Self-service Talend Migration

talend

Moving from On-Premises to the Cloud

talend

Contents

Migration readiness

Introduction

You work at a data-driven organization, and your management has decided to move your operations to the cloud. They've licensed Talend Cloud and have tasked you with migrating your existing Talend projects and Jobs to this new platform.

Terminology

You should be aware of some differences in vocabulary between on-premises Talend and Talend Cloud.

- **Jobs** refer to Studio Job Designs for Standard and Big Data. Jobs also refer to compiled code that is deployed to Talend Administration Center (TAC) for execution in Job Conductor. Once published to Talend Cloud, Studio Jobs become Talend Artifacts.
- **Tasks** are deployed Talend Artifacts in Talend Cloud. That is, once a Studio Job is published to Talend Cloud it's referred to as an artifact until a task is created to orchestrate that item.
- **Plans** are groups of tasks in Talend Cloud. They're analogous to Execution Plans in TAC but have slightly different features.
- **Context Variables** become **Parameters** in Talend Cloud.

The keys to a successful Talend Cloud migration are assessment and planning.

Readiness

The keys to a successful Talend Cloud migration are assessment and planning. We recommend you take the time to understand your existing installation, then create a plan for this project.

Assessment

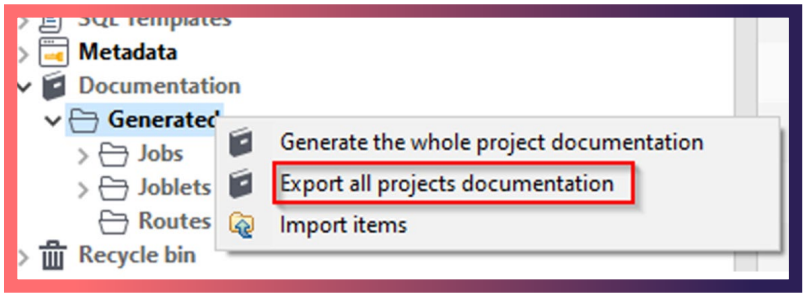
Check your on-premises Talend installation for the following items:

- ☐ Do you have Big Data Jobs that must be migrated to the cloud?
- ☐ Do you deploy Realtime Jobs?
- ☐ Do you use more than one TAC?
- ☐ Do you require or use Talend CI/CD?
- ☐ Do you use more than two or three Talend projects?
- ☐ Do your projects contain more than 100 Jobs?
- ☐ Are there organizational or environmental factors that could impact migration, such as network access, security policy, or administrative approvals?

Items such as Realtime Big Data deployments may require additional considerations for cloud migration. Refer to the last section of this guide and plan accordingly.

Analysis

Look at your Talend projects. An easy way to get a detailed report describing your project and the Talend artifacts within it is to create and export User Generated Project Documentation. In Studio, right-click on Documentation > Generated menu in the Repository. Select “Export all projects documentation” to generate a Zip file containing an HTML report listing for your project.



The user-generated documentation will provide insight into job complexity, component usage, and, most importantly, context variables.

Planning

Once you have a good understanding of your on-premises Talend installation, projects, and artifacts, create a plan that covers tasks and resources for the migration project. When putting your plan together, consider:

- Do you plan to refactor any Jobs (e.g. optimize) during the Talend Cloud migration project?
- Can any obsolete Jobs be removed?
- How many Talend Job Servers do you currently use?
- Do you plan to use Compute Engines to run your Jobs?
- Are you using or do you require Talend CI/CD?
  - **Note:** If your on-premises Talend projects contain lots of Jobs, you should deploy Talend Zero Install CI for Talend Cloud, which will lessen the work required to publish large batches of Jobs to Talend Cloud.

Licensing

Activate your Talend Cloud license and designate one of your technical staff to act as the Talend Cloud Security Administrator for your account. This person should be the technical leader of the migration project and be responsible for managing and provisioning your Talend Cloud implementation. A user with the Project Administator role is also required. By themselves, Security Administrator and Project Administrator do not consume license seats.

Software

Be sure you get from Talend — and nowhere else — all the new software you need. This includes:

- **Talend Studio**—the latest version: If your on-premises Talend release is older than the current Talend Cloud Studio, download a new copy.
- **Remote Engine:** If you elect to deploy your Jobs on Remote Engines, download the Remote Engine installer for the operating system of your choice.

Architecture

While Talend Cloud offers many possibilities, for a self-service migration we work with only Talend Studio, Talend Management Console (TMC), Remote Engines, Cloud Engines, and Git source control. Check for compatible Git versions for Talend Cloud ***before*** you start.

