Business intelligence

Tutorial 3 – Pentaho Data Integration

UNIVERSIDAD DE
MURCIA

Pentaho Data Integrator

UNIVERSIDAD DE MURCIA

- Tools
  - Spoon : GUI for transformations and works design
  - Pan : Running transformations
  - Kitchen : Running works (complex transformations)
  - Carte : Cluster Web Server
- Download:
  - Pentaho Data Integrator (v 7.1)
    - http://sourceforge.net/projects/pentaho/files/
  - (if needed Driver Postgresql-9.42, jre8) dir: $PDI/lib/
    - http://jdbc.postgresql.org/download.html
- Tutorial: lets see some simple tasks.
  - http://wiki.pentaho.com/display/EAI/Pentaho+Data+Integration+Steps

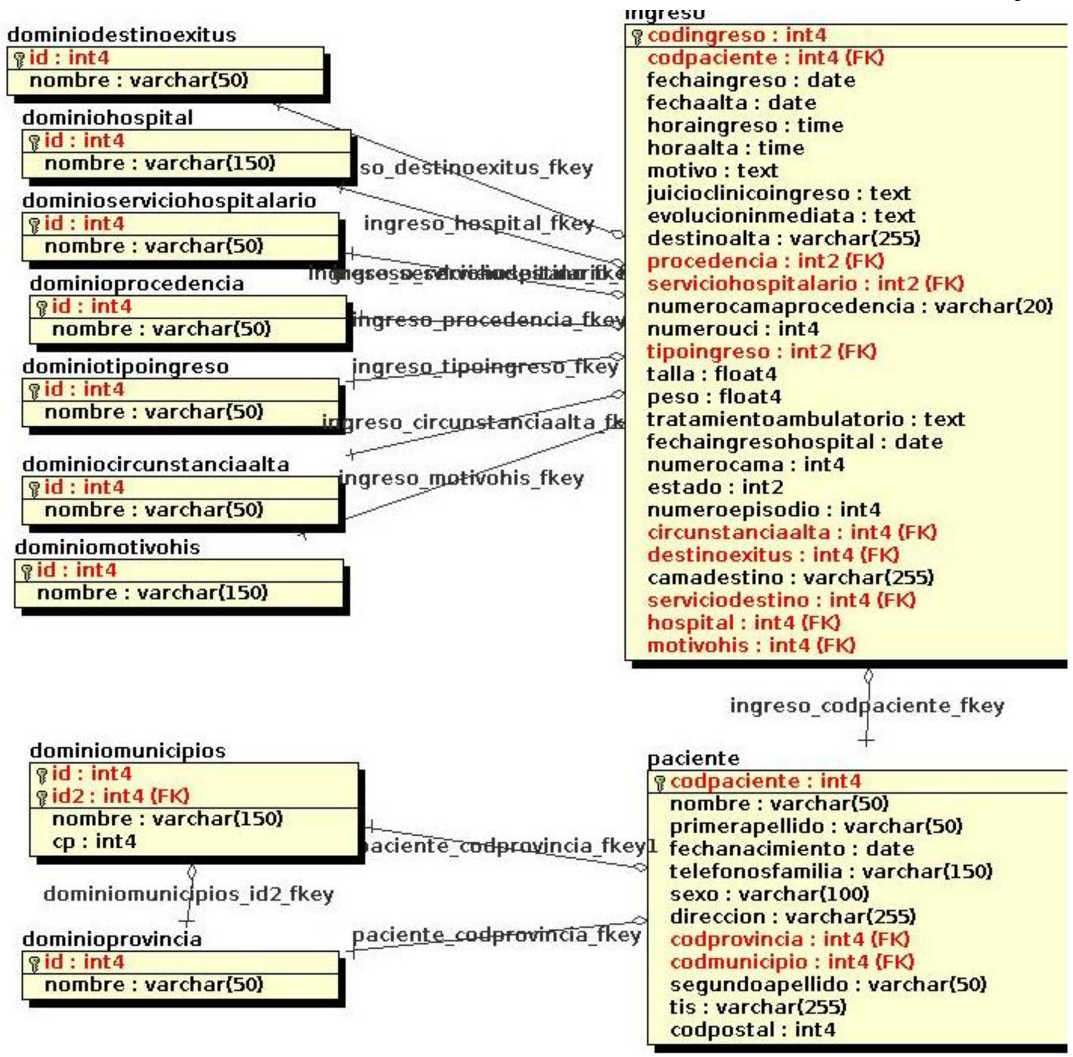
Understanding flows

UNIVERSIDAD DE MURCIA

- Parallel nature:
    - When a transformation is launched, all its steps are started.
    - During the execution, the steps work simultaneously reading rows from the incoming hops, processing them, and delivering them to the outgoing hops. When there are no more rows left, the execution of the transformation ends.
- Streams:
    - A stream is a set of rows all having the same structure or metadata. (Metadata is important)
    - Split, join, inject, …
- Identify rows when reading (rownum)

Database schema

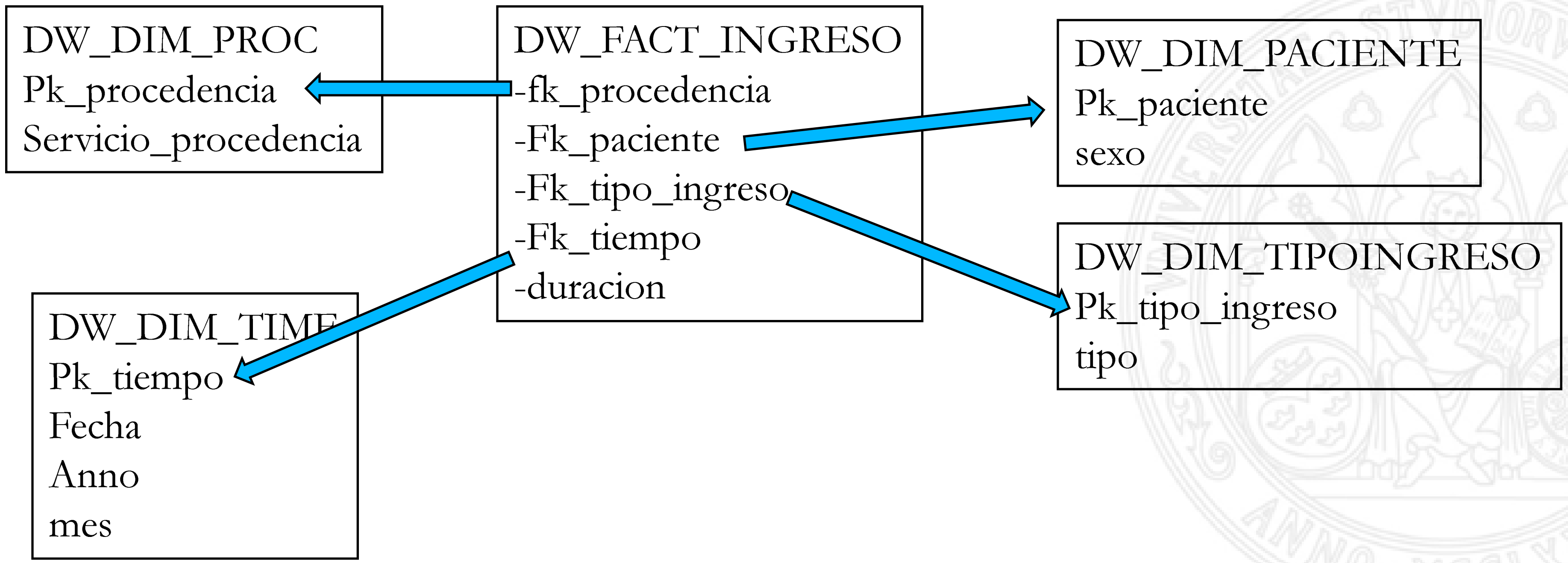
- Source schema: pacientes



- Check or create the connection->
  - Menu “Tools”->Wizard
    - “Create database connection”
      - Postgresql
      - Native (JDBC)
      - Server: given:3128
      - bd: tut3
      - User/pwd: given
  - **Check: Tools->Database->Explore**
  - **Each of you must use your own schema:**
    - **dw_inbdXX**

Database

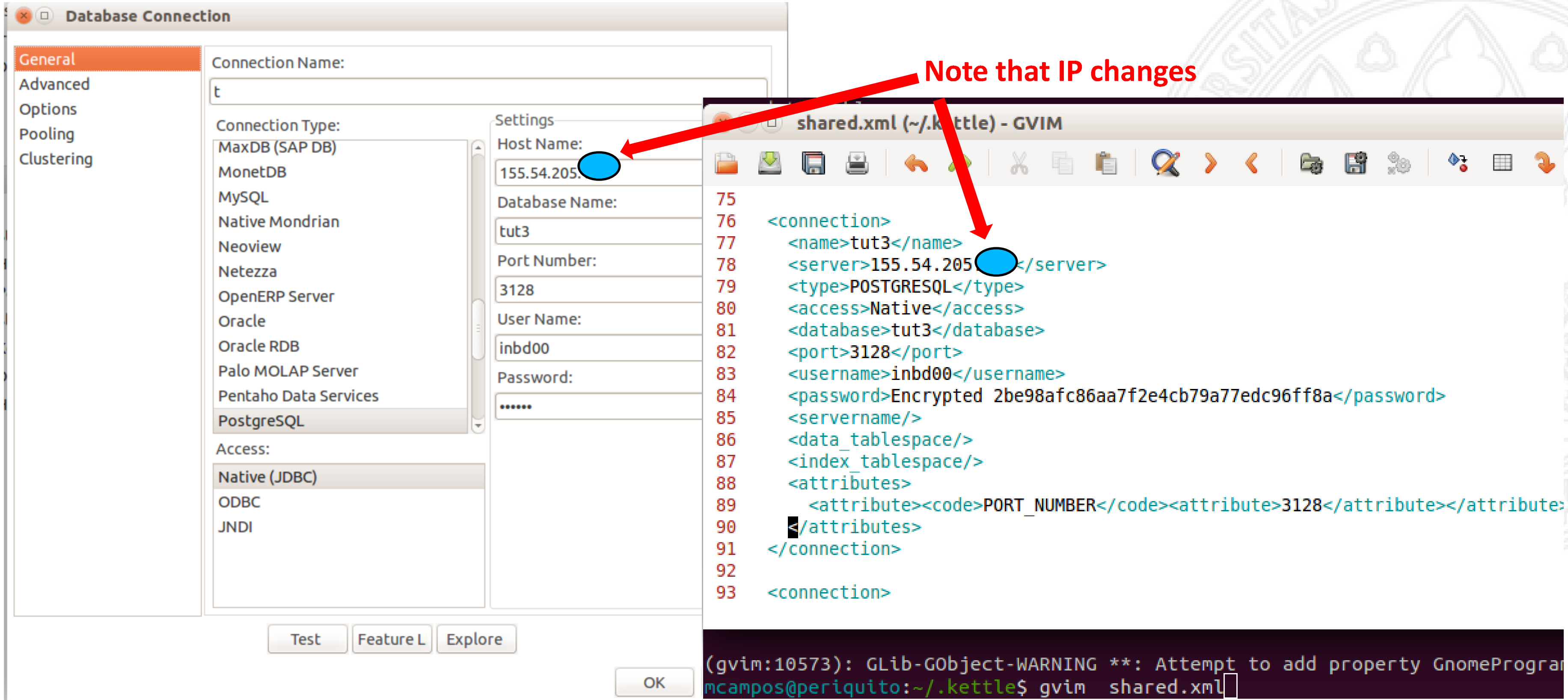
- Target schema



Starting

UNIVERSIDAD DE MURCIA

- Lauch spoon with data_integration/spoon.sh
- Create a database conection
  - Create a first transformation: “File”-> “New”->”Transformation”
  - Menu “Tools” -> “Wizard” -> “Create Connection” and Test it.
  - “View” tab ->Right click on “Database Connection” -> Share: It appear in E/.kettle/shared.xml



