



# Talend Project Audit User Guide

## 7.3.1

# Contents

|  |           |
|--|-----------|
| <b>Copyright.....</b>                          | <b>3</b>  |
| <b>Overview of Talend Project Audit.....</b>   | <b>4</b>  |
| What is Talend Project Audit.....              | 4         |
| Key areas audited in a project.....            | 4         |
| Benefits of auditing a project.....            | 5         |
| Moving forward.....                            | 5         |
| <b>Auditing projects in Talend Studio.....</b> | <b>6</b>  |
| Auditing a project.....                        | 6         |
| Regenerating a report from audit history.....  | 7         |
| Customizing audit database.....                | 8         |
| <b>Talend Project Audit reports.....</b>       | <b>10</b> |
| Project properties.....                        | 10        |
| Job/Component complexity.....                  | 10        |
| Job error management.....                      | 15        |
| Job documentation/versioning.....              | 16        |
| Metadata.....                                  | 18        |
| Layout.....                                    | 20        |
| Job analysis.....                              | 21        |

# Copyright

Adapted for 7.3.1. Supersedes previous releases.

Copyright © 2020 Talend. All rights reserved.

The content of this document is correct at the time of publication.

However, more recent updates may be available in the online version that can be found on [Talend Help Center](#).

## Notices

Talend is a trademark of Talend, Inc.

All brands, product names, company names, trademarks and service marks are the properties of their respective owners.

## End User License Agreement

The software described in this documentation is provided under **Talend**'s End User Software and Subscription Agreement ("Agreement") for commercial products. By using the software, you are considered to have fully understood and unconditionally accepted all the terms and conditions of the Agreement.

To read the Agreement now, visit [http://www.talend.com/legal-terms/us-eula?utm\\_medium=help&utm\\_source=help\\_content](http://www.talend.com/legal-terms/us-eula?utm_medium=help&utm_source=help_content)

# Overview of Talend Project Audit

This document introduces the audit approach used by Talend Project Audit and lists the key areas that are audited in a project.

## What is Talend Project Audit

Talend Project Audit transforms project data flows to valuable business information. It introduces an auditing approach for evaluating various aspects of Jobs implemented in your Talend Studio. The aim is to provide solid quantitative and qualitative evaluation for process-oriented decision support.

A project audit is the process of collecting and evaluating information about Job operations designed in the Studio. The evaluation of obtained information determines if the technical processes and data flows you are using in your audited project are operating effectively and efficiently to achieve the project goals or objectives.

Audit results are presented in printable reports generated to provide extensive data regarding different areas in the audited project.

For more information about the areas audited in a project, see [Key areas audited in a project](#) on page 4.

You can later use the results provided by Talend Project Audit to get a clear picture regarding the status and performance of different elements in the audited project. Using this significant amount of information, you can analyze and monitor the performance of these elements in order to improve efficiency.

For detailed information about the data presented in Talend Project Audit reports, see [Talend Project Audit reports](#) on page 10.

## Key areas audited in a project

Talend Project Audit is an inquiry into the key data processing aspects of a project that is designed in your Talend Studio.

Talend Project Audit provides several functions for auditing a project through investigating different elements in Jobs designed in the Studio. Talend Project Audit reviews:

- degree of difficulty of Jobs and components used in Jobs,
- Job error management,
- documentation and Job versioning,
- usage of metadata items in Job designs,
- layout-related issues in Job and subjob graphical designs,
- Job analysis.

These aspects depend on each other and are correlated accordingly during project investigation in Talend Project Audit.

For more information about the audit results of the above key areas, see [Talend Project Audit reports](#) on page 10.

## Benefits of auditing a project

Talend Project Audit helps to enhance a high level of awareness of the quality and effectiveness of the standards and techniques used throughout your projects realized in your Talend Studio. It provides information that enhances key areas in your project and thus leads to better realization of future data integration projects.

Talend Project Audit also combines the audit results from your projects stored throughout different repositories. It allows to exploit out more of the obtained information with the aim of getting your own points you see in your decision making process.

The evaluation of obtained information determines if the projects designed in a studio operates effectively and efficiently to achieve the organization's objectives.

This opportunity will lead to quick adoption of best practices to help your organization to apply data integration subsequent projects effectively.

To summarize what has been discussed earlier, possible benefits of auditing a project are:

- providing statistical information about key concepts in Jobs included in a project,
- identifying errors and weaknesses, if any, in a project,
- evaluating the performance of the techniques used in the Jobs included in the investigated project,
- improving future project performance by avoiding misused processes and techniques.
- realizing more efficient and effective utilization of your audit data coming from one of your projects,
- customizing your auditing contents to present the most valuable information for your process-oriented decision support.

## Moving forward

Strong project auditing practices are critical to the successful execution of all data-related projects realized in your Talend Studio and, consequently, to the overall accomplishment of your organization.


Periodic audits of the processes of different projects ensure that standards and techniques related to Job realization are identified, investigated, and fixed and the process is improved. As a result, every aspect of the project is equally important and must be reviewed thoroughly.

