talend | Connecting the Data-Driven Enterprise

Talend Activity Monitoring Console

User Guide

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 User Guide explains how to manage *Talend Activity Monitoring Console* functions in a normal operational context.

Information presented in this document applies to *Talend Activity Monitoring Console* **6.4.1**.

1.2. Audience

This guide is for users and administrators of *Talend Activity Monitoring Console*.

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. Introduction to Talend Activity Monitoring Console

Talend Activity Monitoring Console is an add-on tool integrated in the studio and in *Talend Administration Center* for monitoring **Talend** Jobs and projects.

Talend Activity Monitoring Console helps **Talend** product administrators or users to achieve enhanced resource management and improved process performances through a convenient graphical interface and a supervising tool.

Talend Activity Monitoring Console provides detailed monitoring capabilities that can be used to consolidate the collected activity monitoring information, understand the underlying component and Job interaction, prevent faults that could be unexpectedly generated and support system management decisions.

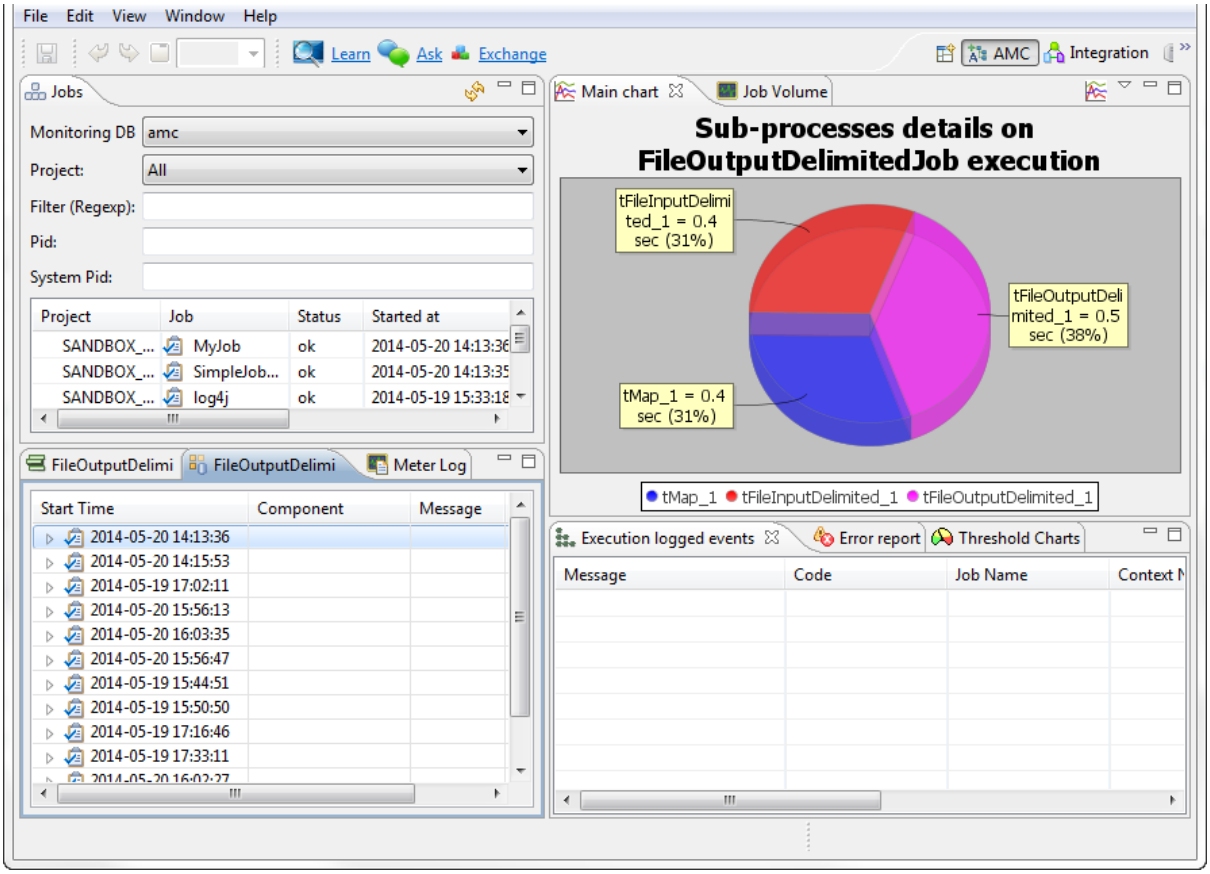
For more information about how to access *Talend Activity Monitoring Console*, see *Accessing the monitoring console*.

For more information about the graphical user interface (GUI) of *Talend Activity Monitoring Console*, see *GUI of the monitoring console*.

1.1. GUI of the monitoring console

The *Talend Activity Monitoring Console* interface consists of the following views:

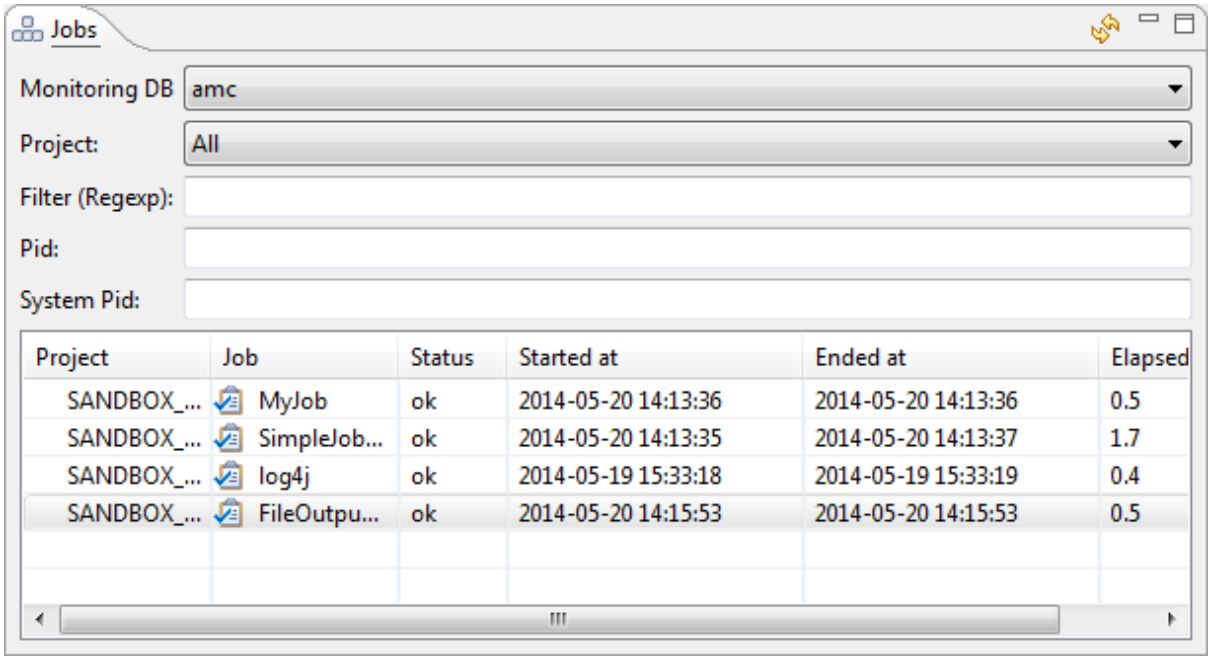
- **Jobs** view,
- **History** and **Detailed history** views,
- **Meter log** view,
- **Main chart** view,
- **Job Volume** view,
- **Logged Events** view,
- **Error report** view,
- **Threshold Charts** view.



You can customize the views to filter the activity monitoring information and display only the information of interest in *Talend Activity Monitoring Console*. For more information about how to customize the views, see *Customizing the monitoring console*.

1.1.1. Jobs view

The **Jobs** view provides the list of Jobs mentioned in the execution log data collected.



On this view, select a particular Job to display its execution details.

The table provides the following information: the **Project** name, the **Job** name, the **Start** and **End** times as well as the Execution **Elapsed time** and the execution **Status**. All this information can be used in regular expressions to drill down to a particular Job instance.

An icon in front of the Job name indicates the completion status. The events related to the Job selected is shown on the **Detailed history** view.

The **History**, **Logged Events** and **Main Chart** views help you have an overall picture of the Job execution selected.

The following analysis-support graphical items can help you to improve your **Talend** processes where needed:

- The **Main Chart** view provides a graph representing the various execution instances of the Job selected in the **Jobs** view.
- The **Job volume** view shows a line chart graph illustrating the variations of the selected Job data flow over the time.
- The **Execution Logged Events** view provides the warning and error messages generated during the Job execution.
- The **Error report** view provides an error-analysis chart.
- The **Threshold Charts** view shows a proportional distribution of the processed flow in the Job selected.

For more information on those views, see:

- *History and Detailed history views*
- *Meter Log view*
- *Logged Events view*
- *Error report view*
- *Main Chart view*

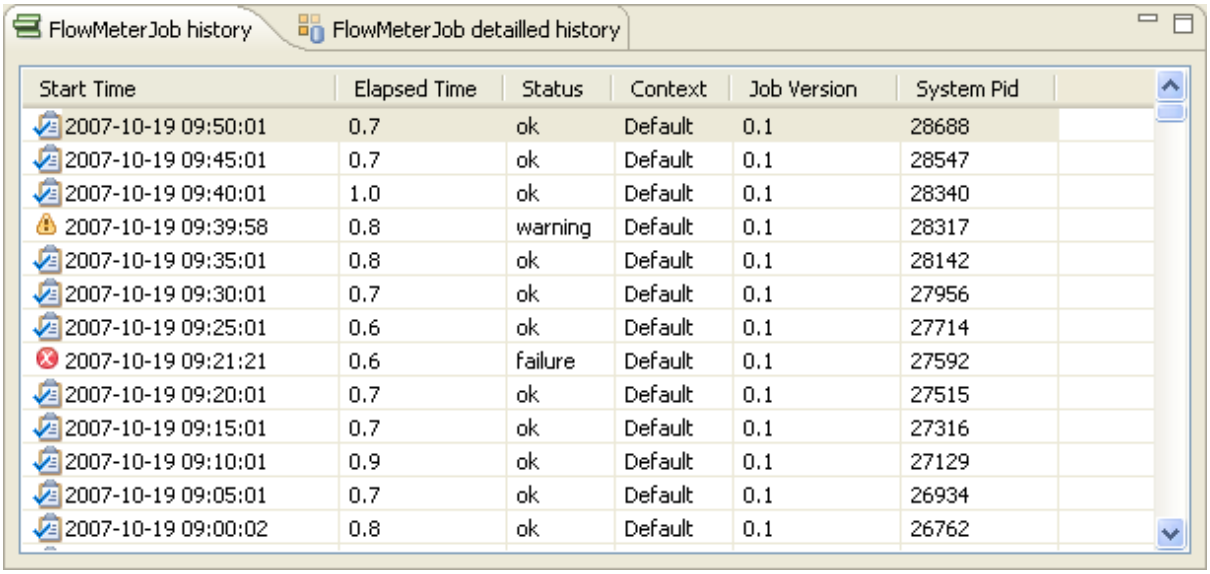
1.1.2. History and Detailed history views

The execution history is shown on two views: **History** and **Detailed History**.

- The **History** view provides a summary of the Job main steps.
- The **Detailed History** view splits up each Job into components and provides the execution details.

1.1.2.1. History view

The **History** view gathers all execution instances of the Job which you selected on the **Jobs** view. The execution instances are ordered following their **Start time** and **System PID**. The latest execution instance is shown at the top of the list.