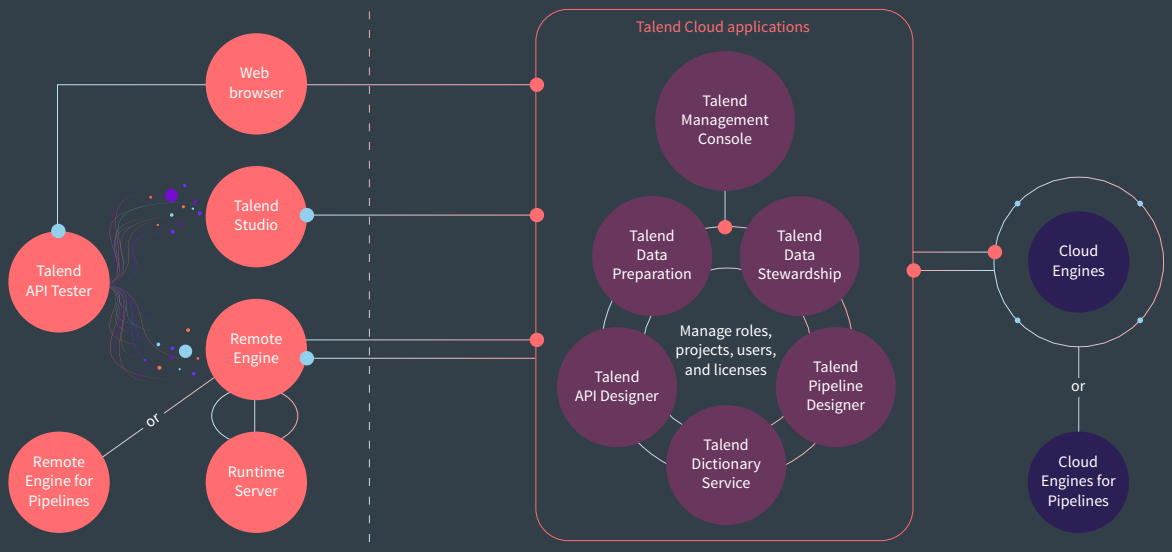
Remote or Cloud Engines

After migration to Talend Cloud, will your Jobs require access to local resources? If so, consider installing one or more Remote Engines. These can be installed anywhere on-premises or in the cloud, and make all your local data accessible by your Talend Jobs. You won't need to upload any data to the cloud, and Remote Engines can be clustered.

If your Jobs are compute-intensive and do not depend on local resources, you can deploy to a Cloud Engine. These are managed by Talend. You don't have to install or configure any additional software, and Talend handles the SLA for these components.

Both Remote Engines and Cloud Engines are licensed by applying tokens when you create and pair an engine with Talend Cloud. Your Talend Cloud license comes with enough tokens to make use of these items. If you need additional execution engines you can purchase more tokens.

Talend Cloud architecture



Sizing

Sizing applies mainly to the compute resources where you deploy your Remote Engines, and is 100% under your control. Jobs run Remote Engines can access the same data as an on-premises Talend Remote Job Server. As you're already running Talend Job Servers, you probably have a good idea about what server resources your Jobs require. If not, think about physical resources for your Remote Engines. A general starting point is 16 GB RAM with 4 CPUs. You can scale up or down as needed.

Preinstalled Remote Engines are available in the AWS Marketplace and the Azure Marketplace, and are the easiest way to deploy a Remote Engine for Talend Cloud. These offerings are both "bring your own license," meaning that if you've purchased tokens for Remote Engines you can use them here. When you opt for preinstalled Remote Engines, you can select the compute instance size and other parameters. For more information, see the Talend guides for AWS and Azure.

Source control

Talend Cloud supports Git/GitHub as the only source control management system for Talend projects. Whether you use Git or Subversion for your on-premises Talend projects, to prepare your projects and Jobs for migration you must:

- Stop all Talend development in on-premises projects.
- Make commits and pushes from all copies of Studio to your repo.
- Create a backup of your repository, or create a tag to label the point of migration.
- Start a single copy of Studio on your on-premises project.
  - Export all Talend project and Job items to a Zip file. Include everything.
- Exit from the on-premises Talend project (exit Studio).
- Create a new project in Talend Cloud and assign a new, empty Git/GitHub repository.
- Start an instance of your new Talend Studio and connect to the new Talend Cloud project. You'll need a Talend Cloud user with developer privileges to connect Studio to Talend Cloud.
- Import the Talend project Zip file into Talend Studio.
- Check that everything was imported.

Once your old Talend project is imported to the Talend Cloud project, check your Jobs to be sure they compile. Identify any cases where errors occur and take the necessary corrective action to ensure a clean compile. If possible, test the Jobs in Talend Studio to be sure they behave as expected.

Setup and configuration

In the first section, we provided key information you need to prepare for a Talend Cloud migration, including assessing your current Talend installation, projects, and Jobs, which should have allowed you to move your Talend projects and Jobs to a Talend Cloud project with Git/GitHub source control. Next, let's look at detailed setup and enablement of your Talend Cloud account, including how to provision users, groups, and roles. We'll add more detail about Talend Cloud projects and other items in the Talend Management Console (TMC).