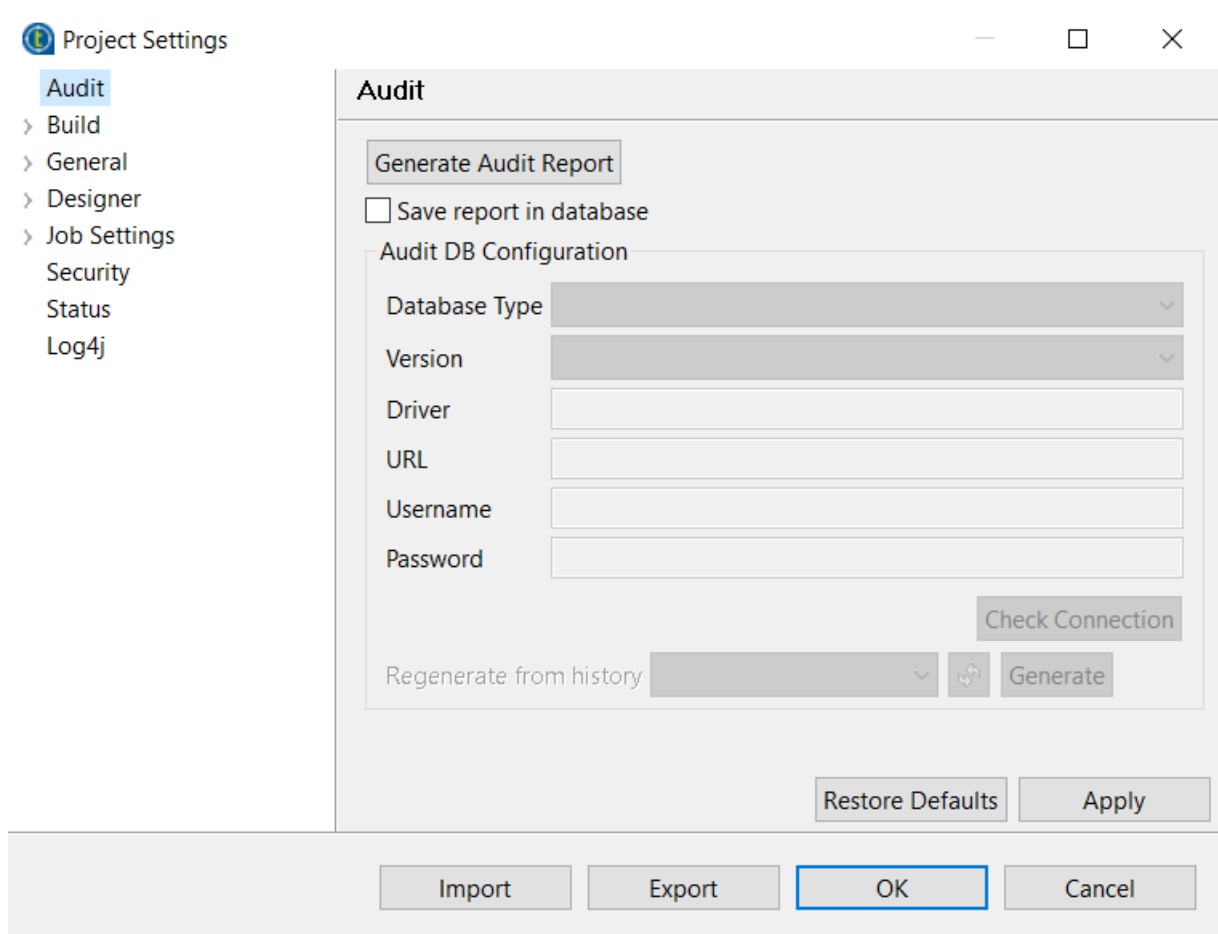
# Auditing projects in Talend Studio

The Talend Project Audit tool evaluates different aspects of the Jobs implemented in a project realized in Talend Studio with the aim of providing solid quantitative and qualitative factors for process-oriented decision support.

## Auditing a project

### Procedure

1. Launch your Talend Studio and log in to the project you want to audit.
2. Click  on the toolbar of the Studio main window, or click **File > Edit Project properties** from the menu to open the **Project Settings** dialog box, and then click **Audit** to open the **Audit** view.



3. Click **Generate Audit Report**.

**Tip:** If you want to save your audit data to the custom audit database you have defined, select the **Save report in databases** check box before clicking **Generate Audit Report**. For more information on how to define a custom audit database, see [Customizing audit database](#) on page 8.

4. In the **Browse for Folder** dialog box, browse to the folder where you want to save your audit report, and click **OK** to start audit report generation.

## Results



Upon successful report generation, a message box appears to inform you where to check your generated audit report.

