait="10000" initialSize="2"
    removeAbandonedTimeout="20000" removeAbandoned="true"
    logAbandoned="true" suspectTimeout="20000"
    timeBetweenEvictionRunsMillis="5000"
    minEvictableIdleTimeMillis="5000"
    jdbcInterceptors="org.apache.tomcat.jdbc.pool.interceptor.ConnectionState;
    org.apache.tomcat.jdbc.pool.interceptor.StatementFinalizer"
    username="syncope" password="syncope"
    driverClassName="com.mysql.jdbc.Driver"
    url="jdbc:mysql://localhost:3306/syncope?characterEncoding=UTF-8"/>
```

   The above example is for MySQL, please adjust the connection parameters to suit your needs.

3.7.6.2. Deploying Talend Identity Management Service

1. Copy the following files from *add-ons/tidm*:

   syncope.war

   syncope-console.war

   to <*TomcatPath*>*/webapps*

2. Launch the Tomcat server.

3. After having launch the server, if Talend Administration Center is not on the default port 8080, stop it and change the port in the *<TomcatPath>/webapps/syncope-console/WEB- INF/classes/configuration.properties* file.

4. If you defined a datasource to be used with Talend Identity Management Service for production purpose, you will need to edit `<TomcatPath>/webapps/syncope-console/WEB-INF/web.xml` and uncomment the `resource-ref` section.

5. Launch the Tomcat server again.

3.7.6.3. Configuring Talend Identity Management Service in a cluster

1. After Talend Identity Management Service has been deployed into Tomcat, to be able to configure it in a cluster, you need to stop Tomcat.

2. Edit `<TomcatPath>/webapps/syncope/WEB-INF/classes/persistence.properties`, and replace the existing configuration with the following:

```
jpa.driverClassName=com.mysql.jdbc.Driver
jpa.url=jdbc:mysql://localhost:3306/syncope?characterEncoding=UTF-8
jpa.username=syncope_user
jpa.password=syncope_pass
jpa.dialect=org.apache.openjpa.jdbc.sql.MySQLDictionary
quartz.jobstore=org.quartz.impl.jdbcjobstore.StdJDBCDelegate
quartz.sql=tables_mysql.sql
logback.sql=mysql.sql
```

3. When deploying multiple Talend Identity Management Service instances, sharing a single database or a single database cluster, it is of fundamental importance that the contained OpenJPA instances are correctly configured for remote event notification. Such configuration, in fact, allows OpenJPA's data cache to remain synchronized when deployed in multiple JVMs, thus enforcing data consistency across all instances.

   The default configuration, adapted for single JVM installations is defined in `<TomcatPath>/webapps/syncope/WEB-INF/classes/persistenceContextEMFactory.xml`, as follows:

```
<entrykey="openjpa.RemoteCommitProvider"value="sjvm"/>
```

   With multiple instances, more options like as TCP or JMS are available. For reference, see http://openjpa.apache.org/builds/2.3.0/apache-openjpa/docs/ref_guide_event.html.

   To use Talend Identity Management Service in a cluster, change the default `sjvm` value with all the IP addresses of all the instances, so they can communicated with each other, in the `<TomcatPath>/webapps/syncope/WEB-INF/classes/persistenceContextEMFactory.xml` file, as follows:

```
<entrykey="openjpa.RemoteCommitProvider"value="tcp(Addresses=10.0.1.10;10.0.1.11)"/>
```

4. Launch the Tomcat server again.

3.7.6.4. Configuring Talend Identity Management Service to use Postgres as internal storage

Prepare Postgres

1. Using pgAdmin III, in the object browser, select the node called **PostgresSQL 9.2 (localhost:5432)/Login-Roles**.

2. Create a new role named `syncope` with password `syncope`. If you use another role and password, you have to adapt the configuration below.
3. Select **PostgresSQL 9.2 (localhost:5432)/Databases**, and create a new database named `syncope`.
4. Assign the **syncope** role to it.

Deploy Talend Identity Management Service

1. If you did not already deploy `syncope.war` and `syncope-console.war` to *<TomcatPath>/webapps*, do it now.
2. Deploy the Postgres JDBC Driver into Tomcat. The Driver can be downloaded at http://jdbc.postgresql.org/download.html.
3. Copy the downloaded driver JAR into *<TomcatPath>/lib*.

Configure Tomcat

To configure Tomcat for Syncope with Mysql backend, look at http://coheigea.blogspot.de/2013/07/apache-syncope-tutorial-part-i_26.html, which is adapted for Talend Identity Management Service using Postgres.

1. Change the content of *<TomcatPath>/webapps/syncope/WEB-INF/classes/persistence.properties* to:

```
jpa.driverClassName=org.postgresql.Driver
        jpa.url=jdbc:postgresql://localhost:5432/syncope
        jpa.username=syncope
        jpa.password=syncope
        jpa.dialect=org.apache.openjpa.jdbc.sql.PostgresDictionary
        quartz.jobstore=org.quartz.impl.jdbcjobstore.PostgreSQLDelegate
        quartz.sql=tables_postgres.sql
        logback.sql=postgresql.sql
```

2. Add a datasource for internal storage in Tomcat's *conf/context.xml*. When Syncope does not find a datasource called *jdbc/syncopeDataSource*, it will connect to internal storage by instantiating a new connection per request, which carries a performance penalty. To avoid this penalty, you need to add the following code to *<TomcatPath>/conf/context.xml*:

```
<Resource name="jdbc/syncopeDataSource" auth="Container"
    type="javax.sql.DataSource"
    factory="org.apache.tomcat.jdbc.pool.DataSourceFactory"
    testWhileIdle="true" testOnBorrow="true"
    testOnReturn="true"
    validationQuery="SELECT 1" validationInterval="30000"
    maxActive="50" minIdle="2" maxWait="10000" initialSize="2"
    removeAbandonedTimeout="20000" removeAbandoned="true"
    logAbandoned="true" suspectTimeout="20000"
    timeBetweenEvictionRunsMillis="5000"
    minEvictableIdleTimeMillis="5000"
    jdbcInterceptors="org.apache.tomcat.jdbc.pool.interceptor.ConnectionState;
    org.apache.tomcat.jdbc.pool.interceptor.StatementFinalizer"
    username="syncope" password="syncope"
    driverClassName="org.postgresql.Driver"
    url="jdbc:postgresql://localhost:5432/syncope"/>
```

3.7.6.5. Configuring Talend Identity Management Service to use Oracle as internal storage

Deploy Talend Identity Management Service

1. If you did not already deploy `syncope.war` and `syncope-console.war` to *<TomcatPath>/webapps*, do it now.

2. Copy the *ojdbcX.jar* file from your Oracle installation into *<TomcatPath>/lib*.

Configure Tomcat

1. Change the content of *<TomcatPath>/webapps/syncope/WEB-INF/classes/persistence.properties* to:

```
jpa.driverClassName=oracle.jdbc.OracleDriver
jpa.url=jdbc:oracle:thin:@<host>:<port>:xe
jpa.username=<user>
jpa.password=<password>
jpa.dialect=org.apache.openjpa.jdbc.sql.OracleDictionary
jpa.pool.validationQuery=SELECT 1 FROM DUAL
#note: other connection pool settings can also be configured here, see
 persistenceContext.xml
quartz.jobstore=org.quartz.impl.jdbcjobstore.oracle.OracleDelegate
quartz.sql=tables_oracle.sql
audit.sql=audit_oracle.sql
database.schema=<schema_name>
```

2. Replace the values between angle brackets to match the configuration of your system.

3. Change the content of *<TomcatPath>/webapps/syncope/WEB-INF/classes/persistenceContextEMFactory.xml* to:

```
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.springframework.org/schema/beans http://
www.springframework.org/schema/beans/spring-beans.xsd">
    <bean id="entityManagerFactory"
    class="org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean">
        <property name="persistenceXmlLocation" value="classpath*:META-INF/spring-
persistence.xml"/>
        <property name="persistenceUnitName" value="syncopePersistenceUnit"/>
        <property name="dataSource" ref="dataSource"/>
        <property name="jpaVendorAdapter">
            <bean class="org.springframework.orm.jpa.vendor.OpenJpaVendorAdapter">
                <property name="showSql" value="false"/>
                <property name="generateDdl" value="true"/>
                <property name="databasePlatform" value="${jpa.dialect}"/>
            </bean>
        </property>
        <property name="jpaPropertyMap">
            <map>
                <!--<entry key="openjpa.Log" value="SQL=TRACE"/>
                <entry key="openjpa.ConnectionFactoryProperties"
                value="PrintParameters=true, PrettyPrint=true,
 PrettyPrintLineLength=80"/>-->

                <entry key="openjpa.jdbc.Schema" value="<schema_name>"/>
                <entry key="openjpa.NontransactionalWrite" value="false"/>
                <entry key="openjpa.AutoDetach" value="close, commit, nontx-read,
 rollback"/>
                <entry key="openjpa.jdbc.SchemaFactory"
 value="native(ForeignKeys=true)"/>
                <entry key="openjpa.jdbc.MappingDefaults"
                value="ForeignKeyDeleteAction=restrict,
 JoinForeignKeyDeleteAction=restrict"/>
                <entry key="openjpa.DataCache" value="true"/>
                <entry key="openjpa.QueryCache" value="true"/>
                <entry key="openjpa.RemoteCommitProvider" value="sjvm"/>
            </map>
        </property>
    </bean>
</beans>
```

4. In the `<entry key="openjpa.jdbc.Schema" value="<schema_name>"/>` line, replace *<schema_name>* with the name of your schema.

3.7.6.6. Testing the access to the Web application

1. Go to `http://localhost:8080/syncope-console/` (assuming that Apache Tomcat is running on *localhost*, port *8080*).
2. Log in with the default credentials:

   login: *admin*

   password: *password*

3.8. Installing and configuring your Talend Data Quality Portal

Using *Talend Installer* is the recommended way to install *Talend Data Quality Portal* but you can perform a manual installation if needed.

To install and configure your *Talend Data Quality Portal*, proceed as follows:

- *Installing your Talend Data Quality Portal*
- *Configuring your Talend Data Quality Portal*

3.8.1. Installing your Talend Data Quality Portal

Prerequisites:

- If you use MySQL, the *my.ini* file of your MySQL server installation directory must be edited in order to increase the size of query packets processed by the server. For more details, please see Table 1.24, "Talend Data Quality Portal".
- The *talend_dq* database exists and is correctly initialized: it contains tables such as the *TDQ_PRODUCT* table. The database can have another name set by the user in the Studio. Use the same name in the Portal configuration (in the *server.xml* file). See Editing the *servertemplate* file for more information.
- The *Talend-DQPortal-YYYYYYYY_YYYY-VA.B.C.zip* file has been downloaded and unzipped.
- The drivers (*.jar* files) corresponding to your Talend Data Quality Portal configuration database are downloaded.

Execute the script

1. Go to the *dqportal* folder.
2. Execute the *install.sh* script and follow the instructions.

   If you answer yes to the last question of the install.sh script which is 'start tomcat?', the HSQL database is started automatically. However, the next time you want to start Talend Data Quality Portal, you will have to manually start the HSQL database first from the database folder of Tomcat.

Editing the *servertemplate* file

1. Open the file *dqportal/resources/hsql/servertemplate* to edit it.
2. Replace `[%TOMCAT_IP%]` and `[%TOMCAT_PORT%]` with the IP address of the Tomcat server and the appropriate Tomcat server port (as set by the user) respectively.

3. Replace `[%TDQ_DB_DRIVER_CLASS_NAME%]` with
   - `com.mysql.jdbc.Driver` if you use MySQL, or
   - `oracle.jdbc.OracleDriver` if you use Oracle, or
   - `net.sourceforge.jtds.jdbc.Driver` if you use MS SQL Server, or
   - `org.postgresql.Driver` if you use PostgreSQL.
4. Replace `[%TDQ_DB_URL%]` with the connection URL to the data mart database.

Database Type	Connection URL
MySQL	`jdbc:mysql://$ip:$dmport/$datamart?characterEncoding=UTF8&useOldAliasMetadataBehavior=true`
Oracle	`jdbc:oracle:thin:@$ip:$dmport:$datamart`
MS SQL Server	`url="jdbc:jtds:sqlserver://$ip:$dmport/$datamart" validationQuery="select 1"`
PostgreSQL	`jdbc:postgresql://$ip:$dmport/$datamart`

   Note that `$ip`, `$dmport` and `$datamart` in the connection URL need to be replaced with the IP address, port number, and name of the data mart database respectively. By default, the database created by the Studio is named *talend_dq*. Make sure you use the same database name in the connection URL and in the Studio. For further information about setting up a distant database from the studio, see *Talend Studio User Guide*.
5. Replace `[%TDQ_DB_USERNAME%]` and `[%TDQ_DB_PASSWORD%]` with the username and password of the data mart database respectively.
6. Copy the content of the *servertemplate* file inside the `<GlobalNamingResources>` part of the *<TomcatPath>/conf/server.xml* file, then save your edits before closing the file.

Changing the default port used by the embedded HSQL database (optional)

1. Open the *<TomcatPath>/database/startup.bat* file to edit it.
2. Add the following at the end of the line:

   ```
   -port <portNumber>
   ```

   where `<portNumber>` is to be replaced with the port number that will be actually used by the database.
3. Open the *<TomcatPath>/conf/server.xml* file and update the port number accordingly.

Starting Tomcat

1. Before starting Tomcat, start the HSQL database:

   Go to the *<TomcatPath>/database* directory and execute the startup script (*startup.sh*).
2. Go to the *<TomcatPath>/bin* directory and launch the *startup.sh*, then log in to the *Talend Data Quality Portal*.

   For more information about how to log into the portal, please see the *Talend Data Quality Portal User and Administrator Guide*.

3.8.2. Configuring your Talend Data Quality Portal

The following sections are designed to help you customize *Talend Data Quality Portal* for your specific environment and make the best use of it.

3.8.2.1. Configuring the portal IP address

In order to be able to work with *Talend Data Quality Portal* when installed using *Talend Installer*, you must edit a file and configure the IP address before you start the Portal.

This configuration is not mandatory, it depends whether the hostname indicated in the file is mapped to a unique IP address or to multiple IP addresses.

To do so, edit the *\tdqp\apache-tomcat\conf\server.xml* file and modify the **value** properties of the **Environment** element: type your IP address where the hostname is indicated (in the below example, the hostname is indicated before the port number, *8580*).

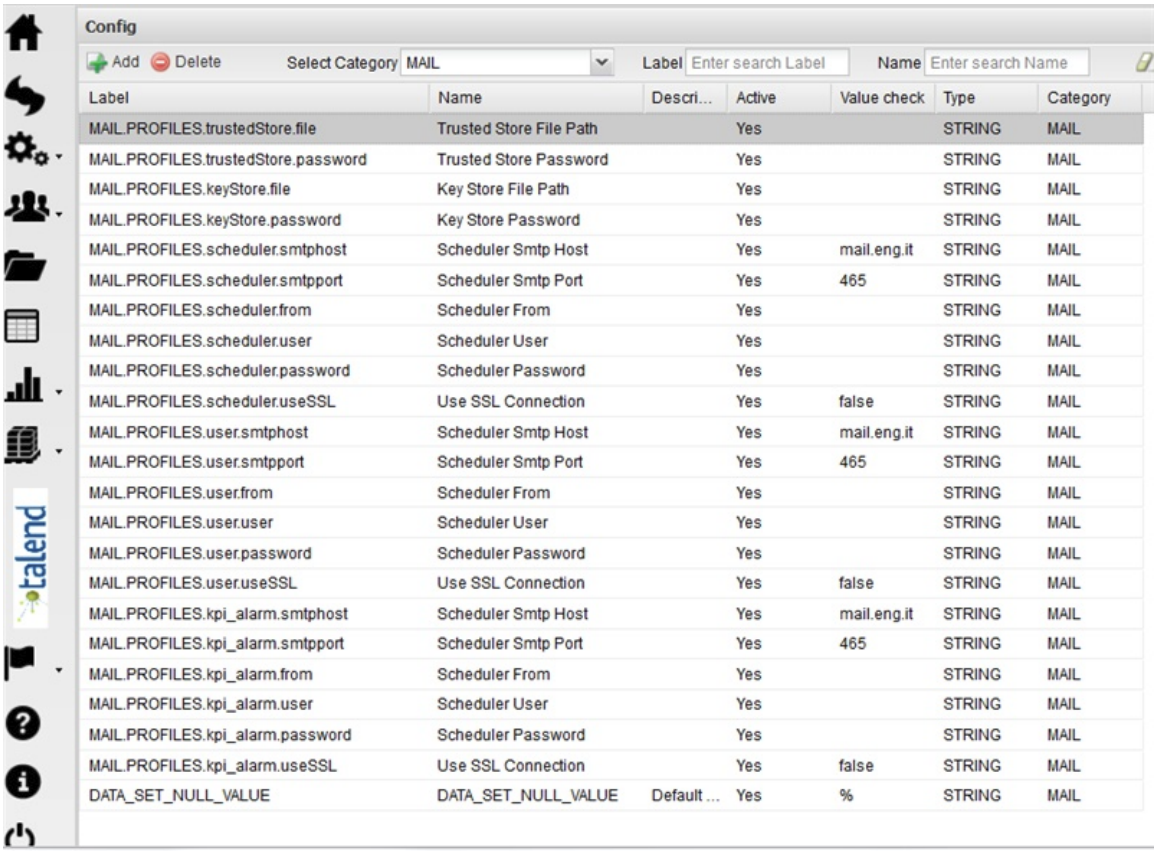
```
<Environment name="spagobi_service_url" type="java.lang.String" value="http://10.42.20.35:8580/tdqportal"/>
<Environment name="spagobi_host_url" type="java.lang.String" value="http://10.42.20.35:8580"/>
<Environment name="talend_logo_path" type="java.lang.String" value="http://10.42.20.35:8580/tdqportal/docs/t
```

3.8.2.2. Configuring the mail server used to send reports

In order to be able to send by email the data quality reports launched from *Talend Data Quality Portal*, you need to configure a mail server.

1. Start Tomcat and log in *Talend Data Quality Portal*.
2. Click the  button and, in the menu that opens, click **Configuration Management** to open the configuration page.
3. Select **MAIL** from the **Select Category** list.

   In this way, SpagoBI show you only the parameters that you have to check/modify.



4. Edit the parameter values corresponding to the SMTP address, the port and the email account that will be used to send the reports.

   Note that you can configure three profiles:

   - **scheduler**: this profile is used to plan the report executions in the scheduler.
   - **user**: this profile is used to allow the user to send documents by e-mail.
   - **kpi_alarm**: this profile is used to send alarm notifications.

3.8.2.3. Configuring LDAP

If you want to use LDAP, note that the properties that need to be configured in *Talend Data Quality Portal* (in **Tools** > **Manages Configurations**) are the following:

- SPAGOBI.SECURITY.PORTAL-SECURITY-CLASS.className=it.eng.spagobi.security.LdapSecurityProviderImpl
- SPAGOBI.SECURITY.USER-PROFILE-FACTORY-CLASS.className=it.eng.spagobi.security.LdapUserProfileFactoryImpl
- SPAGOBI.SECURITY.PORTAL-SECURITY-INIT-CLASS.className=it.eng.spagobi.security.init.LdapSecurityProviderInit

Manage LDAP users

1. Edit the file *<TomcatPath>/webapps/tdqportal/WEB-INF/conf/webapp/ldap_authorizations.xml* and modify the LDAP parameters of your LDAP server.

Attribute Name	Description
HOST	LDAP server host name.
PORT	LDAP server port.
ADMIN_USER	LDAP server administrator user name.
ADMIN_PSW	LDAP server administrator password.
BASE_DN	LDAP server base domain.
USER_SEARCH_PATH	Node under which *Talend Data Quality Portal* looks for users.
USER_OBJECT_CLASS	objectClass corresponding to the user.
USER_ID_ATTRIBUTE_NAME	Name of the attribute containing the user identifier.
USER_NAME_ATTRIBUTE_NAME	Name of the attribute containing the user name.
SUPER_AMDIN_ATTRIBUTE_NAME	Name of the attribute containing the administrator information.
USER_MEMBEROF_ATTRIBUTE_NAME	This attribute must match the USER_ATTRIBUTE one.
USER_ATTRIBUTE	User attributes to be loaded when querying the LDAP server.
GROUP_SEARCH_PATH	Node under which *Talend Data Quality Portal* looks for groups.
GROUP_OBJECT_CLASS	objectClass corresponding to the group.
GROUP_ID_ATTRIBUTE_NAME	Name of the attribute containing the group identifier.
GROUP_ATTRIBUTE	Group attributes to be loaded when querying the LDAP server.
ACCESS_GROUP_NAME	If specified, users must belong to this group in order to access *Talend Data Quality Portal*.
GROUP_MEMBERS_ATTRIBUTE_NAME	If the ACCESS_GROUP_NAME attribute is specified, this attribute has to contain the list of users belonging to this group.

2. Add users using the LDIF File template:

   for example:

we consider DN: `ou=People,dc=talend,dc=com`

and group of users: `ou=/spagobi/talend/tdq_user`

To add a group of users, use the following LDIF template:

...