Users, roles, and groups

Talend users, as provisioned in a TAC, fall into one or more roles:

- Security administrator
- Administrator
- Viewer
- Operation manager
- Auditor
- Designer

Talend Cloud includes different built-in roles:

- Security administrator
- Project administrator
- Environment administrator
- Operator
- Integration developer

We suggest a 1:1 mapping between TAC roles and Talend Cloud roles:

TAC roles		Talend Cloud roles
Security administrator	→	Security administrator
Administrator	→	Project administrator
Viewer		-
Operation manager	→	Operator
Auditor		-
Designer	→	Integration developer

With Talend Cloud you can create roles and assign permissions as you see fit.

Talend Cloud users can belong to one or more groups. For example, you can assign users to a Talend Cloud project by group instead of individually.

For the self-service migration, you need at least one Talend Cloud security administrator, who has these privileges:

Permissions

Management Console

Groups: Manage

Password policy: Manage

Roles: Manage

SSO: Manage

Subscription: Manage

Users: Manage

This user maintains all Talend Cloud users, groups, and roles and is responsible for items that you may not want to share, such as SSO configuration, password policy, and subscription management.

Talend Cloud supports other kinds of administrators as well:

Project administrators manage projects in Talend Cloud and artifact repositories.

Environment administrators manage Talend Cloud environments (for example, development, test, and production) and the creation, maintenance, and execution of Promotion Pipelines, which are the tool used to move Talend Jobs from one environment to another.

Operators publish, schedule, execute, and monitor Jobs, Routes, and Data Services from Talend Studio to Talend Cloud.

Finally, **integration developers** can log in to Studio as developers for a Talend Cloud project. Integration developers cannot publish Jobs to Talend Cloud; an operator must do this on their behalf, or the integration developer must also be assigned the operator role.

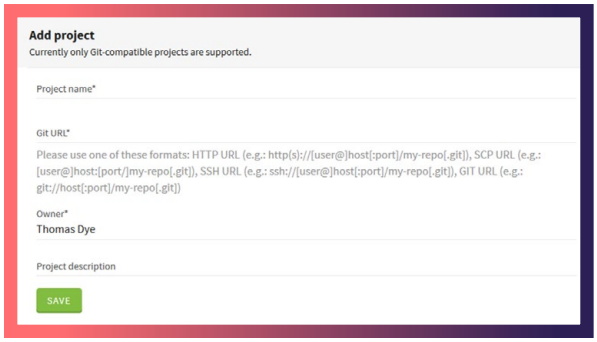
Have your security administrator create a user ID for each user and assign the appropriate Talend Cloud roles.

Talend Cloud projects

When a user logs into Talend Studio and connects to a Talend Cloud project, details about that project are downloaded to Talend Studio from the Git/GitHub repository associated with the Talend Cloud project. To connect Studio to Talend Cloud and begin development, begin by creating a project.

Create project

A user with project administrator privileges must log in to the Talend Cloud web application. Select **Projects** from the left-hand menu bar in the TMC, then **Add Project**:



Enter a project name. We recommend you choose the same name as your project in TAC.

Next, supply a SaaS Git service (like GitHub) or an on-premises Git URL. Your source code repository does not have to be hosted in the cloud, though many organizations choose to do so. The service you select is separate from Talend Cloud. Take care that the Git service you choose, whether SaaS or on-premises, is supported by Talend Cloud. You can find a list of compatible Git services in our online documentation. For your first Talend Cloud project, use an empty Git repository with no existing branches or artifacts.

Talend Cloud environments and workspaces

Talend Cloud is a fully production-capable service that supports a formal software development lifecycle (SDLC). The recommended best practice is to allow for separation of access and responsibilities by Talend Cloud user function.