## Regenerating a report from audit history

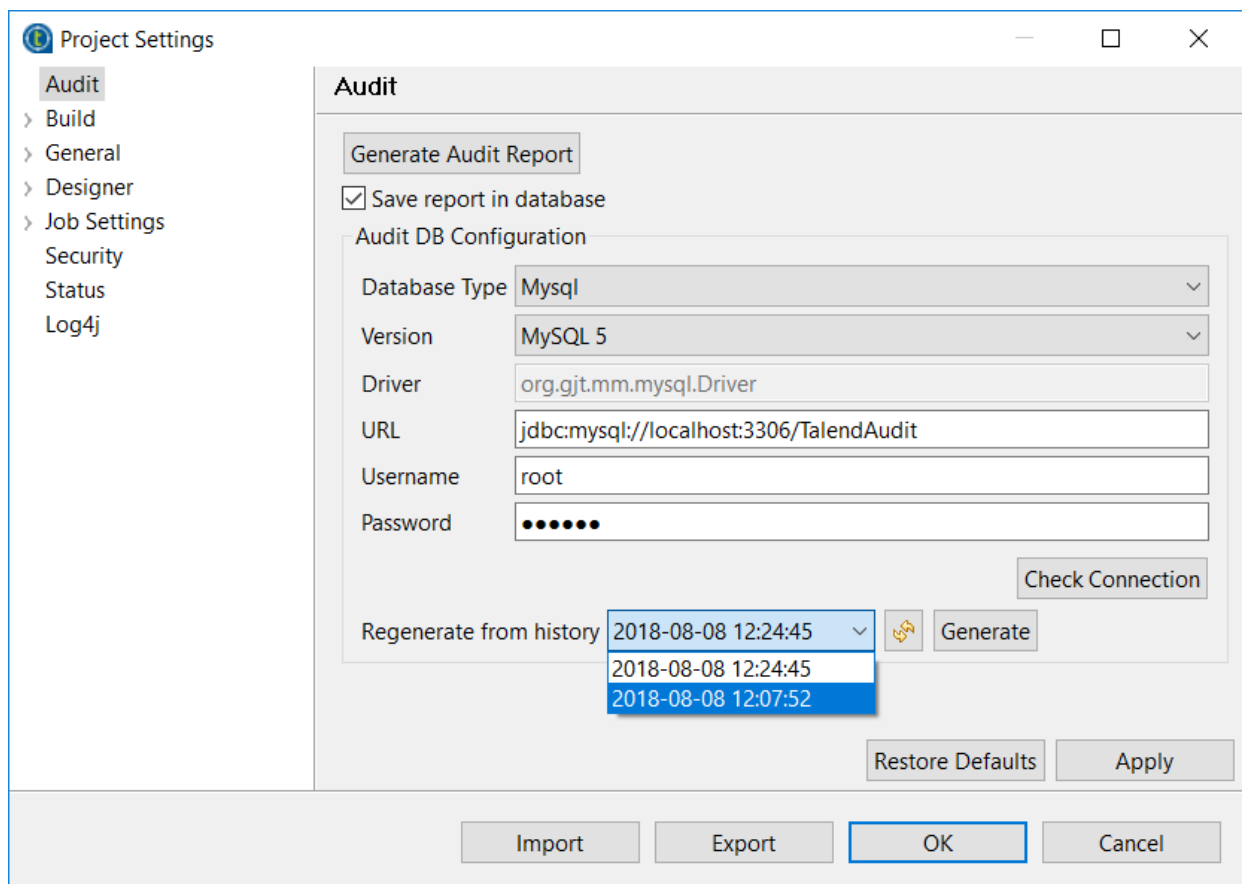
If you have defined a custom audit database and used it when auditing your project, you can regenerate audit reports from history records.

For more information on how to define a custom audit database, see [Customizing audit database](#) on page 8.

### Procedure

1. Click  on the toolbar of the Studio main window, or click **File > Edit Project properties** from the menu to open the **Project Settings** dialog box, and then click **Audit** to open the **Audit** view.
2. Select the **Save report in databases** check box.
3. Click  next to the **Regenerate from history** field to update the list of report generation records.
4. Select a generation record and click the **Generate** button.

### Example



5. In the **Browse for Folder** dialog box, browse to the folder where you want to save your audit report, and click **OK** to start audit report generation.

## Results


Upon successful report generation, a message box appears to inform you where to check the regenerated audit report.

# Customizing audit database

## About this task

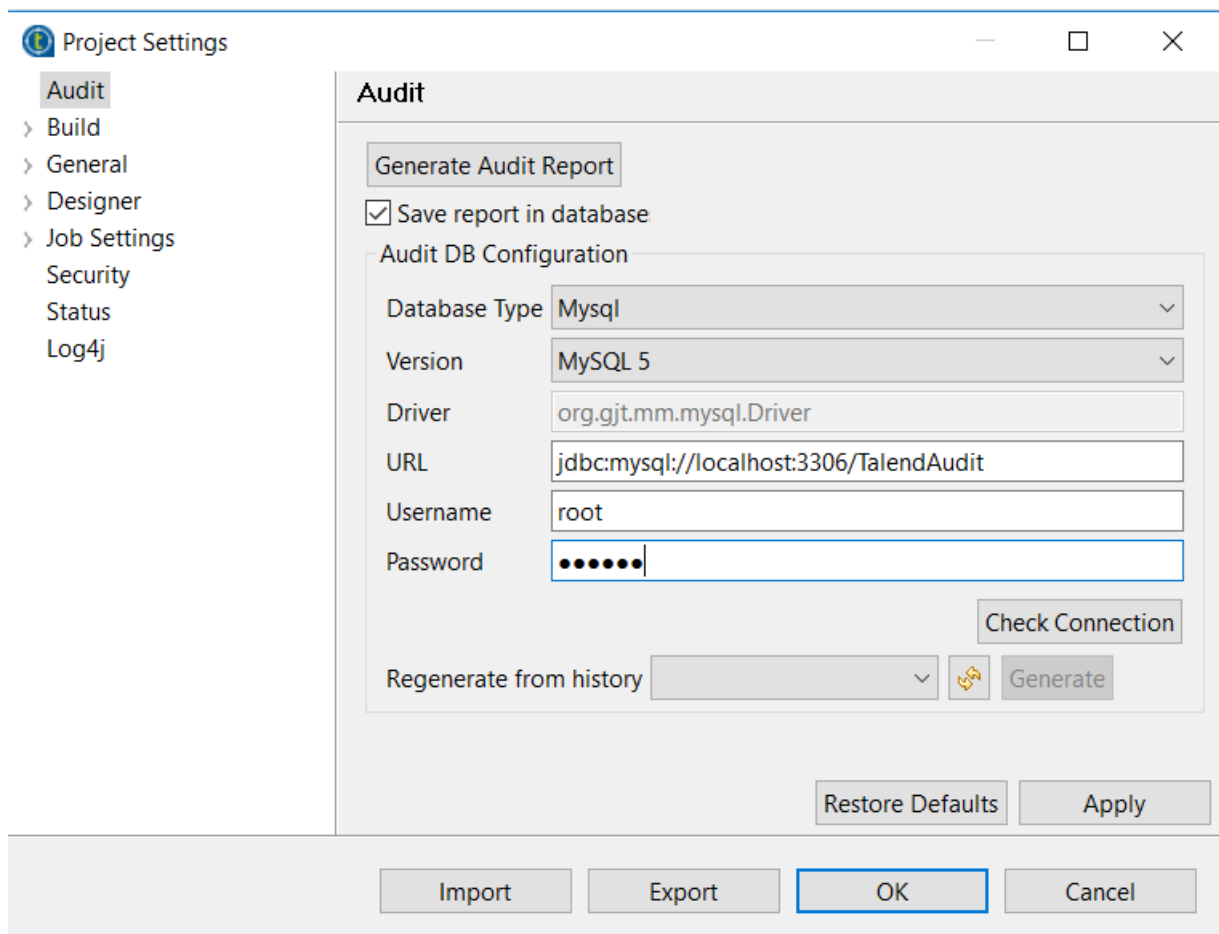
By default, Talend Studio provides an embedded H2 audit database. This is a one-shot database and is cleared once the audit action is complete. To track your project audit history, you need to define a custom database.