```
dn: ou=Group,dc=talend,dc=com
objectClass: top
objectClass: organizationalUnit
ou: Group

dn: ou=/spagobi/admin,ou=Group,dc=talend,dc=com
objectClass: organizationalUnit
objectClass: top
description: /spagobi/admin
ou: /spagobi/admin

dn: ou=/spagobi/dev,ou=Group,dc=talend,dc=com
objectClass: organizationalUnit
objectClass: top
description: /spagobi/dev
ou: /spagobi/dev

dn: ou=/spagobi/test,ou=Group,dc=talend,dc=com
objectClass: organizationalUnit
objectClass: top
description: /spagobi/test
ou: /spagobi/test

dn: ou=/spagobi/user,ou=Group,dc=talend,dc=com
objectClass: organizationalUnit
objectClass: top
description: /spagobi/user
ou: /spagobi/user

dn: ou=/spagobi/talend/tdquser,ou=Group,dc=talend,dc=com
objectClass: organizationalUnit
objectClass: top
description: /spagobi/talend/tdquser
ou: /spagobi/talend/tdquser
```

...

To add new users to the group called *People*, use the following LDIF template:

```
dn: cn=tdq_user,ou=People,dc=talend,dc=com
objectClass: organizationalPerson
objectClass: person
objectClass: top
cn: tdq_user
description: data quality user
ou: ou=/spagobi/talend/tdquser,ou=Group,dc=talend,dc=com
sn: tdq_user
userPassword: tdq_password
superAdmin: false
```

3.8.2.4. Configuring the memory

In order to be able to work with *Talend Data Quality Portal* using Tomcat without having any memory issues, you must increase the heap size memory of Tomcat.

For more information on how to increase the memory size of Tomcat, see *Increasing the memory of Tomcat*.

3.9. Installing and configuring Talend SAP RFC Server

Talend SAP RFC Server is a standalone server that acts as a gateway between *Talend Studio* and an SAP server. It receives SAP IDocs or SAP BW Data Source objects from the SAP server and makes them available for processing by the **tSAPIDocReceiver** component or the **tSAPDataSourceReceiver** component and other components in **Talend** Jobs. For more information about these components, see *Talend Components Reference Guide*.

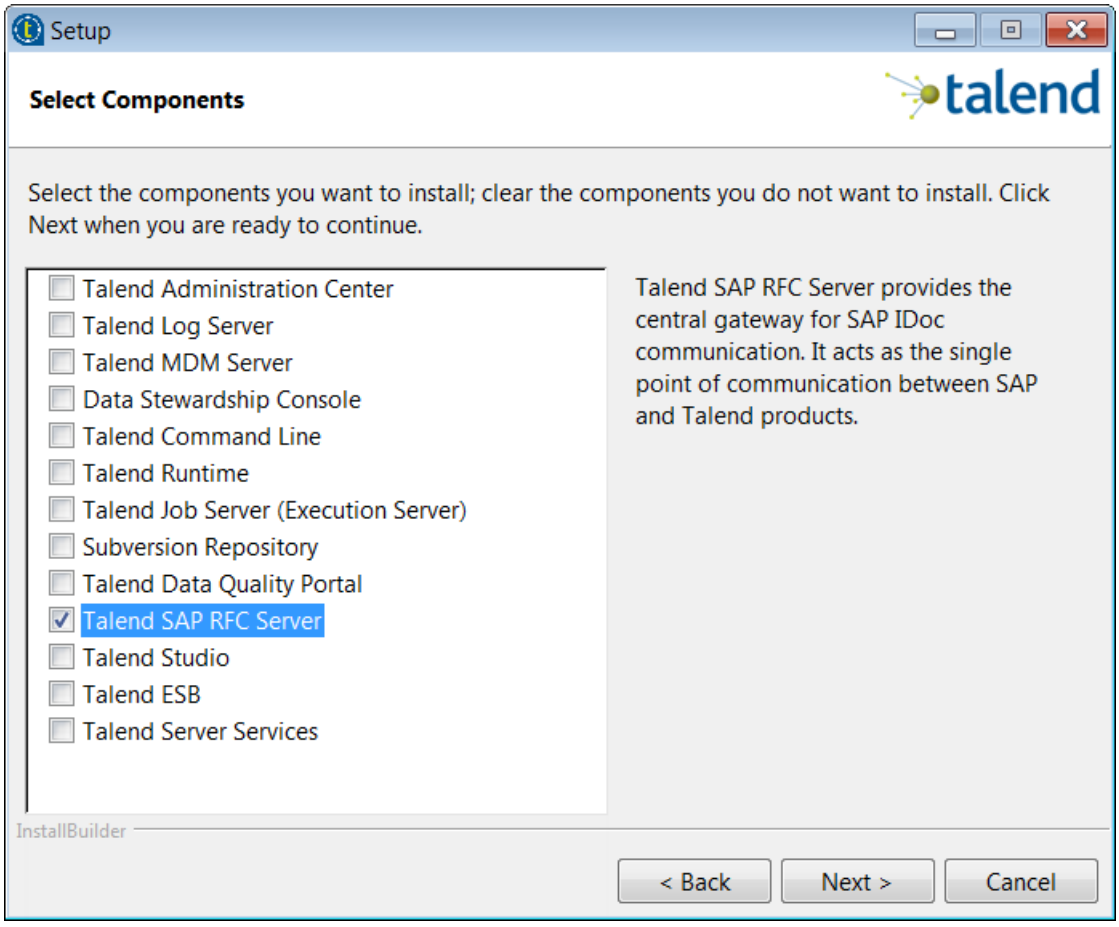
Talend SAP RFC Server is built on an embedded Apache Active MQ. It publishes IDocs or Data Source objects in JMS (Java Message Service) topics or replicates them to JMS queues for batch processing. The **tSAPIDocReceiver** component or the **tSAPDataSourceReceiver** component can then subscribe to these JMS topics or read from the JMS queues.

Note that the SAP server needs to be configured to operate with *Talend SAP RFC Server*. For more information on how to configure SAP, see Talend Help Center.

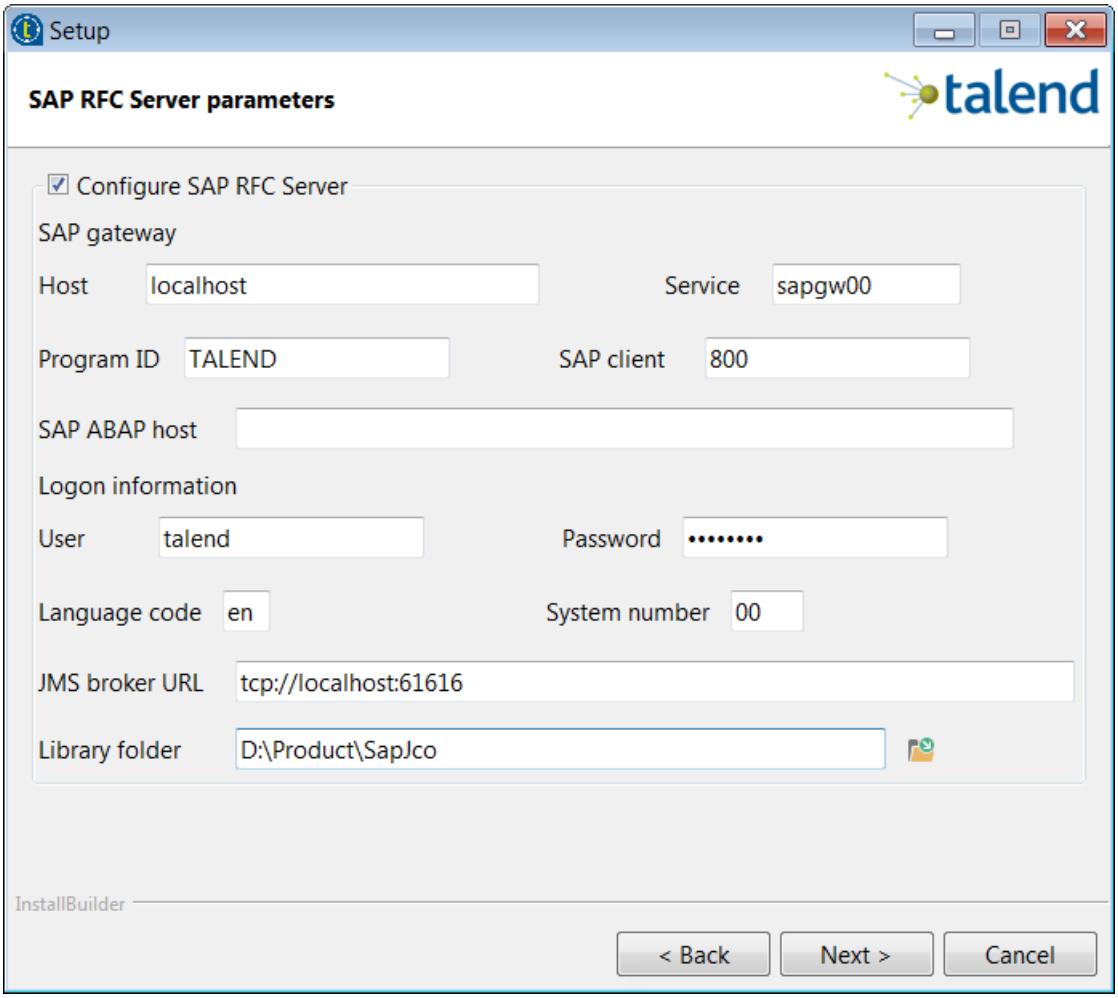
3.9.1. Installing Talend SAP RFC Server

To install *Talend SAP RFC Server* using *Talend Installer*, do the following:

1. Select the **Talend SAP RFC Server** check box in the **[Select Components]** step of the installation process.



2. Select the **Configure SAP RFC Server** check box in the **[SAP RFC Server parameters]** step of the installation process.



3. In the **SAP gateway** section, enter the host name and other identifiers for the SAP system. Contact the administrator of the SAP system for details if you do not already have all this information to hand.

4. In the **Logon information** section, enter the username and password that you use to connect to the SAP system.

5. In the **Language code** field, enter a two-letter language code to specify the language used by the SAP system in its communications with your machine, for instance when displaying error messages or metadata.

6. In the **System number** field, enter the System Number.

7. In the **JMS broker URL** field, enter the host name or IP address of the JMS Broker.

8. In the **Library folder** field, click the folder icon and browse to the location on your local machine where the *sapjco3.jar* file is stored. This proprietary library is required for the *Talend SAP RFC Server* to operate correctly.

9. Click **Next** to proceed to the next step in the installation process.

Once the installation is completed, the *Talend SAP RFC Server* is installed under the *<$INSTALLDIR>\sap-rfc-server* directory, where *<$INSTALLDIR>* is the installer installation directory.

3.9.2. Configuring Talend SAP RFC Server

The configuration file *tsap-rfc-server.properties* for *Talend SAP RFC Server* is created by the installer under the *$TSAPS_HOME\conf* directory (where *$TSAPS_HOME* corresponds to the directory where the *Talend SAP RFC Server* has been installed - by default, this is *<$INSTALLDIR>\sap-rfc-server*). Before starting *Talend SAP RFC Server*, you can configure the file to enable some additional features of the server according to your needs.

Note that any change of the configuration file requires a restart of the *Talend SAP RFC Server*.

- To enable user authentication, you need to uncomment the following three parameters in the configuration file and set their values.

```
# File system directory containing jaas authentication configuration
rfc.server.jms.login.config=conf/user-authentication/login.config

# A jaas username used to authenticate a publisher or sender
rfc.server.jms.login.username=talenduser

# A jaas password used to authenticate a publisher or sender
rfc.server.jms.login.password=talendpassword
```

  Note that the username and password values will be used by the **tSAPIDocReceiver** component or the **tSAPDataSourceReceiver** component to connect to the *Talend SAP RFC Server* and the values must also exist in the *$TSAPS_HOME\conf\user-authentication\users.properties* file. In this file, each row represents a username and password pair, where the username value is on the left side of the equals sign and the password value is on the right side of the equals sign.

```
# username=password
talenduser=talendpassword
```

  Note that if you don't enable user authentication, the **tSAPIDocReceiver** component or the **tSAPDataSourceReceiver** component can also connect to *Talend SAP RFC server* without setting the value for their user and password fields.

- To enable the SSL transport mechanism, you need to copy the key store file for SSL to the *$TSAPS_HOME \conf* folder and then uncomment the following two parameters, the path to the key store file and the password for the key store file, in the configuration file and set their values.

```
# The path to a key store for SSL
rfc.server.jms.ssl.keystore.path=conf/mybroker.ks

# A password for a key store for SSL
rfc.server.jms.ssl.keystore.password=talendkeystore
```

- To set up persistence for SAP IDocs, you need to uncomment the following parameter in the configuration file and set its value to *true*.

```
# Whether JMS messages are persisted or not
rfc.server.jms.persistence=true
```

- To store SAP IDocs in durable queues in *Talend SAP RFC Server*, you need to uncomment the following parameter in the configuration file and set its value to *true*.

```
# Whether JMS messages should be replicated in durable queues
rfc.server.jms.replicate.in.durable.queues=true
```

  This way, the *Talend SAP RFC Server* will keep a copy of all IDocs received in queues named after the IDoc. This is meant to serve the **tSAPIDocsReceiver** component in batch mode. When the receiver runs, it collects all IDocs stored in the durable queues since the last time it ran.

  By default, messages are kept in the queues for up to seven days. You can change the retention period by uncommenting the following parameter in the configuration file and updating its value to meet your own requirement. In this example, the retention period is extended for ten days.

```
# ISO8601 retention period for JMS messages in durable queues
rfc.server.jms.durable.queues.retention.period=P10D
```

3.9.3. Starting Talend SAP RFC Server

To start *Talend SAP RFC Server*, enter the following commands from the command line (where *$TSAPS_HOME* corresponds to the directory where the server has been installed - by default, this is *<$INSTALLDIR>\sap-rfc-server*).

```
cd $TSAPS_HOME/bin
./start-tsaps.sh
```

3.9.4. Stopping Talend SAP RFC Server

To stop *Talend SAP RFC Server*, enter the following commands from the command line (where *$TSAPS_HOME* corresponds to the directory where the server has been installed - by default, this is *<$INSTALLDIR>\sap-rfc-server*).

