HITACHI
Inspire the Next

Pentaho Data Integration

Enterprise Solution

James O'Reilly
Hitachi Vantara Global Learning
Date

Module Objectives

HITACHI
Inspire the Next

When you complete this module, you should be able to:

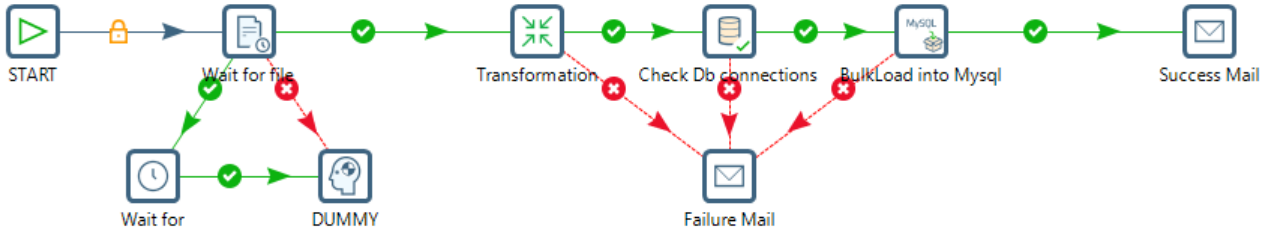
- Implement Jobs to control your ETL workflow
- Deploy your solution with Variables and Parameters
- Scale out your solution with Carte Servers

HITACHI
Inspire the Next

Jobs

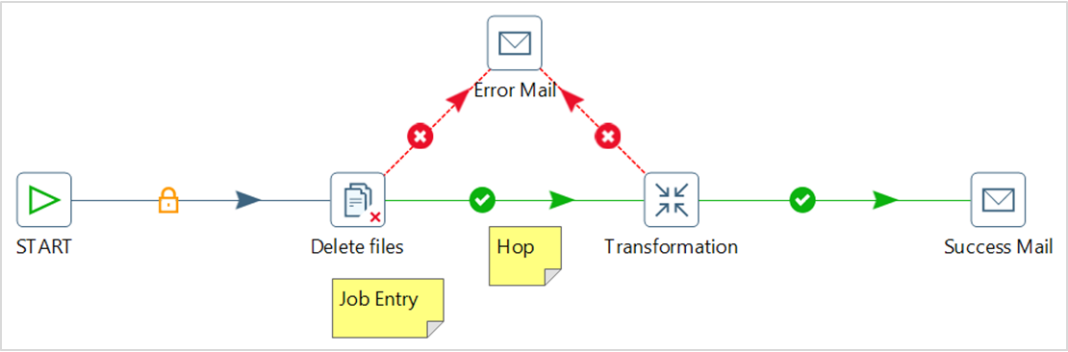
Jobs Workflow

HITACHI
Inspire the Next



- Adding and configuring Job Entries
- Ensure required resources are available
- Using file management Job Entries
- Implement error handling in Jobs
- Implement Checkpoints
- Reviewing execution results

Jobs



- Jobs are workflow-like models for coordinating resources, execution and dependencies of ETL activities
- Consist of job entries, hops and notes
- Aggregate individual transformations to a process and introduce order
- Hops can be conditional

Job Orchestration

HITACHI
Inspire the Next

- In an IT environment, orchestration is the ongoing task of ensuring that technical processes run reliably and according to schedule while producing the correct outputs using acceptable amounts of resources
- Orchestration includes:
  - Ensuring the availability of resources
  - Executing a series of tasks in the right order
  - Detecting and recovering from errors
  - Logging the results
  - Notifying people and other applications