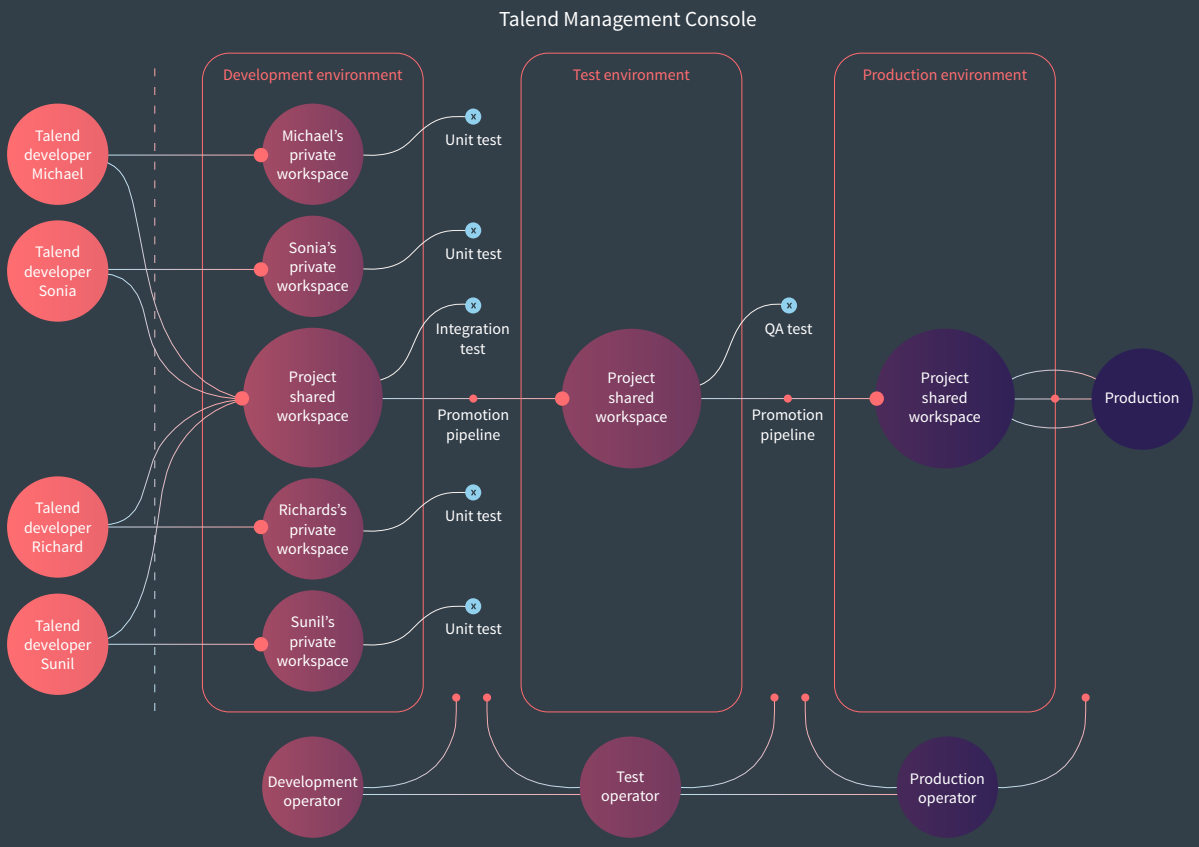
In the diagram below, developer users can create artifacts in Talend Studio and publish them to Talend Cloud. Each developer gets their own private workspace for unit testing but should publish completed artifacts to a project shared workspace. Developer users cannot access or orchestrate in Test or Production environments.

Users with the Test (or QA) role can read from the development shared workspace and publish to the Test environment workspace via a Promotion Pipeline. These users can only orchestrate tasks and plans in the Test environment and have no access to the Production environment.

Production users can read from the Test environment and publish to the Production environment via a Promotion Pipeline and can orchestrate only Production tasks and plans.

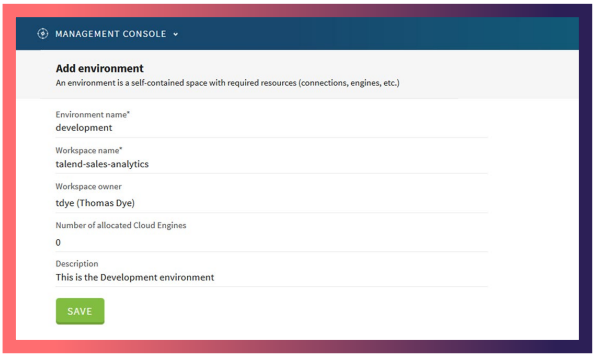
This architecture prevents developers and other unauthorized users from gaining access to the Production environment and ensures traceability of artifacts from development to production.

Talend Cloud environments and workspaces



You must create at least one environment in Talend Cloud and a corresponding workspace. Each Talend Cloud developer has their own workspace, called “default,” which serves as a sandbox area where a developer can test deploy Jobs into Talend Cloud. Environments should have a corresponding public workspace, where Jobs can be published, to make SDLC promotion of Jobs easier.

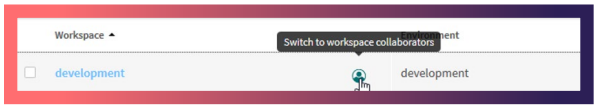
As a Talend Cloud user with the environment administrator role, log in to Talend Cloud. In the Talend Management Console (TMC) select **environments** on the left-hand menu tree, then **Add environment**.



Assign an environment name and a workspace name *that is the same as the environment name.* Assign an owner to the workspace (usually the environment administrator). Don’t allocate any Cloud Engines to the environment; these can be assigned dynamically at task run time. Click **Save**.

From the Environments list that displays when you clicked Save, click the **Workspace permissions** button at the top of the page, next to the **Add environment** button.

Click the workspace collaborators button that displays when you hover the mouse over your new environment



Select any other users that require access to this environment. When you select a user, a permission details dialog that slides in from the right lets you adjust individual users’ permissions on the workspace. Click **Save** for each new user assigned to the workspace.

Users assigned to an environment by this process have at least two workspaces: their personal workspace and the new workspace created when you created the environment.