## Procedure

1. Click  on the toolbar of the Studio main window, or click **File > Edit Project properties** from the menu to open the **Project Settings** dialog box, and then click **Audit** to open the **Audit** view.
2. Select the **Save report in databases** check box.
3. Select your database type and version, and enter the database parameters in the corresponding fields, including the URL, username, and password.

**Warning:** Make sure that the specified database is empty or dedicated to auditing the current project. Otherwise, audit report generation will fail.

## Example



The screenshot shows the 'Project Settings' dialog box with the 'Audit' tab selected. The 'Generate Audit Report' button is visible at the top. Below it, the 'Save report in database' checkbox is checked. The 'Audit DB Configuration' section contains the following fields:

- Database Type: Mysql (dropdown)
- Version: MySQL 5 (dropdown)
- Driver: org.gjt.mm.mysql.Driver (text field)
- URL: jdbc:mysql://localhost:3306/TalendAudit (text field)
- Username: root (text field)
- Password: masked with dots (password field)

At the bottom of the 'Audit DB Configuration' section, there is a 'Check Connection' button. Below that, there is a 'Regenerate from history' dropdown menu, a 'Generate' button, and a 'Restore Defaults' button. At the very bottom of the dialog box, there are buttons for 'Import', 'Export', 'OK', and 'Cancel'.



4. Click **Check Connection** to validate your database connection.
5. Click **Apply** to validate your changes and **OK** to close the dialog box.

### Results

With the **Save report in databases** check box selected, when generating an audit report the audit data is saved to the database you defined.

## Talend Project Audit reports

Auditing specific concepts in a data integration project will result in generating a project analysis report. This report will provide you with information on almost all aspects of the audited project designed in your Talend Studio in order to help you make process-oriented decisions based on the results of the audit.

This chapter aims at helping the interpretation of a Talend Project Audit report. It discusses report content related to each and every audited aspect in a data integration project.

Talend Project Audit report is divided into one introductory section listing the properties of the audited project along with six other sections, each presenting quantitative and qualitative data about the investigated key area in the audited project.

The sections below discuss in details the audit results for every key area in the investigated project.

### Project properties

The table in this section lists identifying information about the audited project:

| Item                   | Description  |
|------------------------|--|
| <b>Project file</b>    | Path that specifies the location of the audited project              |
| <b>Label</b>           | The technical name (used by the system) of the actual project file   |
| <b>Description</b>     | User-defined description of the audited project                      |
| <b>Author</b>          | Login of the user who initially created the project in Talend Studio |
| <b>Product version</b> | The version of the Studio used to build the project to be audited    |

### Job/Component complexity

Talend Project Audit provides a method to assess the complexity of Jobs used in the audited project. The analysis incorporates numeric values for specific complexity elements that are weighted in their contribution to Job overall complexity.

A full description of findings is then presented in tables and charts.

### Project summary

The table in this section lists quantitative indicators of all elements used in the audited project.

| Item                   | Description                                     |
|------------------------|---|
| <b>Job count</b>       | Total number of Jobs used in the project.       |
| <b>Subjob count</b>    | Total number of subjobs used in the project     |
| <b>Component count</b> | Total number of components used in the project. |
| <b>Note count</b>      | Total number of notes used in the project       |

| Item                   | Description  |
|------------------------|--|
| Context variable count | Total number of context variable used in the project                                     |
| Schema count           | Total number of schemas used in the project  |
| Schema column count    | Total number of schema columns used in the project                                       |
| Complexity rating      | Complexity rating for the project that is equal to the sum of all individual Job ratings |

**Note:** Click the highlighted text (clickable text) in your Audit report to display the detailed referenced content without having to scroll down the pdf pages to find it.

## Job rating chart and details

This section provides visual representation of the investigated data related to the complexity rating of all Jobs in the audited project.

The chart represents quantitative and qualitative structures that show the number of Jobs used in the project along with their complexity rating, ranging from very simple to very complex.

Job complexity rating is calculated using numeric values specific to the elements complexity in the investigated Job. Examples of those elements are number of components used in the Job, number of context variables used in the Job, number of mapper components used in the Job and so on.