Job Management

HITACHI
Inspire the Next

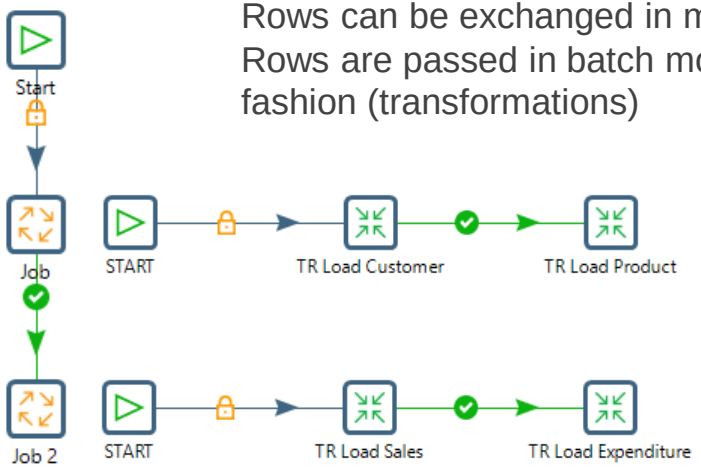
- Always check the status of external systems
  - Host systems
  - Databases / tables
  - File systems / directories
  - FTP sites
  - Web services
- Execute tasks with retry logic
- Notify others when a problem has occurred
- Store variables in a central location and use them consistently

Job Entry

- A job (Job / Transformation) entry is the primary building block of a job.
  - Execute jobs / transformations, retrieve files, generate email, and so on
- Each job can only have one job entry.

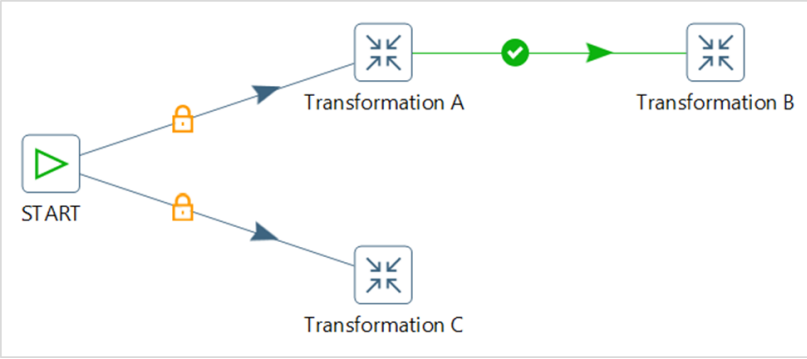
Job entries can pass a result object containing rows

Rows can be exchanged in memory – result rows
Rows are passed in batch mode, not in a streaming fashion (transformations)



Multiple Paths & Backtracking

HITACHI
Inspire the Next



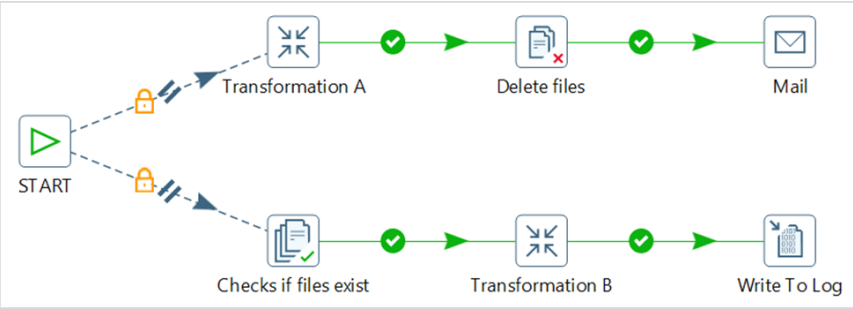
- Jobs are executed using a backtracking algorithm
- The path itself is defined by the actual outcome (success or failure) of the job entries
- The path is always followed until the very end before a next possibility is considered
- Order depends on creation of job entries
- The result of the job (success or failure) depends on the last job entry

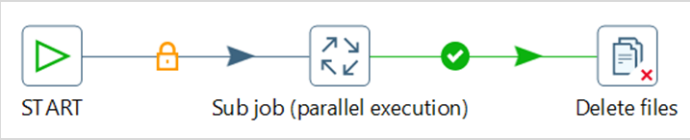
Parallel Execution

HITACHI
Inspire the Next

- A job entry can be told to execute the next job entries in parallel
  - Running in separate threads