```
cd $TSAPS_HOME/bin
./stop-tsaps.sh
```

3.10. Installing and configuring Talend Dictionary Service

Using *Talend Installer* is the recommended way to install *Talend Dictionary Service* but you can perform a manual installation if needed.

When using *Talend Installer*, *Talend Dictionary Service* is automatically installed with *Talend Data Preparation* and *Talend Data Stewardship*.

To manually install and configure *Talend Dictionary Service*, proceed as follows:

- *Installing Talend Dictionary Service manually*
- *Configuring Talend Dictionary Service*

3.10.1. Installing Talend Dictionary Service manually

This procedure contains the steps to manually install *Talend Dictionary Service* on your machine. For the automatic installation procedure using Talend Installer, see *Using Talend Installer graphical installation mode*.

Prerequisites:

- *Talend Administration Center* is installed and running. For more information on *Talend Administration Center* installation, see *Using Talend Installer graphical installation mode* for the automatic installation or *Installing and configuring Talend Administration Center* for the manual installation.
- There are no other instance of MongoDB installed on your machine.

Preparing your installation

1. Download Apache Kafka from https://kafka.apache.org/downloads and install it. For more information on how to install it, see Apache Kafka documentation.
2. Download a MongoDB instance from https://www.mongodb.com/download-center and install it. For more information on how to install it, see MongoDB documentation.

   If you want to secure connections with MongoDB using SSL, MongoDB Enterprise Server has to be manually installed on your machine. For more information, see https://docs.mongodb.com/v3.2/security/.
3. Download Apache Tomcat from http://tomcat.apache.org/download-80.cgi and install it. For more information on how to install it, see Apache Tomcat documentation.

   For production environments, it is recommended to use a separate Tomcat instance for *Talend Dictionary Service*.

To manually install and configure *Talend Dictionary Service*, follow this procedure:

Installing and configuring Talend Dictionary Service

1. Add `mongo` to the `PATH` environment variable.
2. Create the `dqdict` database in MongoDB.
3. Create the following user for the `dqdict` database in MongoDB:
   - Username: dqdict-user
   - Password: duser
4. Stop your Tomcat instance if it was automatically started.
5. Unzip the *DQDictServer-A.B.C.zip* to a *dq_dict* folder.
6. Remove the contents of the *<Tomcat>/webapps* folder.
7. Create a *<Tomcat>/app* folder and copy the *dataquality-semantic-producer-A.B.C.war* file from *dq_dict*.
8. Copy the file contained in *dq_dict/context* to *<Tomcat>/conf/Catalina/localhost*.
9. Copy the configuration file contained in *dq_dict/config* to *<Tomcat>/conf*.
10. Open the *<Tomcat>/conf/data-quality.properties* file and edit the following *Talend Dictionary Service* properties:

    All the passwords entered in the properties file are encrypted when you start your *Talend Dictionary Service* instance.

Field	Action
`tac.url`	Enter the URL to your *Talend Administration Center* followed by a /.
`tac.user-name`	Enter the username of your Data Stewardship user in *Talend Administration Center*.
`tac.password`	Enter the password of your Data Stewardship user in *Talend Administration Center*.
`security.token.secret`	Enter the security token used by *Talend Administration Center*.
`spring.cloud.stream.kafka.binder.brokers`	Enter the host corresponding to your Kafka broker.
`spring.cloud.stream.kafka.binder.defaultBrokerPort`	Enter the port corresponding to your Kafka broker.
`spring.cloud.stream.kafka.binder.zkNodes`	Enter the host corresponding to your Zookeeper node.

Field	Action
`spring.cloud.stream.kafka.binder.defaultZkPort`	Enter the port corresponding to your Zookeeper node.

11. Update the following fields with your MongoDB settings:

Field	Description
`dq.mongo.host`	Host name of your MongoDB instance.
`dq.mongo.port`	Port number of your MongoDB instance.
`dq.mongo.username`	Username used to connect to the database.
`dq.mongo.password`	Password used to connect to the database.
`dq.mongo.database.name`	Name of the database on which *Talend Dictionary Service* is connected, *dqdict* by default.

12. Launch MongoDB.
13. Go to the *dq_dict/database* folder and execute the *semantic-dictionary-import.sh* script.
14. Start *Talend Dictionary Service* by launching, in order:
    1. Apache Zookeeper
    2. Apache Kafka
    3. Apache Tomcat

3.10.2. Configuring Talend Dictionary Service

This section contains information on how to secure connections for *Talend Dictionary Service* and how to configure the application logs.

3.10.2.1. Securing connections for Talend Dictionary Service

To secure connections between *Talend Dictionary Service*, the MongoDB server and *Apache Kafka*, proceed as follows:

1. Open the *<Tomcat>/conf/data-quality.properties* file.
2. To secure connections with MongoDB, edit the following lines:

```
dq.mongo.ssl=true
dq.mongo.ssl.trust-store=<path_to_truststore>
dq.mongo.ssl.trust-store-password=<truststore_password>
```

3. To secure connections with Kafka using communication encryption only, edit the following lines:

```
spring.cloud.stream.kafka.binder.configuration.security.protocol=SSL
spring.cloud.stream.kafka.binder.configuration.ssl.truststore.location=<path_to_truststore>
spring.cloud.stream.kafka.binder.configuration.ssl.truststore.password=<truststore_password>
```

4. To secure connections with Kafka using authentication, edit the following lines:

```
spring.cloud.stream.kafka.binder.configuration.ssl.keystore.location=<path_to_keystore>
spring.cloud.stream.kafka.binder.configuration.ssl.keystore.password=<keystore_password>
spring.cloud.stream.kafka.binder.configuration.ssl.key.password=<key_password>
```

Note that the communication encryption parameters must also be defined to use authentication.

Talend Dictionary Service only supports the Java Key Store (`.jks`) format to store keys and certificates.

To enable HTTPS support on Tomcat, see https://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html.

To enable SSL support on MongoDB, see https://docs.mongodb.com/v3.0/tutorial/configure-ssl/.

To enable SSL support on Kafka, see http://kafka.apache.org/documentation.html#security_ssl.

3.10.2.2. Configuring logs for Talend Dictionary Service

Talend Dictionary Service logs allows you to analyze and debug the activity of *Talend Dictionary Service*.

Talend Dictionary Service logs are located in *<Dictionary_Service_Path>/apache-tomcat/logs/*.

To configure the information level of your log files, proceed as follows:

1. Open the *<Dictionary_Service_Path>/apache-tomcat/conf/data-quality.properties* file.
2. Edit the value of the `logging.level.org.talend` field. For more information on *log4j* log levels, see http://logging.apache.org/log4j/1.2/apidocs/org/apache/log4j/Level.html.

3.11. Installing and configuring Talend Data Preparation

Using *Talend Installer* is the recommended way to install *Talend Data Preparation* but you can perform a manual installation if needed.

To manually install and configure *Talend Data Preparation*, proceed as follows:

- *Installing Talend Data Preparation manually*
- *Configuring Talend Data Preparation*

3.11.1. Installing Talend Data Preparation manually

This procedure contains the steps to manually install *Talend Data Preparation* on your machine. For the automatic installation procedure using Talend Installer, see *Using Talend Installer graphical installation mode*.

Prerequisites:

- *Talend Administration Center* is installed and running. For more information on *Talend Administration Center* installation, see *Using Talend Installer graphical installation mode* for the automatic installation or *Installing and configuring Talend Administration Center* for the manual installation.
- A Talend Data Preparation user exists in *Talend Administration Center*. For more information, see *Talend Administration Center User Guide*.
- There are no other instances of MongoDB installed on your machine.
- To use *Talend Data Preparation* with Big Data, use one of the supported Hadoop distribution. For more information, see *Supported Hadoop distribution versions for Talend Data Preparation with Big Data*.

To manually install and configure *Talend Data Preparation*, follow this procedure:

1. Download a MongoDB 3 instance from https://www.mongodb.com/download-center and install it. For more information on how to install it, see MongoDB documentation.

   If you want to secure connections with MongoDB using SSL, MongoDB Enterprise Server has to be manually installed on your machine. For more information, see https://docs.mongodb.com/v3.4/security/.

2. Unzip the *Talend-DataPreparation-Server-VA.B.C.zip* file where you want *Talend Data Preparation* to be installed.

3. Unzip the *<Data_Preparation_Path>/services/components-api-service-rest-all-components-VA.B.C.zip* file where you want *Components Catalog* to be installed.

4. To use *Talend Data Preparation* in a Big Data context, you need to install two additional tools, *Streams Runner* and *Spark Job Server*.

   1. Unpack *<Data_Preparation_Path>/services/data-streams-streamsrunner-svc-A.B.C.tgz* file where you want *Streams Runner* to be installed.

   2. Unpack the *<Data_Preparation_Path>/services/spark-jobserver-A.B.C.tar.gz* file where you want *Spark Job Server* to be installed. This file contains *Spark Job Server* plus all the required dependencies.

      Note that *Spark Job Server* must be installed on a Linux machine.

      Additionally, you must have already installed *curl*, a command-line tool and library for transferring data with URLs. You can download it from https://curl.haxx.se/ if needed.

5. Add `mongo` to the `PATH` environment variable.

6. Create the `dataprep` database in MongoDB.

7. Create the following user for the `dataprep` database in MongoDB:

   - Username: dataprep_user
   - Password: duser

   You can automatically create the user and password by executing the `<Data_Preparation_Path>/create_mongo_user.sh` file.

   Before you use *Talend Data Preparation* for the first time, you must also perform certain configuration steps. For more information, see *Configuring Talend Data Preparation*, *Configuring the Components Catalog server* and, if appropriate, *Configuring Streams Runner* and *Configuring Spark Job Server*,

3.11.2. Configuring the Components Catalog server

This section contains information on how to configure the *Components Catalog* server for use with *Talend Data Preparation*.

1. Open the *<Components_Catalog_Path>/config/application.properties* file.

2. To change the default port exposed for the *Components Catalog* endpoints, edit the following line.

   ```
   server.port=8989
   ```

3. To change the context path for the *Components Catalog* endpoints, edit the following line.

   ```
   server.contextPath=/tcomp
   ```

Note that the `server.contextpath` and `server.port` properties must match the properties defined for `tcomp.server.url` in the *<Data_Preparation_Path>/config/application.properties* file.

4. To enable the *Components Catalog* server for use with *Talend Data Preparation* in a big data context, add the following line to the file.

   ```
   hadoop.conf.dir=/path/to/Hadoop/configuration/directory
   ```

   This property can also be set as an environment variable. Environment variables take precedence over values set in the *application.properties* file.

5. To use the *Components Catalog* server with a secure Hadoop cluster (using Kerberos), add the following line to the file.

   ```
   krb5.config=/path/to/Kerberos/configuration/file/krb5.conf
   ```

   This property can also be set as an environment variable. Environment variables take precedence over values set in the *application.properties* file.

6. Save your changes to the properties file.

7. Restart *Components Catalog* for your changes to be taken into account.

3.11.3. Configuring Streams Runner

This section contains information on how to configure *Streams Runner* for use with *Talend Data Preparation* in a Big Data context.

1. Open the *<Streams_Runner_Path>/conf/application.conf* file.

2. Edit the following line to specify the IP address that you want the *Streams Runner* service to use. The service will listen on this IP address.

   ```
   play.server.http.host=localhost
   ```

3. Edit the following line to specify the port that you want the *Streams Runner* service to use. The service will listen on this port.

   ```
   play.server.http.port=9060
   ```

4. Edit the following line to specify the *Spark Job Server* hostname.

   ```
   app.svc.jobserver.host=localhost
   ```

5. Edit the following line to specify the port used by *Spark Job Server*.

   ```
   app.svc.jobserver.port=8090
   ```

6. Save your changes to the configuration file.

7. Restart *Streams Runner* for your changes to be taken into account.

3.11.4. Configuring Spark Job Server

This section contains information on how to configure *Spark Job Server* for use with *Talend Data Preparation* in a Big Data context.

1. Open the *<Spark_Job_Server_Path>/settings.sh* file.
2. Edit the following line to specify the IP address that you want the *Spark Job Server* service to use. The service will listen on this IP address.

   ```
   sjs_host=localhost
   ```

3. Edit the following line to specify the port that you want the *Spark Job Server* service to use. The service will listen on this port.

   ```
   sjs_port=8098
   ```

4. Edit the following line to specify the path to the Hadoop cluster settings directory. Note that this must be a local path: you can obtain the settings files (such as *hdfs-site.xml*, *mapred-side.xml*, *core-site.xml*, and *yarn-site.xml*) from your Hadoop cluster and copy them to the machine where you installed *Spark Job Server*.

   ```
   hadoop_conf_dir=/path/to/hadoop/cluster/settings/directory
   ```

5. To use *Spark Job Server* server with a secure Hadoop cluster (using Kerberos), add the following line to the file. Note that this must be a local path: you can obtain the *krb5.conf* file from your Hadoop cluster and copy it to the machine where you installed *Spark Job Server*.

   ```
   krb5.config=/path/to/Kerberos/configuration/file/krb5.conf
   ```

6. Save your changes to the settings file.
7. Restart *Spark Job Server* for your changes to be taken into account.

3.11.5. Configuring Talend Data Preparation

This section contains information on the initial configuration actions to be performed after installation, how to secure connections for *Talend Data Preparation* and how to configure the application logs.

3.11.5.1. Configuring Talend Data Preparation after installation

When you install *Talend Data Preparation*, you need to perform certain configuration steps before use.

1. Open the *<Data_Preparation_Path>/config/application.properties* file and edit the following *Talend Data Preparation* properties:

   All the passwords entered in the properties file are encrypted when you start your Talend Data Preparation instance.

Field	Action
`tac.url`	Enter the URL to your *Talend Administration Center* followed by a /.
`public.ip`	Enter the URL you want to use to access *Talend Data Preparation*.
`server.port`	Enter the port you want to use for *Talend Data Preparation* user interface.
`iam.ip`	Enter the URL to your *Talend Identity and Access Management* instance.
`tac.user-name`	Enter the username of your Data Preparation user in *Talend Administration Center*.

Field	Action
`tac.password`	Enter the password of your Data Preparation user in *Talend Administration Center*.

2. Update the following fields with your MongoDB settings:

Field	Description
`mongodb.host`	Host name of your MongoDB instance
`mongodb.port`	Port number of your MongoDB instance
`mongodb.database`	Name of the database on which *Talend Data Preparation* is connected, *dataprep* by default. The database is created when you first launch *Talend Data Preparation*.
`mongodb.user`	Username used to connect to the database
`mongodb.password`	Password used to connect to the database

3. To enable the interaction between *Talend Data Preparation* and the *Components Catalog* service, edit the following line with your *Components Catalog* server host and port:

   ```
   tcomp.server.url=http://<tcomp_host>:<tcomp_port>/tcomp
   ```

4. To configure the access to *Talend Dictionary Service*, edit the following fields:

Field	Description
`spring.cloud.stream.kafka.binder.brokers`	Enter the host corresponding to your Kafka broker.
`spring.cloud.stream.kafka.binder.defaultBrokerPort`	Enter the port corresponding to your Kafka broker.
`spring.cloud.stream.kafka.binder.zkNodes`	Enter the host corresponding to your Zookeeper node.
`spring.cloud.stream.kafka.binder.defaultZkPort`	Enter the port corresponding to your Zookeeper node.

5. To enable the interaction between *Talend Data Preparation* and *Talend Dictionary Service*, set the `dataquality.semantic.update.enable` property as `true`.

6. Change the value of the `dataquality.indexes.file.location` property from `${java.io.tmpdir}/org.talend.dataquality.semantic` to `<other_location>/org.talend.dataquality.semantic`.

   By default, the custom semantic types that you create using the Dictionary Service server are stored in a `tmp` directory. To avoid losing your changes, it is recommended to change the save location of your custom semantic types. You can set a path to to the location of your choice, as long as it is not in a `tmp` folder.

7. To enable to use of the *Streams Runner* with *Talend Data Preparation*, set the `streams.enable` property as true.

8. To configure the access to the *Streams Runner*, edit the following fields:

Field	Description
`streams.flow.runner.url`	Enter the URL to your *Streams Runner*. The URL is made up of your local machine IP address, and your Big Data Preparation port.
`streams.kerberos.principal`	Enter your Kerberos `principal`.
`streams.kerberos.keytab_path`	Enter the path to your Kerberos `keytab` file.
`streams.hdfs.server.url`	You can optionally set a default URL to be displayed in the input and output **Path** fields, when working with HDFS datasets, in Talend Data Preparation.

   The *<Data_Preparation_Path>/config/tuning.properties* file contains additional parameters for more advanced tuning. Make sure the parameters in this file match the sizing of your cluster.

9. Execute the *start.sh* file to start your *Talend Data Preparation* instance.

3.11.5.2. Configuring an HTTPS connection for Talend Data Preparation

To set up an HTTPS secure connection between the different services, as well as with the MongoDB server, you need to edit the application.properties file.

Note that securing the MongoDB connection is not possible if you selected the embedded MongoDB instance during the installation process.

If you want to secure connections with MongoDB using SSL, MongoDB Enterprise Server has to be manually installed on your machine. For more information, see https://docs.mongodb.com/v3.2/security/.

1. Open the *<Data_Preparation_Path>/config/application.properties* file.
2. To define the path and password of the certificate for the Data Preparation server, edit the following lines:

```
# server TLS setup
tls.key-store=/path/to/key-store.jks
tls.key-store-password=key-store_password
```

3. To define the path and password of the signing Certificate Authority (CA) that issued the server certificate, edit the following lines:

```
tls.trust-store=/path/to/trust-store.jks
tls.trust-store-password=trust-store_password
```

4. To make the security control more flexible regarding the certificate common name and its URL, edit the following lines:

```
# false to disable hostname verification
tls.verify-hostname=false
```

5. To define the path and password of the signing Certificate Authority (CA) that issued the MongoDB server certificate, edit the following lines:

```
mongodb.ssl=true
mongodb.ssl.trust-store=/path/to/trus-store.jks
mongodb.ssl.trust-store-password=trust-store-password
```

6. Change the services URLs from `http` to `https`:

```
dataset.service.url=https://${public.ip}:${server.port}
transformation.service.url=https://${public.ip}:${server.port}
preparation.service.url=https://${public.ip}:${server.port}
```

Talend Data Preparation only supports the Java Key Store (`.jks`) format to store keys and certificates.

3.11.5.3. Configuring Talend Data Preparation when Talend Administration Center is in HTTPS

For *Talend Data Preparation* to be able to connect to a *Talend Administration Center* instance running in `https`, *Talend Data Preparation* must trust the *Talend Administration Center* certificate.

1. Retrieve *Talend Administration Center* certificate, or its Certificate Authority and add it to an existing or new *.jks* file following this example:

```
keytool -import -trustcacerts -alias <cert-alias> -file <tac_certificate.crt> -
keystore <truststore.jks>
```

2. In the *<Data_Preparation_Path>/config/application.properties* file, add the following properties to set the truststore:

```
tls.trust-store=/path/to/<truststore.jks>
tls.trust-store-password=<trust-store_password>

false to disable hostname verification
tls.verify-hostname=false
```

3. Restart *Talend Data Preparation*.

3.11.5.4. Using the tDataprepRun component with an HTTPS connection

In order to make the **tDataprepRun** component work when running *Talend Data Preparation* with an `https` connection, complete the following configuration:

1. Retrieve *Talend Data Preparation* certificate, or its Certificate Authority and add it to an existing or new .jks file following this example:

```
keytool -import -trustcacerts -alias <cert-alias> -file <dp_certificate.crt> -keystore
<truststore.jks>
```

2. To make the Studio trust the *Talend Data Preparation* certificate, edit the *.ini* file used to start the Studio:

```
-Djavax.net.ssl.trustStore=/path/to/<trust-store.jks>
-Djavax.net.ssl.trustStorePassword=<trust-store password>
```

3. When designing your Job in the Studio, connect a **tSetKeystore** component to the data input component with an **OnSubjobOk** link in order for the Job to trust the *Talend Data Preparation* certificate. For more information on how to configure the **tSetKeystore**, see *Talend Components Reference Guide*.

For more information on how to use the **tDataprepRun** component and how to operationalize a recipe in a *Talend* Job, see Talend Help Center (https://help.talend.com).

3.11.5.5. Creating a live dataset with an HTTPS connection

To create a working live dataset when running *Talend Data Preparation* with an `https` connection, complete the following configuration:

1. Retrieve *Talend Data Preparation* certificate, or its Certificate Authority and add it to an existing or new `.jks` file following this example:

```
keytool -import -trustcacerts -alias <cert-alias> -file <dp_certificate.crt> -keystore
<truststore.jks>
```

2. When designing your Job in the Studio, connect a **tSetKeystore** component to the data input component with an **OnSubjobOk** link in order for the Job to trust the *Talend Data Preparation* certificate. For more information on how to configure the **tSetKeystore**, see *Talend Components Reference Guide*.

For more information on how to create a live dataset, see Talend Help Center (https://help.talend.com).

3.11.5.6. Configuring logs for Talend Data Preparation

Talend Data Preparation logs allows you to analyze and debug the activity of *Talend Data Preparation*.

Talend Data Preparation logs are located in *<Data_Preparation_Path>/data/logs/app.log*.

To configure the settings of your log files, edit the *<Data_Preparation_Path>/config/log4j2.xml* file:

- For more information on how to set the *log4j* information level, see http://logging.apache.org/log4j/1.2/apidocs/org/apache/log4j/Level.html.
- For more information on how to set the log rotation, see https://logging.apache.org/log4j/2.x/manual/configuration.html#AutomaticReconfiguration.

3.12. Installing and configuring Talend Data Stewardship

Using *Talend Installer* is the recommended way to install *Talend Data Stewardship* but you can perform a manual installation if needed.

To manually install and configure *Talend Data Stewardship*, proceed as follows:

- *Installing Talend Data Stewardship manually*
- *Configuring Talend Data Stewardship*

3.12.1. Installing Talend Data Stewardship manually

This procedure contains the steps to manually install *Talend Data Stewardship* on your machine. For the automatic installation procedure using Talend Installer, see *Using Talend Installer graphical installation mode*.

Prerequisites:

- *Talend Identity and Access Management* is installed and running. For more information on *Talend Identity and Access Management* installation, see *Using Talend Installer graphical installation mode* for the automatic installation or *Installing and configuring Talend Identity and Access Management* for the manual installation.
- *Talend Administration Center* is installed and running. For more information on *Talend Administration Center* installation, see *Using Talend Installer graphical installation mode* for the automatic installation or *Installing and configuring Talend Administration Center* for the manual installation.
- A Talend Data Stewardship user exists in *Talend Administration Center*. For more information, see *Talend Administration Center User Guide*.
- There are no other instance of MongoDB installed on your machine.

To manually install and configure *Talend Data Stewardship*, follow this procedure:

1. Download Apache Kafka from https://kafka.apache.org/downloads and install it.

   For more information on the supported Apache Kafka version, see *Compatible Messaging Systems*.

   For more information on how to install it, see Apache Kafka documentation.

2. Download a MongoDB instance from https://www.mongodb.com/download-center and install it.

   For more information on the supported MongoDB databases, see *Compatible Databases*.

   For more information on how to install it, see MongoDB documentation.

If you want to secure connections with MongoDB using SSL, MongoDB Enterprise Server has to be manually installed on your machine. For more information, see https://docs.mongodb.com/v3.4/security/.

3. Add `mongo` to the `PATH` environment variable.
4. Create the `tds` database in MongoDB.
5. Create the following user for the `tds` database in MongoDB:
   - Username: tds-user
   - Password: duser

   Download Apache Tomcat from http://tomcat.apache.org/download-80.cgi and install it. For more information on how to install it, see Apache Tomcat documentation.

   For production environments, it is recommended to use a separate Tomcat instance for *Talend Data Stewardship*.
6. Stop your Tomcat instance if it was automatically started.
7. Unzip the *data-stewardship-dist-A.B.C-GA-distribution.zip* to a *TDS_files* folder.
8. Remove the *<Tomcat>/webapps/ROOT.war* file.
9. Create a *<Tomcat>/app* folder and copy the *.war* files from *TDS_files*.
10. Copy the files contained in *TDS_files/context* to *<Tomcat>/conf/Catalina/localhost*.
11. Copy the configuration file contained in *TDS_files/config* to *<Tomcat>/conf*.
12. Open the *<Tomcat>/conf/data-stewardship.properties* file and edit the following *Talend Data Stewardship* properties:

All the passwords entered in the properties file are encrypted when you start your *Talend Data Stewardship* instance.

Table 3.1. MongoDB parameters

Property	Description
`spring.data.mongodb.host`	Host name of your MongoDB instance.
`spring.data.mongodb.port`	Port number of your MongoDB instance.
`spring.data.mongodb.database`	Name of the database on which *Talend Data Stewardship* is connected, *tds* by default.
`spring.data.mongodb.user`	Username used to connect to the database.
`spring.data.mongodb.password`	Password used to connect to the database.

Table 3.2. Gateway configuration

Property	Action
`frontend.url`	Replace *${tinstall.tds.tomcat.port.http}* with Apache Tomcat HTTP port.
`backend.url`	Replace *${tinstall.tds.tomcat.port.http}* with Apache Tomcat HTTP port.
`schemaservice.url`	Replace *${tinstall.tds.tomcat.port.http}* with Apache Tomcat HTTP port.
`semanticservice.url`	Enter the URL to *Talend Dictionary Service*. If your licence does not include *Talend Dictionary Service*, delete this line.

Property	Action
`historyservice.url`	Replace *${tinstall.tds.tomcat.port.http}* with Apache Tomcat HTTP port.

Table 3.3. Kafka configuration

Property	Action
`kafka.broker`	Enter the host and the port corresponding to your Kafka broker.

Table 3.4. Talend Identity and Access Management configuration

Property	Action
`oidc.url`	Enter the URL to *Talend Identity and Access Management*, *http://localhost:9080/oidc* for example.
`oidc.userauth.url`	Enter the URL to *Talend Identity and Access Management* User Authentication, *http://localhost:9080/oidc* for example.
`scim.url`	Enter the URL to *Talend Identity and Access Management* SCIM, *http://localhost:9080/scim* for example.
`oidc.gateway.id`	Enter the *Talend Identity and Access Management* OIDC client identifier.
`oidc.gateway.secret`	Enter the *Talend Identity and Access Management* OIDC password.
`oidc.tds.id`	Enter the *Talend Identity and Access Management* OIDC client identifier.
`oidc.tds.secret`	Enter the *Talend Identity and Access Management* OIDC password.
`oidc.history.id`	Enter the *Talend Identity and Access Management* OIDC client identifier you have generated for *Talend Data Stewardship*.
`oidc.history.id`	Enter the *Talend Identity and Access Management* OIDC password you have generated for *Talend Data Stewardship*.
`oidc.schema.id`	Enter the *Talend Identity and Access Management* OIDC client identifier you have generated for *Talend Data Stewardship*.
`oidc.schema.secret`	Enter the *Talend Identity and Access Management* OIDC password you have generated for *Talend Data Stewardship*.

13. Start *Talend Data Stewardship* by launching, in order:
    1. Apache Kafka
    2. MongoDB
    3. Apache Tomcat

3.12.2. Configuring Talend Data Stewardship

This section contains information on how to secure connections for *Talend Data Stewardship* and how to configure the application logs.

3.12.2.1. Securing connections for Talend Data Stewardship

To secure connections between *Talend Data Stewardship*, the MongoDB server and *Apache Kafka*, you need to edit the application.properties file.

Note that securing the MongoDB connection is not possible if you select the embedded MongoDB instance during the installation process. If you want to secure connections with MongoDB using SSL, MongoDB Enterprise Server has to be manually installed on your machine. For more information, see https://docs.mongodb.com/v3.2/security/.

1. Open the *<Data_Stewardship_Path>/config/data-stewardship.properties* file.
2. To trust the server certificate used by *Talend Data Stewardship*, edit the following lines:

```
http.ssl.truststore.location=<path_to_truststore>
http.ssl.truststore.password=<truststore_password>
```

3. By default, *Talend Data Stewardship* will not verify that the hostname matches the certificate common name.

   To enable this verification, change the value of the following field to *true*:

```
http.ssl.verify.hostname=true
```

4. To allow *Talend Data Stewardship* to use private key authentication, edit the following lines:

```
http.ssl.keystore.location=<path_to_keystore>
http.ssl.keystore.password=<keystore_password>
http.ssl.key.password=<key_password>
```

5. To secure connections with MongoDB, edit the following lines:

```
spring.data.mongodb.ssl=true
spring.data.mongodb.ssl.trust-store=<path_to_truststore>
spring.data.mongodb.ssl.trust-store-password=<truststore_password>
```

6. To secure connections with Kafka using communication encryption only, edit the following lines:

```
kafka.security.protocol=SSL
kafka.ssl.truststore.location=<path_to_truststore>
kafka.ssl.truststore.password=<truststore_password>
```

7. To secure connections with Kafka using authentication, edit the following lines:

```
kafka.ssl.keystore.location=<path_to_keystore>
kafka.ssl.keystore.password=<keystore_password>
kafka.ssl.key.password=<key_password>
```

   Note that the communication encryption parameters must also be defined to use authentication.

8. Change the services URLs from `http` to `https`:

```
tds.history.service.url==https://${public.ip}:${server.port}/data-history-service
schema.service.url=https://${public.ip}:${server.port}/schemaservice
```

Talend Data Stewardship only supports the Java Key Store (`.jks`) format to store keys and certificates.

To enable HTTPS support on Tomcat, see https://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html.

To enable SSL support on MongoDB, see https://docs.mongodb.com/v3.0/tutorial/configure-ssl/.

To enable SSL support on Kafka, see http://kafka.apache.org/documentation.html#security_ssl.

3.12.2.2. Configuring logs for Talend Data Stewardship

Talend Data Stewardship logs allows you to analyze and debug the activity of *Talend Data Stewardship*.

Talend Data Stewardship logs are located in *<Data_Stewardship_Path>/apache-tomcat/logs*.

The *catalina.out* file is an aggregated version of all the available log files.

To configure the information level of your log files, proceed as follows:

1. Open the following files:
   - *<Data_Stewardship_Path>/apache-tomcat/conf/data-stewardship-core-logback.xml* for the core backend service log
   - *<Data_Stewardship_Path>/apache-tomcat/conf/data-stewardship-history-logback.xml* for the history service log
   - *<Data_Stewardship_Path>/apache-tomcat/conf/data-stewardship-schema-logback.xml* for the schemas management service log
2. Add the following line before the `<root>` element: `<logger name="org.talend" level="DEBUG"/>`.

   The log information level is now set to `DEBUG`, but you can set it to another value. For more information on log levels, see http://logging.apache.org/log4j/1.2/apidocs/org/apache/log4j/Level.html.

Appendix A. Appendices

The following appendices contain complementary information to go further with your *Talend* product:

- *Introduction to the Talend products*
- *Architecture of the Talend products*
- *Talend High Availability*
- *Cheatsheet: start and stop commands for Talend server modules*
- *Installing Talend servers as Linux services*
- *H2 Database Administration & Maintenance*
- *Supported Third-Party System/Database/Business Application Versions*

A.1. Introduction to the Talend products

The present section lists all the elements required for using the Talend products. To ease their management, we recommend that you centralize all the server modules on one single system.

All Talend applications to be installed must be the same version.

- An application server (Apache Tomcat server) that hosts *Talend Administration Center*.
- A database server storing the administration metadata of *Talend Administration Center* (by default, an embedded H2 database is used).
- A database server storing data quality reports and analyses managed by *Talend Data Quality Portal*.
- A database server storing golden or master data records managed by *Talend Data Stewardship Console* (if your license includes the Data Stewardship option).

  The *Talend Data Stewardship Console* is deprecated since Talend 6.4. Consider migrating to *Talend Data Stewardship*.
- A version control system for Project metadata.
- A Web browser to access Web application:
  - *Talend Administration Center* where projects, users and processes can be managed and administrated. For more information, see the *Talend Administration Center User Guide*.
  - *Data Quality Portal* that extends the dynamic reporting and monitoring capabilities of Talend Studio. For more information, see the *Talend Data Quality Portal User and Administrator Guide*.
- A Nexus artifact repository in which are stored software updates, external libraries and artifacts.
- Execution servers (JobServers) or Talend Runtime execution containers (based on Apache Karaf) to deploy and execute processes.
- A Studio API to carry out technical processes. For more information, see the *Talend Studio User Guide*.
- A *Talend Activity Monitoring Console*, providing detailed monitoring capabilities that can be used to consolidate the collected log information. For more information, see the *Talend Activity Monitoring Console User Guide*.

Each of these elements is detailed in the following sub-sections.

A.1.1. Apache Tomcat Server

The Apache Tomcat server is an application server that hosts *Talend Administration Center*. This Web application gives access to all management and administration functionalities for an integration project, allowing users to (depending on their role):

- Create and manage projects.
- Create and manage user accounts and roles/rights.
- Access the Publisher to publish and schedule artifact publishing on the Nexus artefact repository.
- Access the Job Conductor to schedule, deploy and execute Jobs.
- Access the ESB Conductor to set the deployment and execution of Services, Routes and Generic OSGi features.

- Access the Monitoring node to monitor the execution of Jobs and visualize the logs.
- Access the ESB Infrastructure to monitor the endpoints activity through Service Locator and monitor all service events through Service Activity Monitoring.

Talend Administration Center can also be hosted by JBoss or Pivotal tc application servers.

For more information on how to install *Talend Administration Center* manually, see *Installing and configuring Talend Administration Center*.

A.1.2. Database

The administration database server is used to store administration information and manage the persistence in Talend Administration Center. By default an embedded H2 database is used, but you can also use MySQL, MS SQL Server, or Oracle to store all cross-project data (users, projects, authorization, license, tasks, triggers, monitoring).

The administration database will be named <*talend_administrator*> in the rest of this document.

The <*talend_administrator*> administration database will contain all the data related to project information and administration including: administration data, project declaration, user declaration and authorization, task list, etc.

The tables in this database are automatically created when connecting for the first time to *Talend Administration Center*. The created tables include (among others):

- a *Users* table,
- a *Projects* table,
- a *Rights* table.

These tables are created, populated and managed automatically by Talend, users do not need to take any action.

For more information on how to run *Talend Administration Center* on another database than the embedded H2, see *Configuring Talend Administration Center to run on a different database than H2*.

Additionally, to perform Data Quality operations, a HSQL, MySQL or Oracle database is used to store report and analysis results and a stewardship console database is used to store golden or master data records. For more information about those databases, see the relevant sections in the present guide.

A.1.3. Version control system

A version control system (either Git or SVN) is used to manage the persistence of all the data relating to the objects (Jobs, Services, Routes, Business Models, Metadata, Routines, Documentation, etc.) stored in "git" or "svn" mode in the shared Repository of the Talend Studio.

We recommend you to store several projects per repository, simply in order not to have too many repositories to deal with. However you can choose to store only one project per Git or SVN repository, if you prefer so.

For more information on how to configure your version control systems, see *Setting up your version control system*.

You can also have several version control repositories each containing several projects. For more information on how to create projects and store them in Git or SVN, see the *Talend Administration Center User Guide*.

A.1.4. Artifact Repository

The artifact repository delivered by *Talend* and based on Sonatype Nexus is a preconfigured application centralizing the management and usage of the **Software Update**, **User libraries** and **snapshots** and **releases** repositories:

- **Software Update** is used to manage application updates (patches) distributed by *Talend*. By default the *talend-updates* repository is embedded within **Software Update** and retrieves the updates published by *Talend*. This repository allows the user to visualize the updates available.

  For detailed information, see *Software update repository*.

- The **User libraries** repository is used to store all external libraries. These libraries are retrieved by Talend Studio at start-up and shared with Talend Administration Center via the *talend-custom-libs* repository.

  For detailed information, see *User Libraries repository*.

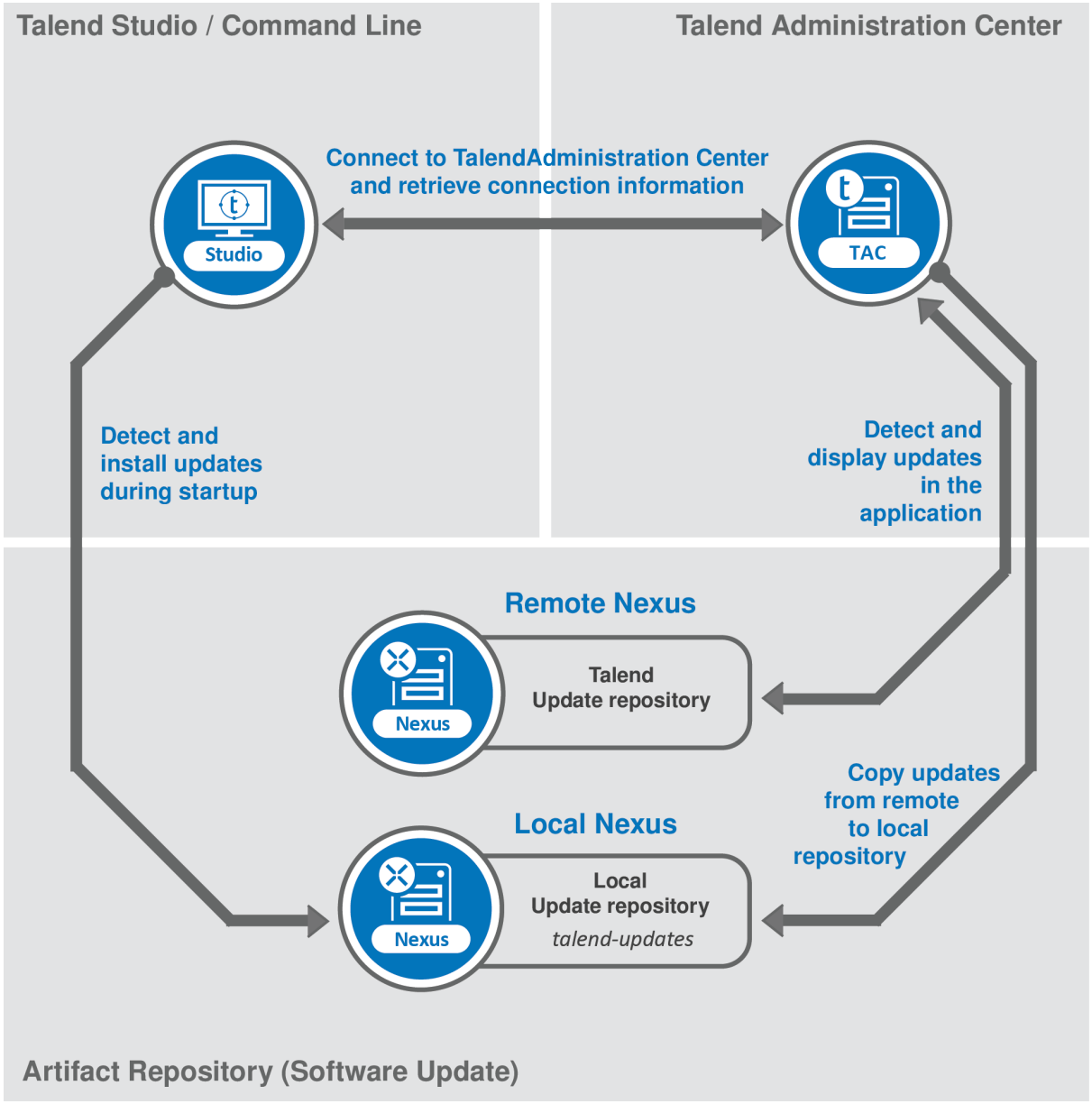
- The **snapshots** and **releases** repositories are used as a catalog in which all artifacts to be deployed and executed are stored. These artifacts are designed by the user from the Studio or any other Java IDE. By default, the **snapshots** repository is used for development purposes and the **releases** repository is used for production. These repositories make artifacts available for deployment and or execution in an execution server.

  For detailed information, see *Snapshots and Releases artifact repositories*.

For more information on how to install and configure Nexus, see *Installing and configuring the Nexus artifact repository*.

A.1.4.1. Software update repository

The following image shows the architecture of **Software Update** linked to *Talend Administration Center* and to the Talend Studio.



To download and install some software updates, you need to connect to **Software Update** (integrated within the Nexus artifact repository) and its embedded repository named *talend-updates*.

To do so, you must install the Nexus artifact repository on your machine and log in its Web interface following the procedure described in *Installing and configuring the Nexus artifact repository*.

The *Artifact-Repository-Nexus-VA.B.C* archive file containing Nexus is embedded in the *Talend Administration Center* archive file that is provided by *Talend*.

In *Talend Administration Center*, the patches available for the current version that have been copied from the Talend remote repository to the local *talend-updates* repository are detected and the administrator can accept them.

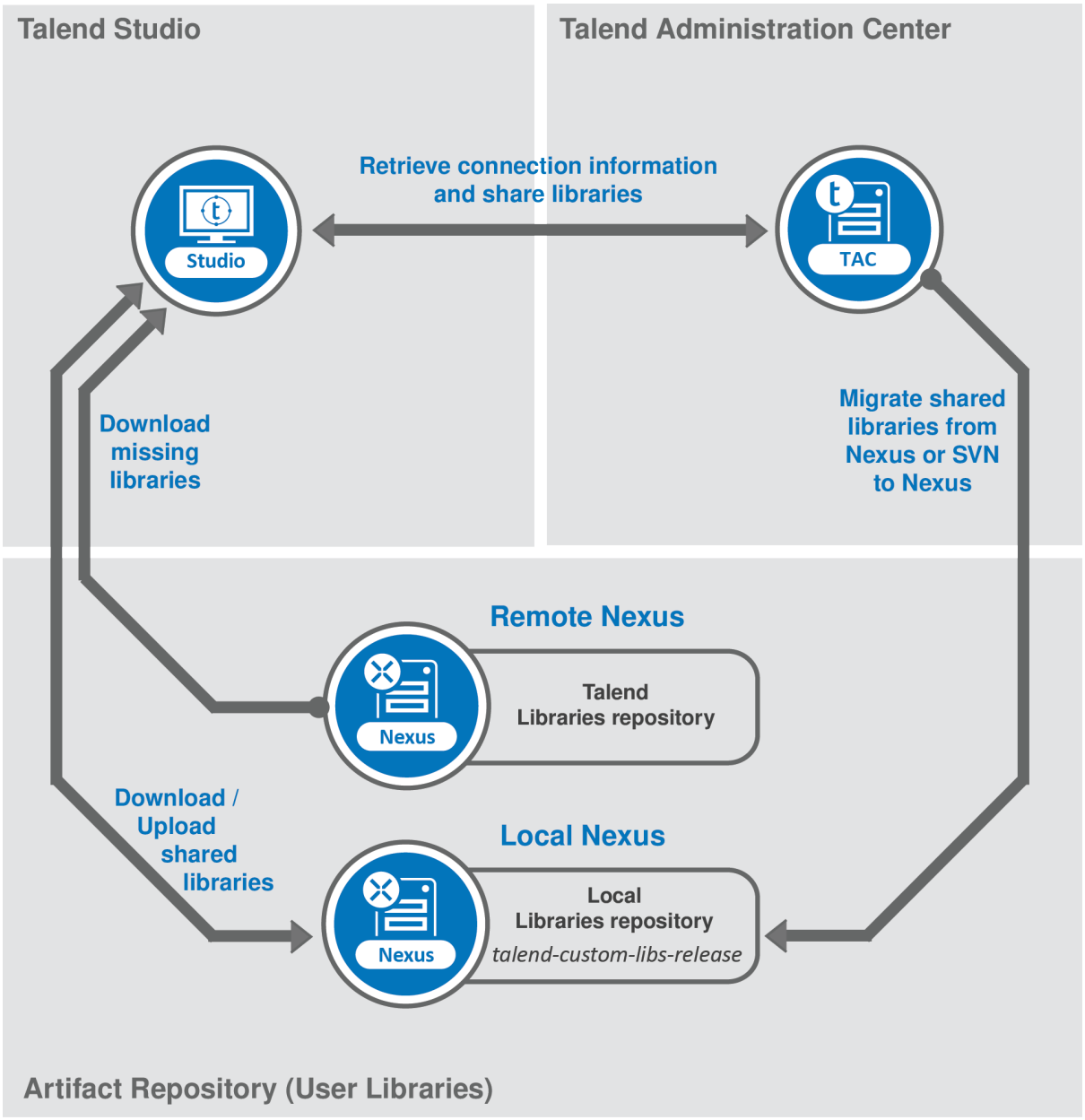
Talend Studio is connected to *Talend Administration Center* to retrieve the repository connection information and the updates are detected and installed automatically.

For more information on how to install and configure the Nexus artifact repository and **Software Update**, see *Installing and configuring the Nexus artifact repository*.

For more information on how to check updates via these repositories, see the *Talend Administration Center* and *Talend Studio User Guides*.

A.1.4.2. User Libraries repository

The following image shows the architecture of the **User Libraries** repository.



To download and install some specific third-party Java libraries or database drivers that are needed by *Talend Studio*, you need to connect to the **User Libraries** repository (integrated within the Nexus artifact repository) and its embedded repository named *talend-custom-libs-release*.

To do so, you must install the Nexus artifact repository on your machine and log in its Web interface following the procedure described in *Installing and configuring the Nexus artifact repository*.

The *Artifact-Repository-Nexus-VA.B.C* archive file containing Nexus is embedded in the *Talend Administration Center* archive file that is provided by *Talend*.

When Talend Studio opens, the external libraries missing from the local *talend-custom-libs-release* repository are detected. You are prompted to download them from the remote Nexus repository, hosted by Talend, and install them.

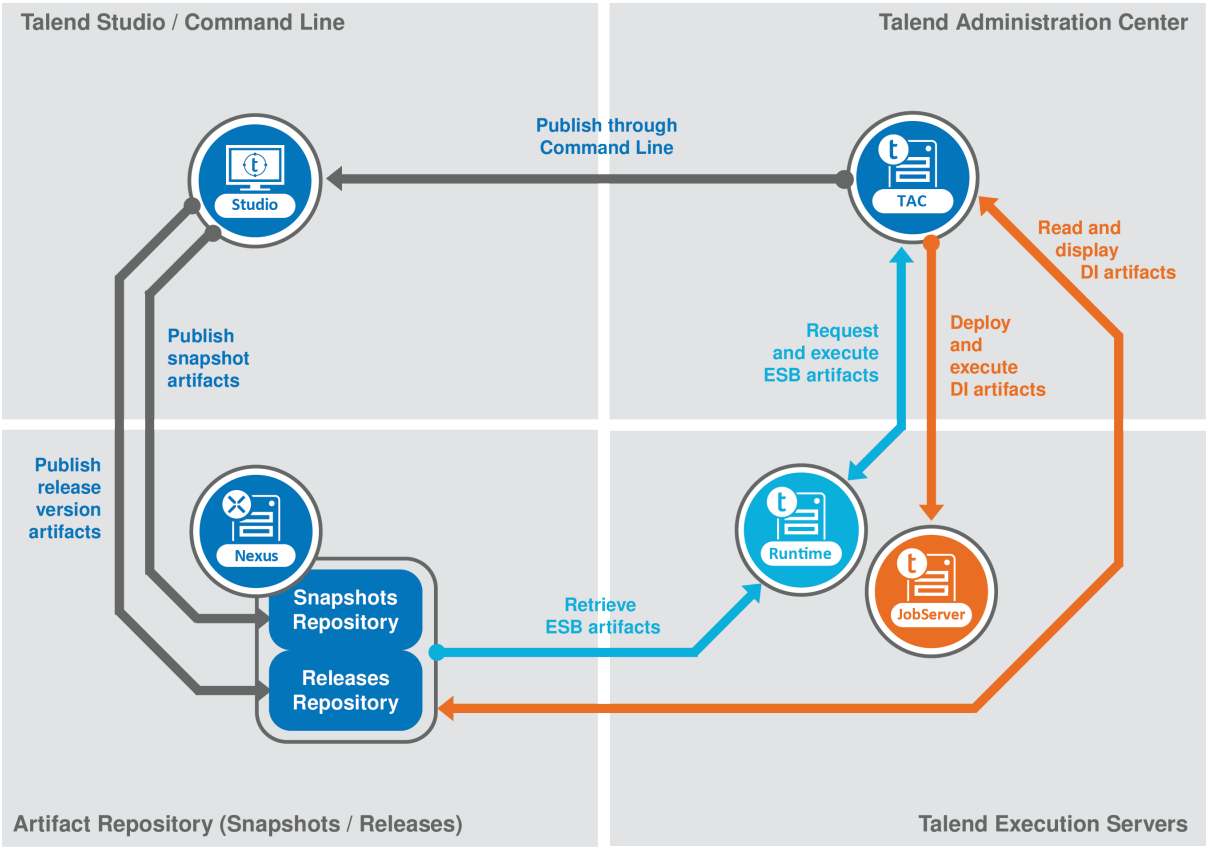
Talend Administration Center is connected to *Talend Studio* and to the local repository and the installed libraries are shared automatically.

For more information on how to install and configure the Nexus artifact repository, see *Installing and configuring the Nexus artifact repository*.

For more information on how to install external libraries, see *Installing external modules*.

A.1.4.3. Snapshots and Releases artifact repositories

The following image shows the architecture of the Snapshots and Releases repositories linked to the Studio, to an execution server and to *Talend Administration Center*.



The artifact repository is also used to store as artifacts all the Services, Routes and Jobs created in Studio or any Generic OSGi Feature created in any other Java IDE.

From the Studio, you can publish those artifacts in the *snapshots* and *releases* repositories (integrated within Nexus). The artifacts are provided to an execution server and then can be selected through *Talend Administration Center* in order to set their deployment.

Talend Administration Center also fetches the artifact status and URLs and display them in its page **Job Conductor** and **ESB Conductor**.

When the deployment of an artifact is initiated in *Talend Administration Center*, the execution server requests the corresponding artifact in the artifact repository. Then, the artifact can be deployed and executed.

Two embedded repositories are provided to store your artifacts:

- a *snapshots* repository to publish snapshot artifacts for development purposes,
- a *releases* repository to publish stable artifacts for production purposes.

The *Artifact-Repository-Nexus-VA.B.C.D.E* archive file containing Nexus is embedded in the *Talend Administration Center* archive file that is provided by *Talend*.

For more information on how to install and configure these artifact repositories, see *Installing and configuring the Nexus artifact repository*.

A.1.5. Talend Runtime

Talend Runtime (based on Apache Karaf) is an execution container in which you can deploy and execute all your Jobs stored on your Git or SVN repository.

It is also used to deploy and execute as OSGi bundles all your Services, Routes and Generic OSGi features that are retrieved from the **Provisioning** repositories.

For more information on the installation of *Talend Runtime*, see *Installing Talend Runtime*.

A.1.6. JobServer

JobServer is an application that allows a system installed on the same network as *Talend Administration Center* to declare itself as an execution server. These systems must obviously have a working JVM. For more information on the installation of **JobServers**, see *Installing and configuring your JobServers*.

A.1.7. Talend Studio

Talend Studio is a rich client that allows the user (such as a project manager, a developer or a DBA) to work on any **Talend** project for which he has authorization.

Talend Studio offers a comprehensive set of tools and functions for all its key capabilities including:

- Integration
- Activity monitoring Console
- Data profiling
- Data quality
- Mediation
- SoapUI

These tools are ALL accessible in different perspectives from one Studio.

The availability of perspectives in your Studio depends either on the license you have when you are working in a local project, or on the type of the remote project itself when you are working in remote projects.

For further information on user authorization on remote project, see the *Talend Administration Center User Guide*.

For further information about the different perspectives available in the studio, see the *Talend Studio User Guide*.

For more information on how to install *Talend Studio*, see *Installing your Talend Studio*.

A.1.8. Talend Activity Monitoring Console log database

If you want to use the *Talend Activity Monitoring Console*, an <*AMC*> log database must be created, which can be installed on any server. This <*AMC*> database will initially be empty. Its name may be modified, but you must take into account this modification in the rest of this document.

The <*AMC*> database will contain three tables that collect data allowing users to monitor Jobs. The three tables will collect data from the following components:

- **tFlowMeterCatcher**,
- **tLogCatcher**,
- **tStatCatcher**.

Instructions on how to create these tables and their structure is provided in the *Talend Activity Monitoring Console User Guide*.

A corresponding SQL user must be created and thus mapped to have access to this database. This user should be granted the "*create*" and "*update*" rights.

The installation of the <*AMC*> database is optional for the solutions that only hold the ESB option, and is only required when using the **tFlowMeterCatcher**, **tLogCatcher** and **tStatCatcher** components. These components can be used in Jobs (for example, ESB Consumer jobs) for REST and Soap ESB Service Providers. Only the **tLogCatcher** component is supported for now and Routes currently can not use these components.

For more information on how to install *Talend Activity Monitoring Console*, see *Installing the Talend Activity Monitoring Console web application*.

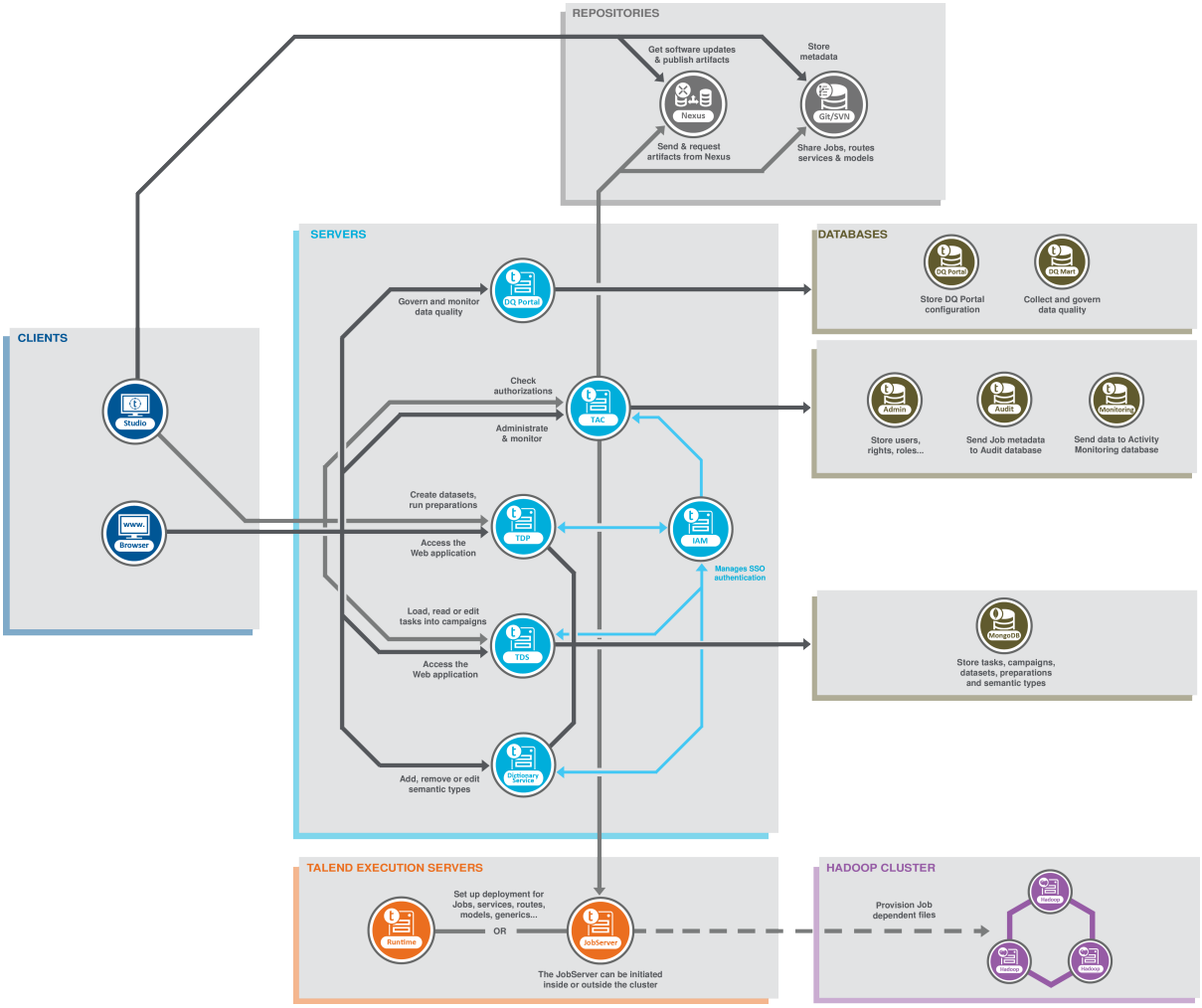
A.2. Architecture of the Talend products

The operating principles of the Talend products could be summarized as briefly as the following topics:

- building technical or business-related processes and data profiling analyses,
- administrating users, projects, access rights and processes and their dependencies,
- deploying and executing technical processes,
- monitoring the execution of technical processes.

Depending on your license, some of the functional blocks may not be available to you.

Each of the above topics can be isolated in different functional blocks and the different types of blocks and their interoperability can be described as in the following architecture diagram :



Building and administrating

- The **CLIENTS** block includes one or more Talend Studio APIs and Web browsers that could be on the same or on different machines.

  From the Studio API, end-users can carry out technical processes: data integration or data service processes, mediation routes and services, and publish them on the Artifact Repository and data profiling analyses and reports regardless of data volume and process complexity. The *Talend Studio* allows the user to work on any project for which he has authorization. For more information, see the *Talend Studio User Guide*.

  From a Web browser, end-users connect to the remotely based Administration Center through a secured HTTP protocol. The end-user category in this description may include developers, project managers, administrators and any other person involved in building data flows, Web, REST and data services, and mediation routes. Each of these end-users will use either the Studio or the Administration Center or both of them depending on the company policy.

  End-users also use a Web browser to connect to *Talend Data Quality Portal* that plugs to the data quality data mart and extends the dynamic reporting and monitoring capabilities of the *Talend Studio*, and to *Talend Data Stewardship Console* to resolve the data issues included in the listed tasks to reach the golden data record (if they subscribed to this option).

  Additionally, from the Web Browser you access the *Talend Data Preparation* Web application. This is where you import your data, from local files or other sources, and cleanse or enrich it by creating new preparations on this data. You can also access the *Talend Data Stewardship* Web application. This is where campaign owners and data stewards manage campaigns and tasks. You can optionally access the *Talend Dictionary Service* server to add, remove or edit the semantic types used on data in the Web applications.

- The **TALEND SERVERS** and **DATABASES** blocks and the **Git/SVN** grey circle include a web-based Administration Center (application server) connected to two shared repositories: one based on a Git or SVN server and one based on a database server (**Admin**).

  The Administration Center enables the management and administration of all projects. Administration metadata (user accounts, access rights and project authorization for example) is stored in the database server and project metadata (Jobs, Business Models, Routines, Routes, Services for example) is stored in the Git or SVN server (to easily share them between the different end-users).

  The Administration Center also enables to configure the tasks that handle job executions and triggers. It also looks after the job generation and deployment to the execution servers. For more information, see the *Talend Administration Center User Guide*.

  The Administration Center also includes the servers used by the Talend Web applications, namely *Talend Data Preparation* and *Talend Data Stewardship*, and also *Talend Dictionary Service*. The *Talend Identity and Access Management* server is used to enable Single Sign-On between those applications.

  Finally, the Administration Center enables you to access and manage the Routes or Services created from *Talend Studio* and published into the **Artifact Repository**, and set up and monitor their deployment and execution into the **Talend Runtime**. For more information, see the *Talend Administration Center User Guide*.

Deploying and executing

- The **Nexus** grey circle represents the artifact repository that stores all the:

  - **Software Updates** available for download.

  - Routes and Services that are published from the Studio and are ready to be deployed and executed in **Talend Runtime**.

- The **TALEND EXECUTION SERVERS** block represents the execution servers that run technical processes according to the execution scheduling set up in the *Talend Administration Center* Web application. Those execution servers can be of:

  - One or more **Talend Runtimes** (execution container) deployed inside your information system. The Talend Runtime deploys and executes the technical processes according to the set up defined in the *Talend Administration Center* Web application. Those processes are Jobs built from the Studio and centralized on the Git or SVN server. Routes and Services retrieved from the artifact repository.

    If you have several Talend Runtimes on which to deploy the Service and Route artifacts, you will be able to load balance their execution according to your needs. All instances of Talend Runtime will communicate between each other via the Service Locator to identify the one more likely to deploy and execute the artifact(s) set to deployment in *Talend Administration Center*. The Talend Runtime elected for the deployment will request for the artifact(s) to deploy and execute from the artifact repository and the artifact repository will thus send the artifact(s) requested along with all the dependencies needed for its/their execution to the Talend Runtime, that will deploy and execute it/them.

  - One or more JobServers deployed inside your information system that run technical processes (Jobs) according to scheduled time, date or event set in the *Talend Administration Center* Web application.

    The end-user can transfer technical processes to a remote execution server directly from the Studio (distant run).

    You must install the JobServer files ("Agent"), delivered by **Talend**, on each of the execution servers to become operational.

    For more information, see *Installing and configuring your JobServers*.

Monitoring

- The **Monitoring** circle represents the monitoring: the Activity Monitoring Console and the Service Activity Monitoring.

  The Activity Monitoring Console allows end-users to monitor the execution of technical processes. It provides detailed monitoring capabilities that can be used to consolidate log information collected, understand the interaction between underlying data flows, prevent faults that could be unexpectedly generated and support system management decisions. For more information on the Activity Monitoring Console, see the *Talend Activity Monitoring Console User Guide*.

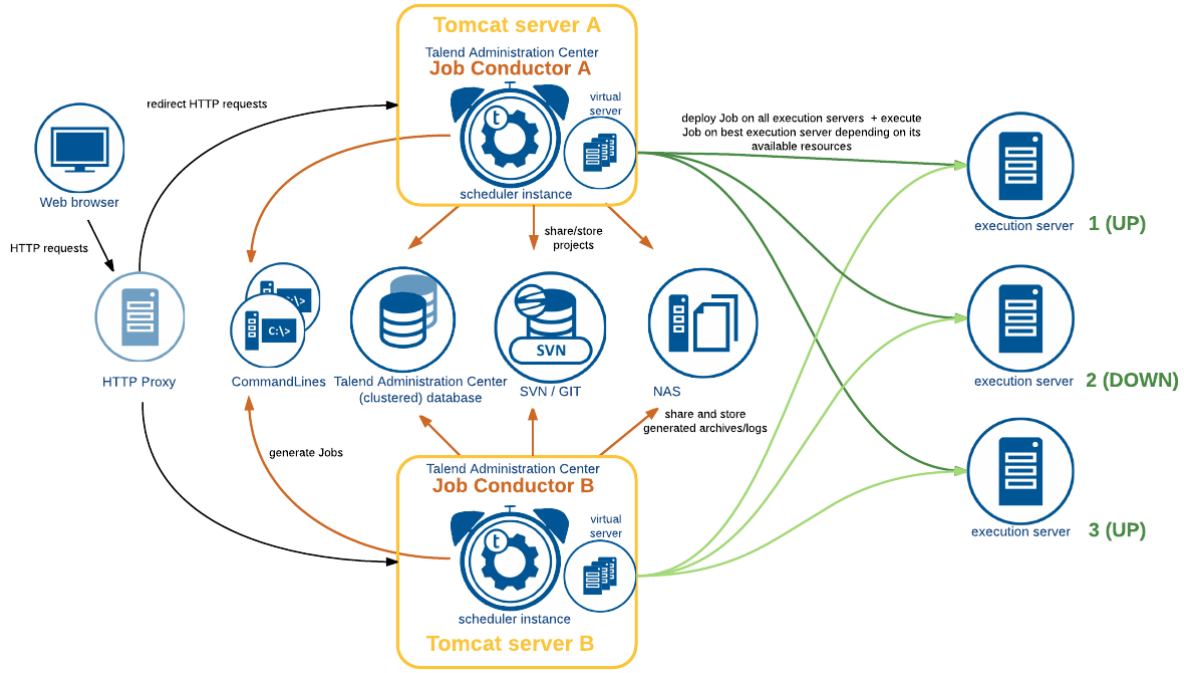
  The Service Activity Monitoring allows end-users to monitor service calls. It provides monitoring and consolidated event information that can be used to understand the underlying requests and replies that compose the event, monitor faults that may be unexpectedly generated and support the system management decisions. For more information on the Service Activity Monitoring, see the *Talend Administration Center User Guide*.

A.3. Talend High Availability

The availability of this feature depends on your license.

You can set up a cluster in your *Talend* system to provide high availability and failover features for task execution scheduling in *Talend Administration Center*. You do this by deploying multiple Job Conductors and Job execution servers on different machines.

High availability in this context refers only to the scheduling of task executions.



To summarize:

- Two application servers (Tomcat or JBoss) holding the *Talend Administration Center* Job Conductors and Virtual Servers, as well as two CommandLine applications are installed (on different machines) and point to the same SVN/GIT shared project.

- All instances of the application server are connected to the project administration database, which is presumably clustered.
- (optional) *Talend Administration Center* users are routed to the same active application instance, for example through an HTTP Proxy (switch). This feature is not provided by Talend and thus needs to be implemented separately.
- The first CommandLine generates the artifacts to be deployed. The second CommandLine is only used when the first one is down.
- When an execution server fails, the other execution servers can recover the interrupted tasks.
- A shared storage is implemented to store and share between active instances all archives and logs generated during each Job execution, for example through a Network-Attached Storage (NAS). This feature is not provided by Talend and thus needs to be implemented separately.

For more information about the failover and the various actions you can perform on a task when a server is unavailable, see the *Talend Administration Center User Guide*.

A.4. Cheatsheet: start and stop commands for Talend server modules

The following table sums up the commands or executables you can use to start and stop *Talend* server modules.

***Talend* server module**	**Start command/executable**	**Stop command/executable**
Tomcat service for *Talend Administration Center*, *Talend Data Quality Portal*, *Talend Data Stewardship Console*, *MDM Server* The *Talend Data Stewardship Console* is deprecated since Talend 6.4. Consider migrating to *Talend Data Stewardship*.	`sh <TomcatPath>/bin/startup.sh`	`sh <TomcatPath>/bin/shutdown.sh`
JBoss service for *Talend Administration Center*	`sh <JBossPath>/bin/run.sh`	`sh <JBossPath>/bin/shutdown.sh`
Talend Runtime	`<TalendRuntimePath>/bin/trun`	**Ctrl+C**
Nexus Artifact Repository	`nexus.sh console`	**Ctrl+C**
JobServer	`<JobServerPath>/start_rs.sh`	`<JobServerPath>/stop_rs.sh`
Talend Log Server	`sh <LogServerPath>/start_logserver.sh`	`sh <LogServerPath>/stop_logserver.sh`
CommandLine	`sh <CommandLinePath>/start_cmdline.sh` or `sh <CommandLinePath>\commandline-linux_x86_64.sh`[1]	`sh <CommandLinePath>/start_cmdline.sh` or **Ctrl+C**[1]
Talend ESB	`tesb:start-all`	`tesb:stop-all`
Event Logging	`tesb:start-el-default`	`tesb:stop-el-default`
Talend Runtime container	`<TalendESBPath>/container/bin/trun`	**Ctrl+C**
Apache ActiveMQ	In *Talend Runtime container*: `feature:install activemq`	**Ctrl+C**
Service Locator	`tesb:start-locator`	`tesb:stop-locator`
Monitoring Server	`tesb:start-sam`	`tesb:stop-sam`
Security Token Service	`tesb:start-sts`	`tesb:stop-sts`

1: The command/executable to use depends whether you installed your Talend product using manual installation or using automatic installation.

A.5. Installing Talend servers as Linux services

The following pages contain procedures on how to install *Talend* servers as Linux services.

- *Installing CommandLine as a service*
- *Installing JobServer as a service*
- *Installing Tomcat as a service*
- *Installing Talend Runtime as a service*
- *Installing Nexus as a service*
- *Installing Talend Log Server as a service*

A.5.1. Installing CommandLine as a service

Installing CommandLine as a service on Ubuntu

1. Create an executable from which commandline can be run in the directory */usr/bin*.

   To do this, you need to create two files, *commandline_start* and *commandline_stop* containing the following:

   For *commandline_start*:

   ```
   #!/bin/sh
   cd <CommandLinePath>
   nohup sh commandline-linux.sh&
   ```

   The path <CommandLinePath> should be absolute.

   For *commandline_stop*:

   ```
   #!/bin/sh
   (echo stopServer ; sleep 2) | telnet localhost 8002
   ```

2. Ensure that the two files above are executable. To do this, you can execute the two commands below in the */usr/bin* directory:

   ```
   # chmod +x /usr/bin/commandline_start