Connection Name: t

Connection Type: MaxDB (SAP DB), MonetDB, MySQL, Native Mondrian, Neoview, Netezza, OpenERP Server, Oracle, Oracle RDB, Palo MOLAP Server, Pentaho Data Services, PostgreSQL

Access: Native (JDBC), ODBC, JNDI

Settings

Host Name:	155.54.205[redacted]
Database Name:	tut3
Port Number:	3128
User Name:	inbd00
Password:	••••••

```
  <connection>
    <name>tut3</name>
    <server>155.54.205[redacted]</server>
    <type>POSTGRESQL</type>
    <access>Native</access>
    <database>tut3</database>
    <port>3128</port>
    <username>inbd00</username>
    <password>Encrypted 2be98afc86aa7f2e4cb79a77edc96ff8a</password>
    <servername/>
    <data_tablespace/>
    <index_tablespace/>
    <attributes>
      <attribute><code>PORT_NUMBER</code><attribute>3128</attribute></attribute>
    </attributes>
  </connection>

  <connection>
```

```
(gvim:10573): GLib-GObject-WARNING **: Attempt to add property GnomeProgra
mcampos@periquito:~/.kettle$ gvim  shared.xml
```

Load dimension: Procedence

UNIVERSIDAD DE MURCIA

- Create new transformation: Save it as “LoadProcedence”
- Select “Design” tab-> Input->Table Input , and Drag & Drop into the right
  - Select your Connection
  - Define query (we can also browse with “Obtain query”).
    - `Select id,nombre from pacientes.dominioprocedencia;`
  - Check with preview

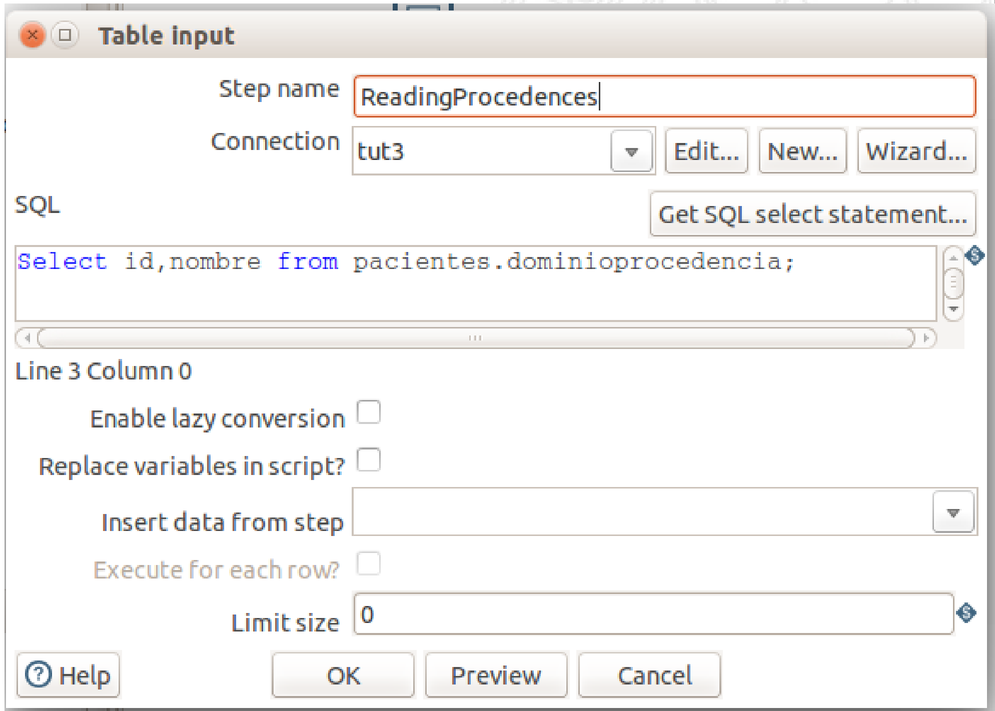


Table input

Step name: ReadingProcedences

Connection: tut3

SQL

```
Select id,nombre from pacientes.dominioprocedencia;
```

Line 3 Column 0

Enable lazy conversion: ☐

Replace variables in script?: ☐

Insert data from step:

Execute for each row?: ☐

Limit size: 0

Load dimension: Procedence

UNIVERSIDAD DE MURCIA

- Select Design → Out → Out table
  - Define **YOUR** target schema "dw_inbd_XX" and table "dw_dim_proc".
  - Note: Batch inserts for efficiency/
  - Also "Insert/Update" step.

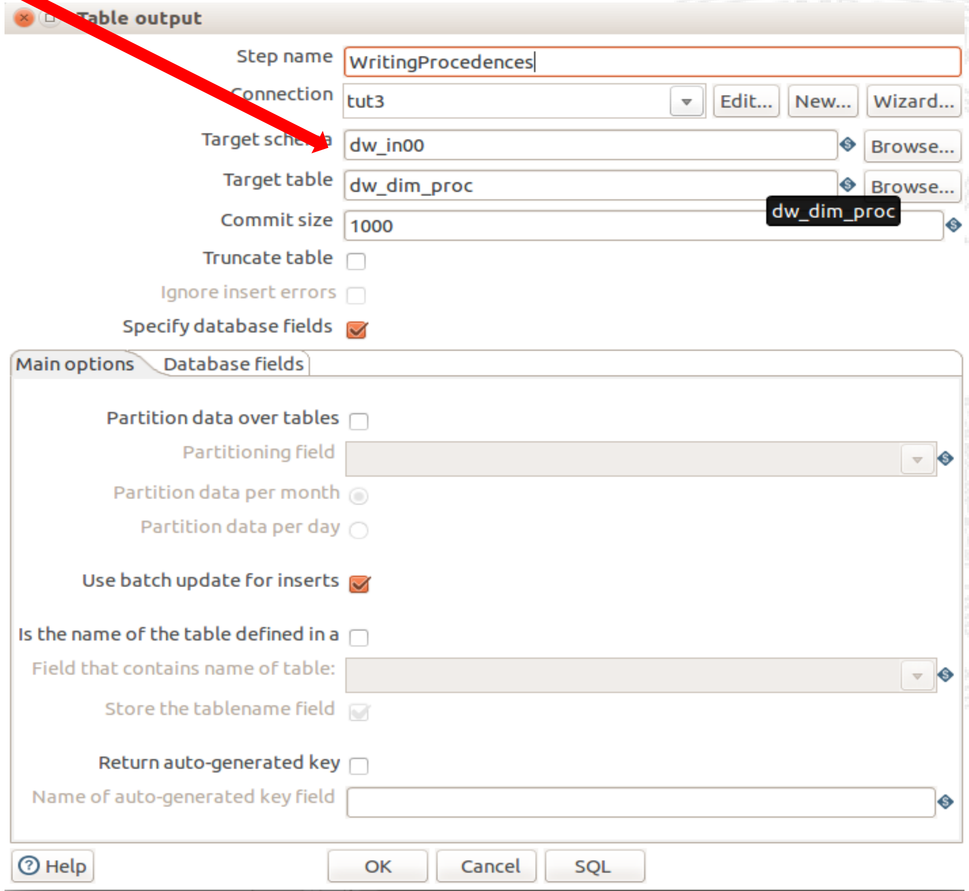


Table output

Step name	WritingProcedences
Connection	tut3
Target schema	dw_in00
Target table	dw_dim_proc
Commit size	1000
Truncate table	☐
Ignore insert errors	☐
Specify database fields	✓

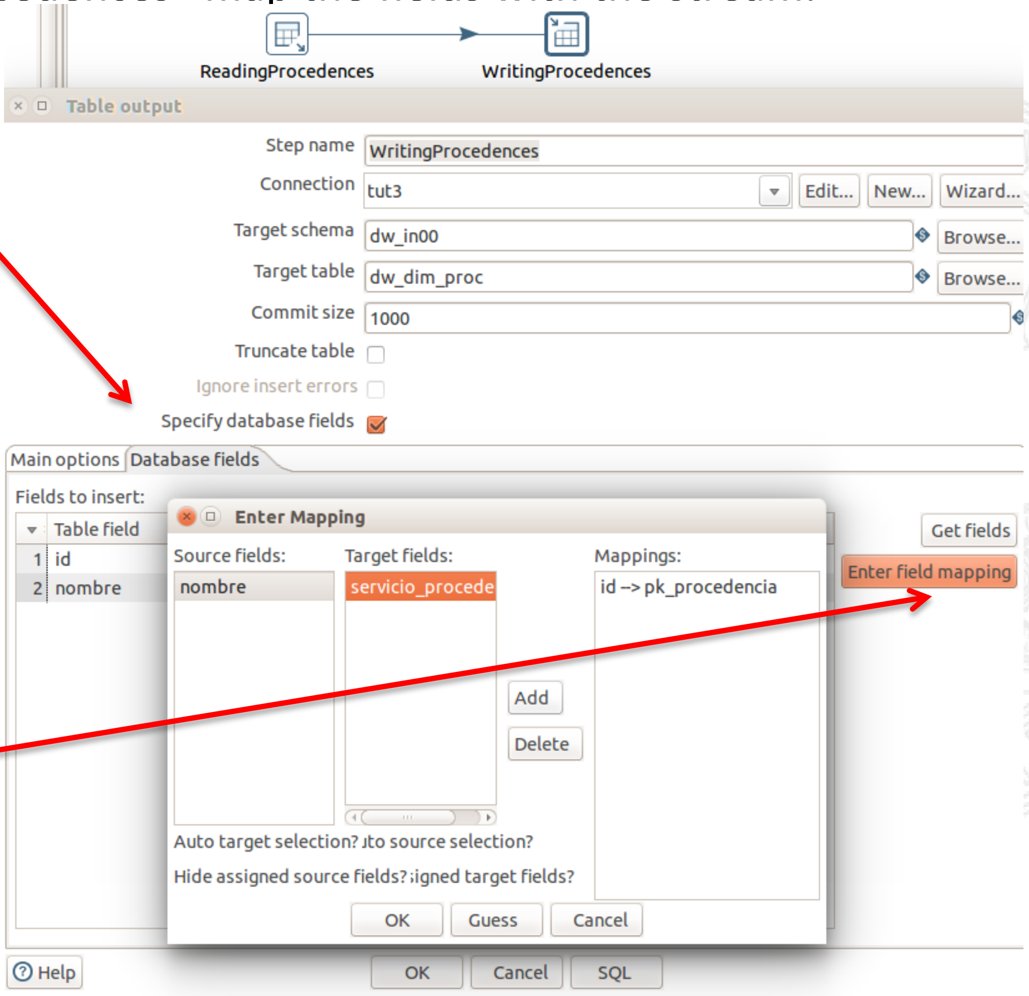
Main options | Database fields

Partition data over tables	☐
Partitioning field	
Partition data per month	●
Partition data per day	○
Use batch update for inserts	✓
Is the name of the table defined in a	☐
Field that contains name of table:	
Store the tablename field	✓
Return auto-generated key	☐
Name of auto-generated key field	

Load dimension: Procedence

UNIVERSIDAD DE MURCIA

- We have to link the stepts:
  - Pressing shift, central button, drag&drop, ….
  - Go to previous step and "WritingProcedences" map the fields with the stream:
  - Choose "Specify database fields"
    - Tab: "Database fields".
  - If needed
    - Use "Transform-> Select values"
    - to rename or metadata



Step name: WritingProcedences
Connection: tut3
Target schema: dw_in00
Target table: dw_dim_proc
Commit size: 1000
Truncate table: ☐
Ignore insert errors: ☐
Specify database fields: ✓

Fields to insert:

	Table field
1	id
2	nombre

Enter Mapping

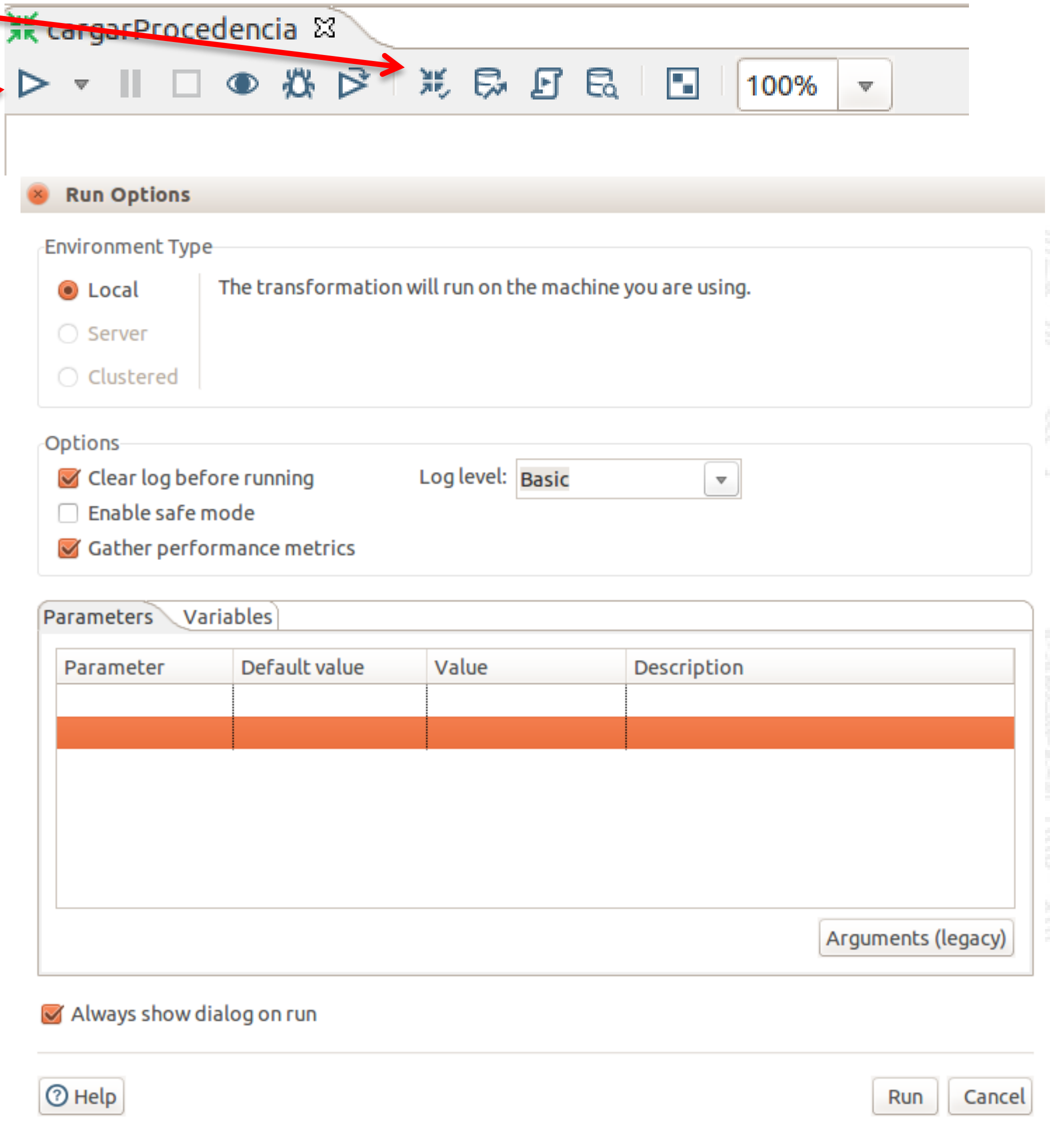
Source fields: nombre
Target fields: servicio_procede
Mappings: id --> pk_procedencia

Auto target selection? ıto source selection?
Hide assigned source fields? ;igned target fields?

Run transformation

UNIVERSIDAD DE MURCIA

- **IMPORTANT:** Before running: Verify transformation
- Several running options
- Parameter passing
- Log level
- Safe mode:
    - Extra row checking in runtime



Run Options

Environment Type

[✓ Local] The transformation will run on the machine you are using.
Server
Clustered

Options

[✓] Clear log before running
[ ] Enable safe mode
[✓] Gather performance metrics

Log level: Basic

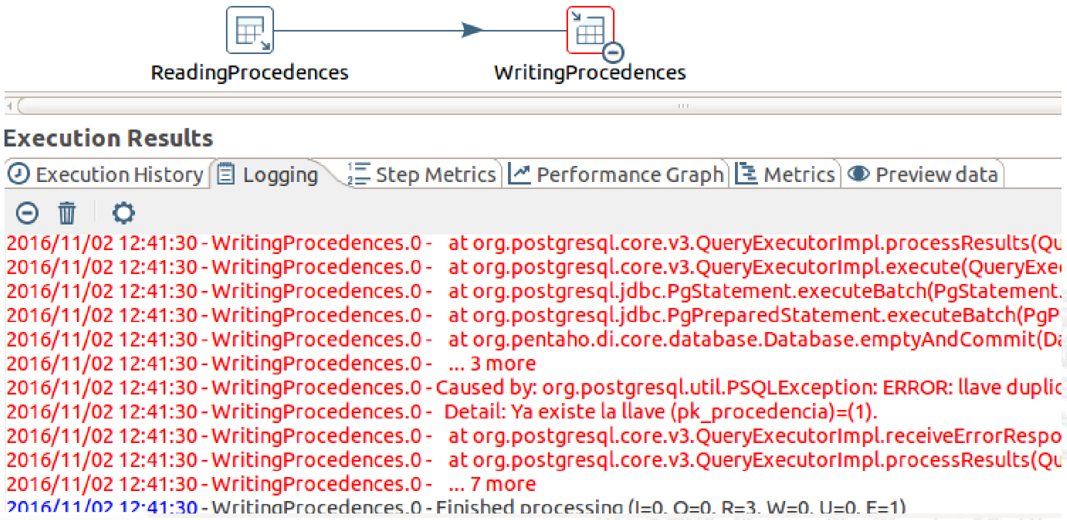
Parameters | Variables

Parameter	Default value	Value	Description

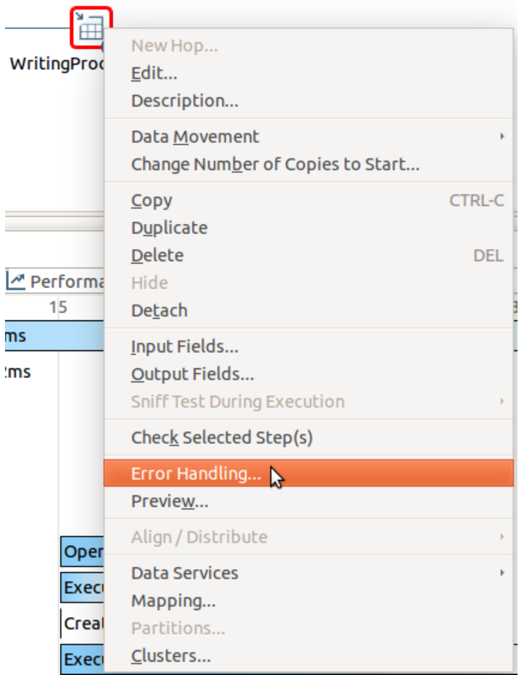
[✓] Always show dialog on run

Run transformation

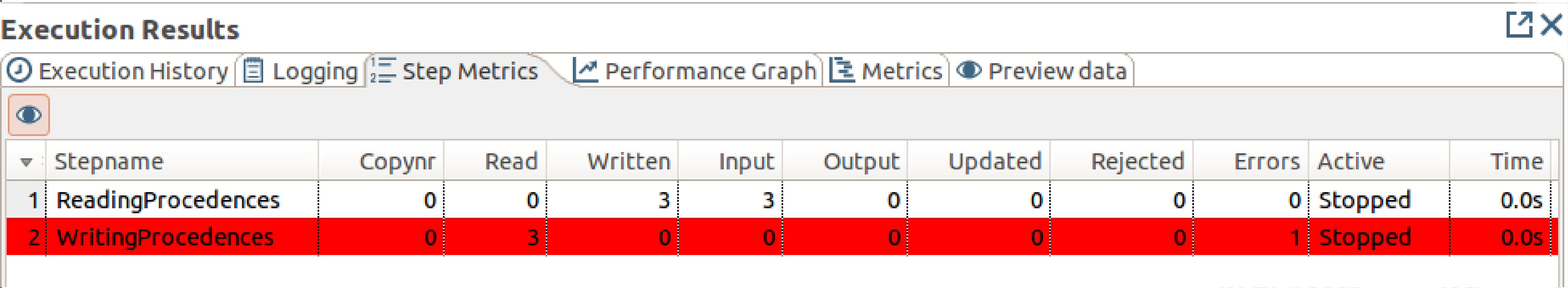
- It can be debugged
  - See “Write to log”
  - Error handling in some steps. Rigth click



```
2016/11/02 12:41:30 - WritingProcedences.0 -   at org.postgresql.core.v3.QueryExecutorImpl.processResults(Qu
2016/11/02 12:41:30 - WritingProcedences.0 -   at org.postgresql.core.v3.QueryExecutorImpl.execute(QueryExe
2016/11/02 12:41:30 - WritingProcedences.0 -   at org.postgresql.jdbc.PgStatement.executeBatch(PgStatement.
2016/11/02 12:41:30 - WritingProcedences.0 -   at org.postgresql.jdbc.PgPreparedStatement.executeBatch(PgP
2016/11/02 12:41:30 - WritingProcedences.0 -   at org.pentaho.di.core.database.Database.emptyAndCommit(Da
2016/11/02 12:41:30 - WritingProcedences.0 -   ... 3 more
2016/11/02 12:41:30 - WritingProcedences.0 - Caused by: org.postgresql.util.PSQLException: ERROR: llave duplic
2016/11/02 12:41:30 - WritingProcedences.0 -  Detail: Ya existe la llave (pk_procedencia)=(1).
2016/11/02 12:41:30 - WritingProcedences.0 -   at org.postgresql.core.v3.QueryExecutorImpl.receiveErrorRespo
2016/11/02 12:41:30 - WritingProcedences.0 -   at org.postgresql.core.v3.QueryExecutorImpl.processResults(Qu
2016/11/02 12:41:30 - WritingProcedences.0 -   ... 7 more
2016/11/02 12:41:30 - WritingProcedences.0 - Finished processing (I=0, O=0, R=3, W=0, U=0, E=1)
```