- When you have a number of sequential job entries, these are executed in parallel as well





- If you want to continue in sequence after this parallel execution you need to put the parallel part of the job in a sub-job

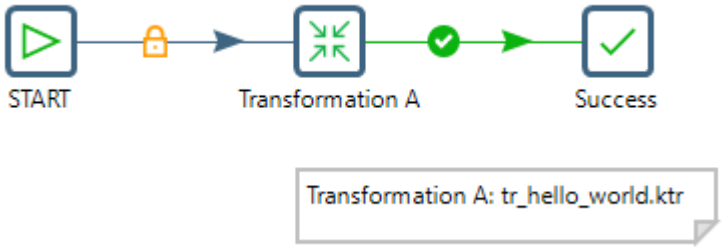
HITACHI
Inspire the Next

Lab 1 – Create a Job

Lab 1 - Hello World - Job

HITACHI
Inspire the Next

- Simple Job



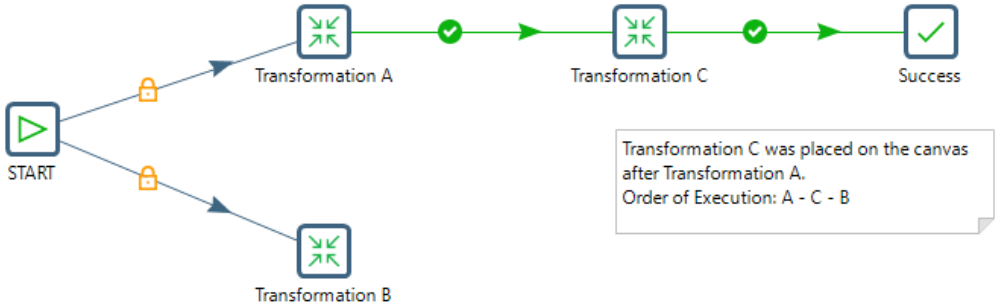
HITACHI
Inspire the Next

Lab 2 – Order of Execution

Lab 2 – Order of Execution

HITACHI
Inspire the Next

- Backward Algorithm



- Parallel

HITACHI
Inspire the Next

Variables & Parameters

Variables, Parameters

HITACHI
Inspire the Next

When you Run a Transformation or Job, there are several ways you can define the required settings.

- Variables allow you to dynamically enter the required setting.
- Parameters statically sets the condition of the Transformation or Job.

Variables

HITACHI
Inspire the Next

- Variables can be used throughout Pentaho Data Integration, including in transformation steps and job entries. You define variables by setting them with the Set Variable step in a transformation or by setting them in the kettle.properties file in the directory:
  - $HOME/.kettle (Unix/Linux/OSX)
  - C:\Documents and Settings\<username>\.kettle\ (Windows XP)
  - C:\Users\<username>\.kettle\ (Windows Vista, 7 and later)

- The way to use them is either by grabbing them using the Get Variable step or by specifying meta-data strings like:

  ${VARIABLE}

  or:

  %%VARIABLE%%

Parameters

HITACHI
Inspire the Next

- Parameters are special in the sense that they are explicitly named command line arguments. If you pass on a lot of arguments to your Kettle Job or Transformation, it might help to assign those values to an explicitly named parameter.
- Parameters have following advantages:
  - On the commandline you assign the value directly to a parameter, hence there is zero chance of a mix-up.
  - A default value can be defined for a named parameter
  - A description can be provided for a named parameter
  - No need for an additional transformation that sets the variables for the job