The list below gives the basic rating factors for calculating Job complexity:

- $\text{Components\_rating} = \text{simpleComponents\_rating} + \text{complexComponents\_rating}$ , where:
  - $\text{simpleComponents\_rating} = n * \text{number of simple components used in the Job}$ , where simple components refer to all the components other than those listed in the file `<studio>/plugins/org.talend.commandline.audit_<version>/complexComponents.properties`
  - $\text{complexComponents\_rating} = n * \text{number of complex components used in the Job}$ , where complex components refer to the components listed in the file `<studio>/plugins/org.talend.commandline.audit_<version>/complexComponents.properties`
- $\text{parameterCount\_rating} = n * \text{number of context variables used in the Job} * \text{number of contexts}$
- $\text{Transformer\_rating} = \text{mapCount\_rating} + \text{mapColumnsCount\_rating} + \text{mapInput\_rating} + \text{mapOutput\_rating}$ , where:
  - $\text{mapCount\_rating} = n * \text{number of the mapper components used in the Job}$ , where mapper components refer to **tMap** and **tXMLMap**
  - $\text{mapColumnsCount\_rating} = n * \text{number of columns of input and output schemas}$
  - $\text{mapInput\_rating} = n * \text{number of input connections in the mapper components(s) used in the Job}$
  - $\text{mapOutput\_rating} = n * \text{number of output connections in the mapper components(s) used in the Job}$
- $\text{customCodeComponents} = \text{number of components with Java code used in the Job}$ , such as **tJava**.

where "n" is a coefficient that helps balancing the actual "weight" of each criteria in the audit results. For the coefficient values, check the file `<studio>/plugins/org.talend.commandline.audit_<version>/rating.properties`.

**Warning:** Manual modification to the `.properties` files is discouraged as that will change project audit results, which may be required for a consultant to compare projects on site.

The formula for rating a Job is:

$$\text{Job\_Rating} = \text{Components\_rating} * 2 + \text{parameterCount\_rating} + \text{Transformer\_rating} + \text{customCodeComponents}$$

The resulting figures are defined as the following to give the different Job complexity ratings used:

- 0 - 50: very simple,
- 51 - 100: simple,
- 101 - 300: moderate,
- 301 - 500: complex,
- > 500: very complex.

The Job rating chart is accompanied (in the **Job rating - details** section in Talend Project Audit) by Job rating details providing numeric values for each complexity element shown.

The below table lists the numeric values for different complexity elements of the investigated Job.

| Item              | Description   |
|-------------------|---|
| <b>Identifier</b> | Name of the Job   |
| <b>Auth.</b>      | Email of the author of the Job                              |
| <b>Creation</b>   | Creation date of the Job                                    |
| <b>Update</b>     | Date of the last modification done on the Job               |
| <b>Status</b>     | Status of the Job   |
| <b>Version</b>    | Version of the Job  |
| <b>Components</b> | Number of components used in the Job                        |
| <b>Context</b>    | Number of context variables used in the Job                 |
| <b>Notes</b>      | Number of notes used in the Job                             |
| <b>Rating</b>     | Complexity rating for the Job based on the defined criteria |

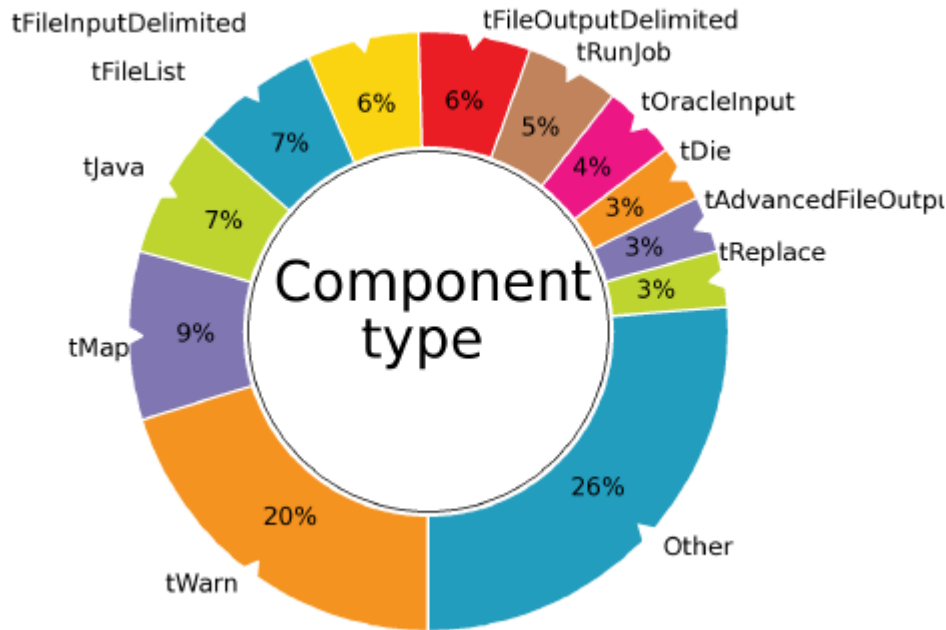
**Note:** Click the highlighted text (clickable text) to display the detailed referenced content without having to scroll down the pdf pages to find it.

## Component usage pie chart

The components pie chart (pie-shaped chart) is used to give a quick overall visual impression of the audit results regarding proportions of each of the component types used in the audited project.

It is split into segments, illustrating percentages of components based on their types. It uses those percentages or fractions to compare component types used in the project. The whole is equal to 100%.

Below is an example of a pie chart for an audited project.



**Note:** All component types with less than 2.5% proportion will be grouped under **Other**.

## Components usage details

The component pie chart is accompanied by component usage details providing numeric values for each component type shown on the pie chart.