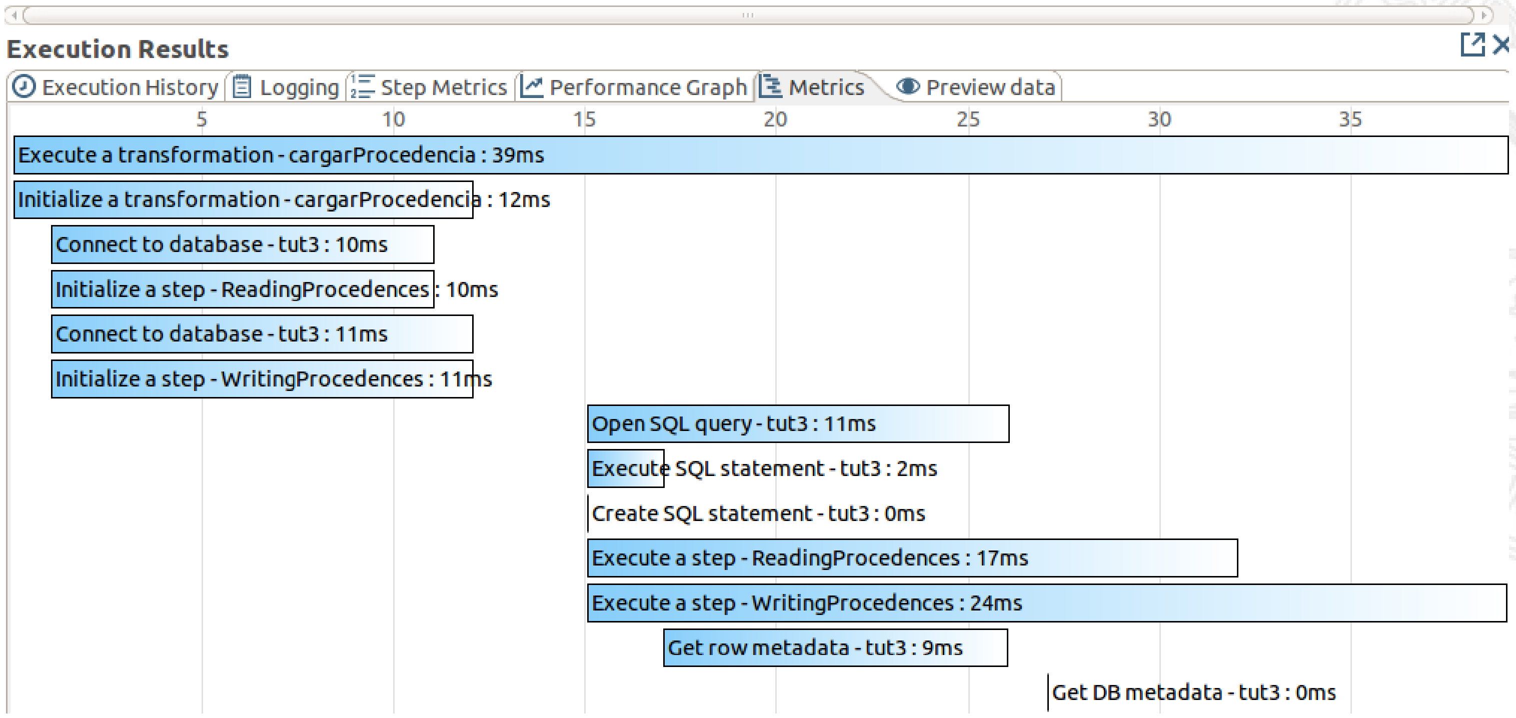


Run transformation: Metrics



Execution Results

	Stepname	Copynr	Read	Written	Input	Output	Updated	Rejected	Errors	Active	Time
1	ReadingProcedences	0	0	3	3	0	0	0	0	Stopped	0.0s
2	WritingProcedences	0	3	0	0	0	0	0	1	Stopped	0.0s



Execution Results

Execute a transformation - cargarProcedencia : 39ms
Initialize a transformation - cargarProcedencia : 12ms
Connect to database - tut3 : 10ms
Initialize a step - ReadingProcedences : 10ms
Connect to database - tut3 : 11ms
Initialize a step - WritingProcedences : 11ms
Open SQL query - tut3 : 11ms
Execute SQL statement - tut3 : 2ms
Create SQL statement - tut3 : 0ms
Execute a step - ReadingProcedences : 17ms
Execute a step - WritingProcedences : 24ms
Get row metadata - tut3 : 9ms
Get DB metadata - tut3 : 0ms

Load dimensions

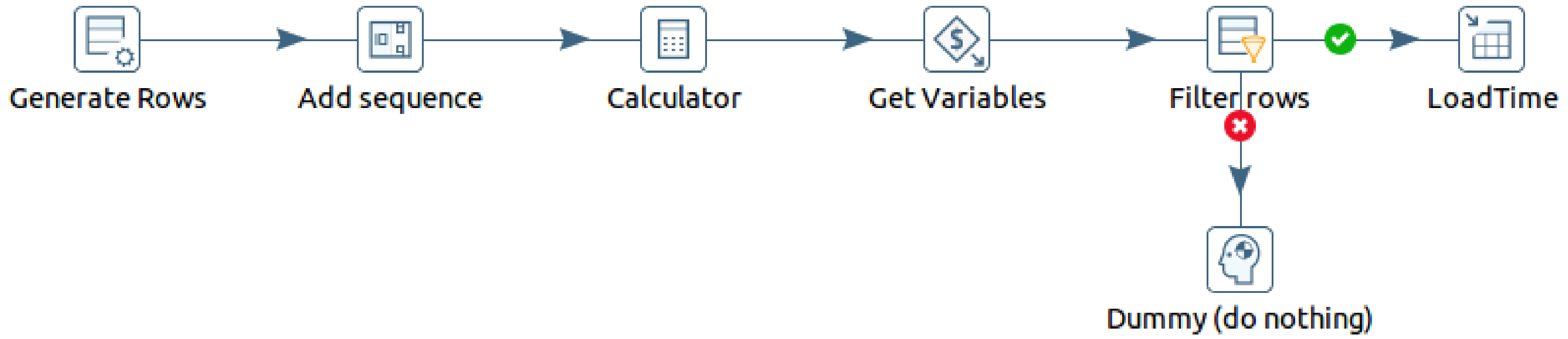
- We do the same for
  - Admission type
    - Source table: pacientes.dominiotipoingreso
    - Target table: dw_YOUR_SCHEMA.dw_dim_tipoingreso
    - Map both fields
    - *Save it as "LoadAdmissionType"*
  - Patient:
    - Source table: pacientes.paciente
      - Source fields: "codpaciente", "sexo"
    - Target table: dw_YOUR_SCHEMA.dw_dim_patient
    - Map both fields
    - *Save it as "LoadPatient"*

Load time dimension

- Objective (step):
  - Generate 1000 dates (generate rows)
  - Create a sequence to be added (sequence)
  - Add to every date the i-th number of the sequence (calculator)
  - Extract year and month (calculator)
  - Create subrogate key (calculator) -> the same sequence
  - Read parameter (read variable)(define the variable in the transformation)
  - Filter rows older than the parameter date
  - Store in the "time" table using the consecutive pk

Load time dimension

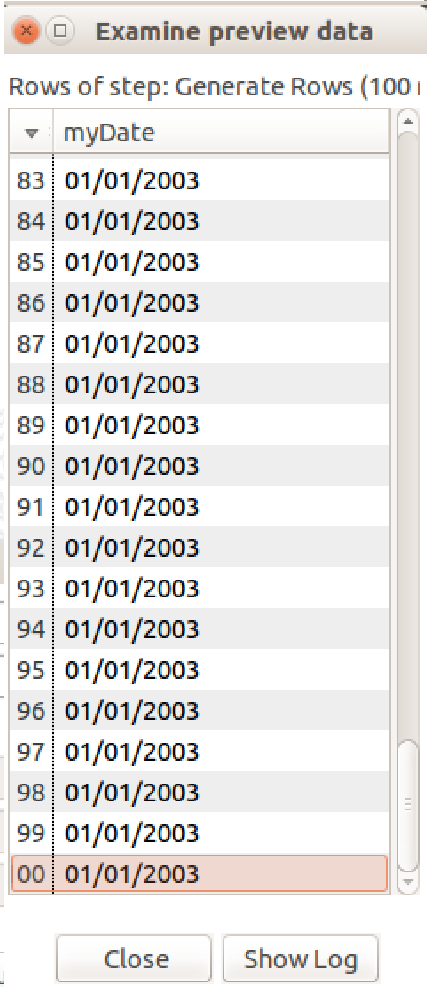
- We enter the following steps
  1. Input – Generate rows
  2. Transform – Add Sequence
  3. Transform – Calculator
  4. Job – Get variables
  5. Flow – Filter rows
  6. Flow – Dummy transformation
  7. Out – Table writer



Load time dimension

UNIVERSIDAD DE MURCIA

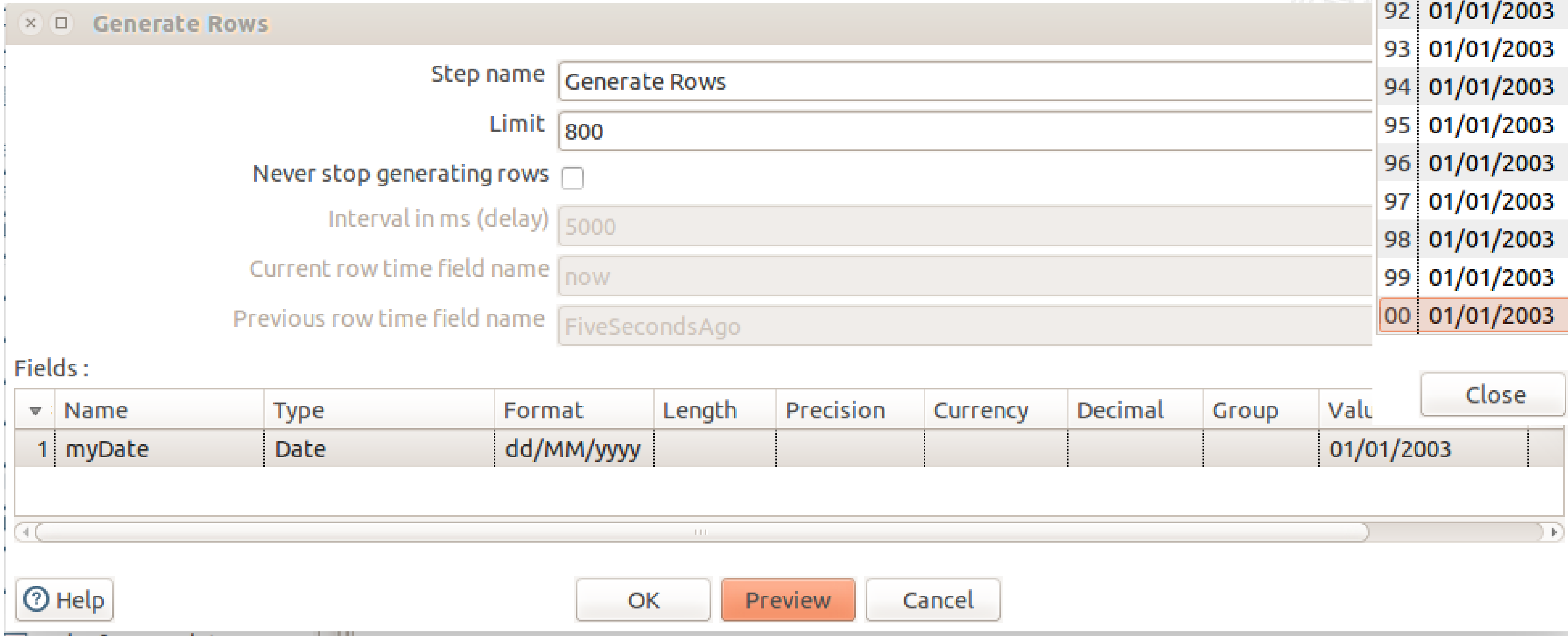
- Generate rows
  1. Input – Generate rows
  2. Set format and initial value
  3. Limit the number of rows: 800



Examine preview data

Rows of step: Generate Rows (100

	myDate
83	01/01/2003
84	01/01/2003
85	01/01/2003
86	01/01/2003
87	01/01/2003
88	01/01/2003
89	01/01/2003
90	01/01/2003
91	01/01/2003
92	01/01/2003
93	01/01/2003
94	01/01/2003
95	01/01/2003
96	01/01/2003
97	01/01/2003
98	01/01/2003
99	01/01/2003
00	01/01/2003