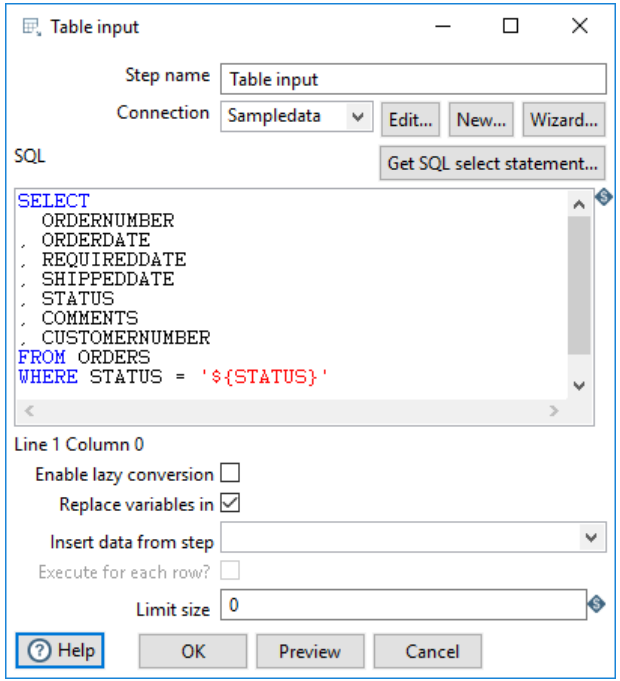
HITACHI
Inspire the Next

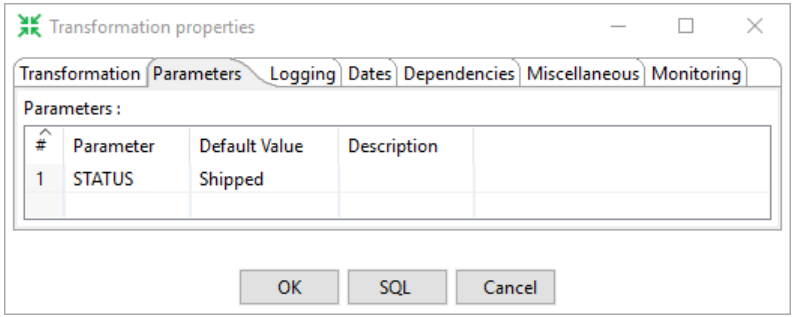
Lab 1 – Parameters & Variables

Lab 1 - Variables and Parameters

HITACHI
Inspire the Next

- So lets modify an existing Transformation ..
- Open Module 3 / Lab 2: Reading from a Database Table