Execution Engines

Cloud and Remote Engines

You can run your Jobs in the Talend Cloud or keep your Jobs within your on-premises environment using Remote Engines. Let's look at the characteristics and advantages of each.

Cloud Engines

Cloud Engines are created and maintained by Talend.

Cloud Engines don't need to be installed. One Cloud Engine is always available and can be assigned to tasks at run time, assuming enough tokens are available. While you can add Cloud Engines to an Environment Configuration from within the TMC, we recommend you add Cloud Engines to individual tasks. This way, Cloud Engines are allocated at run time, and your token count is lowered only while the task runs.

Cloud Engines are more expensive than Remote Engines, but they are created and maintained by Talend and are always available across environments or can be assigned to a workspace.

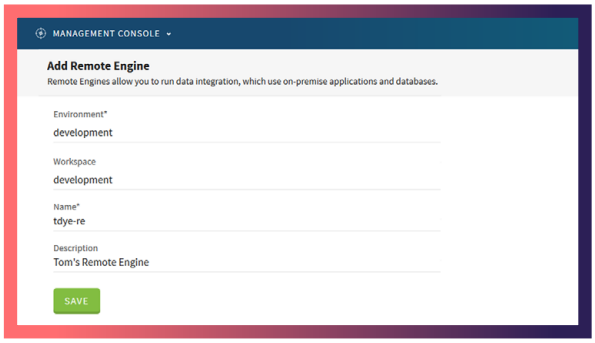
Cloud Engines do not have access to your on-premises data. Cloud Engines shine for compute-intensive Jobs, or Jobs that utilize data sources and destinations that are 100% in the cloud, such as Salesforce.com.

Remote Engines

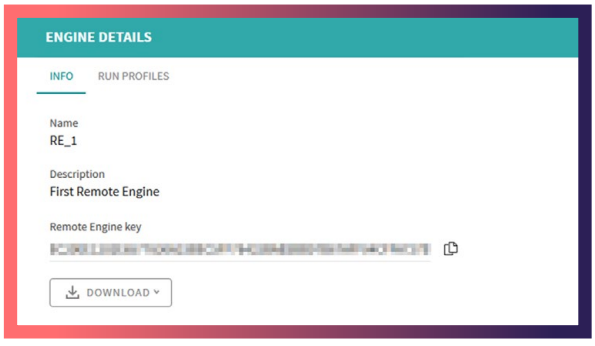
Remote Engines are installed on a server you control, which may be a physical server or virtual machine (VM) in your data center, or on a compute node in a cloud service such as AWS, Azure, or GCP. See the section on Readiness for information regarding sizing and the AWS and Azure marketplace offerings for Remote Engines.

If you elect to install Remote Engines, consider the following:

- Download the Remote Engine installer for your operating system from Talend Cloud.
- Run the installer. Reference the Talend Cloud Installation Guide for additional information. Then choose the link for platform of your choice (Windows, Linux, or Mac).
- Alternatively, you can create a Remote Engine by accessing an image from the AWS or Azure Marketplace. You'll only need to supply a pairing key (see below) when you create a Remote Engine via this method.
- A Remote Engine server requires access to the public internet. No inbound ports are required to be opened, only outbound traffic on the SSL port (443). The Remote Engine will initiate contact with Talend Cloud over that port.
- Remote Engines are paired with a Talend Cloud account. When you add one to Talend Cloud, you are supplied with a **pairing key** — a text string that you must supply to the Remote Engine when you install it. Pairing is how Talend Cloud knows the Remote Engine belongs to you and only you.
- From the Talend Management Console (TMC), select **Engines** on the left-hand menu. Click the **Add** button, then select **Remote Engine** to bring up a screen like the one below.



- Assign the environment to the Remote Engine, give it a name and description, and click **Save**.
- From the TMC Engines page, select the Remote Engine you just created. It should show as "not paired." Click the Remote Engine name to display details, including the pairing key.
- Copy this key and provide it to the Remote Engine installer or to the AWS or Azure create dialogs if you opt for a Remote Engine from these marketplaces. Then, after the Remote Engine runs for a few minutes, the status should change to "paired."



Your remote engine is now ready to run a Talend Job.

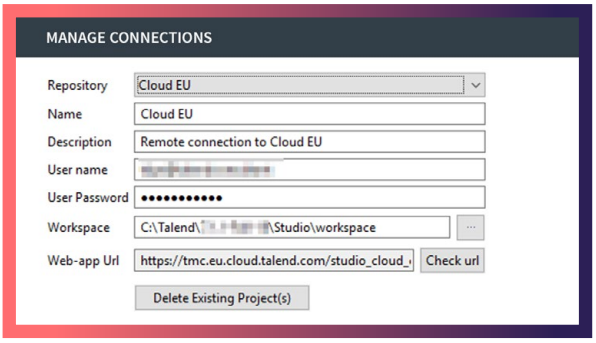
Studio

Install Studio

Install your new Talend Cloud copy of Studio just as you would for an on-premises version.