The table below lists numeric values for components types.

| Item                  | Description  |
|-----------------------|--|
| <b>Component type</b> | Name of components used in the audited project   |
| <b>Count</b>          | Number of occurrences of the same component per project  |
| <b>Percentage</b>     | The fraction of 100 representing the frequency of each of the components used in the audited project |

**Note:** Click the highlighted text (clickable text) to display the detailed referenced content without having to scroll down the pdf pages to find it.

## Jobs grouped by components

The tables in this section provide lists of Jobs using the same components and indicate, at the bottom of each table, the total number of a component's occurrences per Job.

| Item         | Description   |
|--------------|---|
| <b>Job</b>   | Name of the Job that uses a specific component      |
| <b>Count</b> | Number of occurrences of the same component per Job |

**Note:** Click the highlighted text (clickable text) to display the detailed referenced content without having to scroll down the pdf pages to find it.

## Schema column chart and details

This section provides visual representation of the investigated data related to the complexity rating of the schema and schema column count used in the audited project.

The chart represents quantitative and qualitative structures to show in one instance the number of columns used in each schema along with complexity rating, ranging from *very low* to *very high*.

Schema column complexity rating is calculated using numeric values for the number of columns used in each schema in the audited project.

Schema column complexity ratings are defined as:

- 0 - 10: *very low*,
- 11 - 30: *low*,
- 31 - 60: *moderate*,
- 61 - 100: *high*,
- 101 - 999999: *very high*.

The schema column rating chart is accompanied (in the **Schema columns-details** section in Talend Project Audit report) by schema columns details providing numeric values for each element shown in the chart.

The table below lists the numeric values for column count per schema and schema count per project.

| Item                       | Description   |
|----------------------------|---|
| <b>Schema column count</b> | Number of columns per schema  |
| <b>Schema count</b>        | Number of schema per project, grouped by number of columns used in schema   |
| <b>Percentage</b>          | The fraction of 100 representing the frequency of each group of schemas (grouped by column count) used in the audited project |

## Jobs with 10 most schema column count

Another method used in Talend Project Audit for assessing the complexity of the audited project is counting schema columns in all Jobs in the investigated project and then grouping Jobs in groups relative to the 10 highest schema column figures.

Consequently, this section in Talend Project Audit report will provide 10 tables presented in ascending order. Each table groups all Jobs that hold the same figure among the 10 highest schema column figures.

Each of the 10 tables is preceded by the schema column figure used as the grouping factor for all Jobs listed in the table. The number of schemas used in each of the listed Jobs is given in a separate column.

| Item       | Description                            |
|------------|--|
| <b>Job</b> | Name of the Job in the audited project |

| Item         | Description              |
|--------------|--------------------------|
| Schema count | Number of schema per Job |

**Note:** This report is parameterized. It is possible to define the number used as grouping factors.

This type of investigation will help diagnosing system performance problems, for example through identifying the columns that are present in the Job design but not really used during Job execution.

An example of this is when you use the **Lookup** flow in your Job design. Usually, a lookup schema has numerous columns and you only use limited number in your Job.

## Jobs with 5 least schema column count

The method used here for assessing the complexity of the audited project is calculating schema columns in all the Jobs used in the project and then grouping Jobs in groups relative to the 5 lowest column figures.

Consequently, this section in Talend Project Audit report will provide 5 tables presented in ascending order. Each table groups all Jobs that hold the same figure among the 5 lowest schema column figures.

This type of investigation will help identifying Jobs where schema is not defined, for example.

Each of the 5 tables is preceded by the schema column figure used as a grouping factor for all Jobs listed in the table. The number of schemas used in each of the listed Jobs is given in a separate column.

| Item         | Description                                 |
|--------------|---|
| Job          | Name of the Job used in the audited project |
| Schema count | Number of schema per Job                    |

**Note:** This report is parameterized. It is possible to define the number of figures used as the grouping factor.

## Job error management

This section provides detailed information about trigger types and trigger numbers used in the investigated project. It also details what trigger types are used in what Jobs.

### Trigger usage in a project

The table in this section lists numeric values for trigger types used in the audited project.

| Item          | Description   |
|---------------|---|
| Trigger type  | Triggers can be any of the following types: <ul style="list-style-type: none"> <li>• <b>On Subjob Ok</b></li> <li>• <b>On Subjob Error</b></li> <li>• <b>Run if</b></li> <li>• <b>On Component Ok</b></li> <li>• <b>On Component Error</b></li> </ul> |
| Trigger count | Number of triggers per project  |
| Percentage    | The fraction of 100 representing the frequency of each trigger type in the audited project  |

## Trigger usage in Jobs

The table in this section lists numeric values for trigger types used in Jobs part of the audited project.

| Item         | Description   |
|--------------|---|
| Trigger type | Triggers can be any of the following types: <ul style="list-style-type: none"> <li>• <b>On Subjob Ok</b></li> <li>• <b>On Subjob Error</b></li> <li>• <b>Run if</b></li> <li>• <b>On Component Ok</b></li> <li>• <b>On Component Error</b></li> </ul>   |
| Job count    | Number of Jobs using at least one trigger type  |
| Percentage   | The fraction of 100 representing the frequency of each trigger type in a Job fraction relative to Job total number in the project.<br><br>For example, if <b>On Subjob Ok</b> is used in 25 Jobs where global Job count in the project is 154 and the number of Jobs using at least one trigger type is 48, the percentage can be figured by the proportion $25 \times 100 / 154$ which works out to be about 16.23 |

## Job documentation/versioning

This section provides detailed information about folder level, item type, item status, and item versioning used in the investigated project. It also details the usage of Job folders and Job versioning in the investigated project.

The audit results will show if you profit from the documentation and versioning functions in the Studio to correctly classify different types and versions of documents and Jobs in the investigated project.