Generate Rows

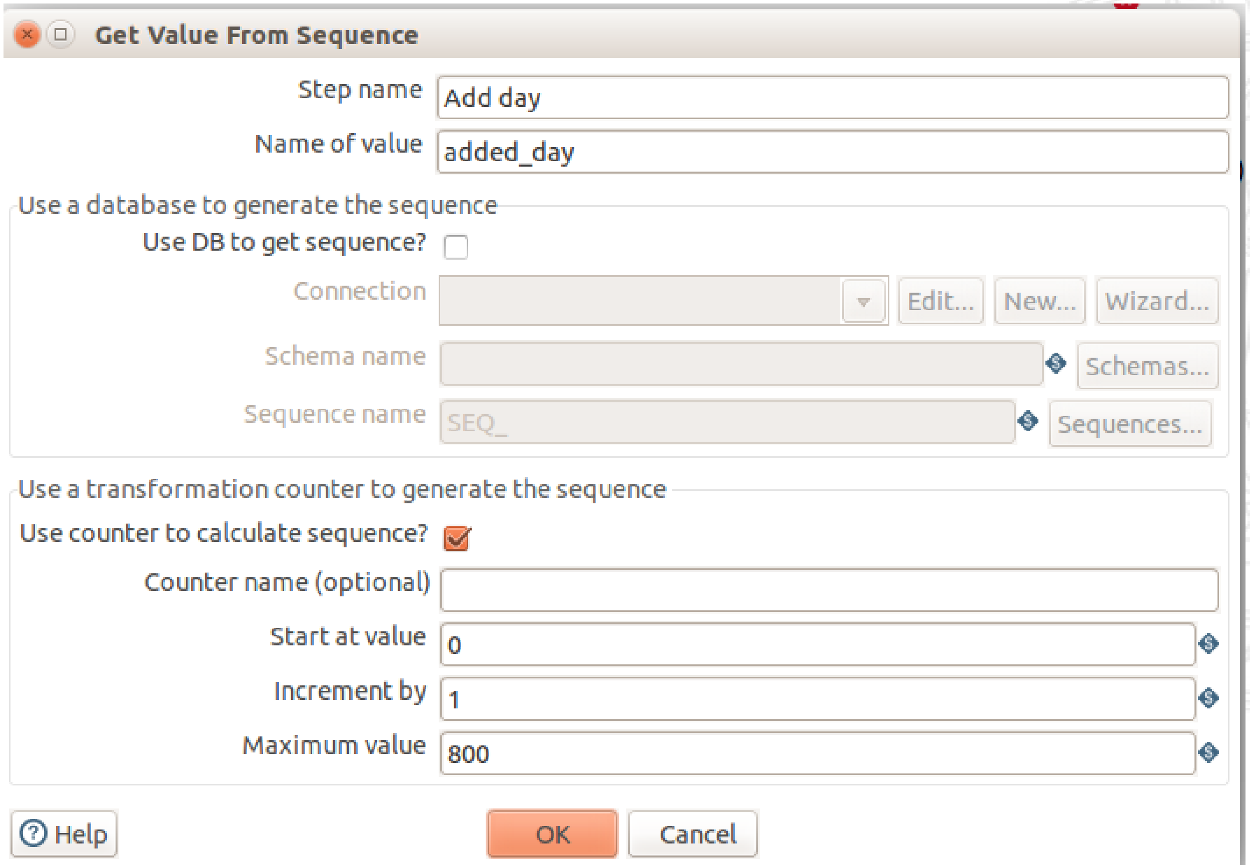
Step name	Generate Rows
Limit	800
Never stop generating rows	☐
Interval in ms (delay)	5000
Current row time field name	now
Previous row time field name	FiveSecondsAgo

Fields :

	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Valu
1	myDate	Date	dd/MM/yyyy						01/01/2003

Load time dimension

- Add a day
  1. Transform – Add Sequence
  2. Set a name for the value
  3. Set inital, increment and max value



Get Value From Sequence

Step name	Add day
Name of value	added_day

Use a database to generate the sequence

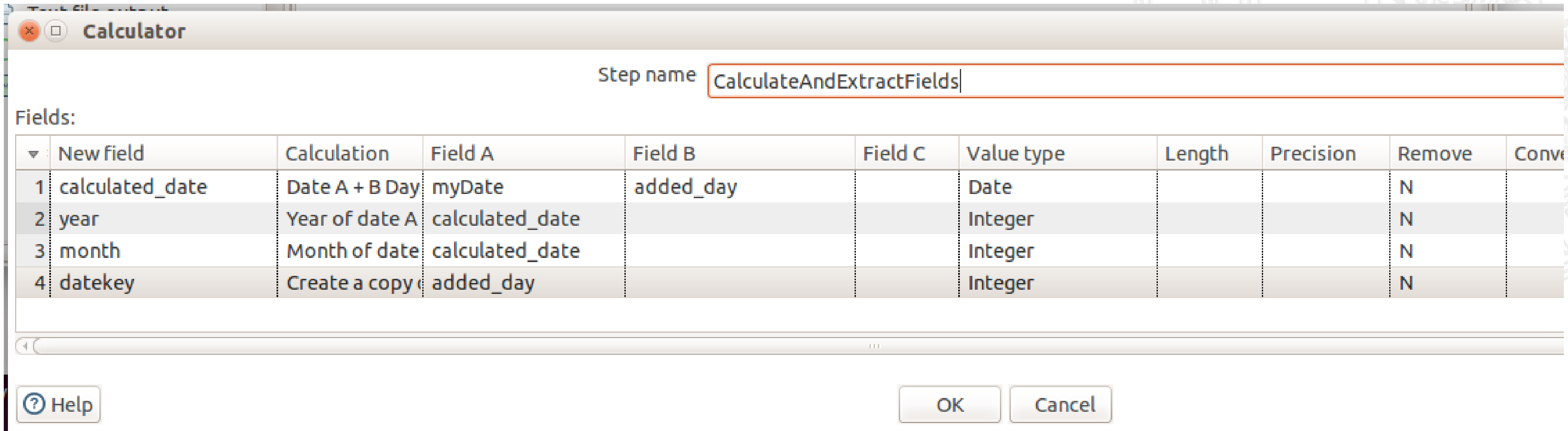
Use DB to get sequence?	☐
Connection	
Schema name	
Sequence name	SEQ_

Use a transformation counter to generate the sequence

Use counter to calculate sequence?	✓
Counter name (optional)	
Start at value	0
Increment by	1
Maximum value	800

Load time dimension

- Transform – Calculator
  1. Calculate new date
  2. Extract year and month
  3. We also create the subrogate key (comment from lecturer)
  4. Note:
     1. Important: Date type
     2. Remove?



Step name: CalculateAndExtractFields

Fields:

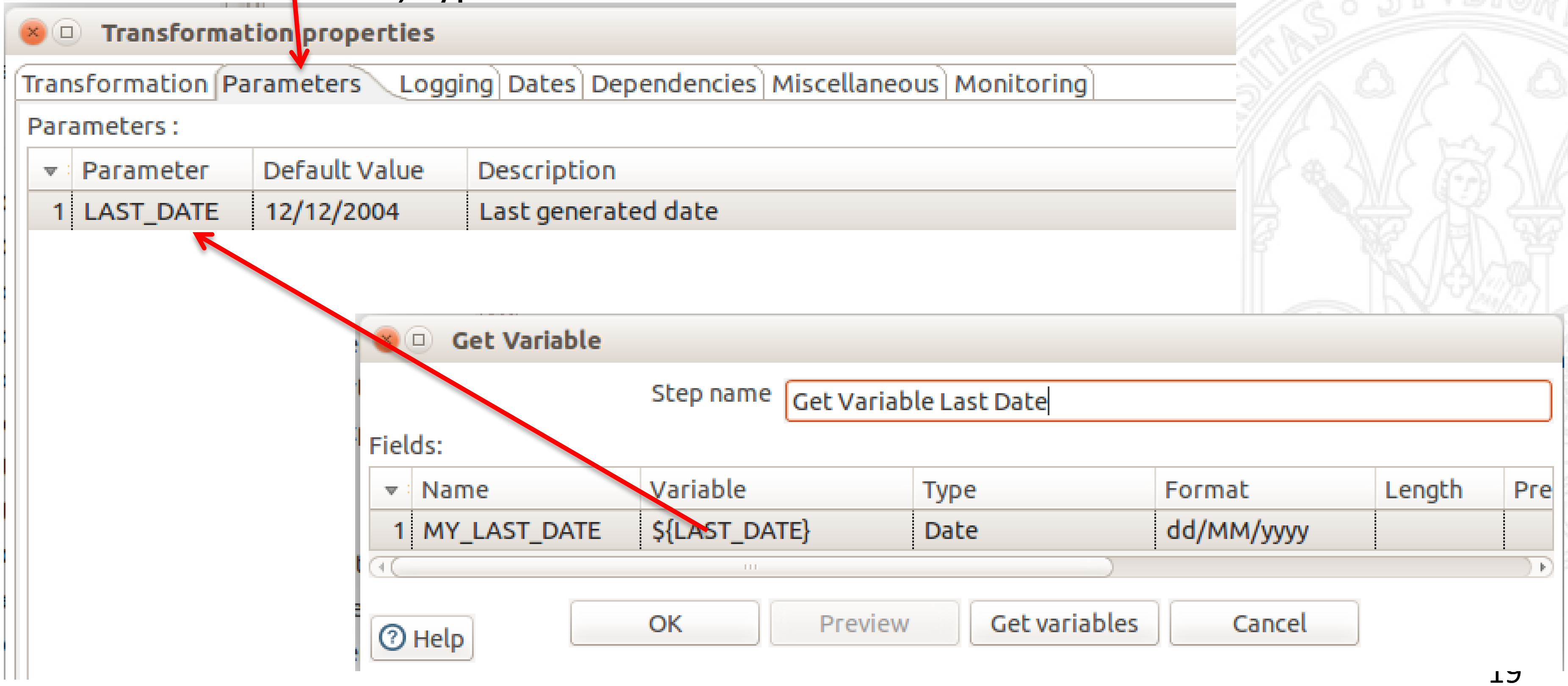
	New field	Calculation	Field A	Field B	Field C	Value type	Length	Precision	Remove	Conve
1	calculated_date	Date A + B Day	myDate	added_day		Date			N	
2	year	Year of date A	calculated_date			Integer			N	
3	month	Month of date	calculated_date			Integer			N	
4	datekey	Create a copy	added_day			Integer			N	

Load time dimension

- Job – Read variable

1. (background of transformation) Right Click->Properties: Define in “Parameters” tab the variable and value in the transformation

2. Read variable, type and include format



Transformation properties

Transformation | Parameters | Logging | Dates | Dependencies | Miscellaneous | Monitoring

Parameters :

	Parameter	Default Value	Description
1	LAST_DATE	12/12/2004	Last generated date

Get Variable

Step name: Get Variable Last Date

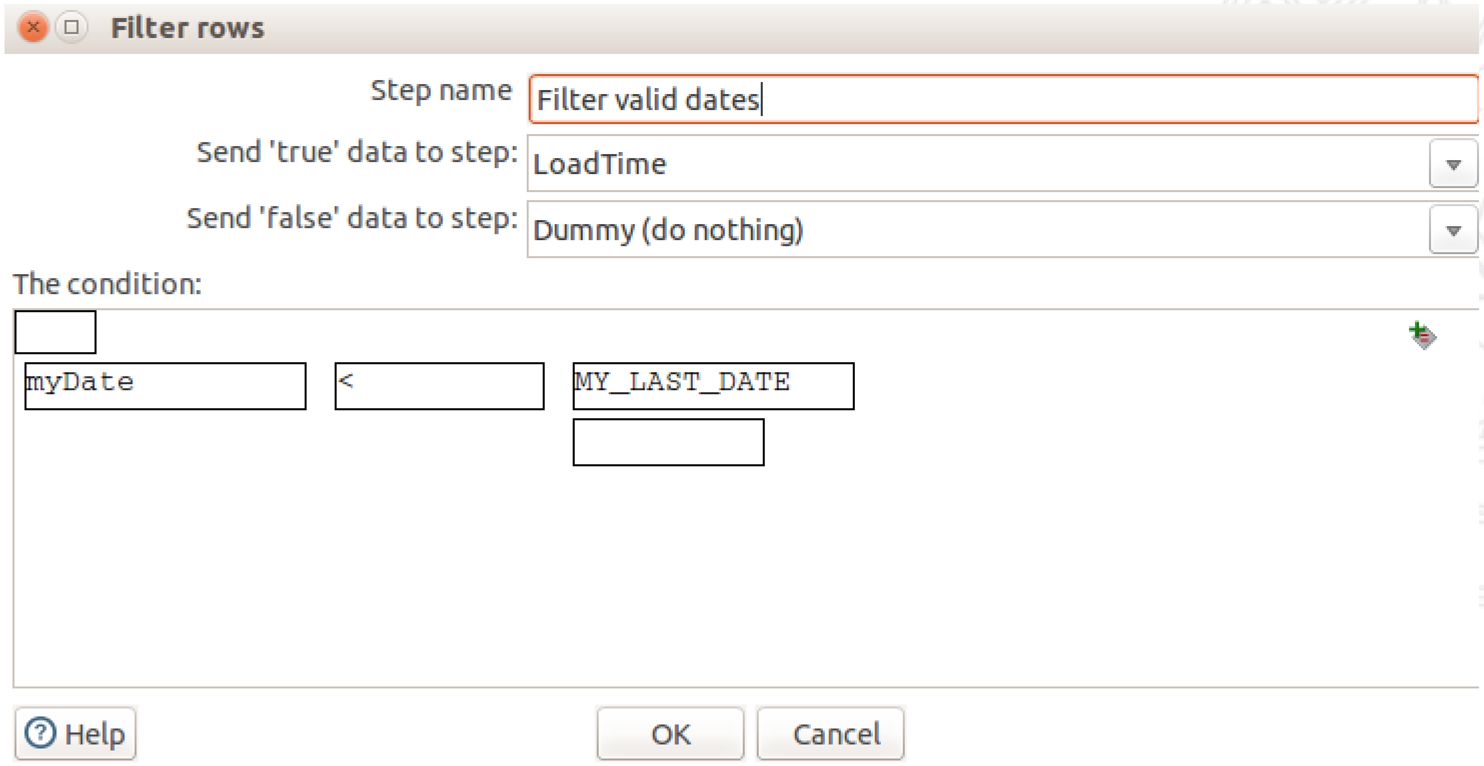
Fields:

	Name	Variable	Type	Format	Length	Pre
1	MY_LAST_DATE	${LAST_DATE}	Date	dd/MM/yyyy		

Load time dimension

UNIVERSIDAD DE MURCIA

- Filter rows
  1. Flow – Filtrer rows
  2. Flow – Dummy transformation: “Discard dates”
  3. Sent to dummy transformation rows after last date



Step name: Filter valid dates

Send 'true' data to step: LoadTime

Send 'false' data to step: Dummy (do nothing)

The condition:

myDate < MY_LAST_DATE

Load time dimension

UNIVERSIDAD DE MURCIA

- Out – Table writter

1. Mapping database fields

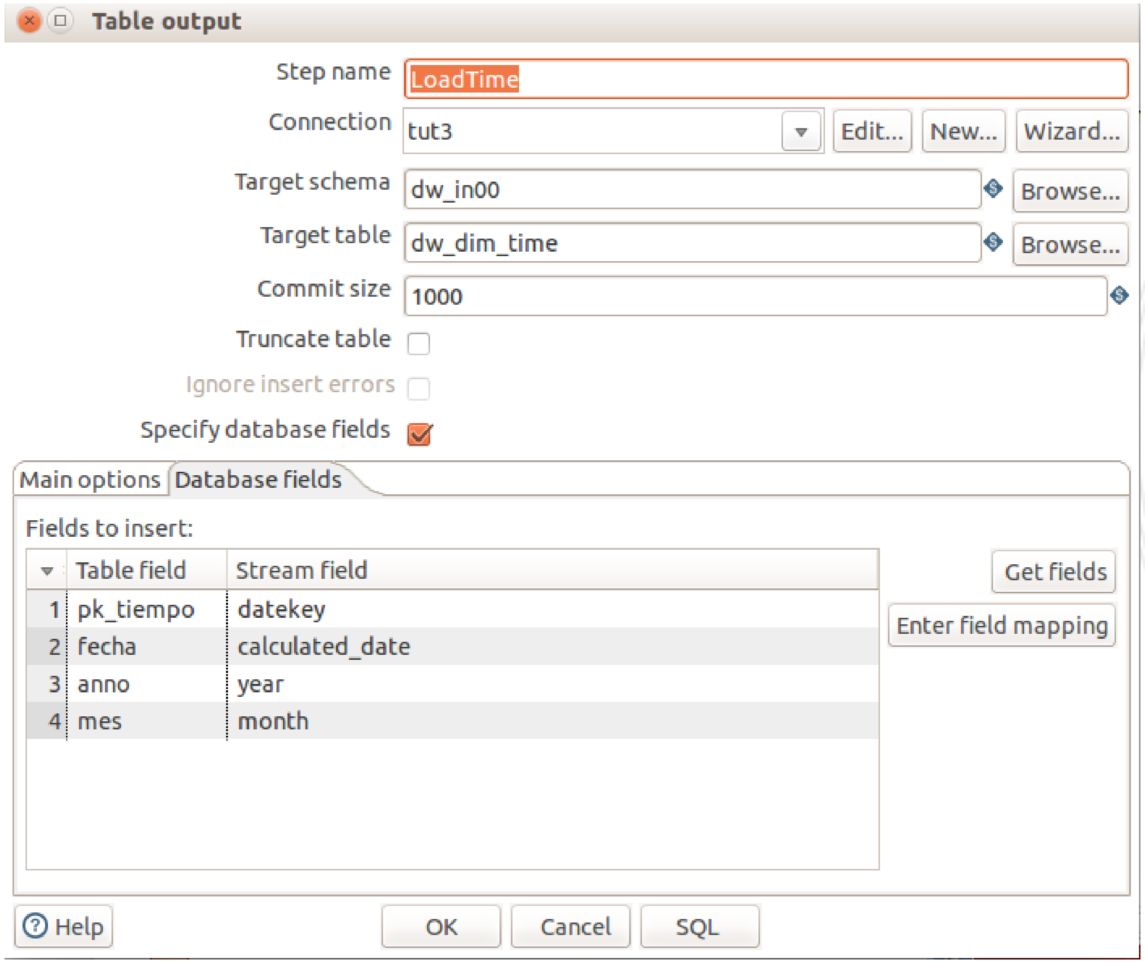


Table output

Step name: LoadTime

Connection: tut3

Target schema: dw_in00

Target table: dw_dim_time

Commit size: 1000

Truncate table: ☐

Ignore insert errors: ☐

Specify database fields: ✓

Main options | Database fields

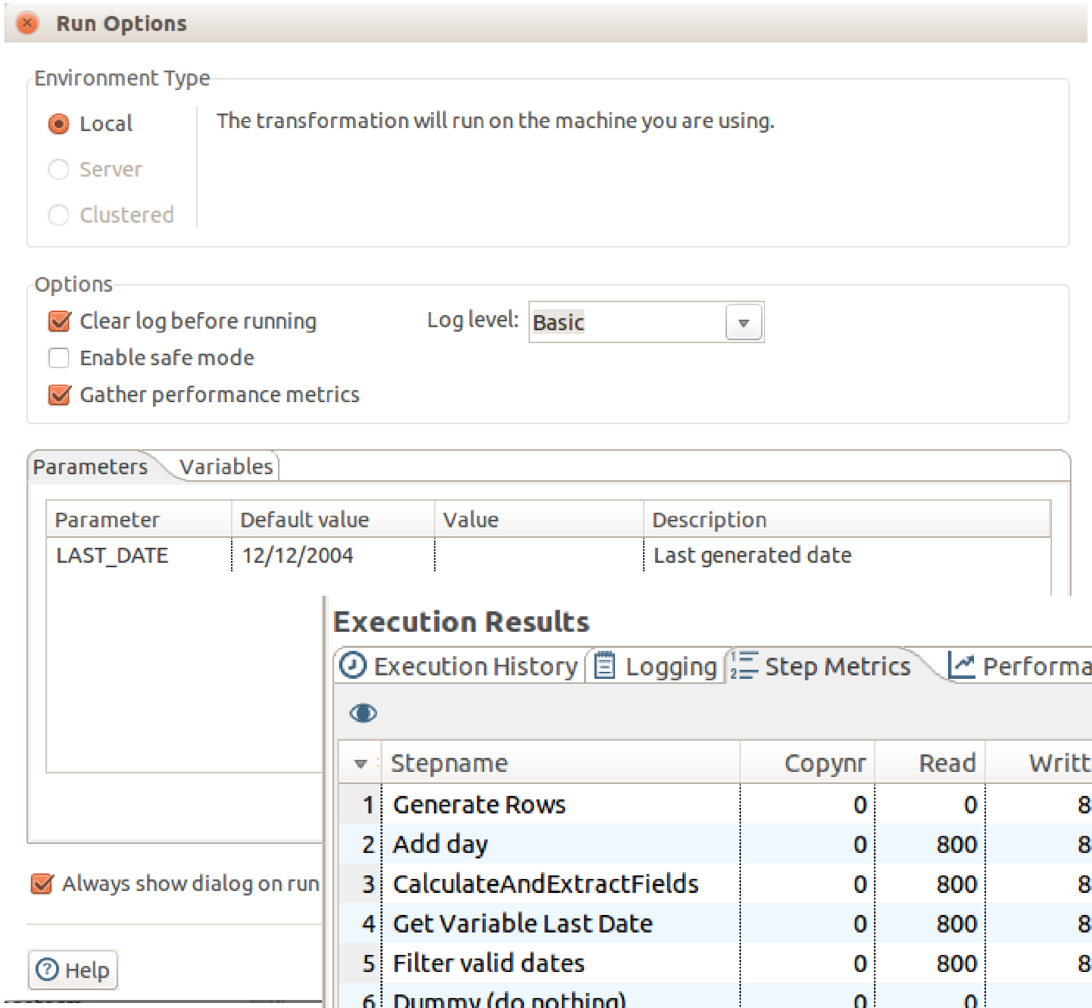
Fields to insert:

	Table field	Stream field
1	pk_tiempo	datekey
2	fecha	calculated_date
3	anno	year
4	mes	month

Load time dimension

UNIVERSIDAD DE MURCIA

- Run job with parameters



Run Options

Environment Type

- [✓ Local] The transformation will run on the machine you are using.
- Server
- Clustered

Options

- [✓] Clear log before running
- [ ] Enable safe mode
- [✓] Gather performance metrics

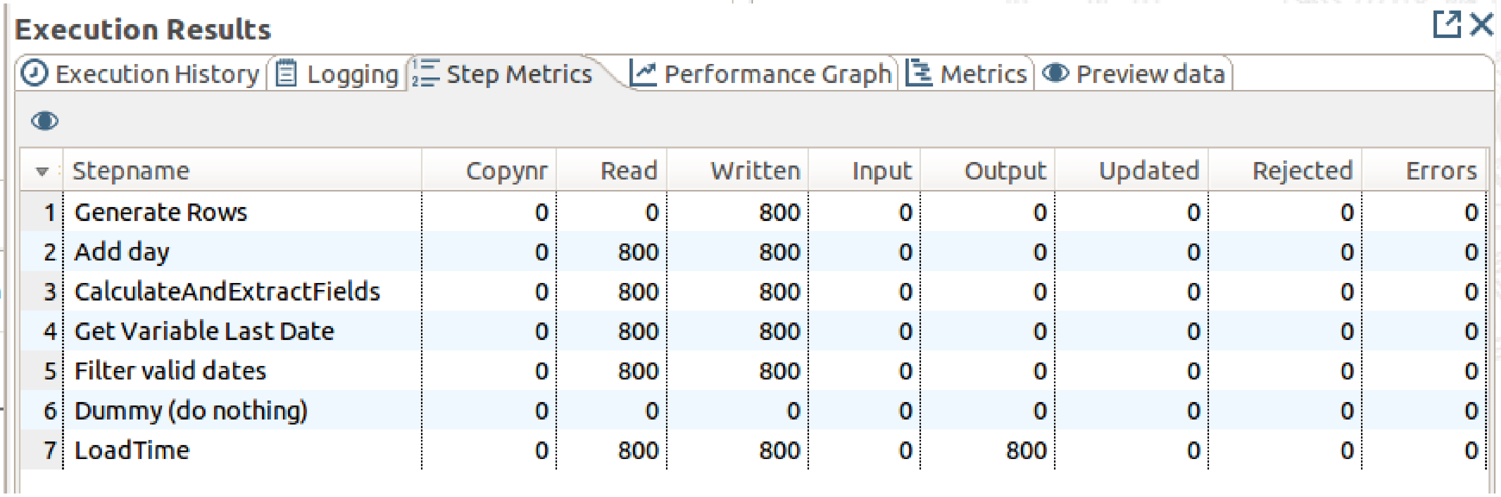
Log level: Basic

Parameters | Variables

Parameter	Default value	Value	Description
LAST_DATE	12/12/2004		Last generated date