Connect to Talend Cloud

Upon startup of Studio, you're prompted for a connection to Talend Cloud. Click **Manage Connections** and select the region of Talend Cloud (US, EU, or APAC) that matches your Talend Cloud account.



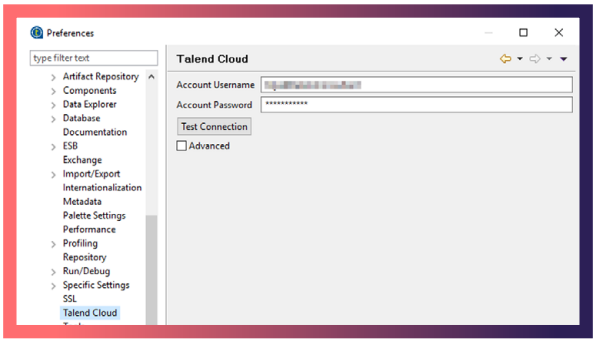
Supply your account credentials and then click **Check url** before continuing. Be sure your workspace is separate from any other Talend Studio workspaces.

Connect for the first time

Once you've saved the connection, you can log in to your Talend Cloud project from Studio. Only one person should connect at first. Talend Studio may upgrade your Git project to the latest release of Talend Cloud automatically when you first connect to the project.

Set preferences

Developers who will publish Talend Jobs to Talend Cloud need to set preferences for Talend Cloud in Studio. Click on **Window > Preferences**, and then select **Talend Cloud**. Supply your login credentials for Talend Cloud and click the **Test Connection** button.



Check Jobs

Now you have your Talend Cloud project created in Git/GitHub. You've connected to Studio and imported your on-premises Talend Jobs. Build and run your Jobs within Studio to be sure they compile and run correctly.

Context variables

Existing Talend Job context variables potentially need treatment before publishing to Talend Cloud. Context variables are known as **parameters** in Talend Cloud, and there are several types. For now we'll discuss **connection parameters**, which are sourced from context variables that are concerned with connecting to a data resource.

We strongly recommend that connection parameter context variables be stored in the Context Repository and reused among Talend Jobs.

Usually you will create a set of custom connection parameters. The naming convention is:

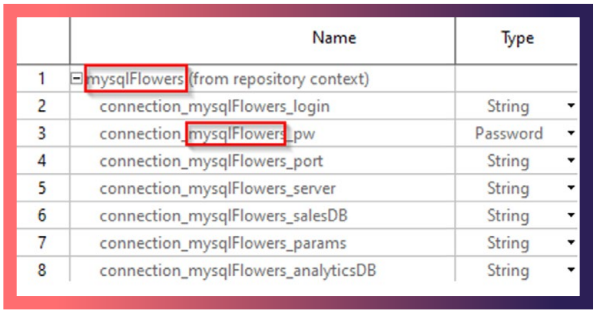
connection_<application_name>_<parameter_name>

where

connection_
is a fixed prefix

<application_name>
is the name of the system to which you want to connect

<parameter_name>
is the variable name of the connection parameter



	Name	Type
1	mysqlFlowers (from repository context)	
2	connection_mysqlFlowers_login	String
3	connection_mysqlFlowers_pw	Password
4	connection_mysqlFlowers_port	String
5	connection_mysqlFlowers_server	String
6	connection_mysqlFlowers_salesDB	String
7	connection_mysqlFlowers_params	String
8	connection_mysqlFlowers_analyticsDB	String

To promote reuse across Talend Cloud tasks, the application name should be the same as the name of the repository context group. Talend recommends you name the context group using the convention:

<resource_type><resource_purpose>

where

<resource_type>
is the data service type, such as mysql or ftp, in lower case

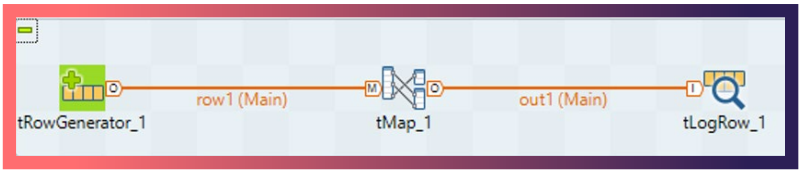
<resource_purpose>
is the purpose of the resource in your Talend Job. In this example the purpose is a database of available flower types for a floral supply company

You must follow a naming standard for connection parameters for three reasons:

1. **Password encryption.** Passwords values can only be hidden for context variables that conform to the connection parameter naming convention.
2. **Sharing of parameters across environements.** The parameters themselves can be shared across Talend Cloud environments, but not the values. Each environment can have a different set of values germane to that environment.
3. **Sharing of parameters across tasks.** Connection parameters that conform to the naming convention can be shared across TMC tasks.