### Documentation item type

The table in this section lists numeric values for item types used in the audited project.

| Item               | Description               |
|--------------------|---------------------------|
| Document item type | Type of the document item |



| Item       | Description   |
|------------|---|
| Item count | Number of items of the same type per project  |
| Percentage | The fraction of 100 representing the frequency of each item type in the audited project |

## Documentation folder level

The table in this section lists numeric values for folder hierarchy levels used in the audited project.

| Item                  | Description   |
|-----------------------|---|
| Document folder level | Numeric value that represents folder level: <ul style="list-style-type: none"> <li>• <b>0</b>: the item is not attached to a folder</li> <li>• <b>1</b>: the item is attached to a one-level-deep folder</li> <li>• <b>2</b>: the item is attached to a two-level-deep folder</li> <li>• ...</li> </ul> |
| Folder count          | Number of folders (of the same level) per project   |
| Percentage            | The fraction of 100 representing the frequency of each folder level in the audited project  |

## Documentation item status

The table in this section lists numeric values for different item statuses used in the audited project.

| Item                 | Description   |
|----------------------|---|
| Document item status | Status of the item used: <b>checked, unchecked, validated</b>                             |
| Item count           | Number of items of the same status per project  |
| Percentage           | The fraction of 100 representing the frequency of each item status in the audited project |

## Documentation item versioning

The table in this section lists numeric values for item versions used in the audited project.

**Note:** If you create many versions of the same item, only the last item version is taken into account.

| Item                     | Description  |
|--------------------------|--|
| Document item versioning | Last version of the added item   |
| Item count               | Number of items of the same version per project  |
| Percentage               | The fraction of 100 representing the frequency of each item version in the audited project |

## Job folder usage

The table in this section lists numeric values for Job folder hierarchy levels used in the audited project.

| Item                | Description  |
|---------------------|--|
| <b>Folder level</b> | Numeric value that represents Job folder level: <ul style="list-style-type: none"> <li>• <b>0</b>: the Job is not attached to a folder</li> <li>• <b>1</b>: the Job is attached to a one-level-deep folder</li> <li>• <b>2</b>: the Job is attached to a two-level-deep folder</li> <li>• ...</li> </ul> |
| <b>Job count</b>    | Number of Jobs of the same level per project   |
| <b>Percentage</b>   | The fraction of 100 representing the frequency of each Job level in the audited project  |

## Job versioning usage

The table in this section lists numeric values for Job versions used in the audited project.

**Note:** If you create many versions of the same Job, only the last Job version is taken into account.

| Item               | Description   |
|--------------------|---|
| <b>Job version</b> | Last version of each of the Jobs in the audited project                                   |
| <b>Job count</b>   | Number of Jobs of the same version in the audited project                                 |
| <b>Percentage</b>  | The fraction of 100 representing the frequency of each Job version in the audited project |

## Metadata

Talend Project Audit calculates the percentage of the usage of repository, property, and schema metadata in the investigated project.

The audit results will show if you use the **Metadata** repository integrated in Talend Studio to store your predefined metadata and thus be able to reuse it in different Jobs.

### Repository metadata

The table in this section lists numeric values for metadata types used in the audited project.

| Item                 | Description  |
|----------------------|--|
| <b>Metadata type</b> | <p>Metadata can be of one of the following types:</p> <ul style="list-style-type: none"> <li>• <b>DB connections</b></li> <li>• <b>File delimited</b></li> <li>• <b>SAP connections</b></li> <li>• <b>File Excel</b></li> <li>• <b>File regex</b></li> <li>• <b>File positional</b></li> <li>• <b>File Idif</b></li> <li>• <b>File xml</b></li> <li>• <b>LDAP schema</b></li> <li>• <b>WSDL schema</b></li> <li>• <b>Generic schema</b></li> <li>• <b>Salesforce schema</b></li> </ul> |
| <b>Count</b>         | Number of each of the above metadata types used in the audited project   |
| <b>Percentage</b>    | The fraction of 100 representing the frequency of each metadata type in the audited project  |

## Property metadata

The table in this section lists numeric values for property metadata types used in the audited project.

| Item                 | Description   |
|----------------------|---|
| <b>Metadata type</b> | <p>Property metadata can be of one of the following two types:</p> <ul style="list-style-type: none"> <li>• <b>Built in</b></li> <li>• <b>Repository</b></li> </ul> |
| <b>Count</b>         | Number of each of the above property metadata types used in the audited project   |
| <b>Percentage</b>    | The fraction of 100 representing the frequency of each property metadata type in the audited project  |

## Schema metadata

The table in this section lists numeric values for schema metadata types used in the audited project.

| Item                 | Description   |
|----------------------|---|
| <b>Metadata type</b> | <p>Schema metadata can be of one of the following two types:</p> <ul style="list-style-type: none"> <li>• <b>Built in</b></li> <li>• <b>Repository</b></li> </ul> |
| <b>Count</b>         | Number of each of the above schema metadata types used in the audited project   |
| <b>Percentage</b>    | The fraction of 100 representing the frequency of each schema metadata type in the audited project  |