[✓] Always show dialog on run

Help



Execution Results

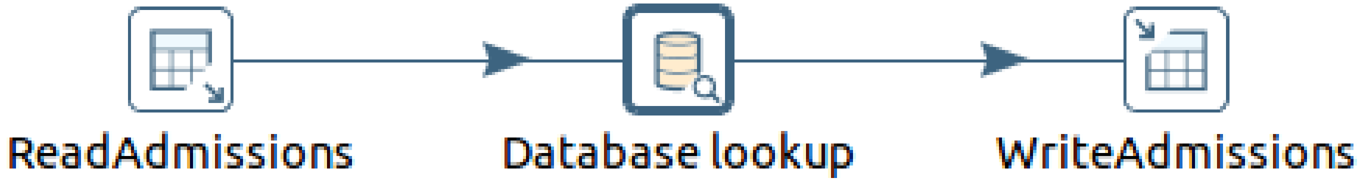
Execution History | Logging | Step Metrics | Performance Graph | Metrics | Preview data

	Stepname	Copynr	Read	Written	Input	Output	Updated	Rejected	Errors
1	Generate Rows	0	0	800	0	0	0	0	0
2	Add day	0	800	800	0	0	0	0	0
3	CalculateAndExtractFields	0	800	800	0	0	0	0	0
4	Get Variable Last Date	0	800	800	0	0	0	0	0
5	Filter valid dates	0	800	800	0	0	0	0	0
6	Dummy (do nothing)	0	0	0	0	0	0	0	0
7	LoadTime	0	800	800	0	800	0	0	0

Load facts

UNIVERSIDAD DE MURCIA

- Lets do the following
    1. Input -> Input table: Select some admissions
    2. Lookoup -> Database lookup: Foreign key in the time table
    3. Ouput -> Ouput table: Store the facts
- Note: this is not the normal procedure, just an example, we are not creating subrogate keys except for time.

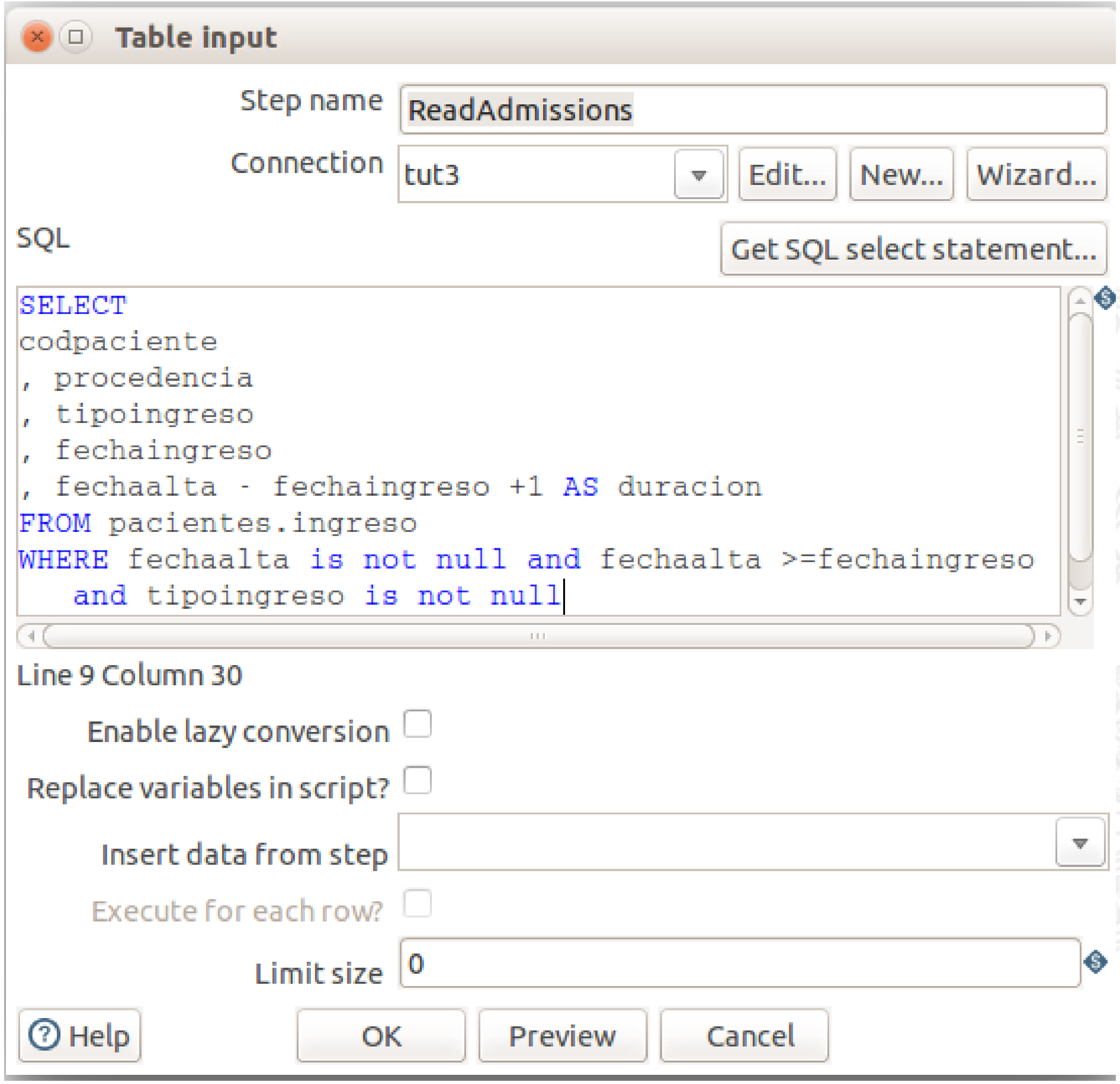


Load facts

- In fact table load:
  1. codpaciente
  2. procedencia
  3. tipoingreso
  4. fechaingreso,
  5. Meassure: duration

Conditions:

1. Discharged
   - fechaalta is not null
2. Correct dates
   - discharge >= admission
3. Correct tipoingreso
   - tipoingreso is not null



Step name: ReadAdmissions

Connection: tut3

SQL

```
SELECT
codpaciente
, procedencia
, tipoingreso
, fechaingreso
, fechaalta - fechaingreso +1 AS duracion
FROM pacientes.ingreso
WHERE fechaalta is not null and fechaalta >=fechaingreso
   and tipoingreso is not null
```

Line 9 Column 30

Enable lazy conversion ☐

Replace variables in script? ☐

Insert data from step:

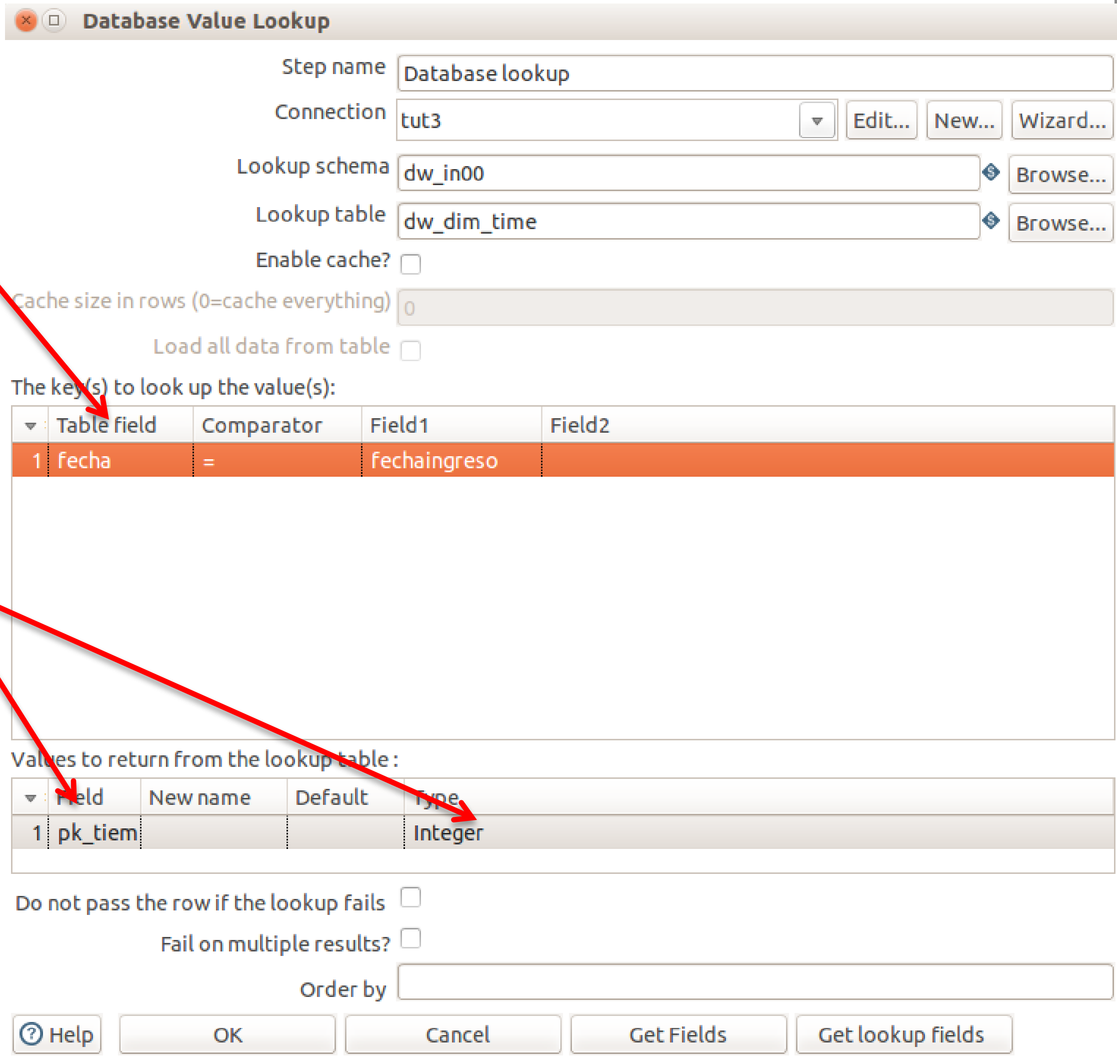
Execute for each row? ☐

Limit size: 0

Load facts

UNIVERSIDAD DE MURCIA

- Look up the primary key in time dimension matching the admission date
  1. Search – Database lookup
  2. Select lookup fields
  3. Select extracted field:
     1. Dimension pk
  - Remaining fields: optional
  - Careful with type



Database Value Lookup

Step name	Database lookup
Connection	tut3
Lookup schema	dw_in00
Lookup table	dw_dim_time
Enable cache?	☐
Cache size in rows (0=cache everything)	0
Load all data from table	☐

The key(s) to look up the value(s):

#	Table field	Comparator	Field1	Field2
1	fecha	=	fechaingreso	

Values to return from the lookup table :