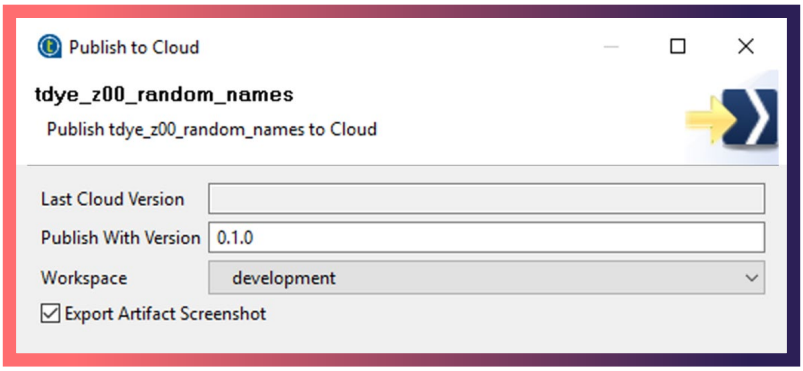
Identify candidate for first go-live

The first Job published to Talend Cloud can be a simple existing Job or a new Job you create for this task. A good candidate for an existing Job for the initial go-live would be a Job with no external data connections, since the idea of the first Job is to get familiar with Talend Cloud end to end. You can address complexity such as data connections after you familiarize yourself with Talend Cloud orchestration.



Publish to Talend Cloud

Once you have a Job ready, be sure your developer has the proper Talend Cloud privileges to publish: Integration Developer and Operator. Then right-click on the Job on the Repository tab and select **Publish to Cloud**.

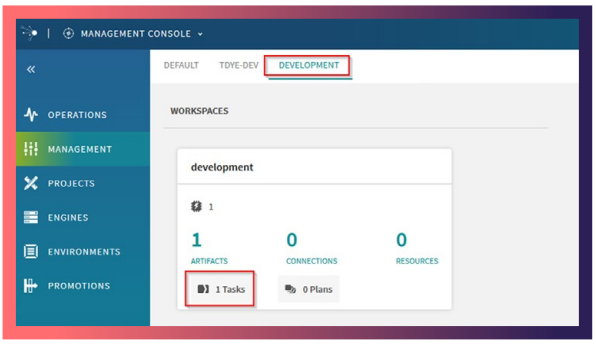


Because your Job has never been published, there won't be a value for **Last Cloud Version**. Leave the **Publish With Version** field as it is, and select the shared workspace in your development environment. Click **Finish**. The Job will be published to Talend Cloud in the workspace you selected.

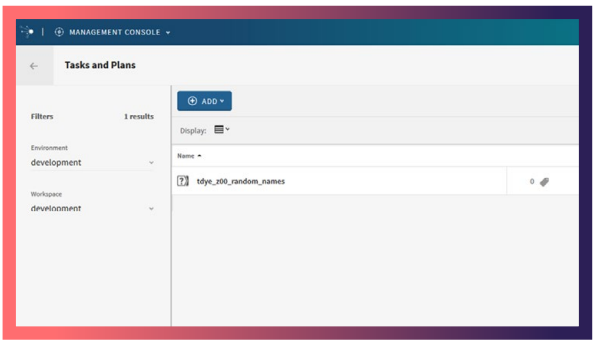
Orchestration

Modify initial task

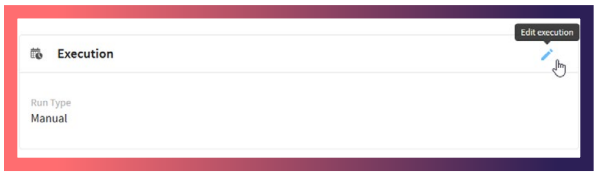
Log in to the Talend Cloud web UI. Access the TMC and select the **Management** menu. If necessary, change your workspace to “development”. You’ll see the job you uploaded (artifact) and note that an initial task has been created for you. Click on the **Tasks** button (highlighted in the red box in the figure below).



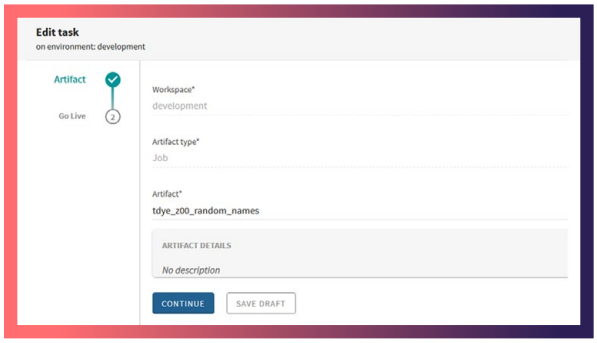
The Tasks and Plans page will display.



Note the Environment and Workspace settings, but leave everything as it is, and click the name of the task to display the Tasks Information page. Hover the mouse over the word **Execution** and a pencil icon will appear. Click the pencil.