Start Time	Elapsed Time	Status	Context	Job Version	System Pid
2007-10-19 09:50:01	0.7	ok	Default	0.1	28688
2007-10-19 09:45:01	0.7	ok	Default	0.1	28547
2007-10-19 09:40:01	1.0	ok	Default	0.1	28340
2007-10-19 09:39:58	0.8	warning	Default	0.1	28317
2007-10-19 09:35:01	0.8	ok	Default	0.1	28142
2007-10-19 09:30:01	0.7	ok	Default	0.1	27956
2007-10-19 09:25:01	0.6	ok	Default	0.1	27714
2007-10-19 09:21:21	0.6	failure	Default	0.1	27592
2007-10-19 09:20:01	0.7	ok	Default	0.1	27515
2007-10-19 09:15:01	0.7	ok	Default	0.1	27316
2007-10-19 09:10:01	0.9	ok	Default	0.1	27129
2007-10-19 09:05:01	0.7	ok	Default	0.1	26934
2007-10-19 09:00:02	0.8	ok	Default	0.1	26762

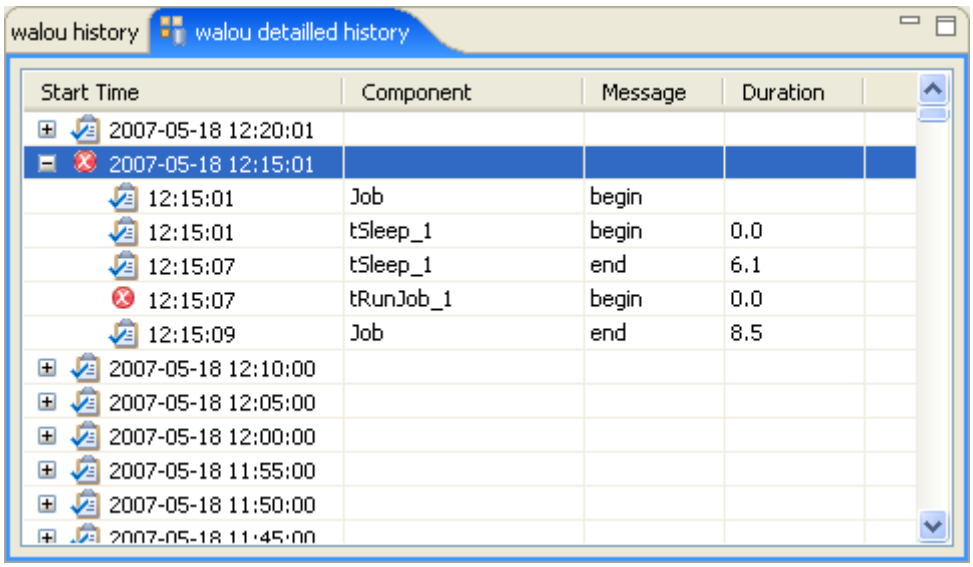
This view provides generic execution information which includes: **Start time**, **Elapsed time**, **Status**, **Context**, **Job version** and **System PID**.

The **Status** column displays the state of the Job execution instance for each of the steps and shows any possible error the Job may have fallen in or failure that may have occurred.

If any warning message was generated while executing the Job, the error icon would show in front of the execution instance or step. And the full message would be displayed on the **Execution Logged Events** tab.

1.1.2.2. Detailed History view

The **Detailed History** view is also based on the **Start Time** and **PID** of each Job execution. Similarly, the latest execution instance is shown at the top of the list.



Click on the relevant entry to expand the node and display the details.

The details include the name of each component involved in the instance selected and the corresponding message (**begin** or **end**). These components appear in a logical sequence based on the order, and each component is run in the Job.

Make sure the **tStatCatcher Statistics** box is checked in the **Advanced settings** tab of each component that you want to let show on *Talend Activity Monitoring Console*.

For all components, the execution **message** is displayed. If one message (most likely the **end** message) is missing, this often corresponds to an execution error, that should be illustrated with the red icon.

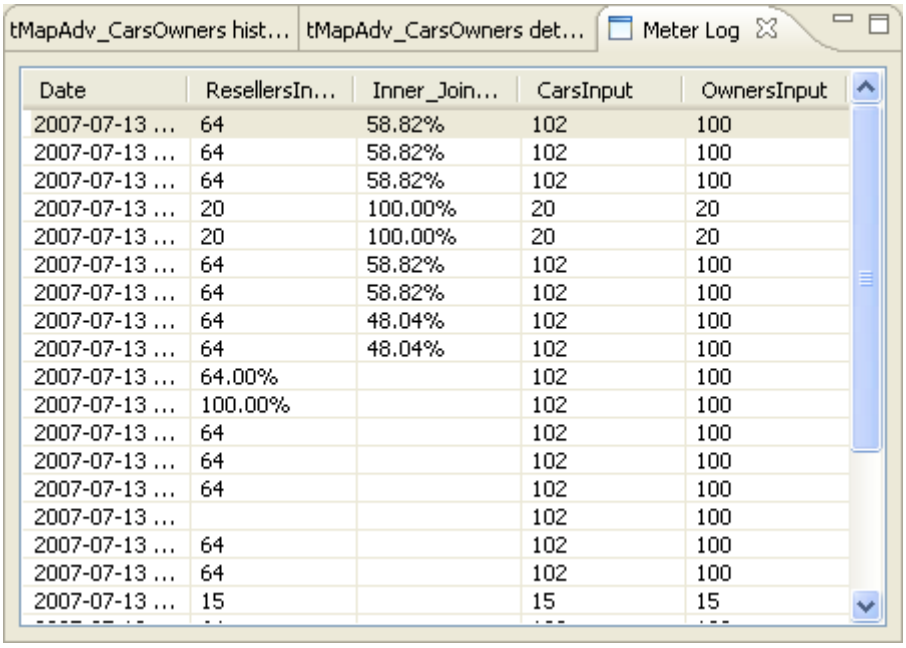
Also the component execution duration is displayed (**Duration**).

The **Main Chart** view displays a pie chart representing for each component, its respective share of the execution time.

For more information, see *Main Chart view*.

1.1.3. Meter Log view

The **Meter Log** view displays the detailed information of the various flows processed in the Job selected on the **Jobs** view.



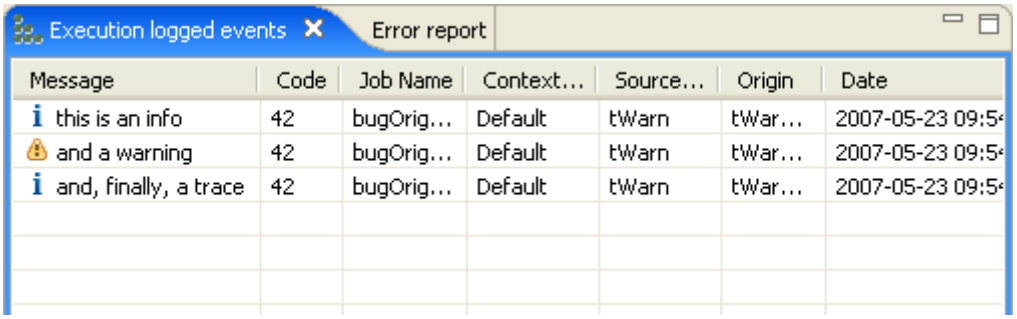
Date	ResellersIn...	Inner_Join...	CarsInput	OwnersInput
2007-07-13 ...	64	58.82%	102	100
2007-07-13 ...	64	58.82%	102	100
2007-07-13 ...	64	58.82%	102	100
2007-07-13 ...	20	100.00%	20	20
2007-07-13 ...	20	100.00%	20	20
2007-07-13 ...	64	58.82%	102	100
2007-07-13 ...	64	58.82%	102	100
2007-07-13 ...	64	48.04%	102	100
2007-07-13 ...	64	48.04%	102	100
2007-07-13 ...	64.00%		102	100
2007-07-13 ...	100.00%		102	100
2007-07-13 ...	64		102	100
2007-07-13 ...	64		102	100
2007-07-13 ...	64		102	100
2007-07-13 ...			102	100
2007-07-13 ...	64		102	100
2007-07-13 ...	64		102	100
2007-07-13 ...	15		15	15

The number of columns displayed depends on the **tFlowMeter** components used in the selected Job.

The additional column **Date** provides you with the date and time stamp of the execution of the Job.

1.1.4. Logged Events view

The **Logged Events** view displays in full the messages generated through **tWarn** or **tDie** components as well as **Java Exception**.



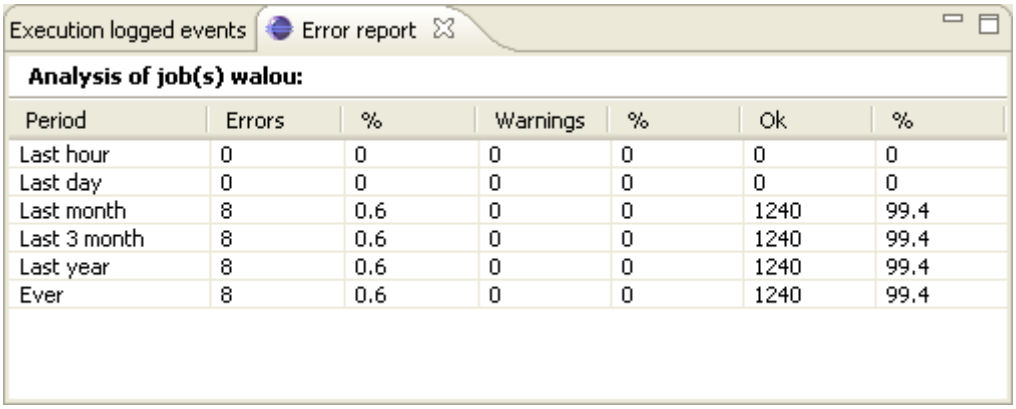
Message	Code	Job Name	Context...	Source...	Origin	Date
this is an info	42	bugOrig...	Default	tWarn	tWar...	2007-05-23 09:5
and a warning	42	bugOrig...	Default	tWarn	tWar...	2007-05-23 09:5
and, finally, a trace	42	bugOrig...	Default	tWarn	tWar...	2007-05-23 09:5

It provides the following information: **Message**, **Code**, **Job Name**, **Context**, **Source type**, **Origin** and **Date**.

Message	Message displayed is defined in the Properties of the **tWarn** and **tDie** components in **Talend** or automatic messages of **Java Exception**.
Code	Code level you defined in the **Properties** of the **tWarn** or **tDie** component
Job Name	Name of the Job where the message is generated from.
Context	Name of the context if a specific context has been defined for the Job other than **Default**.
Source type	Type of message component. It can be **tWarn**, **tDie** or **Java Exception** according to the Properties you set and the message type.
Origin	Name of the component where the message originated from.
Date	Moment of the Job execution.

1.1.5. Error report view

The **Error Report** view provides an analysis of the proportion of errors that occurred over a number of Job executions.



Period	Errors	%	Warnings	%	Ok	%
Last hour	0	0	0	0	0	0
Last day	0	0	0	0	0	0
Last month	8	0.6	0	0	1240	99.4
Last 3 month	8	0.6	0	0	1240	99.4
Last year	8	0.6	0	0	1240	99.4
Ever	8	0.6	0	0	1240	99.4

The report covers a defined range of time periods and provides the number of **Errors**, **Warnings** or **OK** messages that have been generated as well as the corresponding percentage.

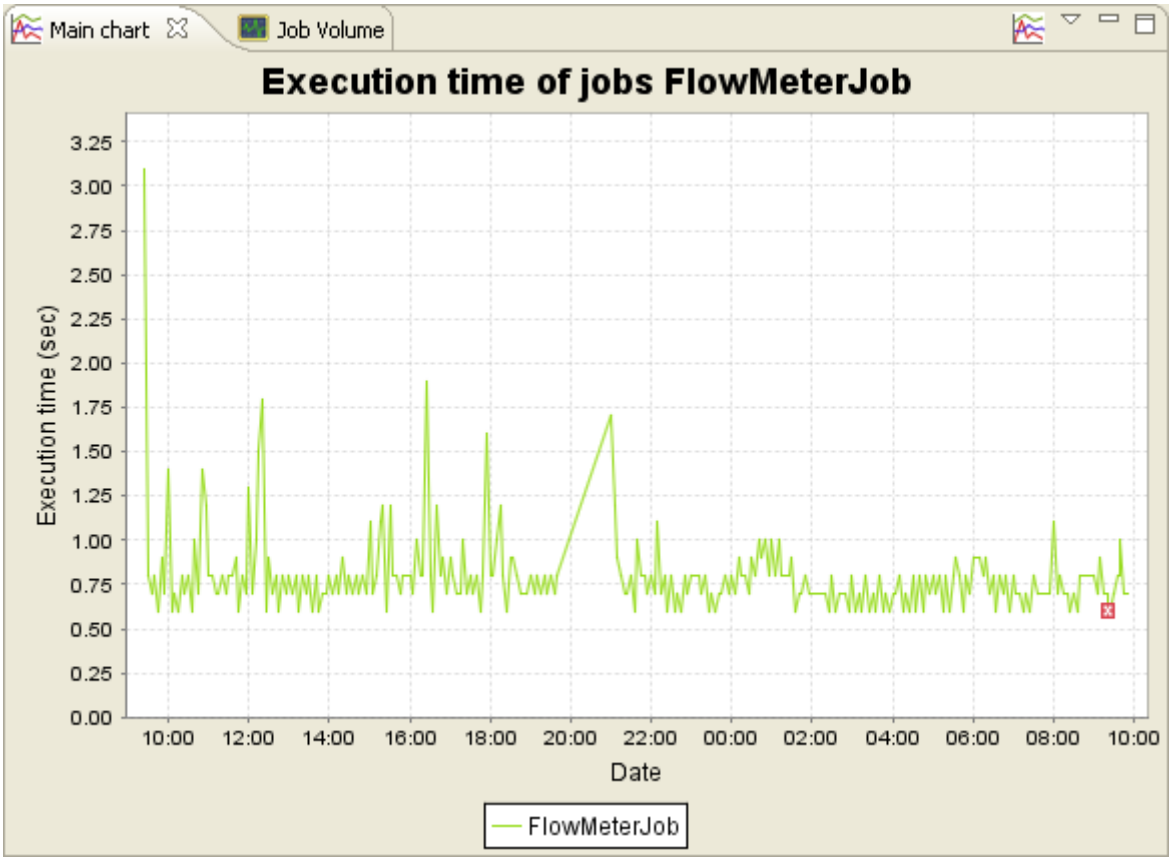
1.1.6. Main Chart view

On the **Main Chart** view of the *Talend Activity Monitoring Console* window, various charts are provided to illustrate the Job instance you selected. These charts help you to quickly and intuitively understand the behaviour of **Talend** Jobs illustrated by the collected log data.