#	Field	New name	Default	Type
1	pk_tiem			Integer

Do not pass the row if the lookup fails	☐
Fail on multiple results?	☐
Order by	

Load facts

- Store output in table
  - Map the fields

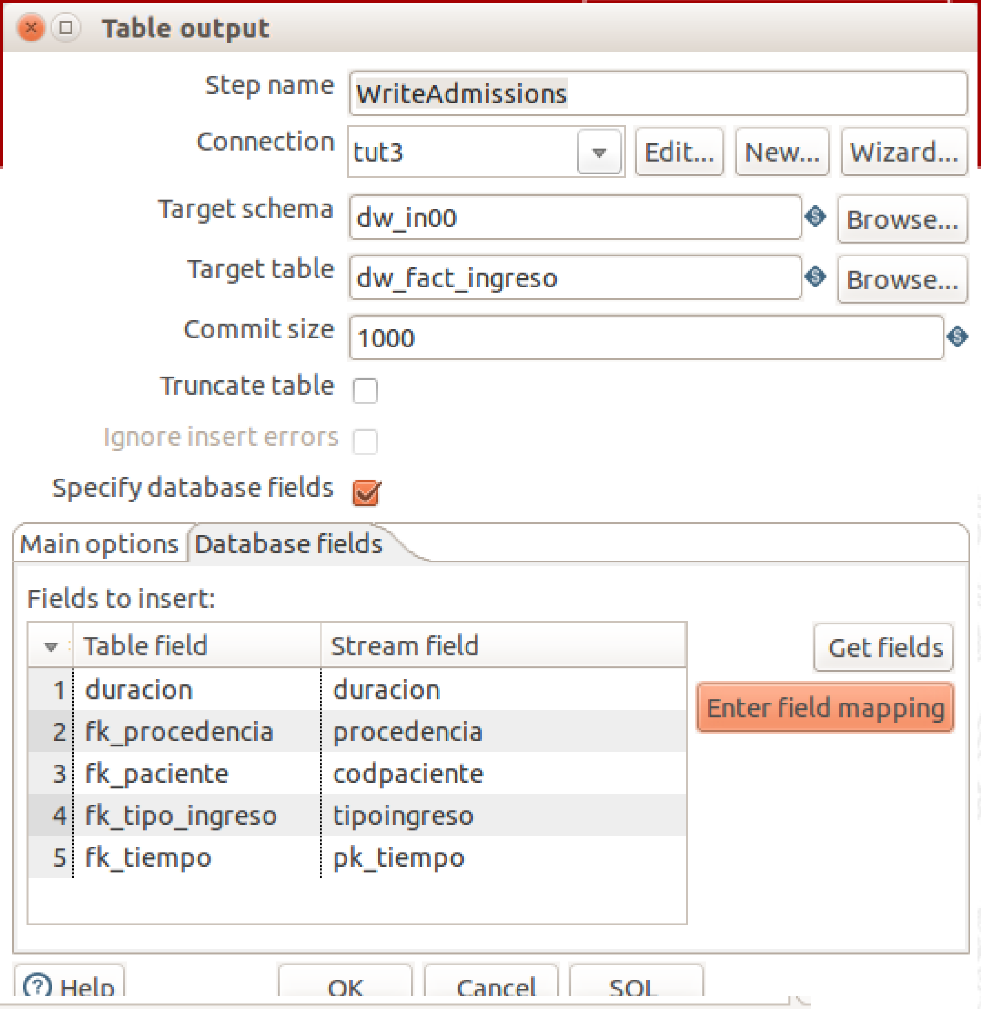
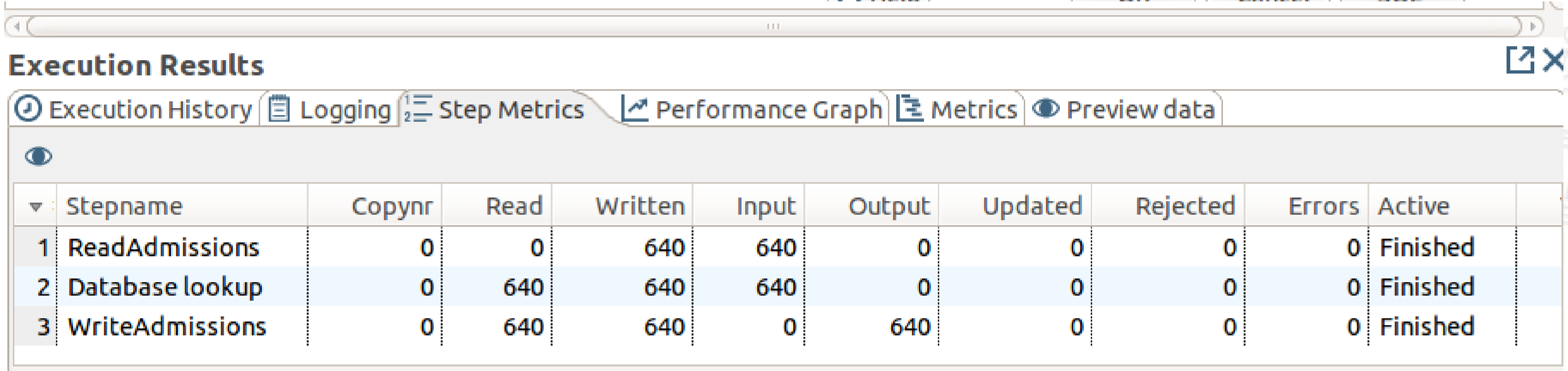


Table output

Step name	WriteAdmissions
Connection	tut3
Target schema	dw_in00
Target table	dw_fact_ingreso
Commit size	1000
Truncate table	☐
Ignore insert errors	☐
Specify database fields	✓

Fields to insert:

	Table field	Stream field
1	duracion	duracion
2	fk_procedencia	procedencia
3	fk_paciente	codpaciente
4	fk_tipo_ingreso	tipoingreso
5	fk_tiempo	pk_tiempo



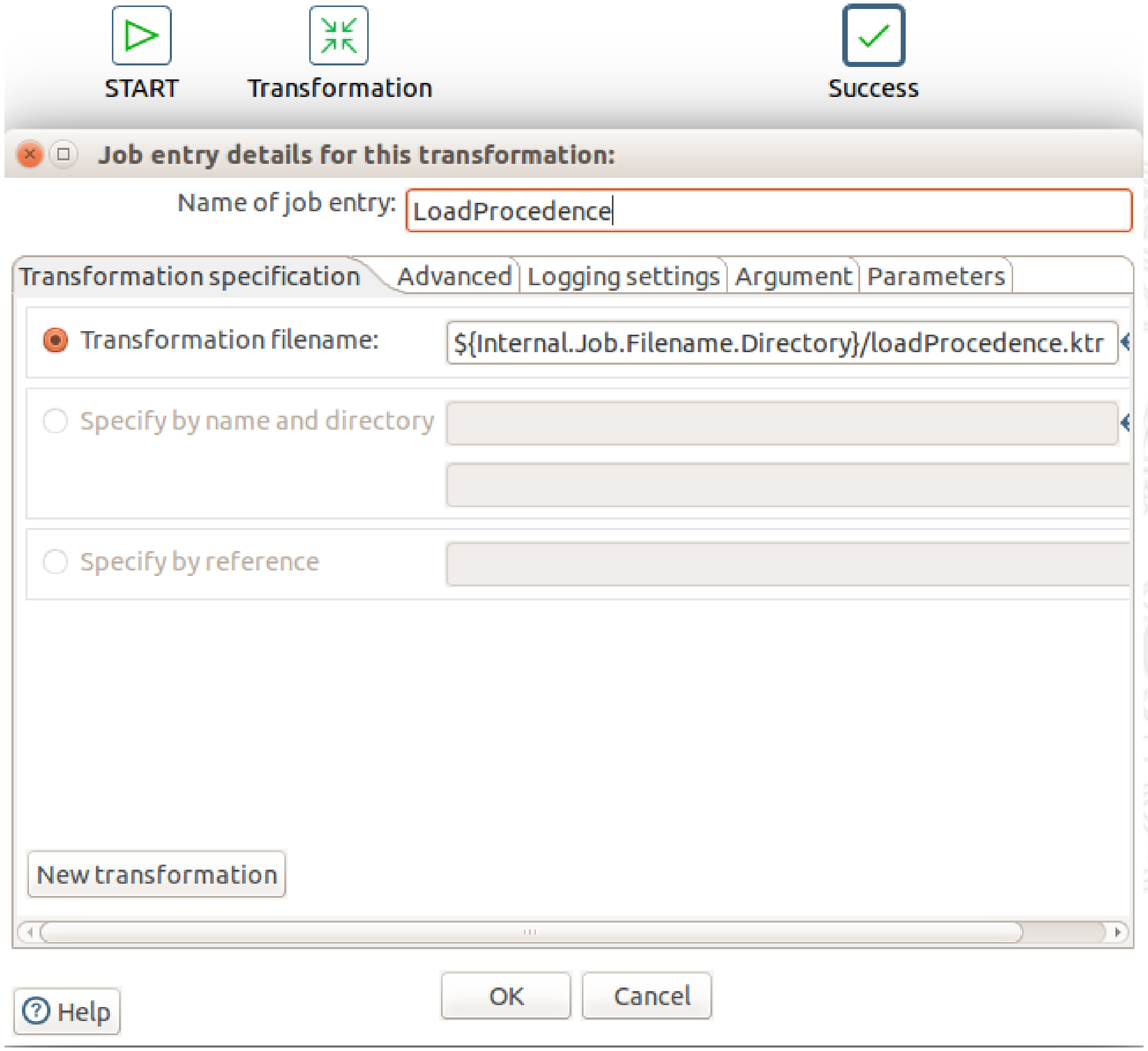
Execution Results

	Stepname	Copynr	Read	Written	Input	Output	Updated	Rejected	Errors	Active
1	ReadAdmissions	0	0	640	640	0	0	0	0	Finished
2	Database lookup	0	640	640	640	0	0	0	0	Finished
3	WriteAdmissions	0	640	640	0	640	0	0	0	Finished

Create job

UNIVERSIDAD DE MURCIA

- Put all transformations in a job
  - New File -> Job
  - General -> Start
  - General ->Transformation
  - General -> Success



Job entry details for this transformation:

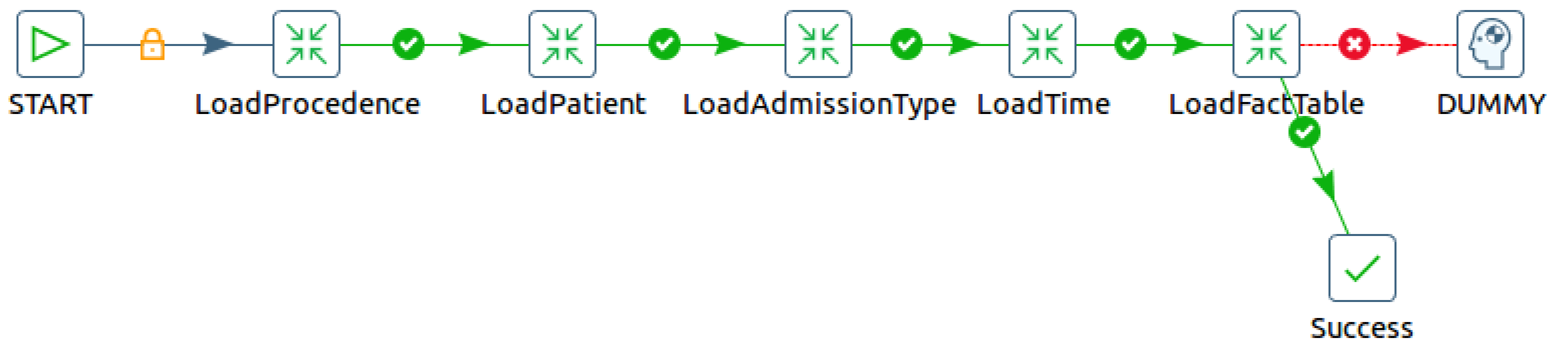
Name of job entry: LoadProcedence

● Transformation filename: ${Internal.Job.Filename.Directory}/loadProcedence.ktr

○ Specify by name and directory

○ Specify by reference

Create job



- Condition for links
- We can add a new transformation for creating or rebuilding the database
- Include also an error node to abort the job

Interesting Steps

- Input/output: CSV, XML, datagrid, Property files
- Transform: Select values
- Scripting: SQL Scripting, JavaScript
- Lookup: Database join
- Mapping: to create “subtransformations”
- Job: Get/set files/variables from results
- Filemanagement, filetransfer, mail (for jobs)

Tutorial

UNIVERSIDAD DE MURCIA

- Upload to the task in Aula Virtual a compressed file with the following:
  - Transformation definitions (.ktr files)
    - Load procedence
    - Load admission type
    - Load patient
    - Load time
    - Load Fact tables
    - Create and drop schema
  - Job definition (.kjb file)
    - Load all