HITACHI
Inspire the Next

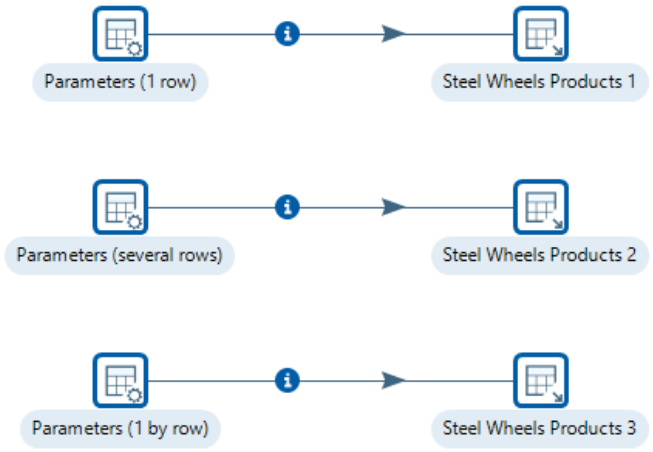
Lab 2 – Parameters and SQL

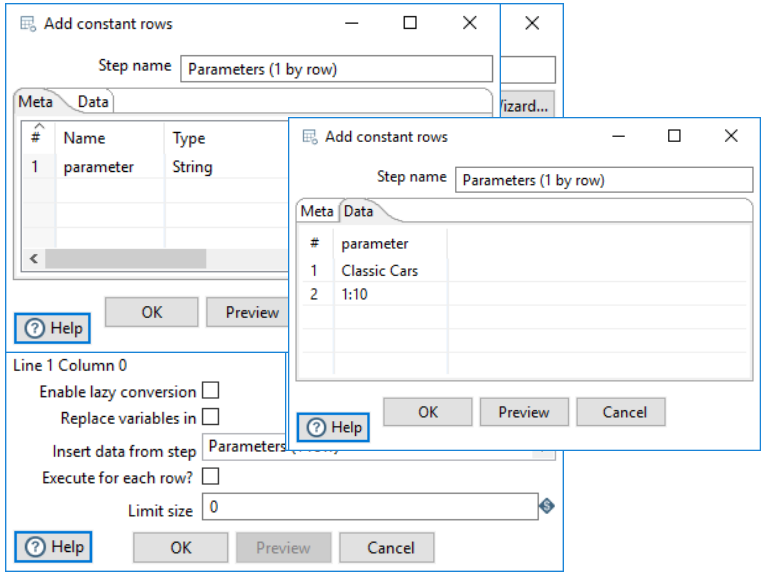
Lab 2 – Parameters in SQL

HITACHI
Inspire the Next

- One of the ways you have to make your queries more flexible is by passing it through some parameters.

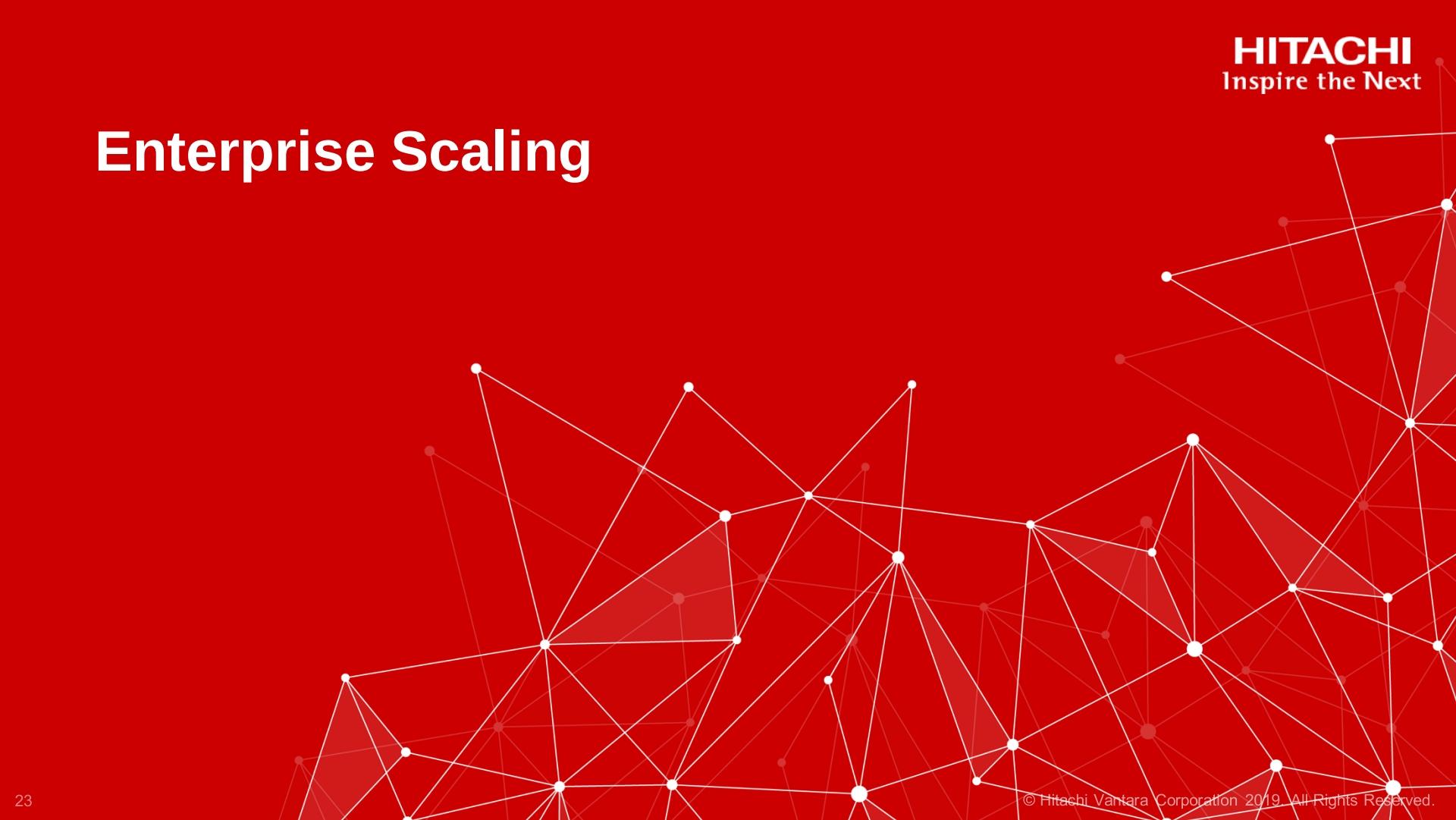
This demonstration illustrates the different options for passing parameters.