The chart varies depending on the Job selected on the **Jobs** view or on the **Detailed History** view.

1.1.6.1. Line chart (standalone Job)

Select a Job instance on the **Jobs** view to display on the **Main Chart** view, the **line chart** corresponding to the various Job execution points plotted on the graph and joined together by a line.

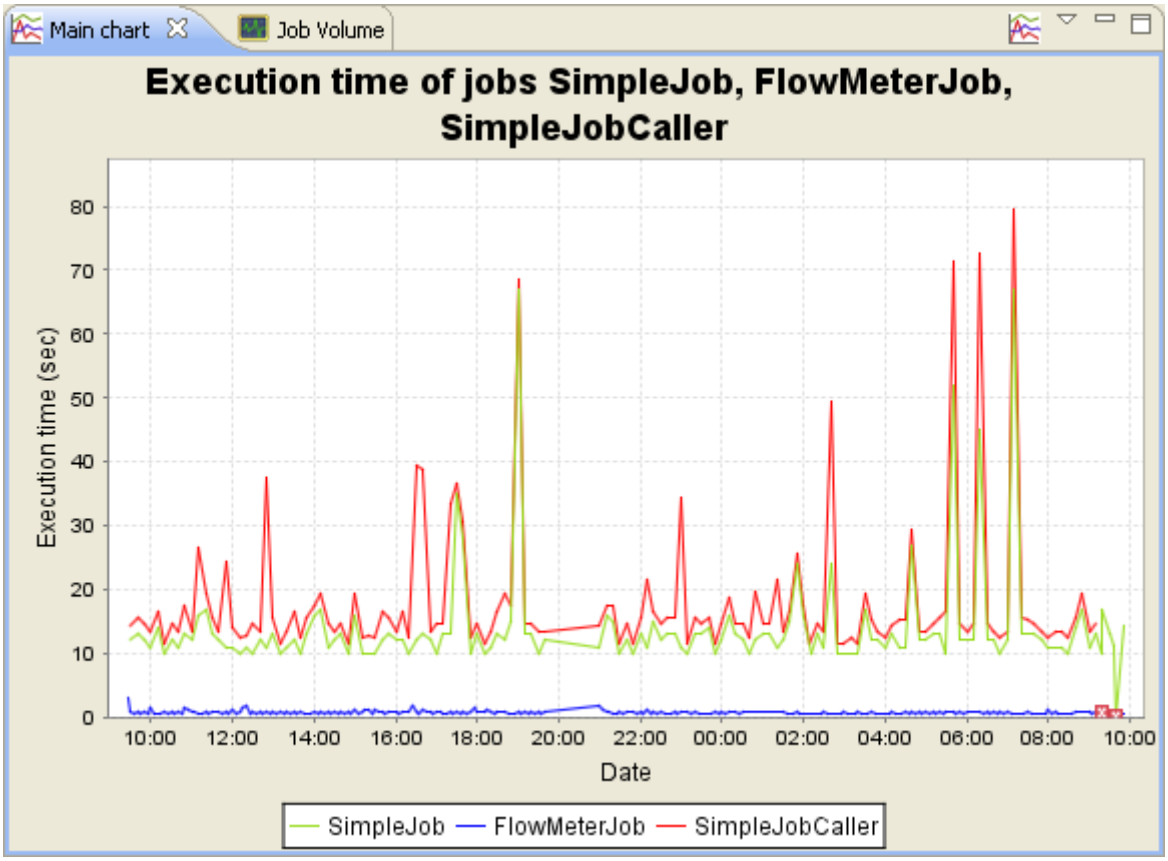


The vertical (y) axis of the chart corresponds to the scale of time (in seconds) needed for the execution of the Job selected.

The horizontal (x) axis of the chart corresponds to the moment (date and time) of each execution.

Multiple-line charts

On the **Jobs** view, select several Job instances using the **Shift** key, to display a **multiple-line chart** on the **Main Chart** view.



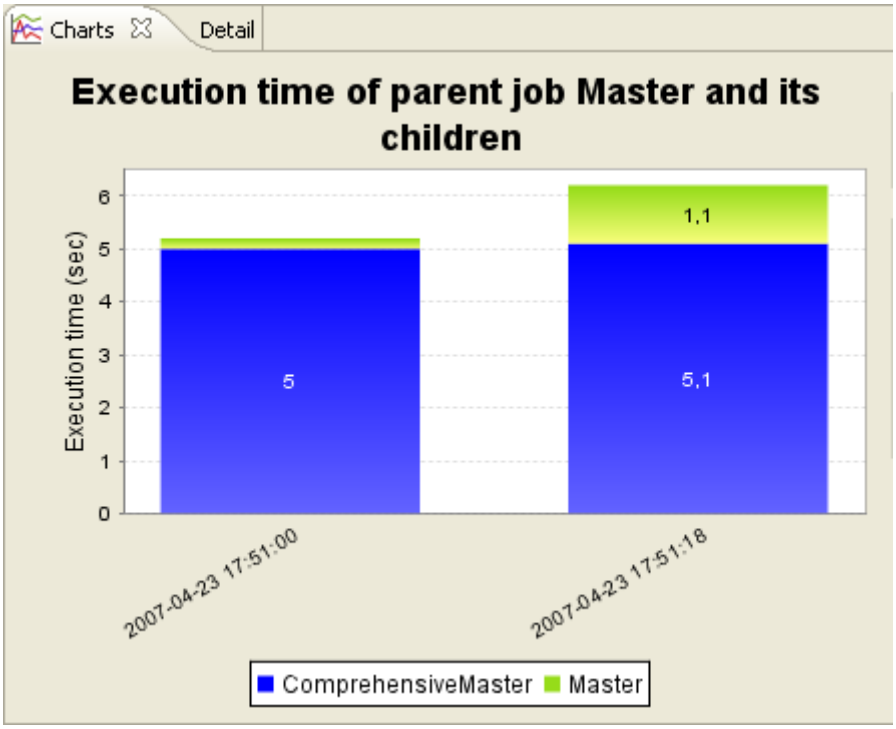
Each instance shows in a different color, defined in a caption at the bottom of the chart.

1.1.6.2. Stacked bar chart (Parent Job)

A parent Job is a Job that runs one or several subjobs. These subjobs are called child Jobs. The parent Job (or master Job) contains the **tRunJob** component.

If the Job you selected on the **Job Information** table is a parent Job, the **Charts** view changes to a **Bar chart** stacking all child Jobs above the parent Job.

This **Stacked bar graph** shows obvious differences in respective time execution and thus help comparing the master Jobs and its child Jobs.

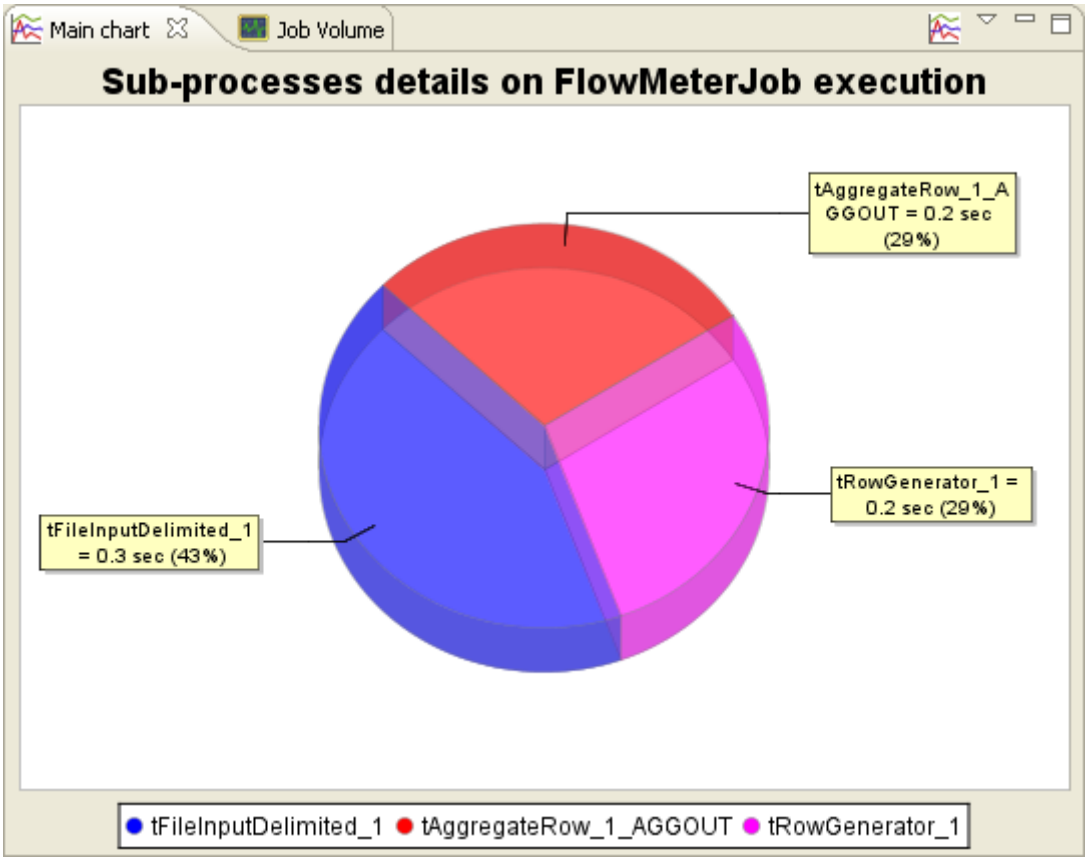


The vertical (y) axis of the chart shows the scale of the time (in seconds) elapsed for the overall execution of the master Job selected including its related child Jobs.

The horizontal (x) axis of the chart shows the moment (date and time) of each execution allowing you to see the evolution and possible changes over a set amount of days.

1.1.6.3. Pie chart (components)

In the **Detailed History** view, select an instance of the Job execution. The **Main Chart** view displays the time share taken by each component in the overall Job execution time.



The pie chart is thus divided among all components composing the Job and having the **tStatCatcher statistics** option checked (on in their respective **Component** view of your *Talend Studio*).

Each component shows in a different color, defined in a caption at the bottom of the chart. A detailed information about each share is provided next to the pie chart.