   # chmod +x /usr/bin/commandline_stop
   ```

3. Paste the *sh* file: /<CommandLinePath> */addons/scripts/talend_commandline* into the directory: */etc/init.d* in order to create the service related to the two executables above.

4. Make the file executable using:

   ```
   # chmod +x /etc/init.d/talend_commandline_commandline
   ```

5. Execute the following command:

   ```
   # update-rc.d talend_commandline_commandline defaults 60
   ```

Now you have created the service, related to the two executable files *commandline_start* and *commandline_stop*.

Installing the service on RedHat/CentOS 7 Systems

All the following commands have to be executed with super-user privileges.

1. Create the service file with the following command:

   ```
   touch /etc/systemd/system/Talend-CommandLine.service
   ```

2. Assign the relevant rights to the file you created:

   ```
   chmod 664 /etc/systemd/system/Talend-CommandLine.service
   ```

3. Paste the following content in the file while adapting it to your configuration:

   ```
   [Unit]
   Description=Talend CommandLine Service
   After=network.target
   [Service]
   WorkingDirectory=<CommandLinePath>
   ExecStart=/bin/bash commandline-linux_x86_64.sh start
   Type=simple
   [Install]
   WantedBy=default.target
   ```

4. Reload the service daemon:

   ```
   systemctl daemon-reload
   ```

5. Start the service:

   ```
   systemctl start Talend-CommandLine.service
   ```

Installing the service on RedHat/CentOS 6 Systems

1. Create/Copy the following script to the */etc/init.d/commandline* file:

   ```
   #!/bin/sh
   ### BEGIN INIT INFO
   # Default-Start:     2 3 4 5
   # Default-Stop:      S 0 1 6
   # Short-Description: Example initscript
   # Description:       This file should be used to construct scripts to be placed in /
   etc/init.d.
   ### END INIT INFO

   # Author: jsd03

   # Do NOT "set -e"

   # PATH should only include /usr/* if it runs after the mountnfs.sh script
   PATH=/usr/sbin:/usr/bin:/sbin:/bin
   COMMANDLINE=/oa/talend/commandline303
   CMD_WORKSPACE=$COMMANDLINE"/commandline-workspace"
   STARTUP=commandline.sh
   USER=cxp
   NAME=commandline
   PORT=10004


   # Read configuration variable file if present
   [ -r /etc/default/$NAME ] && . /etc/default/$NAME

   # Load the VERBOSE setting and other rcS variables
   [ -f /etc/default/rcS ] && . /etc/default/rcS

   # Define LSB log_* functions.
   ```

```
# Depend on lsb-base (>= 3.0-6) to ensure that this file is present.
. /lib/lsb/init-functions

#
# Function that starts the daemon/service
#
do_start()
{
echo -n $"Starting commandline service: "
 su - $USER -c "rm -Rf $CMD_WORKSPACE"
 su - $USER -c "cd $COMMANDLINE && screen -dmS cmdLine ./$STARTUP"
#screen -dmS cmdLine $DAEMON_START
 RETVAL=$?
 echo

}
```

```
#
# Function that stops the daemon/service
#
do_stop()
{
 echo -n $"Stopping commandline service: "
 su - $USER -c "( echo 'stopServer' ; sleep 2 ) | telnet localhost $PORT"
 RETVAL=$?
 echo

}

case "$1" in
start)
do_start
 ;;
stop)
 do_stop
 ;;
*)
 echo $"Usage: $0 {start|stop}"
 exit 1
esac

exit 0
```

2. Edit the *USER* and *COMMANDLINE* variable values in the script (with the dedicated user to run Talend, and the CommandLine path respectively).
3. To make sure that the script is executable, type:

   ```
   # chmod 0755 /etc/init.d/commandline
   ```

4. Type in:

   ```
   chkconfig --list

   chkconfig --add commandline
   ```

   Now, you can test this self-defined service.

A.5.2. Installing JobServer as a service

Installing JobServer as a service on RedHat/CentOS 7 Systems

All the following commands have to be executed with super-user privileges.

1. Create the service file with the following command:

   ```
   touch /etc/systemd/system/Talend-JobServer.service
   ```

2. Assign the relevant rights to the file you created:

   ```
   chmod 664 /etc/systemd/system/Talend-JobServer.service
   ```

3. Paste the following content in the file while adapting it to your configuration:

   ```
   [Unit]
   Description=Talend JobServer Service
   After=network.target

   [Service]
   WorkingDirectory=<JobServerPath>
   ExecStart=/bin/bash start_rs.sh
   ExecStop=/bin/bash stop_rs.sh
   Type=simple

   [Install]
   WantedBy=default.target
   ```

4. Reload the service daemon:

   ```
   systemctl daemon-reload
   ```

5. Start the service:

   ```
   systemctl start Talend-JobServer.service
   ```

Installing JobServer as a service on RedHat/CentOS 6

1. Create/Copy the following script to the */etc/init.d/jobserver* file:

   ```
   # chkconfig: 345 91 10
   # description: Starts and stops the jobserver daemon.
   #

   # Source function library.
   . /etc/rc.d/init.d/functions

   # Get config.
   . /etc/sysconfig/network
   ```

   ```
   # Check that networking is up.
   [ "${NETWORKING}" = "no" ] && exit 0

   user=cxp
   jobserver=/u/bin/Talend/jobserver_3.0.1
   startup=start_rs.sh
   shutdown=stop_rs.sh


   start(){
    echo -n $"Starting jobserver service: "
    su - $user -c "cd $jobserver && sh $startup &"
    RETVAL=$?
    echo
   }

   stop(){
    echo -n $ "Stopping jobserver service: "
    su - $user -c  "cd $jobserver && sh $shutdown"
    RETVAL=$?
    echo
   ```