HITACHI
Inspire the Next

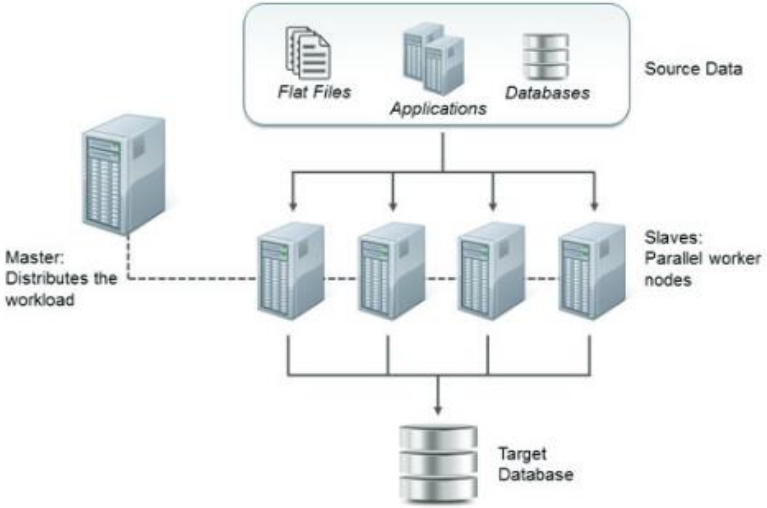
Enterprise Scaling



Clustering Carte Slave servers

HITACHI
Inspire the Next

- Clustering is a technique that can be used to scale out transformations to make them run on multiple servers, in parallel.
- The Cluster is defined with a Schema, with one master server acting as the controller for the cluster.
- The Cluster schema also contains metadata on how master and slaves pass data back and forth.



Carte Configuration

HITACHI
Inspire the Next

Master DI server

- Default configuration for low volume and process mix

Master DI server with Independent number Carte servers

- Used for large amount of transformations of lesser volume
- Master acts as a load balancer for transformations

Master DI server with slave servers (Clustering)

- Used for large volumes with fewer transformations
- Distributes set of rows across slave servers
- Can be fixed cluster size or dynamic

Carte as a Slave Server

HITACHI
Inspire the Next

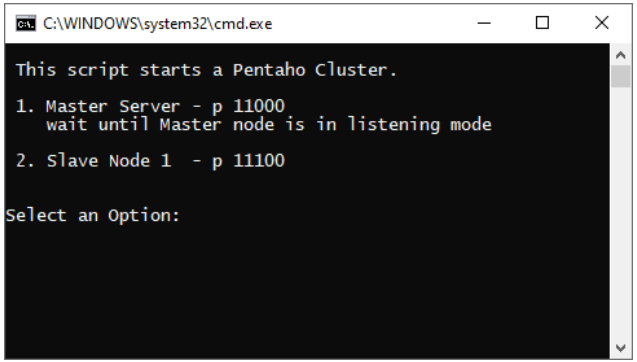
- Carte is a lightweight HTTP server that accepts commands from remote clients.
- These commands control the deployment, management, and monitoring of Jobs and Transformations.
- You can configure the carte server with an XML file (examples found in \pwd directory)