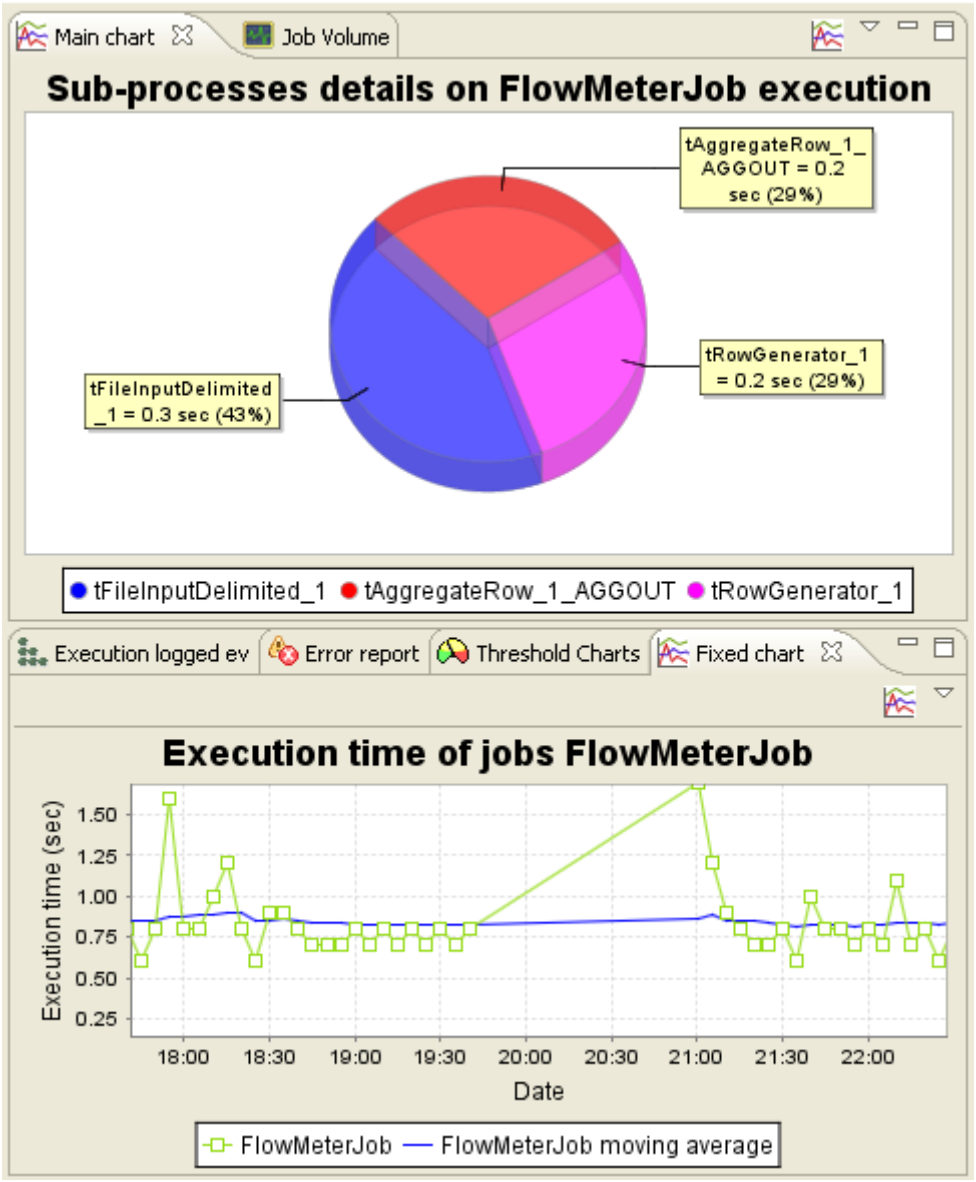
If the Job instance is a parent Job, the **pie chart** shows the execution time share between each children Job if you selected the *tStatCatcher Statistics* check box on each of **tRunJob** component calling in a new child Job.

1.1.6.4. Multiple-chart views

You can have several **Main Chart** views opened at the same time on your *Talend Activity Monitoring Console*, in order to overlay different aspects of your Jobs in a mosaic of charts. This feature removes the need for manually switching between individual charts.

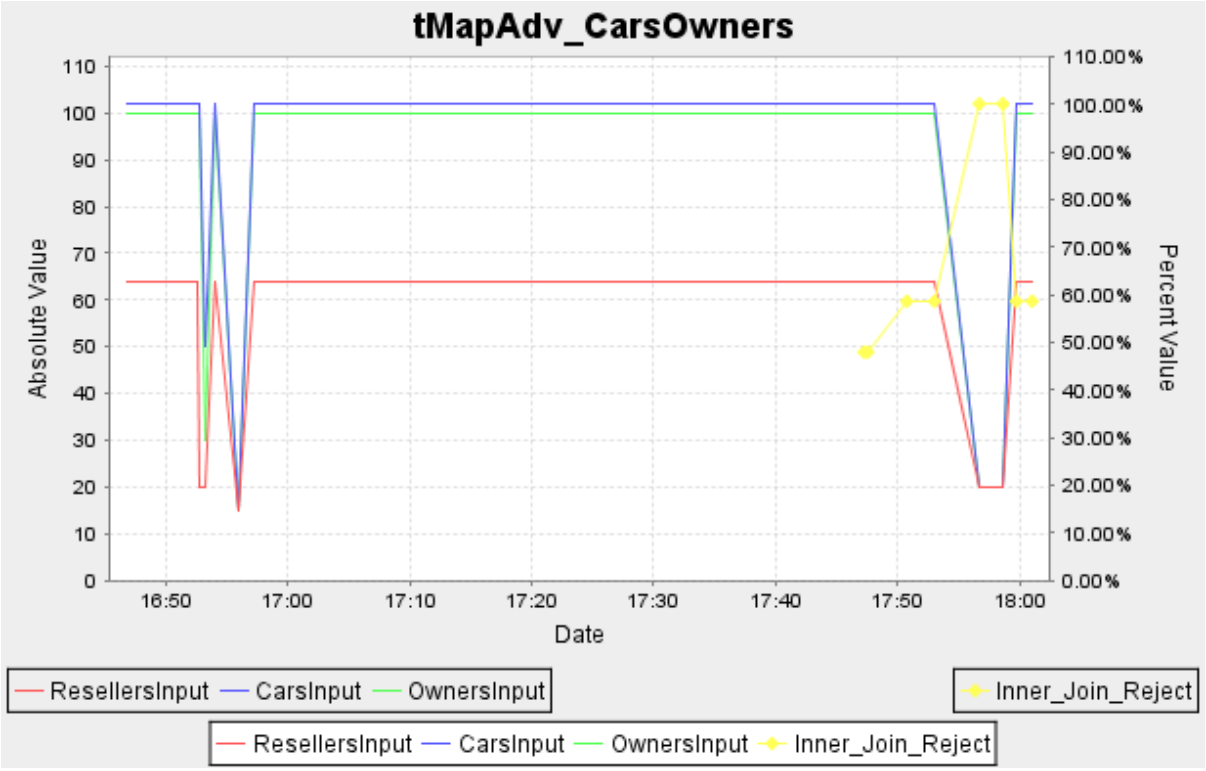
To duplicate the **Main Chart** view, click on the **Chart** button (**Open as a fixed chart**) at the top right hand side.

The **Fixed Chart** view opens up at the top of the window, but cannot be moved around as you can do with the **Main Chart** view.



1.1.7. Job Volume view

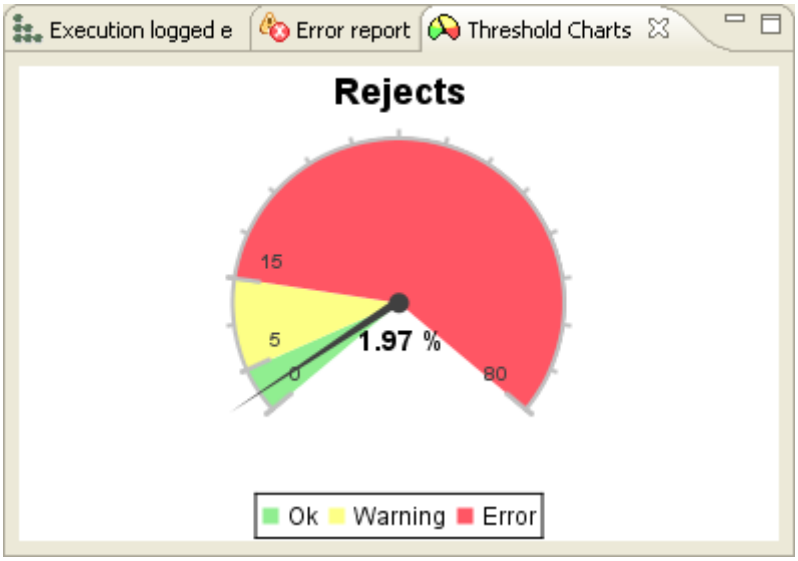
The **Job Volume** view displays a line chart representing the volumetrics of the flow being processed. Depending on the **tFlowMeter** component settings on your **Talend** Job design, the scale and units will differ.



Each flow tracked down using a **tFlowMeter** component will be displayed in a different color on the **Job Volume** chart. The **Relative Value** flows depend on the **Percent Values** axis showing on the right side of the chart whereas the **Absolute Value** flows follow the left range axis.

1.1.8. Threshold chart view

The **Threshold** chart view displays the proportion of the flow measured.



The chart displays a speedometer showing various thresholds that you defined on the **tFlowmeter** component, helping to visualize the acceptable or non acceptable flow volume thresholds.

Chapter 2. Before using Talend Activity Monitoring Console

Activity monitoring information can be stored in delimited files or database tables.

Before collecting and reusing the activity monitoring information of your **Talend** Jobs, you have to:

- Create files or database tables to be used as datasources for the activity monitoring information. For more information, see *Creating files or database tables*.
- Enable activity monitoring either by configuring the **Stats & Logs** settings at the project level or Job level or by adding the relevant components to your Jobs in order to catch and record the activity monitoring information and deliver it to the defined output (files or database tables). For more information, see *Enabling activity monitoring*.
- Configure the datasources to retrieve the activity monitoring information, which can be displayed on *Talend Activity Monitoring Console* either from the studio or from the **Monitoring** module of *Talend Administration Center*. For more information, see *Configuring datasources for the monitoring console*.

In *Talend Administration Center*, only database tables are supported as datasources. You can connect to these tables via the **Connections** area in the top of the **Activity Monitoring Console** page in the web application.

For more information about the database supported by *Talend Activity Monitoring Console*, see the section about compatible databases in the *Talend Installation Guide*.

2.1. Creating files or database tables

Talend Activity Monitoring Console is an application that allows you to monitor Job executions. The Job executions are monitored using three files or database tables that relate to the following data:

- collection of logs,
- component statistics,
- data flow volumes.

To store this data, you need to create three files or database tables respectively using the schema of the **tLogCatcher**, **tStatCatcher**, **tFlowMeterCatcher** components (present in the **Palette** of your Talend Studio).

The files can be empty, and their structure will be generated automatically if you use files as datasources to store the activity monitoring information.

The installation of the <*AMC*> Database and the use of the *Talend Activity Monitoring Console* in the studio is optional for *Talend ESB* and is only required if the **tFlowMeterCatcher**, **tLogCatcher** and **tStatCatcher** components are used. These components can be used in Jobs (for example, ESB Consumer jobs) for REST and Soap ESB Service Providers. Only the **tLogCather** is supported for now as the other components are not fully supporting the "Keep listening" environments, and Routes currently can not use these components.

To create the database tables to be used as data sources for the monitoring console, do the following:

1. Create a database that you name *amc*, for example.
2. Launch *Talend Studio*.
3. Create a Job that contains three **tCreateTable** components.
4. Define the connection parameters to the *amc* in the three **tCreateTable** components.
5. Define the data structure by taking the schema provided in the components: **tLogCatcher**, **tStatCatcher** and **tFlowMeterCatcher**.

   Note that after creating the **tStatCatcher**, **tLogCatcher** and **tFlowMeterCatcher** tables, you have to make sure the **moment** data type is set to datetime instead of date.

To create these tables in a MySQL database, you can also use the following script:

```
-- MySQL dump 10.13  Distrib 5.5.24, for Win32 (x86)
--
-- Host: 127.0.0.1    Database: amc
-- ------------------------------------------------------
-- Server version 5.5.24-log

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--
-- Current Database: `amc`
--

CREATE DATABASE /*!32312 IF NOT EXISTS*/ `amc` /*!40100 DEFAULT CHARACTER SET latin1 */;

USE `amc`;
```

```
--
-- Table structure for table `tflowmetercatcher`
--

DROP TABLE IF EXISTS `tflowmetercatcher`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `tflowmetercatcher` (
  `moment` datetime DEFAULT NULL,
  `pid` varchar(20) DEFAULT NULL,
  `father_pid` varchar(20) DEFAULT NULL,
  `root_pid` varchar(20) DEFAULT NULL,
  `system_pid` bigint(8) DEFAULT NULL,
  `project` varchar(50) DEFAULT NULL,
  `job` varchar(255) DEFAULT NULL,
  `job_repository_id` varchar(255) DEFAULT NULL,
  `job_version` varchar(255) DEFAULT NULL,
  `context` varchar(50) DEFAULT NULL,
  `origin` varchar(255) DEFAULT NULL,
  `label` varchar(255) DEFAULT NULL,
  `count` int(3) DEFAULT NULL,
  `reference` int(3) DEFAULT NULL,
  `thresholds` varchar(255) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `tflowmetercatcher`
--

LOCK TABLES `tflowmetercatcher` WRITE;
/*!40000 ALTER TABLE `tflowmetercatcher` DISABLE KEYS */;
/*!40000 ALTER TABLE `tflowmetercatcher` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `tlogcatcher`
--

DROP TABLE IF EXISTS `tlogcatcher`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `tlogcatcher` (
  `moment` datetime DEFAULT NULL,
  `pid` varchar(20) DEFAULT NULL,
  `root_pid` varchar(20) DEFAULT NULL,
  `father_pid` varchar(20) DEFAULT NULL,
  `project` varchar(50) DEFAULT NULL,
  `job` varchar(255) DEFAULT NULL,
  `context` varchar(50) DEFAULT NULL,
  `priority` int(3) DEFAULT NULL,
  `type` varchar(255) DEFAULT NULL,
  `origin` varchar(255) DEFAULT NULL,
  `message` varchar(255) DEFAULT NULL,
  `code` int(3) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `tlogcatcher`
--

LOCK TABLES `tlogcatcher` WRITE;
/*!40000 ALTER TABLE `tlogcatcher` DISABLE KEYS */;
/*!40000 ALTER TABLE `tlogcatcher` ENABLE KEYS */;
UNLOCK TABLES;

--
```

```
-- Table structure for table `tstatcatcher`
--

DROP TABLE IF EXISTS `tstatcatcher`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `tstatcatcher` (
  `moment` datetime DEFAULT NULL,
  `pid` varchar(20) DEFAULT NULL,
  `father_pid` varchar(20) DEFAULT NULL,
  `root_pid` varchar(20) DEFAULT NULL,
  `system_pid` bigint(8) DEFAULT NULL,
  `project` varchar(50) DEFAULT NULL,
  `job` varchar(255) DEFAULT NULL,
  `job_repository_id` varchar(255) DEFAULT NULL,
  `job_version` varchar(255) DEFAULT NULL,
  `context` varchar(50) DEFAULT NULL,
  `origin` varchar(255) DEFAULT NULL,
  `message_type` varchar(255) DEFAULT NULL,
  `message` varchar(255) DEFAULT NULL,
  `duration` bigint(8) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `tstatcatcher`
--

LOCK TABLES `tstatcatcher` WRITE;
/*!40000 ALTER TABLE `tstatcatcher` DISABLE KEYS */;
/*!40000 ALTER TABLE `tstatcatcher` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;
```

Now that the *Talend Activity Monitoring Console* database is created, you can monitor the Job executions either from *Talend Studio* or from the **Monitoring** page of *Talend Administration Center*.