```
}

restart() {
 stop
 start
}

# See how we were called.
case "$1" in
start)
 start
 ;;
stop)
 stop
 ;;
restart)
 restart
 ;;
*)
 echo $"Usage: $0 {start|stop|restart}"
 exit 1
esac

exit 0
```

2. Edit the *user* and *jobserver* variable values in the script (with the dedicated user to run Talend, and the JobServer path respectively).
3. To make sure that the script is executable, type:

   ```
   # chmod 0755 /etc/init.d/jobserver
   ```

4. Type in:

   ```
   chkconfig --list

   chkconfig --add jobserver
   ```

   Now, you can test this self-defined service.

Installing JobServer as a service on Ubuntu

1. Create/Copy the following script to the */etc/init.d/jobserver* file as explained in Installing JobServer as a service on RedHat/CentOS 6.
2. Edit the *user* and *jobserver* variable values in the script (with the dedicated user to run Talend, and the JobServer path respectively).
3. To make sure that the script is executable, type:

   ```
   # chmod 0755 /etc/init.d/jobserver
   ```

4. Execute the following command:

   ```
   # update-rc.d jobserver defaults 60
   ```

Install JobServer as a service on OpenSuse

The following procedure needs to be performed with root privileges.

This procedure was tested on OpenSuse 11.2 x64 bits.

1. Make sure that the three scripts *jobserver_start*, *jobserver_stop* and *jobserver* are executable.

2. Copy *usr/bin/jobserver_start* and *usr/bin/jobserver_stop* into:

   /usr/bin/

3. Copy *etc/ini.d/jobserver* in:

   /etc/init.d/

4. Edit the configuration file *etc/sysconfig/jobserver* and set the path to your installation directory.

5. Copy this file into:

   /etc/sysconfig/

6. Execute the following command to create a link called *rcjobserver*:

   ```
   ln -s /etc/init.d/jobserver /usr/sbin/rcjobserver
   ```

7. To start or stop the JobServer manually, use:

   ```
   rcjobserver start
   ```

   ```
   rcjobserver stop
   ```

8. Install the service using:

   ```
   Yast > System > System Services
   ```

9. Type in:

   ```
   chkconfig -e jobserver
   ```

10. Set the variable to *ON*

11. Run SuSEconfig.

The JobServer installation path can be edited through *Yast > /etc/sysconfig Editor* in *Applications/Talend*.

A.5.3. Installing Tomcat as a service

Installing the service on RedHat/CentOS 7 Systems

All the following commands have to be executed with super-user privileges.

1. Create the service file with the following command:

   ```
   touch /etc/systemd/system/tomcat.service
   ```

2. Assign the relevant rights to the file you created:

   ```
   chmod 664 /etc/systemd/system/tomcat.service
   ```

3. Paste the following content in the file while adapting it to your configuration:

```
[Unit]
Description=Apache Tomcat Web Application Container
After=syslog.target network.target

[Service]
Type=forking
```

```
Environment=JAVA_HOME=/usr/lib/jvm/jre
Environment=CATALINA_PID=/opt/tomcat/temp/tomcat.pid
Environment=CATALINA_HOME=/opt/tomcat
Environment=CATALINA_BASE=/opt/tomcat
Environment='CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC'
Environment='JAVA_OPTS=-Djava.awt.headless=true -Djava.security.egd=file:/dev/./
urandom'

ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/bin/kill -15 $MAINPID

[Install]
WantedBy=multi-user.target
```

4. Reload the service daemon:

```
systemctl daemon-reload
```

5. Start the service:

```
systemctl start tomcat.service
```

Installing the service on RedHat/CentOS 6 and Ubuntu Systems

1. Create/Copy the following script to the */etc/init.d/tomcat* file:

```
# chkconfig: 345 91 10
# description: Starts and stops the Tomcat daemon.
#

# Source function library.
. /etc/rc.d/init.d/functions

# Get config.
. /etc/sysconfig/network

# Check that networking is up.
[ "${NETWORKING}" = "no" ] && exit 0

user=cxp
tomcat=/u/bin/Tomcat/apache-tomcat-8.0.33/
startup=$tomcat/bin/startup.sh
shutdown=$tomcat/bin/shutdown.sh
#export JAVA_HOME=/usr/local/java

status(){
ps ax --width=1000 | grep "[o]rg.apache.catalina.startup.Bootstrap start"
| awk '{printf $1 " "}' | wc | awk
'{print $2}' > /tmp/tomcat_process_count.txt
read line < /tmp/tomcat_process_count.txt
if [ $line -gt 0 ]; then
echo -n "tomcat ( pid "
ps ax --width=1000 | grep "[o]rg.apache.catalina.startup.Bootstrap start"
| awk '{printf $1 " "}'
echo -n ") is running..."
echo
else
echo "Tomcat is stopped"
fi
}

start(){
 echo -n $"Starting Tomcat service: "
 #daemon -c
 su - $user -c "$startup"
 RETVAL=$?
```

```
 echo
}

stop(){
 action $"Stopping Tomcat service: " su - $user "$shutdown"
 RETVAL=$?
 echo
}

restart(){
  stop
  start}
```

```
# See how we were called.
case "$1" in
start)
 start
 ;;
stop)
 stop
 ;;
status)

 status tomcat
 ;;
restart)
 restart
 ;;
*)
 echo $"Usage: $0 {start|stop|status|restart}"
 exit 1
esac

exit 0
```

2. Edit the *user* and *tomcat* variable values in the script to match your configuration.
3. To make sure that the script is executable, type:

   ```
   # chmod 0755 /etc/init.d/tomcat
   ```

4. Type in:

   ```
   chkconfig --list

   chkconfig --add tomcat
   ```

   Now, you can test this self-defined service.

A.5.4. Installing Talend Runtime as a service

The Talend Runtime container is based on Apache Karaf. Karaf Wrapper (for service wrapper) makes it possible to install the Talend Runtime container as a service.

To install Talend Runtime as a service, you first have to install the wrapper, which is an optional feature:

Installing the wrapper

1. Browse to the *bin* folder of the Talend Runtime installation directory, then launch the container by executing the *trun* file as root.
2. To install the wrapper feature, type:

```
karaf@trun> feature:install wrapper
```

Once installed, wrapper feature will provide `wrapper:install` new command in the *trun*, which allows you to install Talend Runtime as a service.

3. To install the service, type in the following command:

```
karaf@trun> wrapper:install
```

Alternatively, to register the container as a service in automatic start mode, simply type:

```
karaf@trun> wrapper:install -s AUTO_START -n TALEND-CONTAINER -d Talend-Container -
D "Talend Container Service"
```

where `TALEND-CONTAINER` is the name of the service, `Talend-Container` is the display name of the service and `"Talend Container Service"` is the description of the service.

Here is an example of `wrapper:install` command executing on Linux:

```
karaf@trun()> feature:install wrapper
karaf@trun()> wrapper:install -s AUTO_START -n TALEND-CONTAINER \
-d Talend-Container -D "Talend Container Service"
Creating file: <TalendRuntimePath>/bin/TALEND-CONTAINER-wrapper
Creating file: <TalendRuntimePath>/bin/TALEND-CONTAINER-service
Creating file: <TalendRuntimePath>/etc/TALEND-CONTAINER-wrapper.conf
Creating file: <TalendRuntimePath>/lib/libwrapper.so
Creating file: <TalendRuntimePath>/lib/karaf-wrapper.jar
Creating file: <TalendRuntimePath>/lib/karaf-wrapper-main.jar
Setup complete. You may want to tweak the JVM properties in the wrapper
configuration file:
<TalendRuntimePath>/etc/TALEND-CONTAINER-wrapper.conf
before installing and starting the service.
```

The wrapper files are installed, you now have to install the Talend Runtime service.

In the following procedure, `TALEND-CONTAINER` is the name of the service and is only given as an example. Note also that <TalendRuntimePath> is the Talend Runtime installation directory.

Installing the service on RedHat/CentOS 7 Systems

All the following commands have to be executed with super-user privileges.

1. Create the service file with the following command:

```
touch /etc/systemd/system/Talend-Container.service
```

2. Assign the relevant rights to the file you created:

```
chmod 664 /etc/systemd/system/Talend-Container.service
```

3. Paste the following content in the file while adapting it to your configuration:

```
[Unit]
Description=Talend Runtime Service
After=network.target

[Service]
ExecStart=<TalendRuntimePath>/bin/trun
Type=simple

[Install]
WantedBy=default.target
```

4. Reload the service daemon:

```
systemctl daemon-reload
```

5. Start the service:

```
systemctl start Talend-Container.service
```

Installing the service on Redhat/CentOS 6 Systems

- To install the service:

```
$ ln -s /<TalendRuntimePath>/bin/TALEND-CONTAINER-service /etc/init.d/
$ chkconfig TALEND-CONTAINER-service --add
```

- To start the service when the machine is rebooted:

```
$ chkconfig TALEND-CONTAINER-service on
```

- To disable starting the service when the machine is rebooted:

```
$ chkconfig TALEND-CONTAINER-service off
```

- To start the service:

```
$ service TALEND-CONTAINER-service start
```

- To stop the service:

```
$ service TALEND-CONTAINER-service stop
```

- To uninstall the service:

```
$ chkconfig TALEND-CONTAINER-service --del
$ rm /etc/init.d/TALEND-CONTAINER-service
```

Installing the service on Linux Ubuntu distributions

- To install the service:

```
$ ln -s /<TalendRuntimePath>/bin/ TALEND-CONTAINER-service /etc/init.d/
```

- To start the service when the machine is rebooted:

```
$ update-rc.d TALEND-CONTAINER-service defaults
```

- To disable starting the service when the machine is rebooted:

```
$ update-rc.d -f TALEND-CONTAINER-service remove
```

- To start the service:

```
$ /etc/init.d/TALEND-CONTAINER-service start
```

- To stop the service:

```
$ /etc/init.d/TALEND-CONTAINER-service stop
```

- To uninstall the service:

```
$ rm /etc/init.d/TALEND-CONTAINER-service
```

A.5.5. Installing Nexus as a service

To install Nexus artifact repository as a service, proceed as follows:

1. As root, copy *Artifact-Repository-Nexus-VA.B.C.D.E/nexus-A.B.C-D/bin/nexus* to */etc/init.d*.
2. Make the *nexus* script executable by running the following command:

   ```
   # chmod 755 /etc/init.d/nexus
   ```

3. Edit the *nexus* script by changing the following values:
   - Change *NEXUS_HOME* to the absolute folder location, for example *NEXUS_HOME="usr/local/nexus"*.
   - Set the *RUN_AS_USER* option to a user with restricted rights that you want to use to run the service. Note that you should not be running Nexus as root.
   - Change *PIDDIR* to a directory where the user has read/write permissions, for example *./*.
4. Change the owner and group of your Nexus-related directories to the user that you want to use to run the service.
5. From the commandline, browse to */etc/init.d*.
6. On Ubuntu, install the service by running the following command:

   ```
   # update-rc.d nexus defaults
   ```

   On other Linux distributions, use the following commands:

   ```
   $ chkconfig --add nexus
   $ chkconfig --levels 345 nexus on
   ```

A.5.6. Installing Talend Log Server as a service

To install Talend Log Server as a service, proceed as follows:

Installing Log Server as a service on RedHat/CentOS 7 Systems

All the following commands have to be executed with super-user privileges.

1. Create the service file with the following command:

   ```
   touch /etc/systemd/system/Talend-LogServer.service
   ```

2. Assign the relevant rights to the file you created:

   ```
   chmod 664 /etc/systemd/system/Talend-LogServer.service
   ```

3. Paste the following content in the file while adapting it to your configuration:

   ```
   [Unit]
   Description=Talend Log Server Service
   After=network.target

   [Service]
   WorkingDirectory=<LogServerPath>
   ExecStart=/bin/bash start_logserver.sh
   ExecStop=/bin/bash stop_logserver.sh
   Type=simple
   ```

```
[Install]
WantedBy=default.target
```

4. Reload the service daemon:

```
systemctl daemon-reload
```

5. Start the service:

```
systemctl start Talend-LogServer.service
```

Installing Log Server as a service on RedHat/CentOS 6 and Ubuntu Systems

1. Create a script from which Talend Log Server can be run in the directory */etc/init.d/tlogserver*, like the following:

```
#!/bin/sh
#
# tlogserver: this script starts and stops the monolithic jar
#
# chkconfig: - 85 15
# description: logstash is an open source log management system.
# processname: tlogstash
# config: %%%LOGSERV_CONFIG%%%
# binary: %%%LOGSERV_JAR%%%
prog=tlogserver
PATH=%%%INSTALLDIR%%%/logserv:/sbin:/bin:/usr/sbin:/usr/bin
NAME=tlogserver

test -x $DAEMON || exit 0

set -e

start() {
        echo -n $"Starting $prog: "
        %%%INSTALLDIR%%%/logserv/start_logserver.sh
}

stop() {
        echo -n $"Stopping $prog: "
        %%%INSTALLDIR%%%/logserv/stop_logserver.sh
}


case "$1" in
  start)
        start
        ;;
  stop)
        stop
        ;;
  restart)
        stop
        start
        ;;
     *)
            N=/etc/init.d/$NAME
            echo "Usage: $N {start|stop|restart}" >&2
            exit 1
            ;;
    esac

    exit 0
```

2. Ensure that the file above is executable. To do this, you can execute the commands below in the */etc/init.d/tlogserver* directory:

```
# chmod +x /etc/init.d/tlogserver
```

3. Execute the following command to activate the startup script:

```
# update-rc.d tlogserver defaults 60
```

A.6. H2 Database Administration & Maintenance

This Chapter provides information about how to manage and back up the H2 embedded database.

For more information about how to use the H2 database and web console, refer to the H2 database documentation at http://www.h2database.com.

A.6.1. About H2 embedded database

H2 is a relational database management system written in Java. It can be embedded in Java applications or run in the client-server mode.

This database is the default solution embedded in *Talend Administration Center* to store all cross-project information such as users, authorizations, projects...

If you do not want to use this default database, you can set up a database server (MySQL, MSSQL or Oracle).

The benefits of using this H2 embedded database is that it simplifies the installation process.

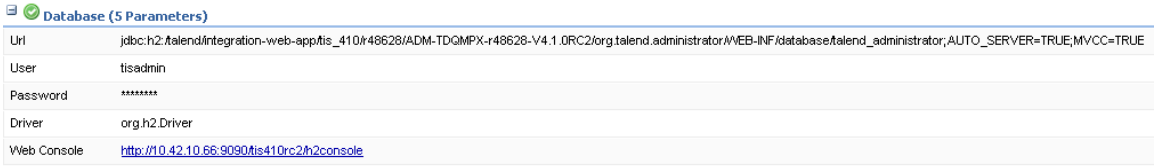
A.6.2. Administrating the H2 database through the Web console

To help you administrate the H2 embedded database, a dedicated Web console is available directly from *Talend Administration Center*.

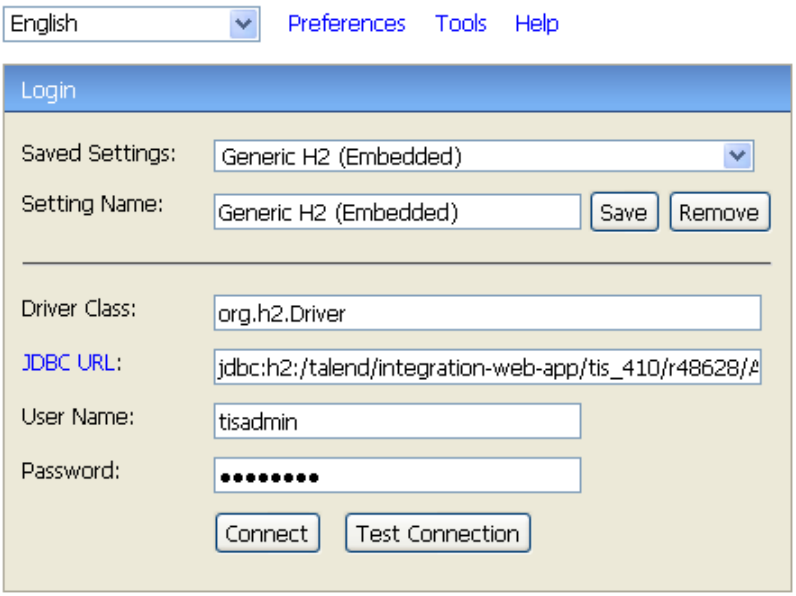
A.6.2.1. Connecting to the H2 Web Console

From *Talend Administration Center*, you can access the H2 administration console:

1. From the main **Menu**, click **Configuration** to access the **Configuration** page.
2. On the **Configuration** page, expand the **Database** node to display the parameters.



3. In the **Web Console** field, click the link to access the H2 Web Console.
4. The H2 Web Console's **Login** page displays:



5. In the **User Name** and **Password** fields, type in the connection login and password to the database, by default talend_admin and talend_admin.

6. The **JDBC URL** field reads by default:

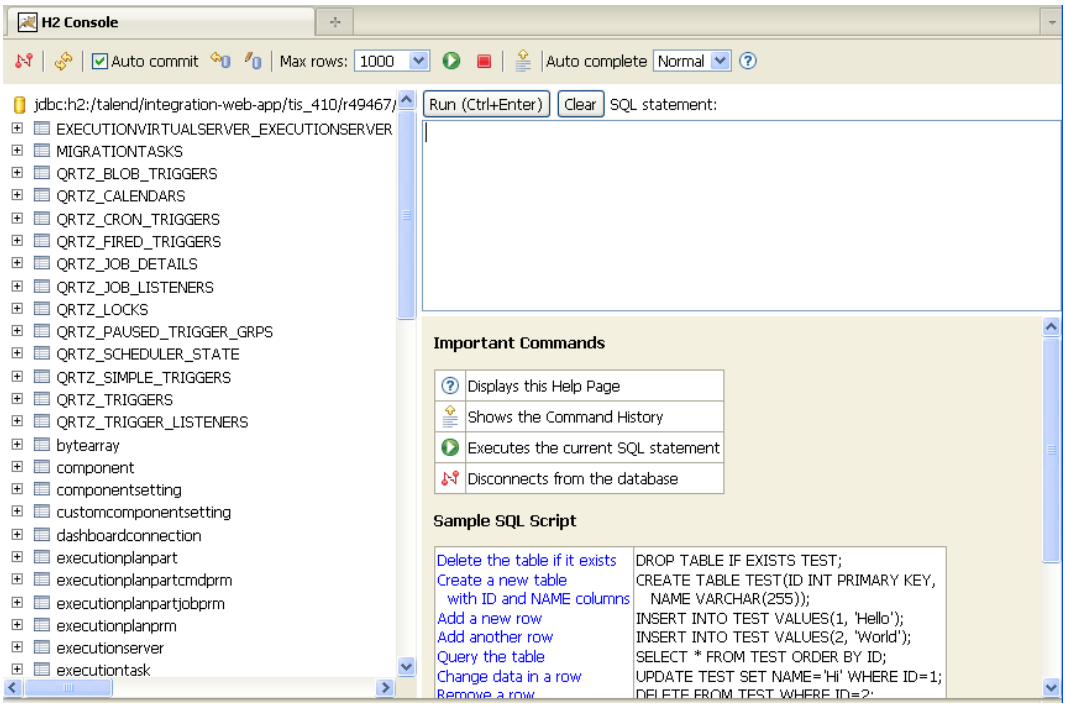
   jdbc:h2:/<ApplicationPath>/WEB-INF/database/
   talend_administrator;AUTO_SERVER=TRUE;MVCC=TRUE;LOCK_TIMEOUT=15000

   where <*ApplicationPath*> is the location where *org.talend.administrator* was deployed.

   If you have moved the H2 embedded database location, then fill out the JDBC URL field with the updated URL information. Prior to clicking Connect, click the Test Connection button in order to check the new URL. In case of a mistyped URL, the JDBC URL will revert back to the original URL information.

7. Click **Connect**.

   The Web database administration page displays.



For more information about H2 use and troubleshooting, please refer to the H2 online documentation on http://www.h2database.com.

A.6.2.2. Backing up the H2 database

The configuration parameters of the H2 database backup is already set by default so that the backup occurs on an daily basis.

If you need or want to make edits to this setting, edit the configuration file:

<ApplicationPath>/WEB-INF/classes/configuration.properties

The cron-based backup of the embedded database triggers everyday at 3.45am all year round. The syntax reads as follows "*Seconds Minutes Hours Day-of-month Month Day-of-week Year*", such as for example:

`0 45 3 ? * * *` (default setting - trigger every day at 3.45am)
`0 45 5 ? * MON-FRI` (every Monday, Tuesday, Wednesday, Thursday and Friday at 5.45 am)

More examples are available on http://www.quartz-scheduler.org/documentation/quartz-2.2.x/tutorials/tutorial-lesson-06.html.

Other automatic backups are performed at startup and shutdown of the application server:

```
database.embedded.backup.doBackupAtStartup=true
database.embedded.backup.doBackupAtShutdown=true
```

The backup files are stored at the following location, up to the 30 latest backups:

<ApplicationPath>/WEB-INF/database/backups

A.6.3. Setting up the H2 database for access from other machines

To allow other users to access the H2 database for centralized storage of cross-project information, do the following:

1. Stop Tomcat service if it is running.
2. Unzip your H2 database server package to any of your local drives. The latest H2 database server package is available at http://www.h2database.com/html/download.html.
3. Open a CMD window, navigate to the drive where the H2 database server package was unzipped, and change directory to *h2\bin*, which contains the *h2*.jar* file.
4. Start the H2 server as a service using the following command:

   ```
   java -cp h2*.jar org.h2.tools.Server -tcp -tcpAllowOthers
   -tcpPort <port_number>
   ```

Now other users can access the H2 database, but you still need to edit the database URL to make *Talend Administration Center* work.

To do so, proceed as follows:

1. Open the *configuration.properties* file in the *<ApplicationPath>/WEB-INF/Classes* folder, and edit the H2 database URL setting as follows:

```
database.url=jdbc:h2:tcp://<IP_address>:<port_number>/file:<ApplicationPath>/WEB-
INF/database/talend_administrator;AUTO_SERVER=TRUE;IFEXISTS=TRUE;MVCC=TRUE;
LOCK_TIMEOUT=15000
```

where <*IP_address*> is your IP address, <*port_number*> is the TCP port number specified in the command used to start the H2 server, and <*ApplicationPath*> is the location where *org.talend.administrator* was deployed.

2. Start the Tomcat service.
3. Start your *Talend Administration Center* Web application.

Now others can access and use the H2 database through the URL address.

A.7. Supported Third-Party System/Database/ Business Application Versions

This document provides the information about the versions of the systems or databases or business applications supported by Talend Studio.

A.7.1. Supported systems, databases and business applications by Talend components

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

The access to these systems, databases and business applications varies depending on the Studio you are using.

Systems/Databases	Versions	OS	Available with...
Alfresco	2.1	N/A[1]	All **Talend** products
Amazon Redshift	Initial release of Amazon Redshift	N/A[1]	All **Talend** products
AS/400	V5R2 to V5R4 V5R3 to V6R1 V6R1 to V7R2	N/A[1]	All **Talend** products
Access[2]	2003 2007	Windows	**Talend** products with Data Integration (DI), Master Data Management (MDM), Enterprise Service Bus (ESB) or Big Data
Bonita	5.2.3 5.3.1 5.6.1 5.10.1 6.5.2 7.2.4	N/A[1]	All **Talend** products
Cassandra	2.0.0 3.0/3.1/3.2/3.3/3.4	Windows + Linux	**Talend** products with Big Data

Systems/Databases	Versions	OS	Available with...
	(Deprecated versions: 1.1.2/1.2.2)		
CouchBase	2.0	Windows	**Talend** products with Big Data
CouchDB	1.0.2	Windows	**Talend** products with Big Data
DB Generic	ODBC	Windows	All **Talend** products
DB2	10.5 10.1	Windows + Linux	**Talend** components with all products. **Talend** products with MDM or ESB.
DynamoDB	No specified version	N/A[1]	**Talend** products with Big Data
EXASolution	6.0 and earlier	Windows	**Talend** products with DI, MDM, ESB or Big Data
Elasticsearch	Until 2.3.X	N/A[1]	**Talend** products with Big Data
FireBird	2.1	Windows + Linux	**Talend** products with DI, MDM, ESB or Big Data
Greenplum	4.2.1.0	Windows (client only) + Linux	**Talend** products with DI, MDM, ESB or Big Data
HSQLDb	1.8.0	N/A[1]	**Talend** products with DI, MDM, ESB or Big Data
Informix	11.50	Windows + Linux	All **Talend** products
Ingres	9.2	Windows + Linux	All **Talend** products
Interbase	7 and above	N/A[1]	**Talend** products with DI, MDM, ESB or Big Data
JavaDB	6	Windows + Linux	**Talend** products with DI, MDM, ESB or Big Data
Kafka	0.8.2.0 0.9.0.1[3] 0.10.0.1[3]	Windows + Linux	**Talend** products with Big Data
LDAP	No version limitation	Windows + Linux	All **Talend** products
Microsoft AX	Dynamics AX 4.0 Dynamics AX 2012	N/A[1]	All **Talend** products
Microsoft CRM	4.0 2011 2013 2015 2016	N/A[1]	All **Talend** products
MS SQL Server	2000 2003 2005 2008 2012 2014 [4] 2016 [4]	Windows + Linux	All **Talend** products
MaxDB	7.6	N/A[1]	**Talend** products with DI, MDM, ESB or Big Data
MongoDB	2.5.X 2.6.X	Windows + Linux	**Talend** products with Big Data

Systems/Databases	Versions	OS	Available with...
	3.0.X 3.2.X		
MySQL	Mysql4 Mysql5 MariaDB	Windows + Linux	All **Talend** products
Netezza	7.2	Windows + Linux	All **Talend** products
NetSuite	2014 2016	Windows + Linux	All **Talend** products
Neo4j	1.X.X 2.X.X/2.2.X/2.3	Linux	**Talend** products with Big Data
OleDb	2000 2003 2005 2007 2010	N/A[1]	All **Talend** products
Oracle	Oracle 8i/9i/10g/11g/11g (11.6)/12c	Windows + Linux	All **Talend** products
ParAccel	3.1 3.5	N/A[1]	**Talend** products with DI, MDM, ESB or Big Data
PostgreSQL	9.X	Windows + Linux	All **Talend** products
PostgresPlus	9.X	Windows + Linux	**Talend** products with DI, MDM, ESB or Big Data
Red Hat BRMS	6.1	Windows + Linux	**Talend** products with DI, MDM, ESB or Big Data
Salesforce	V39 and earlier	Windows + Linux	All **Talend** products
SAP	ECC 6.0 EhP6	Windows	All **Talend** products
SAP BW	7.3 7.4 7.5	Windows	All **Talend** products
SAP Hana	1.0	Windows	All **Talend** products
SAS	9.1 9.2	Windows + Linux	**Talend** products with DI, MDM, ESB or Big Data
SQLite	3.6.7	Windows + Linux	All **Talend** products
Sybase	12.5 12.7 15.2 15.5 15.7 16.0	Windows + Linux	All **Talend** products
SybaseIQ	12.5 12.7	Windows + Linux	All **Talend** products

Systems/Databases	Versions	OS	Available with...
	15.2 16.0		
Teradata	12 13 14 15	Windows + Linux	All **Talend** products
VectorWise	2	Windows + Linux	**Talend** products with DI, MDM, ESB or Big Data
Vertica	3 3.5 4 4.1 5.0 5.1 6.0 6.1.X 7.0.X 7.1.X	Windows + Linux	**Talend** products with DI, MDM, ESB or Big Data
VtigerCRM	Vtiger 5.0 Vtiger 5.1	N/A[1]	All **Talend** products

1. The test information is not available yet.

2. When working with Java 8, only the General collation mode is supported.

3. For information about the security options supported by the Kafka components, see Talend Help Center.

4. No new feature introduced by MS SQL Server 2014/2016 is supported.

A.7.2. Supported Hadoop distribution versions for Talend Jobs

- ✅: officially supported.
- ❎: not officially supported.
- 🔒: the Kerberos kinit option is supported by the Studio.
- 🔒🔒: the Kerberos kinit option and the Kerberos keytab option are both supported by the Studio.

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

If support for the Hadoop distribution you want to use is not yet available in your *Talend Studio*, it may be available via an update. For related information, see Talend Help Center.

		HBase	HCatalog	HDFS	Hive[5]		Oozie	Pig	Spark		Sqoop	Talend MapReduce
					Hive1 [2]	Hive2 [2]			Standalone	YARN		
Google Dataproc	V1.1	✗	✗	✗	✗	✓	✗	✗	✗	✓	✗	✓
HDP	V1.2.0 (Deprecated)	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
	V1.3.0 (Deprecated)	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
	V2.0.0 (Deprecated)	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
	V2.1.0 (Deprecated)[3]	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓
	V2.2.0 (Deprecated)[3]	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
	V2.3.2 (Deprecated)	✓	✓	✓	✗	✓	✓	✓	✗	✓	✓	✓
	V2.4.0	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓
	V2.5.0	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓
	V2.6.0	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓
Apache	1.0.0 (deprecated)	✓ 0.92.0	✗	✓	✓ 0.9.0	✗	✗	✓	✗	✗	✗	✓
Cloudera	CDH4 (deprecated)	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
	CDH4.3 + (deprecated)	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
	CDH 5.0 (deprecated)	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
	CDH 5.1 (deprecated)	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
	CDH 5.1 (deprecated)	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
	CDH 5.4 (deprecated)	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓

		HBase	HCatalog	HDFS	Hive[5]		Oozie	Pig	Spark		Sqoop	Talend MapReduce
					Hive1 [2]	Hive2 [2]			Standalone	YARN		
	CDH 5.5 (YARN mode)	✅	✅	✅	❌	✅	✅	✅	✅	✅	✅	✅
	CDH 5.6 (YARN mode)	✅	✅	✅	❌	✅	✅	✅	✅	✅	✅	✅
	CDH 5.7 (YARN mode)	✅	✅	✅	❌	✅	✅	✅	✅	✅	✅	✅
	CDH 5.8 (YARN mode)	✅	✅	✅	❌	✅	✅	✅	✅	✅	✅	✅
	CDH 5.10 (YARN mode)	✅	✅	✅	❌	✅	✅	✅	✅	✅	✅	✅
MapR	2.0.0 (deprecated)	✅	❌	✅	✅	❌	✅	✅	❌	❌	✅	✅
	2.1.2 (deprecated)	✅	❌	✅	✅	❌	✅	✅	❌	❌	✅	✅
	2.1.3 (deprecated)	✅	❌	✅	✅	✅	✅	✅	❌	❌	✅	✅
	3.0.1 (deprecated)	✅	❌	✅	✅	✅	✅	✅	❌	❌	✅	✅
	3.1.0 (deprecated)	✅	✅	✅	✅	✅	✅	✅	❌	❌	✅	✅
	4.0.1 (deprecated)[4]	✅	✅	✅	✅	✅	✅	✅	❌	❌	✅	✅
	4.1.0 (deprecated)[4]	✅	✅	✅	✅	✅	✅	✅	✅	✅	✅	✅
	5.0.0 (YARN mode)[4]	✅	✅	✅	❌	✅	✅	✅	✅	✅	✅	✅
	5.1.0 (YARN mode)[4]	✅	✅	✅	❌	✅	✅	✅	✅	✅	✅	✅
	5.2.0 (YARN mode)[4]	✅	✅	✅	❌	✅	✅	✅	✅	✅	✅	✅
Amazon EMR	Apache 1.0.3 (deprecated)	✅	❌	✅	✅	❌	❌	✅	❌	❌	❌	✅
	Apache 2.4.0 (deprecated)	✅	❌	✅	✅	❌	❌	✅	❌	❌	✅	✅
	EMR 4.0.0 (deprecated)	❌	❌	✅	❌	✅	❌	✅	❌	✅	❌	✅
	EMR 4.5.0 (Apache 2.7.2)	❌	✅	✅	❌	✅	✅	✅	❌	✅	✅	✅

		HBase	HCatalog	HDFS	Hive[5]		Oozie	Pig	Spark		Sqoop	Talend MapReduce
					Hive1 [2]	Hive2 [2]			Standalone	YARN		
	EMR 4.6.0 (Apache 2.7.2)	✓	✓	✓	✗	✓	✓	✓	✗	✓	✓	✓
	EMR 5.0.0 (Apache 2.7.2)	✓	✓	✓	✗	✓	✓	✓	✗	✓	✓	✓
	EMR 5.4/5.5	✓	✓	✓	✗	✓	✓	✓	✗	✓	✓	✓
Pivotal HD	1.0.1 (deprecated)	✓	✗	✓	✓	✗	✗	✓	✗	✗	✓	✓
	2.0 (deprecated)	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
Microsoft HD Insight	3.1 (deprecated)	✗	✗	✗	✓	✓	✗	✓	✗	✗	✗	✓
	3.2 (deprecated)	✗	✗	✗	✓	✓	✗	✓	✓	✗	✗	✓
	3.4	✗	✗	✗	✓	✓	✗	✓	✓	✓	✗	✓
Custom[1]												

1. This enables the connection between the Studio and a custom Hadoop distribution not yet officially supported in the Studio. For further information, see the sections describing how to connect to a custom Hadoop distribution of the Talend Big Data Getting Started Guide or the documentation of any related component that creates the connection to a Hadoop distribution, such as tHDFSConnection.

2. In the Standalone mode, Hive 1 does not support the Kerberos security.

3. This option also allows you to connect to a Teradata Hadoop platform. For further information about the version of the Hortonworks Data Platform used in the Teradata platform you are using, see Teradata's related documentation.

4. For this MapR version, the MapR security ticket mechanism is supported by the Studio.

5. The Profiling perspective does not support the Embedded connection mode on Hive distributions. This mode is available mainly for test purposes done by Hadoop developers. The studio may not be able to run correctly with the embedded mode.

For further information about what Hadoop distributions are supported from the **Profiling** perspective, see *Supported Hive distributions for profiling data*.

For further information about the versions of all the supported third-party systems/databases, see *Supported systems, databases and business applications by Talend components*.

A.7.3. Supported Hadoop distribution versions for Talend Data Preparation with Big Data

The following table lists the supported Hadoop distributions for Talend Data Preparation with Big Data.

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

Distribution	Supported version
HDP	2.4 and above
Cloudera	5.7 and above

Distribution	Supported version
EMR	4.5 and above
Hadoop	2.7 and above

A.7.4. Supported databases for profiling data

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

The table below lists the databases supported from the **Profiling** perspective of the studio. For a complete list about supported third-party systems, see *Supported systems, databases and business applications by Talend components*.

Database name	Database version
Amazon Redshift	Initial release of Amazon Redshift
AS/400	V5R2 to V7R2
Hive	For a complete list of the supported Hive distributions, see *Supported Hive distributions for profiling data*.
IBM DB2	10.5
IBM DB2 ZOS	2000/2005/2008
Impala (a sub-module of Cloudera)	CDH5 5.0/5.1
Informix	11.50
Ingres	9.2
Microsoft SQL Server	2000/2003/2005/2008/2012
MySQL	5.1/5.5/5.6
Netezza	6
Oracle with SID	9i to 11g
Oracle with service name	9i to 11g
PostgreSQL	8.3/9.1+
SQLite	3.6.7
Sybase (ASE and IQ)QLite	12.5/12.7/15.2
Teradata	12/13/14/15
Vertica	6.0/7.0

A.7.5. Supported Hive distributions for profiling data

The information contained in the following table is applicable for the 6.4.1 version of your Talend product at the time of its release. For updated information on the latest supported versions of the third-party systems, see the online version of this page on Talend Help Center.

Below is a list of the supported distributions for Hive profiling.

For the time being, the embedded mode on Hive distributions is available mainly for test purposes done by Hadoop developers. The studio may not be able to run correctly with the embedded mode.

Hive distribution		version	
		Hive 1	**Hive 2**
HortonWorks	**HDP 1.0.0** (deprecated)	Embedded and Standalone	No

Hive distribution		version	
		Hive 1	**Hive 2**
	HDP 1.2	Embedded and Standalone	Embedded and Standalone
	HDP 1.3	Embedded and Standalone	Embedded and Standalone
	HDP 2.0	Embedded (only Linux) and Standalone	Embedded (only Linux) and Standalone
	HDP 2.1	Embedded (only Linux) and Standalone	Embedded (only Linux) and Standalone
	HDP 2.2	Embedded (only Linux) and Standalone	Embedded (only Linux) and Standalone
Cloudera	**CDH4** (Kerberos authentication is supported)	Embedded and Standalone	Embedded and Standalone
	CDH5 (Kerberos authentication is supported)	Embedded and Standalone	Embedded and Standalone
	CDH5.1 MR1 (Kerberos authentication is supported)	No	Standalone
	CDH5.4 YARN (Kerberos authentication is supported)	No	Standalone
MapR	**MapR 1.2** (deprecated)	Standalone	No
	MapR 2.0	Embedded and Standalone	No
	MapR 2.1.2	Embedded and Standalone	No
	MapR 3.0.1	Embedded and Standalone	Embedded and Standalone
	MapR 3.1	Embedded and Standalone	Embedded and Standalone
Apache	**Apache 1.0.0 (Hive 0.9.0)**	Embedded and Standalone	No
	Apache 0.20.23 (Hive 0.7.1)	Standalone	No
Pivotal HD	**Pivotal HD 1.0.1**	Standalone	No
	Pivotal HD 2.0.1	Embedded (only Linux) and Standalone (Linux and Windows)	Embedded and Standalone (only Linux)