

tConnectionPool – Documentation

- Component allows you to create a connection pool inside a Talend DI Job
- Supported databases:
 - o MySQL
 - o Oracle
- Two operation modes:
 - o Create Connection Pool
 - o Close Connection Pool
- Uses the “specify a data source alias” feature in other database components
- Can also be used over different jobs, if they are combined via tRunJob

Basic Settings

Item	Description
Database Version	MySQL Oracle <ul style="list-style-type: none">- Supported Version: 11-6,11,10- Connection Types: SID, ServiceName
Host	"127.0.0.1"
Port	"3306"
Database	"myDatabase"
User	"databaseUser"
Password	"password"
Alias	Enter a job wide unique name. You have to use this alias in every other database component e.g. "MyConnectionPool"

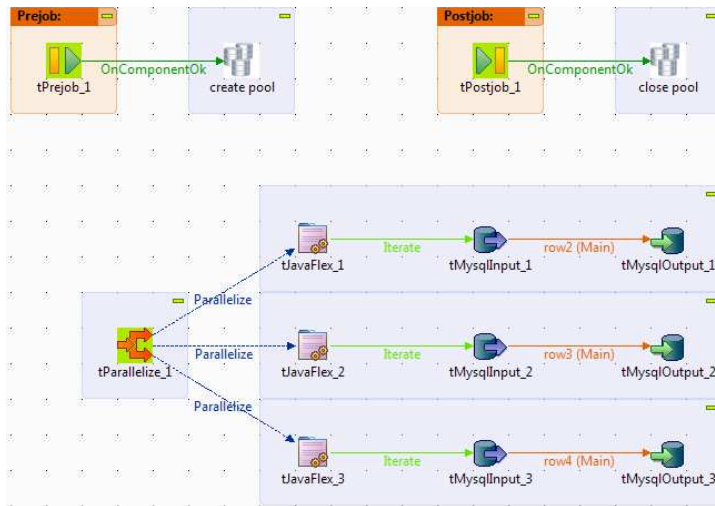
Advanced Settings

- Please note: All time declaration for
 - o MySQL in milliseconds
 - o Oracle in seconds

Item	Description
Additional JDBC Parameter	Specify additional properties property_1=value1;property_2=value_2; ...
Test On Borrow	If checked, every connection is validated before it is used
Time Between Checks	Idled connections are checked during a validate run. Here you set the time between that runs.
Max Idle Time	Maximum time an connection can stay in idle mode, before it is removed from the pool.
Number Checked Connections	Only available for MySQL. Defines the number of connections that are checked during a validation run.
Set Initial Size	Initial amount of available connections.
Set Max Connection Size	Maximum amount of connections those are available within the pool.
Set Max Time To Wait For Connection	Maximum time a connection request can wait if all connections are in use.
Set Initial SQL	Only available for MySQL. Sets a list of SQL statements to be executed when a connection is first created. Separated via semicolon

Usage

Job implementation:



Create Connection Pool:

The screenshot shows the configuration window for the **create pool(tConnectionPool_1)** component. The **Basic settings** tab is selected. The **Mode** is set to **Create Connection Pool**. The **Database Version** is set to **MySQL**. The **Host** is **"127.0.0.1"**, **Port** is **"3306"**, and **Database** is **"database"**. The **User** is **"user"** and the **Password** is masked with ********. The **Alias** is **"DataSource"**.

Usage in other database components: Leave connection parameters blank and specify the data source alias.

The screenshot shows the configuration window for the **tMysqlInput_1** component. The **Basic settings** tab is selected. The **Host** and **Port** fields are empty. The **Benutzername** field is empty. The **Schema** is set to **Built-In**. The **Tabellenname** is **"customer"**. The **Abfragetyp** is set to **Built-In**. The **Abfrage** is **"select * from table1"**. The **Datenquelle** section is expanded, showing the option **Specify a data source alias** checked, with the **Data source alias** set to **"DataSource"**.

Close Connection Pool:

The screenshot shows the configuration window for the **close pool(tConnectionPool_2)** component. The **Basic settings** tab is selected. The **Mode** is set to **Close Connection Pool**. The **Alias** is **"DataSource"**.