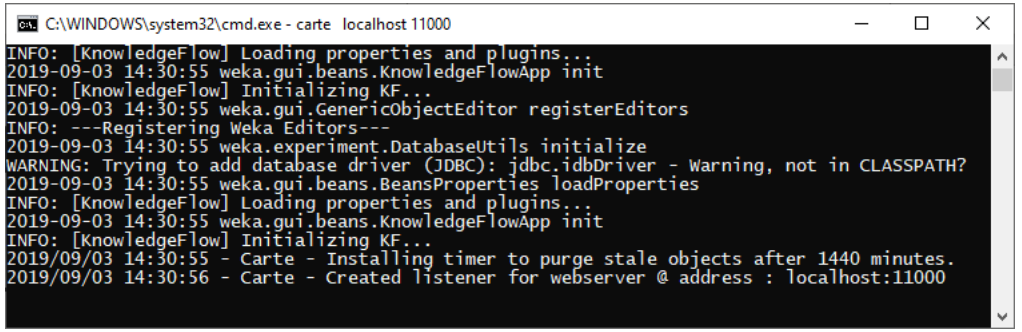
Lab 1 – Implement Carte Servers

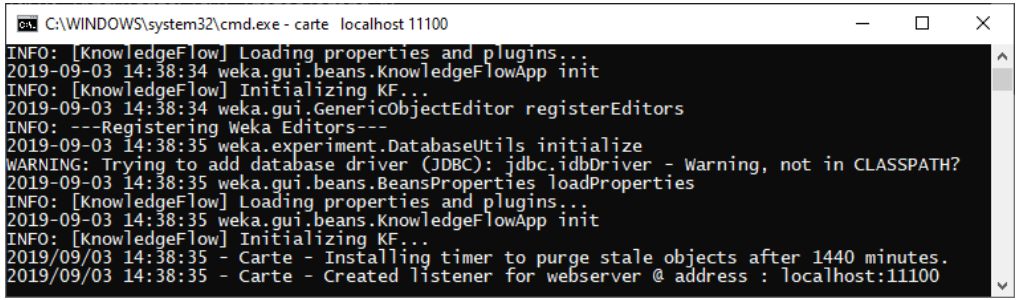
Lab 1 – Start Master / Slave Nodes

HITACHI
Inspire the Next

- Script to start a Master and Slave node
  - Topic 3 - Scalability\Lab 1 - Master - Slave Nodes\config_nodes.cmd