To do so, you simply need to tell *Talend Studio* and *Talend Administration Center* about the connection parameters to these tables. For more information, refer to *Configuring datasources for the monitoring console*.

2.2. Enabling activity monitoring

To reuse the activity monitoring information and display it on *Talend Activity Monitoring Console*, you need to enable activity monitoring using either the **Stats & Logs** settings at the project or Job level or the **tStatCatcher**, **tLogCatcher** and **tFlowMeterCatcher** components for an individual Job.

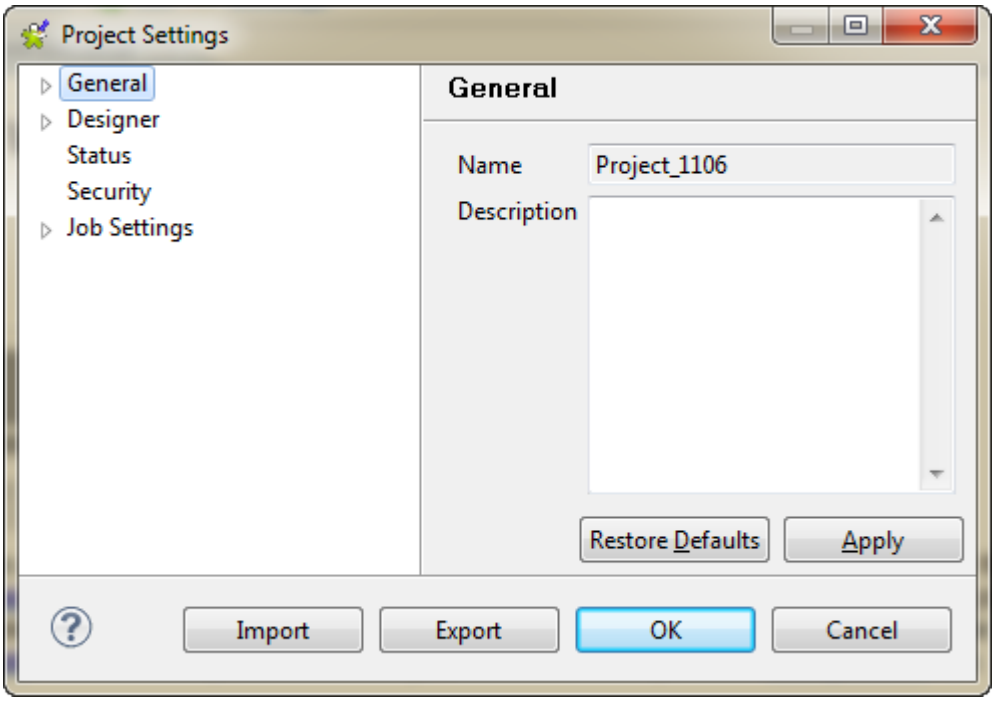
The use of the **Stats & Logs** settings is recommended to prevent your Job from being overloaded with numerous components.

2.2.1. Enabling activity monitoring for a project

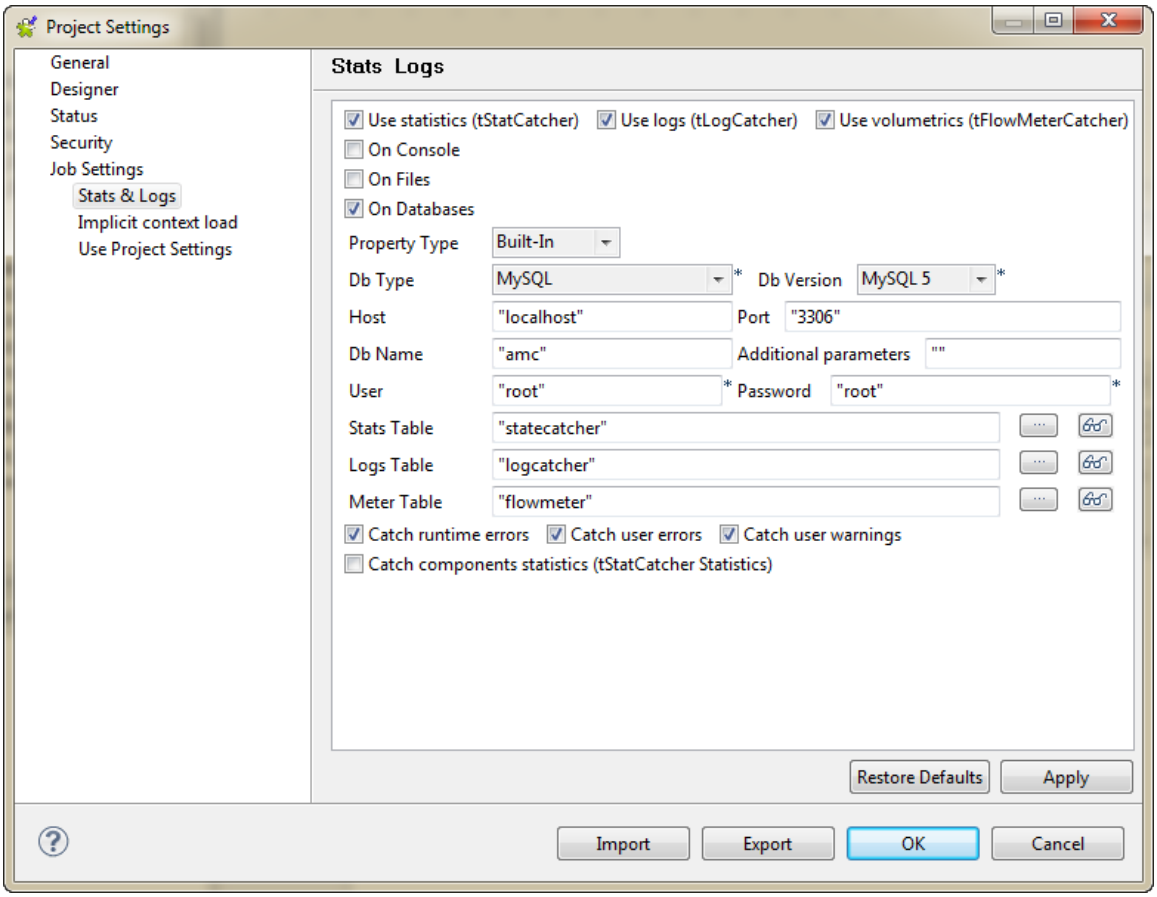
To enable activity monitoring for a project in your studio, do the following:

1. Select **File** > **Edit Project properties** from the menu to open the **[Project Settings]** dialog box.

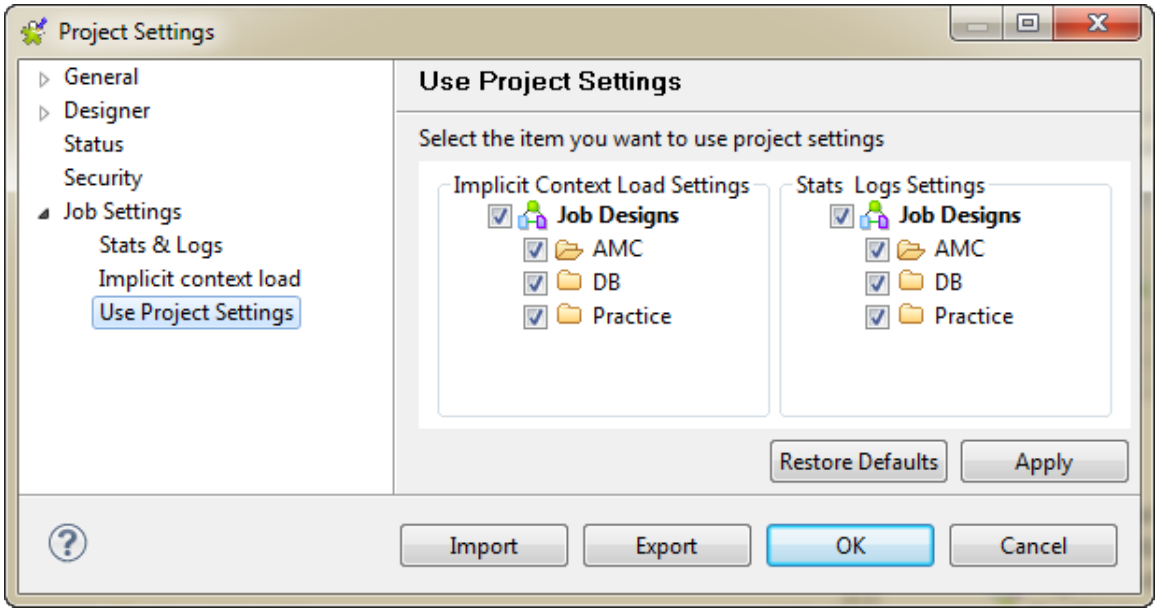
   Alternatively, you can click  to open the dialog box directly.



2. Expand the **Job Settings** node and select **Stats & Logs** to open the **Stats Logs** page. Then, customize the configuration according to your needs.



3. If needed, click **Use Project Settings** and select the Jobs to which you want to apply the settings.



For more information on project settings, see the section on customizing project settings of your *Talend Studio User Guide*.