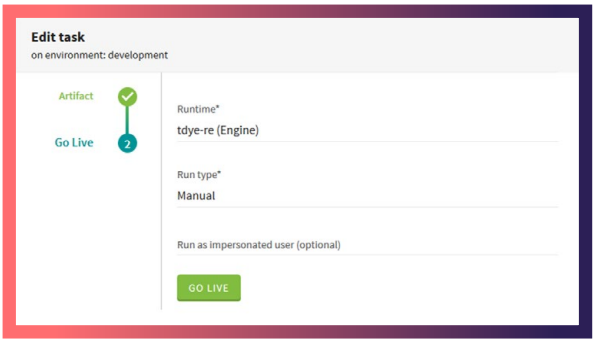
This brings up the Edit Task page. Check that all the items on the page are as expected, then click **Continue**.



This brings us to the Go Live page.

Select your desired runtime from the dropdown box and set the run type to **Manual**. By default, the task is assigned to and run on a Cloud Engine, but if you choose, you can assign and run the task on a Remote Engine.

Click **Go Live** to run the Job.



Run and check logs

Congratulations! You've run your first Job in Talend Cloud. When it completes, the Job results will display on the task page. You can also go back to the main TMC page and click on the Operations left-hand menu to monitor the Job. TMC will display the run status of your Job, and it lists every run attempt.

When the Job is finished, you can show the detail by clicking the down chevron in the upper right corner of the last run list:

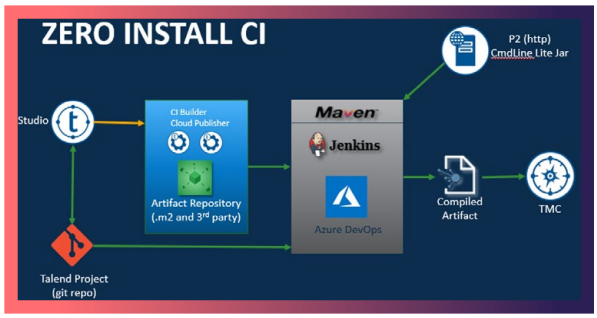
On the resulting detail information dialog, click the **View Logs** button and select the All log. Scroll down to see the output of the Job.

CI/CD with Talend Cloud

If your on-premises Talend project contains many Jobs, we recommend you use continuous integration to automate the compilation and publishing of all jobs from a Talend project into Talend Cloud. Talend offers a new type of CI called Zero Install CI that has no dependency on a separate command-line process, or on Studio running. The compile function is contained in jar files that are supplied in a P2 repository, downloadable from Talend Cloud. This P2 repo must be made available to the CI process via http (commonly via Apache Tomcat). The Talend Maven plugins for CI Builder and Cloud Publisher are now in Studio's .m2 repository and are pushed to the Artifact Repository (e.g. Nexus or jFrog) at first Studio login.

The new CI process should be configured with a build management tool such as Azure's DevOps, or Jenkins, or any tool that supports builds with Maven. Talend documentation contains installation and configuration use cases for Jenkins and Azure DevOps.

The Talend CI process can enable unit testing of Talend Jobs or subjobs in Studio. Talend CI can fail a build if any test cases fail.



The Talend Cloud Swagger API offers methods to list and execute promotion pipelines. Developers can use the Swagger API to enable continuous delivery by executing promotion pipelines to move Talend Cloud artifacts from development to test, test to production, and so on.

Conclusion

Now you should be up and running in Talend Cloud. If you need additional help, Talend has more resources for you:

- Talend Professional Services is the best resource to assist with Talend Cloud migrations. Take advantage of our expert guidance for your complex use cases. Contact your sales account rep or customer success manager.
- Licensed users can turn to Talend Support when things don't work as expected.
- To learn more about Talend Cloud, check the new Talend Academy and Talend Training.
- Visit the Talend Community for forums where you can post questions on a variety of topics. Experts in the greater Talend community are eager to help.
- help.talend.com contains documentation for all Talend products.
- Watch a recorded webinar on migrating from Talend on-prem to Talend Cloud, with a demo of a sample project migration.

About Talend

Talend (NASDAQ: TLND), a leader in data integration and data integrity, enables every company to find clarity amidst the data chaos. Talend is the only company to bring together in a single platform all the necessary capabilities that ensure enterprise data is complete, clean, compliant, and readily available to everyone who needs it throughout the organization. With Talend, organizations are able to deliver exceptional customer experiences, make smarter decisions in the moment, drive innovation, and improve operations.

For more information, please visit www.talend.com

About the author

Thomas Dye is a Talend Technical Manager, responsible for strategies and tactics for Talend Cloud migration. Mr. Dye has extensive experience with Talend Cloud and all the large cloud platforms, and has worked with many Big Data, ETL, Data Quality, MDM, CRM, BI, ERP, and OLTP systems for global markets. Previously, Mr. Dye was a Principal Consultant and Technical Lead for Talend Professional Services. He is a Certified Computing Professional.


talend