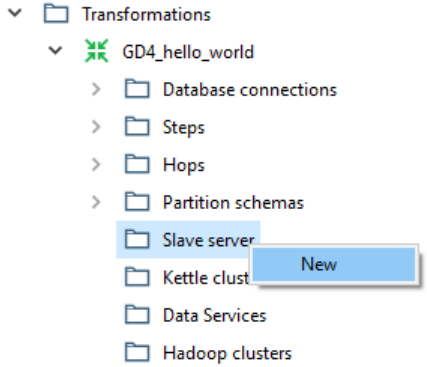


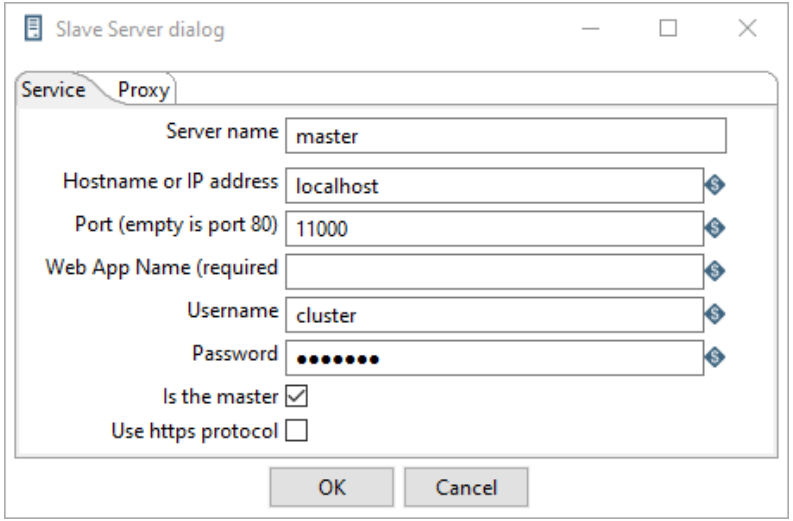


Lab 1 - Master DI Carte Server

HITACHI
Inspire the Next

- Open tr_hello_world transformation
- On the View tab, select: Slave > New



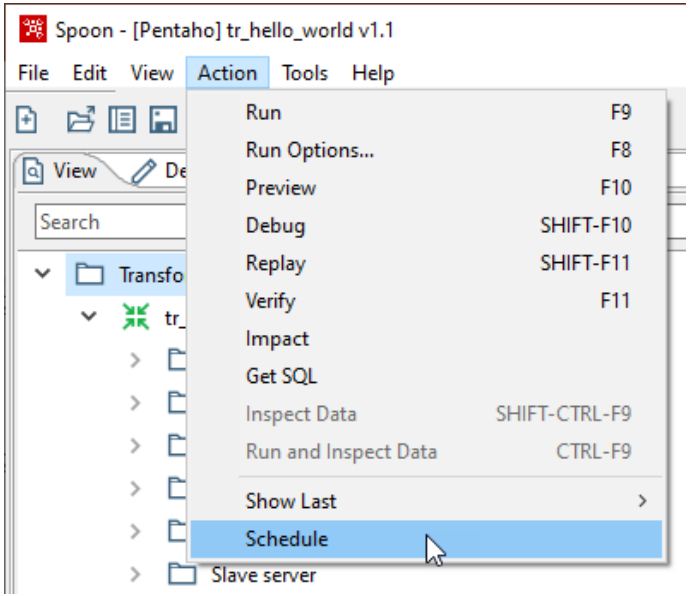


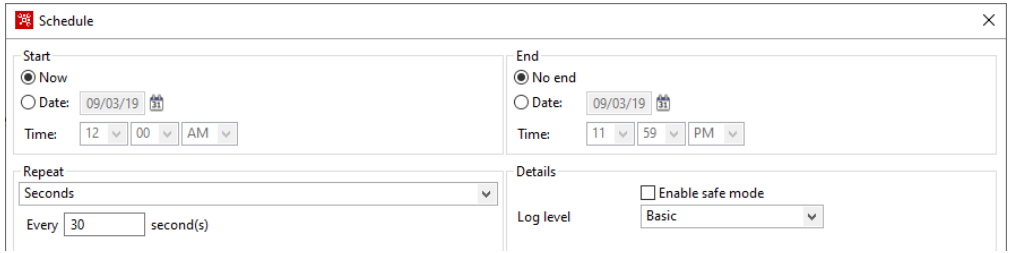
- Right mouse click and select: Monitor
- Ensure that you have connected to the server.
- Access the server: http://localhost:8080/kettle/status

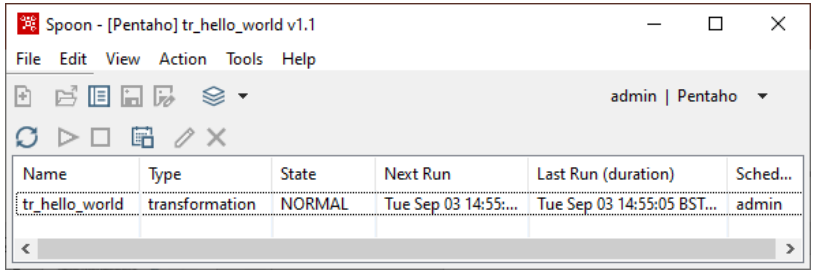
Username: cluster
Password: cluster

Lab 1 - Monitor & Schedule

- Export to Repository







Module Recap

HITACHI
Inspire the Next

In this module, you should have learned to:

- Implement Jobs to control your ETL workflow
- Deploy your solution with Variables and Parameters
- Scale out your solution with Carte Servers

HITACHI
Inspire the Next

Thank You

HITACHI

Inspire the Next