2.2.2. Enabling activity monitoring for a Job

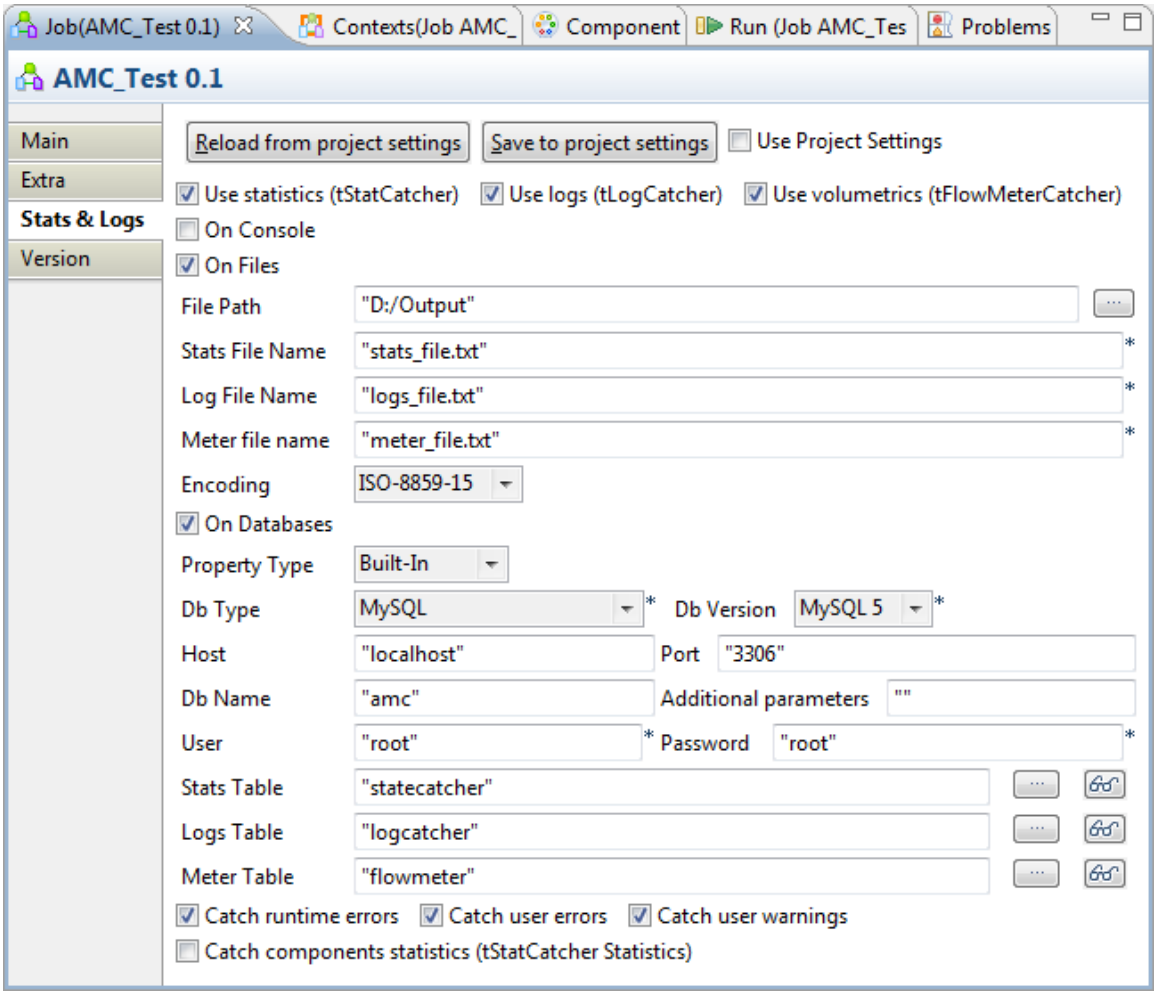
You can enable activity monitoring for a Job using either Job settings or components.

2.2.2.1. Using Job settings

Proceed as follows:

1. Open the Job in your *Talend Studio*, go to the **Job** tab view, and click **Stats & Logs**.

2. If the **Use Project Settings** check box is selected, clear it. Then, customize the configuration according to your needs.



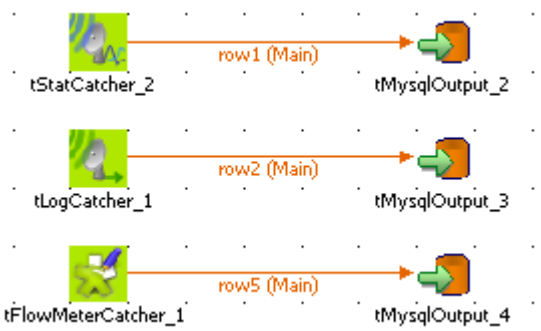
For more information on Job settings, see the section on automating the use of statistics & logs of your *Talend Studio User Guide*.

2.2.2.2. Using components

Proceed as follows:

1. Add the **tStatCatcher**, **tLogCatcher** and **tFlowMeterCatcher** components as needed to your Job.

2. Link them to the relevant output (either files or database tables).



tLogCatcher, **tStatCatcher** and **tFlowMeterCatcher** are the typical components required to record respectively various logs, statistics or flow information that will be interpreted and displayed on *Talend Activity Monitoring Console*.

- **tLogCatcher** can be triggered by a Java exception, **tWarn** or **tDie**.
- **tStatCatcher** can be triggered by the **tStatCatcher Statistics** check box in a Job or individual components.
- **tFlowMeterCatcher** can be triggered by **tFlowMeter**.

More specifically,

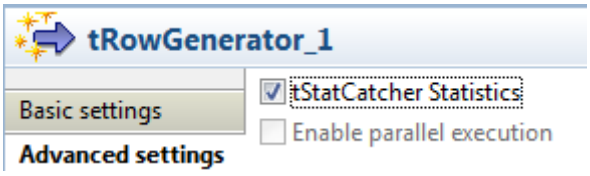
- **tWarn** can be configured to display or log a warning or other notification messages describing the behaviour of a Job without any impact on the Job progress. **tWarn** is usually part of a subjob dedicated to the log.
- **tDie** is used to stop the Job or subjob. This component is commonly used for error handling. **tDie** is usually part of a subjob to catch the message related to the die operation (set by the user at the component-level) and deliver it to the files or database.
- **tFlowMeter** is used to measure the number of rows being processed. This ratio or absolute number can be logged to feed the files or database to set up charts. **tFlowMeter** has to be part of the Job and directly plugged to the **Row** > **Main** connections composing the Job to measure the flow.

If you want to avoid using the **tFlowMeter** component, see *Enabling connection monitoring*.

For more information regarding those components, see the *Talend Components Reference Guide*.

2.2.3. Enabling component statistics

To trace the behaviour of each component involved in the Job in *Talend Activity Monitoring Console*, select the **tStatCatcher Statistics** check box in the **Advanced settings** view of the relevant components of your Job.

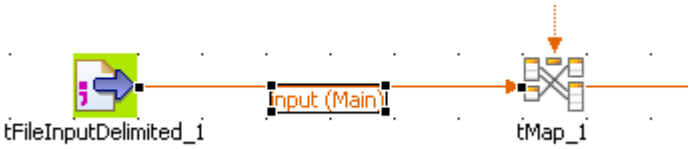


2.2.4. Enabling connection monitoring

In order to ease the reading of a complex Job based on numerous connections, you can use the connection settings to avoid using the **tFlowMeter** component.

Proceed as follows:

1. Select the connection linking the two components that you want to monitor.



2. On the **Component** view, select the **Advanced settings** tab.
3. Select the **Monitor this connection** check box.

4. Fill in the required parameters, as detailed in the **tFlowMeter** section of the *Talend Components Reference Guide*.

2.3. Configuring datasources for the monitoring console

In *Talend Studio*, the **Stats & Logs** settings or the **tStatCatcher**, **tLogCatcher** and **tFlowMeterCatcher** components help you easily collect the typical activity monitoring data into the files or the database tables of your choice.

In order to reuse the data in *Talend Activity Monitoring Console*, you have to configure the datasources.

To access the configuration page, select **Window** > **Preferences** from the menu, then expand the **AMC** node and select **Datasource Type**.

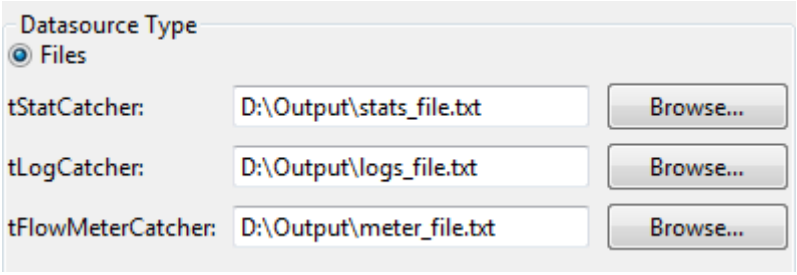
Note that only database tables are supported as datasources in *Talend Administration Center*. For more information, see *Database*.

2.3.1. Files

To use files as datasources, select **Files** and specify the files you created in the corresponding fields.

In the area, carry out the following operations:

- In the **tStatCatcher** field, browse to the file from which the statistics data will be retrieved.
- In the **tLogCatcher** field, browse to the file from which the log data will be retrieved.
- In the **tFlowMeterCatcher** field, browse to the file from which the data flow volumes will be retrieved.



If you use project settings or Job settings for activity monitoring, make sure that the files selected here match what you have configured in the **Stats & Logs** settings. For more information about the project settings or Job settings. For more information, see *Enabling activity monitoring*.

If you use the **tStatCatcher**, **tLogCatcher** and **tFlowMeterCatcher** components for activity monitoring, make sure that the connected Output components are of delimited type (**File > Output** folders, on the **Palette**).

2.3.2. Database

To use database tables as datasources, select **Database** and define the database and tables.

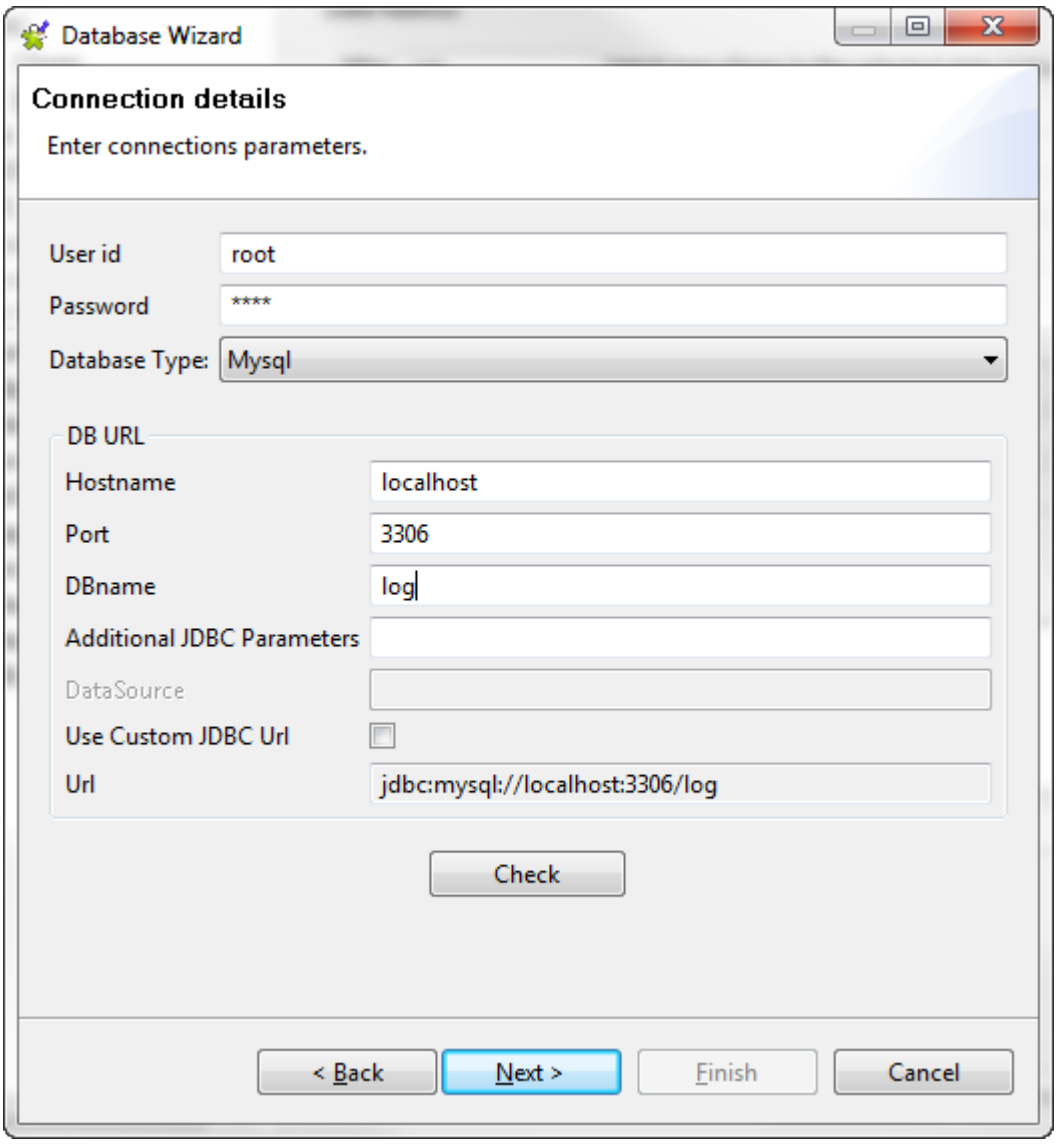
For more information about the databases and their versions supported by *Talend Activity Monitoring Console*, see the section about compatible databases in the *Talend Installation Guide*.

To add a new connection, proceed as follows:

1. Click **Add** to open the **[Database Wizard]** dialog box.

2. Type in a name for the bookmark in the **Name** field, which will show on the list of database connections available in the **[Preferences]** window. Click **Next** to continue.