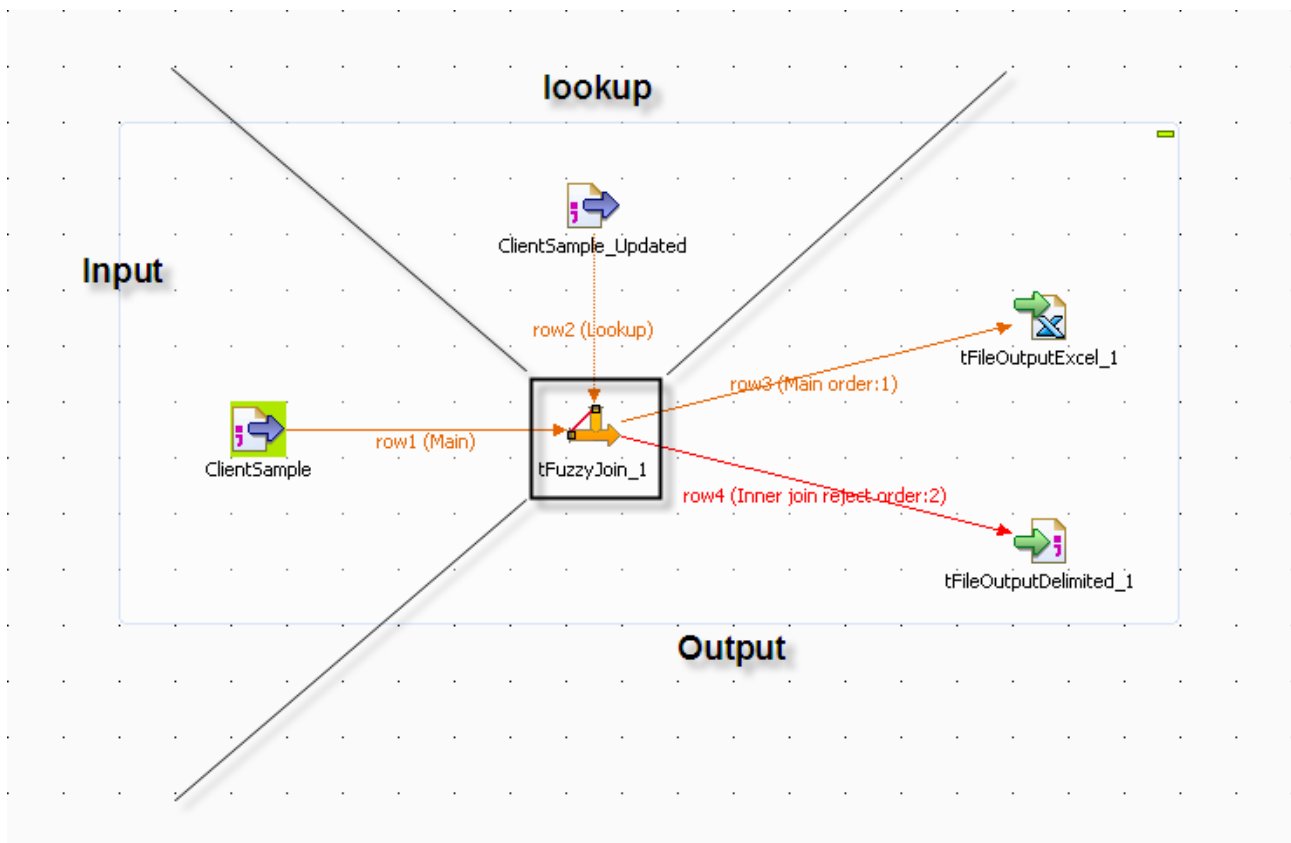
## Layout

The placement of different types of components in a Job design is very important to simplify understanding the process in the Job.

In a Job design, all types of components are placed in relation to the main flow component. Input components should be placed to the left, output components should be placed to the right, and a subflow component with a **Lookup** link should come higher above.

To assess the quality of the placement of different components in Jobs part of the audited project, Talend Project Audit analyzes misplaced input and output components and components with a **Lookup** links in subjobs.

The below figure illustrates an example of the placement of different components in a specific Job.



A full description of findings is then presented in a table.

## Misplaced components

The table in this section lists numeric values for misplaced components used in subjobs:

| Item             | Description  |
|------------------|--|
| Job              | Name of Job  |
| Subjob           | Name of subjob used in the specified Job   |
| Input component  | Number of input components in the specified subjob that are not placed to the left   |
| Lookup component | Number of subflow components with a <b>lookup</b> link that are not placed above the main flow component in the specified subjob |

| Item                    | Description  |
|-------------------------|--|
| <b>Output component</b> | Number of output components in the specified subjob that are not placed to the right |

**Note:** Click the highlighted text (clickable text) in your Audit report to display the detailed referenced content without having to scroll down the pdf pages to find it.

## Job analysis

To assess the quality and the importance of all Jobs used in the audited project, Talend Project Audit conducts a Job analysis depending on certain synthetic elements, like number of components, schemas and triggers used in every Job of the audited project. The results of the investigation of these synthetic elements are weighted in their contribution to Job overall importance.

Full description of findings is then presented in separate tables, one table per Job. These tables are listed in descending order: more important to less important.

### Job synthetic view

Each table in this section lists numeric values on different synthetic elements of the investigated Job:

| Item             | Description   |
|------------------|---|
| <b>Indicator</b> | Investigated synthetic elements can be one of the following types: <ul style="list-style-type: none"> <li>• <b>Job rating</b></li> <li>• <b>Component count</b></li> <li>• <b>Schema column count</b></li> <li>• <b>Context variable count</b></li> <li>• <b>Job notes count</b></li> <li>• <b>Misplaced input component</b></li> <li>• <b>Misplaced lookup component</b></li> <li>• <b>Misplaced output component</b></li> <li>• <b>Trigger usage</b></li> </ul> |
| <b>Value</b>     | Number of each of the above investigated elements in the investigated Job of the audited project  |

**Note:** Click the highlighted text (clickable text) in your Audit report to display the detailed referenced content without having to scroll down the pdf pages to find it.