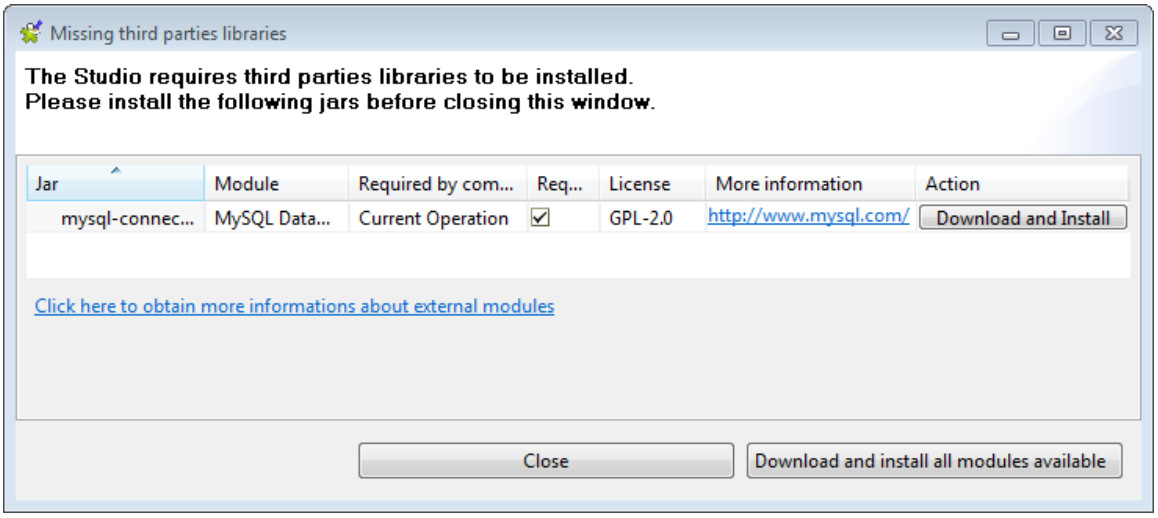
3. Fill in the **User id** and **Password** fields. Select the relevant **Database Type** from the list. Then, type in specific database URL information: **Hostname, Port**, and **DBname**.



If you want to use a custom JDBC URL, select the **Use Custom JDBC Url** check box and fill the **Url** field with the relevant JDBC URL.

4. Click **Check** to check the connection.

   If a library or driver (*.jar* file) is missing, a dialog box is displayed inviting you to install it.



When the required library or driver is installed, close the window and click **Check** again to check the connection parameters.

The Studio provides multiple approaches to automate the installation. For further information, see the chapter describing how to install external modules of the *Talend Installation Guide*.

5. Click **Next** to continue.

6. Click **Select** to define in the list the **tStatCatcher**, **tLogCatcher** and **tFlowMeterCatcher** tables where the activity monitoring information will be retrieved from.

   An error message will appear in case of any connection error or failure.

7. Click **Finish** to validate your configuration.

If you use project settings or Job settings for activity monitoring, make sure that the database tables selected here match what you have configured in the **Stats & Logs** settings. For more information about the project settings or Job settings, see *Enabling activity monitoring*.

If you use the **tStatCatcher**, **tLogCatcher** and **tFlowMeterCatcher** components for activity monitoring, make sure that the connected **Output** components are of database type (**Databases** folder, on the **Palette**).

As mentioned before, in *Talend Administration Center*, only database tables are supported as datasources. To define the connection parameters to the database tables from the Web application (*Talend Administration Center*), do the following:

1. Select the **Monitoring** menu, click **Activity Monitoring Console** then add a connection to the **Connections** list.

2. Set the connection parameters to the *LogCatcher* and *StatCatcher* tables.

   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 the Talend Administration Center User Guide. So, you have to connect to Talend Administration Center as an Operation Manager to be able to configure the Activity Monitoring Console monitoring database connection information.

For more details on how to use the **Monitoring** module, refer to the *Talend Administration Center User Guide*.

Chapter 3. Using Talend Activity Monitoring Console

This chapter first introduces how to access *Talend Activity Monitoring Console* and then describes how to customize the different views in the GUI of the *Talend Activity Monitoring Console*.

3.1. Accessing the monitoring console

You are allowed to access *Talend Activity Monitoring Console* from your *Talend Studio* and from the *Talend Administration Center* Web application.

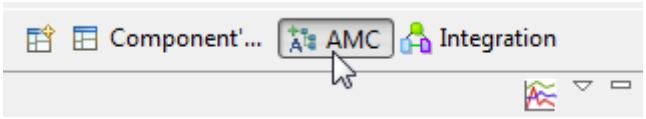
The following sections explain the steps to access *Talend Activity Monitoring Console* from your *Talend Studio* and from the *Talend Administration Center* Web application.

3.1.1. Accessing the monitoring console from the studio

You can monitor your **Talend** Job activities from your *Talend Studio* when either files or database tables are used as the datasources.

To access *Talend Activity Monitoring Console* through the studio interface, select **Window** > **Perspective**> **AMC** from the menu.

Alternatively, click the **AMC** button in the upper right corner of the studio.



3.1.2. Accessing the monitoring console from Talend Administration Center

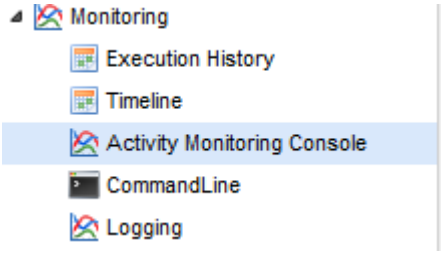
In *Talend Administration Center*, *Talend Activity Monitoring Console* is a Web application that has to be deployed in the Tomcat server before it can appear in the *Talend Activity Monitoring Console* menu. For more information, see the *Talend Installation Guide*.

You can monitor your **Talend** Job activities from *Talend Administration Center* when database tables are used as the datasources.

To access *Talend Activity Monitoring Console* through the *Talend Administration Center* interface for the first time, proceed as follows:

Prerequisite: Make sure that the AMC URL has been defined on the **Configuration** page of *Talend Administration Center* before. For more information, see *Talend Administration Center User Guide*.

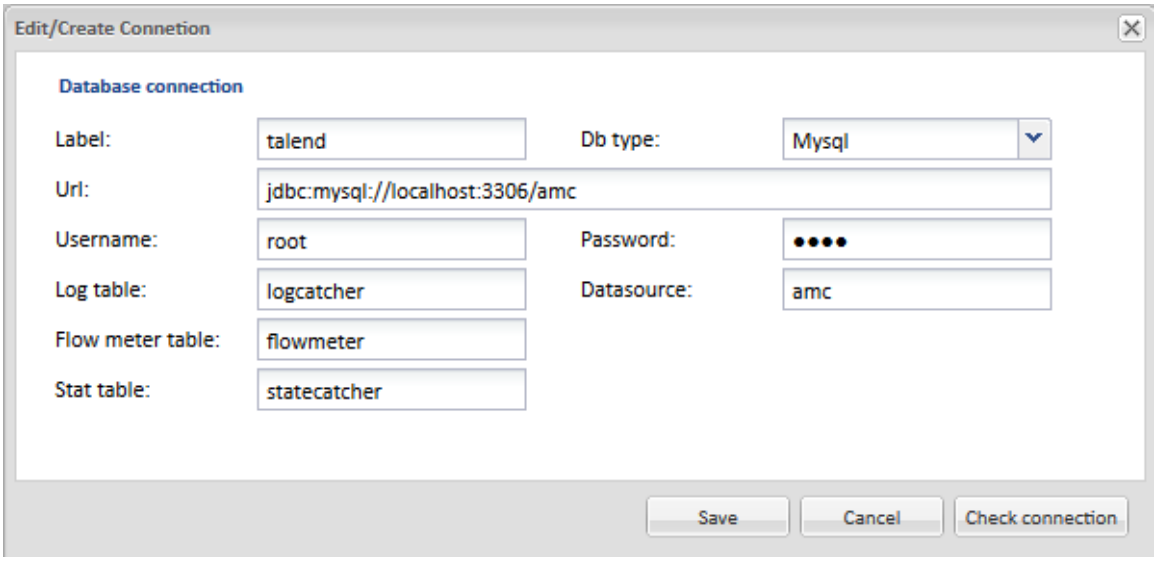
1. In *Talend Administration Center*, expand the **Monitoring** node. Then, click **Activity Monitoring Console**.



2. Next to the **Connections** list on the **Activity Monitoring Console** page, click the **[+]** button to open the **[Edit/Create Connection]** configuration window.

Later, you can click to edit the connection or to delete the connection if needed.

By default, the Monitoring node allows you to connect to the MySQL database. If you want to establish a connection to any other database, you must first put the corresponding jar file in the Tomcat lib folder. Otherwise, your database connection cannot be initialized.

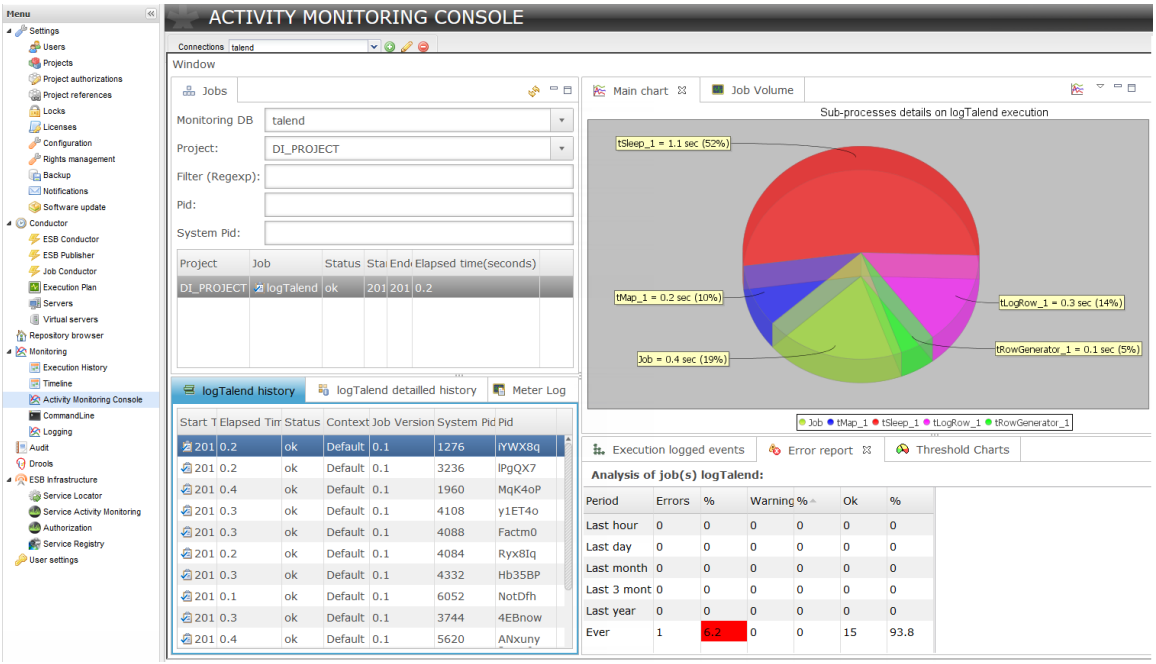


3. Enter the relevant connection information.

Field	Description
Label	Name of the connection entry in the connection list.
DB type	The database type to which you want to connect.
URL	URL address the server hosting the log database, including the host, port and the name of the log database to connect to.
Datasource	If required, set the data source for the selected database.
Username	Login name to the database.
Password	Password to the database.
Log table	Name of the table gathering the information caught through the **tLogCatcher** component in the Job.
Flow meter table	Name of the table gathering the information caught through the **tFlowMeterCatcher** component in the Job.
Stat table	Name of the table gathering the information caught through the **tStatCatcher** component in the Job.

4. Click **Save** to add the database connection to the **Connections** list. You can click **Check connection** to test the database connection or click **Cancel** to close the window without saving any changes.

5. Select the connection from the **Connections** list. The corresponding view is displayed with the connection.



If more than one connection exists in the **Connections** list, select the active connection of the database where you save the log tables and then view the monitored information.

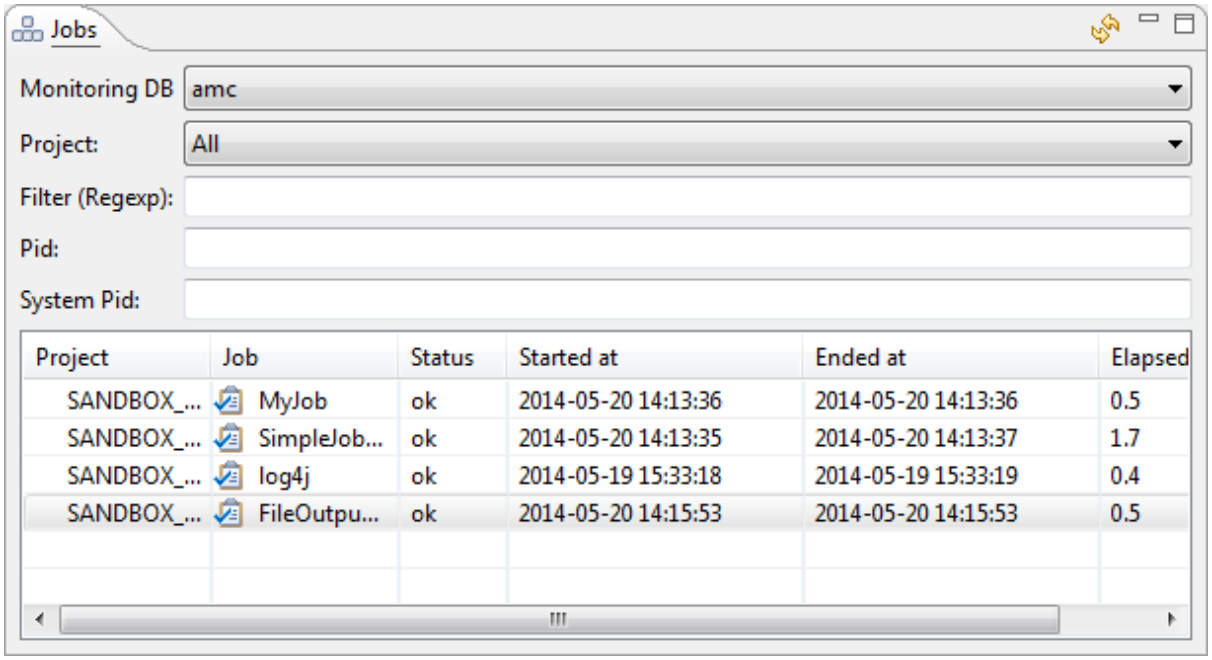
For more information about the various monitoring views, see *GUI of the monitoring console*.

3.2. Customizing the monitoring console

To display the information of interest in *Talend Activity Monitoring Console*, you can customize the display options of each view as needed.

3.2.1. Customizing Jobs view

The **Jobs** view provides the list of Jobs mentioned in the execution log data collected.



On this view, select a particular Job to display its execution.

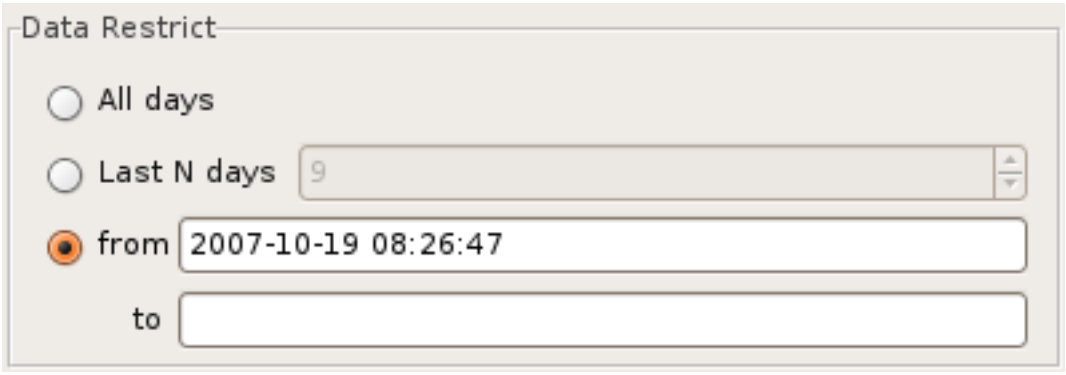
- You can filter Job instances following the **Project** they belong to, so that only the selected project's Jobs are displayed in the information list. By default all projects contained in the log sources will show up.
- You can also filter on other information if you need. In the **Filter** field, type in the corresponding regular expression to fine-tune your selection.
- More precisely, you can select one single execution. In the **Talend Pid** field, type in the corresponding **Talend** PID, execution PID (Process ID) given by your **Talend** application.
- You can as well type in a system Pid in the **System Pid** field to select Job(s) according to their execution PIDs given by the operating system(s) where the Jobs run.

3.2.2. Customizing History and Detailed history views

The **History** and **Detailed History** views display all execution entries that are contained in the files or database tables that you set as datasources.

To display only the execution entries that match some date or time period criteria, do the following:

1. Select **Window** > **Preferences** from the menu to open the **[Preferences]** window.
2. Expand the **AMC** node and select **Datasource Type**.
3. In the **Data Restrict** area, set the filtering preferences.



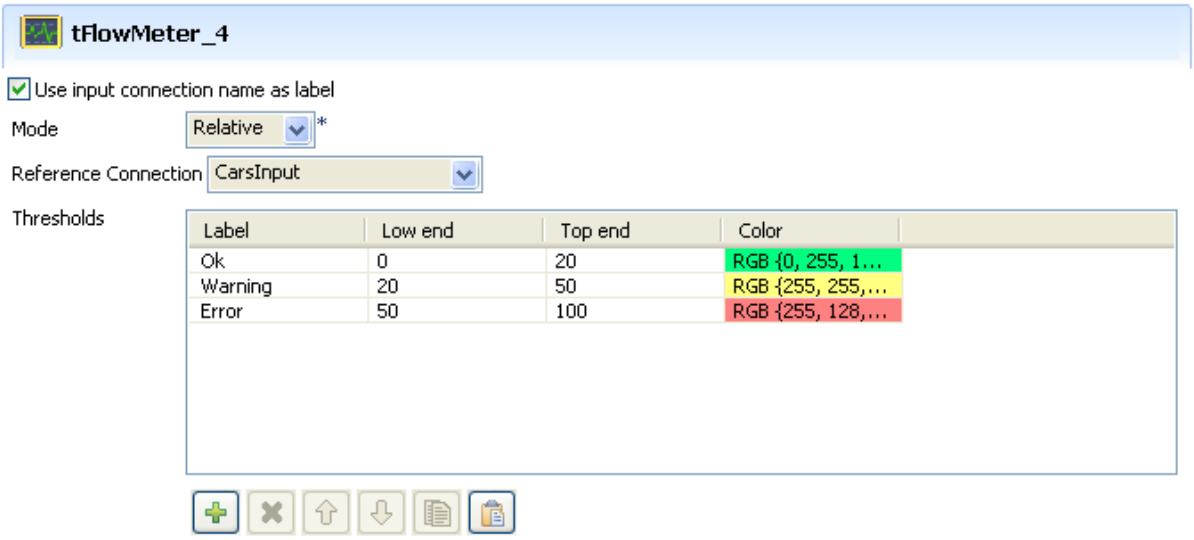
The three options are described in the following table:

Option	Description
All days	Default selection. All the recorded entries are displayed with no filtering.
Last N days	Only the entries of the last days will show. Select the number of days to be taken into account.
from ... to...	Only the entries between the dates defined here will show. Select the relevant boundary dates and times respectively in the **from** and **to** fields.

3.2.3. Customizing Meter Log view

Depending on the **tFlowMeter** setting in the *Talend Studio* Job Designer, each flow measure can display either an absolute or a relative value.

- The absolute value is the exact number of rows being processed.
- The relative value is the ratio of rows being processed, against a reference flow you define on the **tFlowMeter** component.

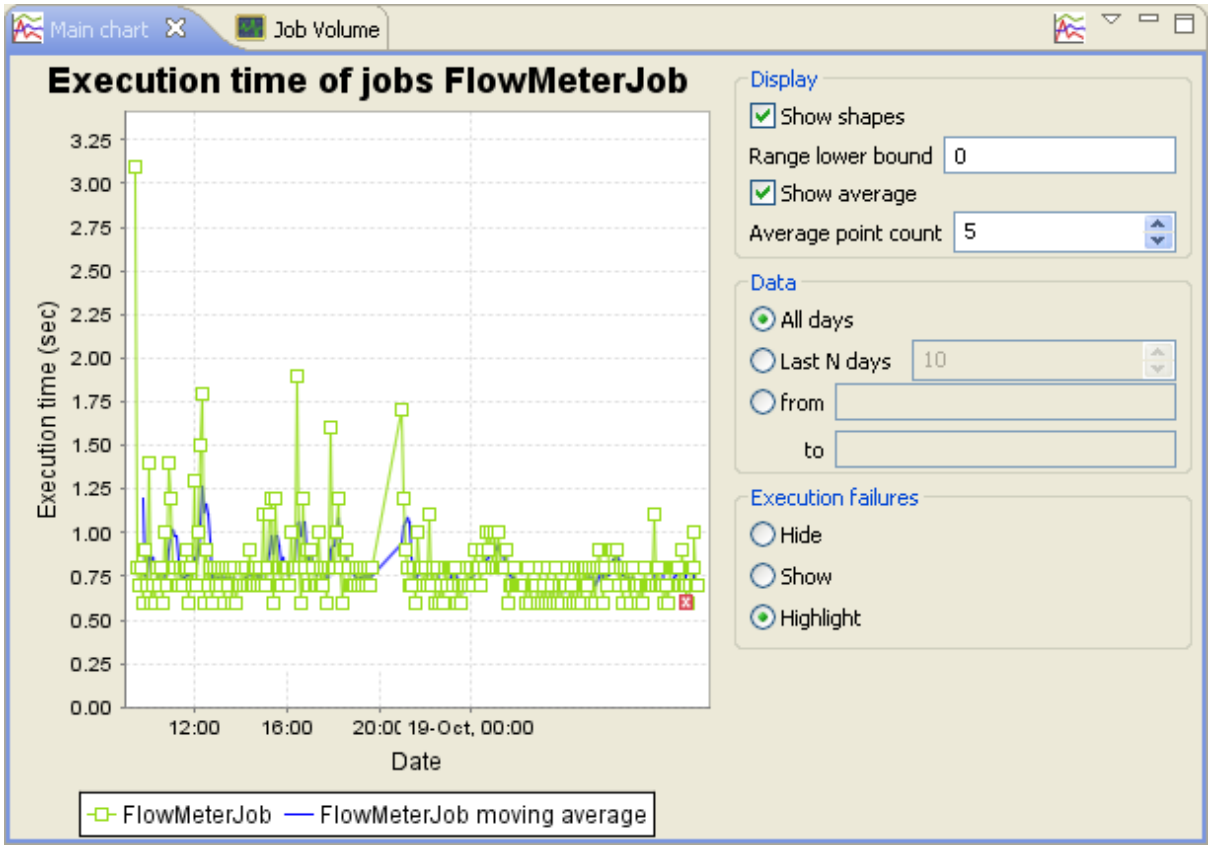


On the **Component** view of the **tFlowMeter** component of your *Talend Studio*, select the **Mode** (**Relative** or **Absolute**) you want to display the data flow processed in *Talend Activity Monitoring Console*. For the relative mode, point out also the **Reference Connection** the ratio is calculated against.

3.2.4. Customizing the Main Chart and the Job Volume views

Depending on the type of view, you have access to various graph and data display options.

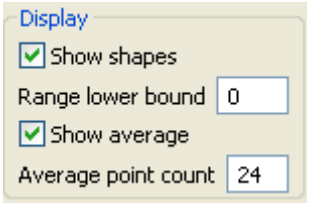
To customize the **Main chart** and **Job Volume** views, click the down arrow docked at the top right end of *Talend Activity Monitoring Console* main interface to show the display options.



Alternatively, you can customize the main chart display in the **[Preferences]** window. To do this, select **Window** > **Preferences** from the menu, then expand the **AMC** node and select **Charts**.

3.2.4.1. Shape display

When the **Main Chart** or **Job Volume** view displays a **line chart** or a **multiple line chart**, each execution instance is represented by a shape as a milestone on the line graph.



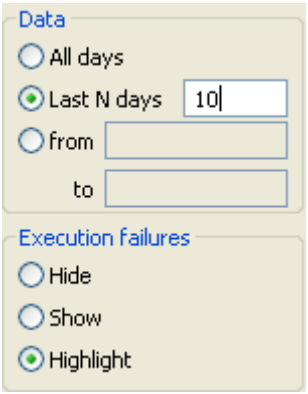
You can choose to show or hide these execution nodes from the graph, by selecting or clearing the **Show shapes** check box.

You can also define the **Range Lower Bound** to rescale the chart at your convenience on the vertical (y) execution time axis. Reset this boundary to fit best the line chart on the graph.

Select the **Show average** check box to display a new line chart presenting the average of the last N points, where N is the number defined in the **Average point count** field.

3.2.4.2. Data display

You can also fine-tune the **chart moment array** to be displayed on the horizontal (x) axis of the line chart or multiple line chart to narrow down to the most relevant execution moments.



On the **Data** area, change the radio button selection from **All days** to the **Last N Days** or define the array of moments to display in the **From... to** date settings.

You can smooth the line chart displayed by selecting the **Hide** check box, in the **Execution failures** area. All failures are thus excluded from the chart, or you can select the **Highlight** radio button to highlight the execution failures and spot them more easily.

3.2.5. Customizing Threshold charts view

To customize the threshold chart view, you need to define the threshold value and their graphical representation on the **Basic settings** tab of the **tFlowMeter Component** view.