Informatica PC Training

Day-4

Agenda:
➢ Aggregator Transformation
➢ Sorter Transformation
➢ Sequence Generator Transformation
➢ Stored Procedure Transformation
➢ Rank Transformation
➢ Demo

Debadatta Mohanty

Housekeeping Tips

- **Please mute your phone during the presentation.**

- **If there is too much noise, participants will be put on auto-mute.**

- **We shall open up the table for Q&A at the end of the session.**

- **Please feel free to post your questions over Chat as well.**

- **This session will be recorded and an email will be sent with links to the recordings after the session.**

- **At the end of the course, TEX will request you to provide feedback on the training.**

HCL

Aggregator Transformation

Aggregator Transformation

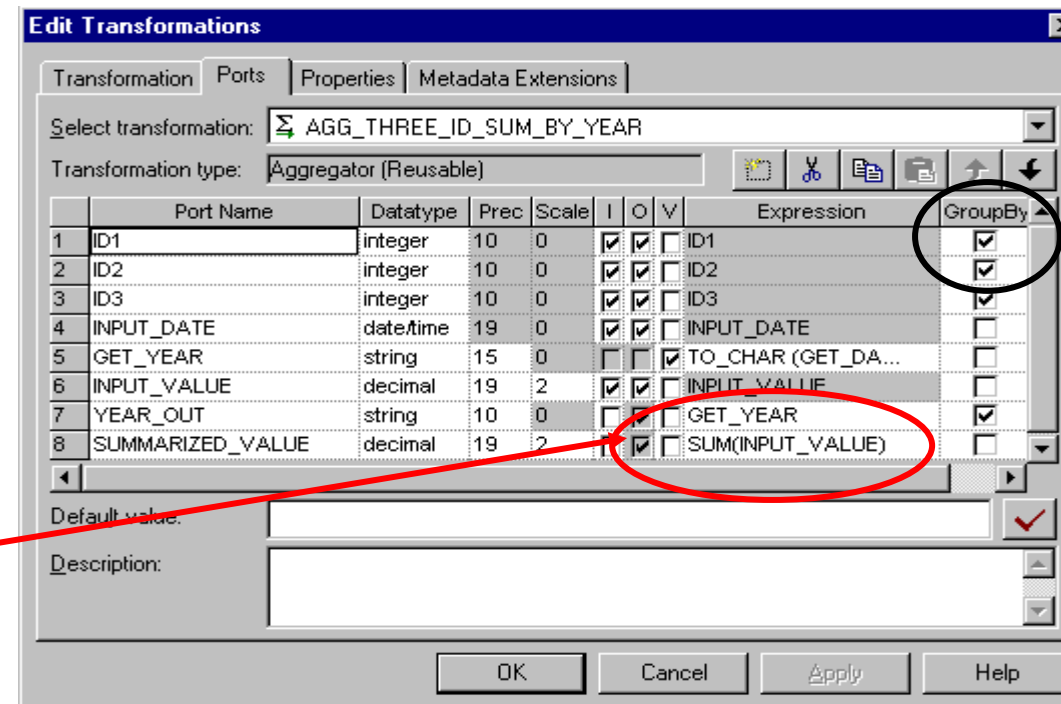Performs aggregate calculations

Active Transformation
Connected

Ports
- Mixed
- Variables allowed
- Group By allowed

Create expressions in output or variable ports

Usage
- Standard aggregations

Informatica Functions – Aggregator T/F

AVG
COUNT
FIRST
LAST
MAX
MEDIAN
MIN
PERCENTILE
STDDEV
SUM
VARIANCE

Aggregate Functions

- Return summary values for non-null data in selected ports
- Use only in Aggregator transformations
- Use in output ports only
- Calculate a single value (and row) for all records in a group
- Only one aggregate function can be nested within an aggregate function
- Conditional statements can be used with these functions

Aggregate Expressions

Aggregate functions are supported only in the Aggregator Transformation
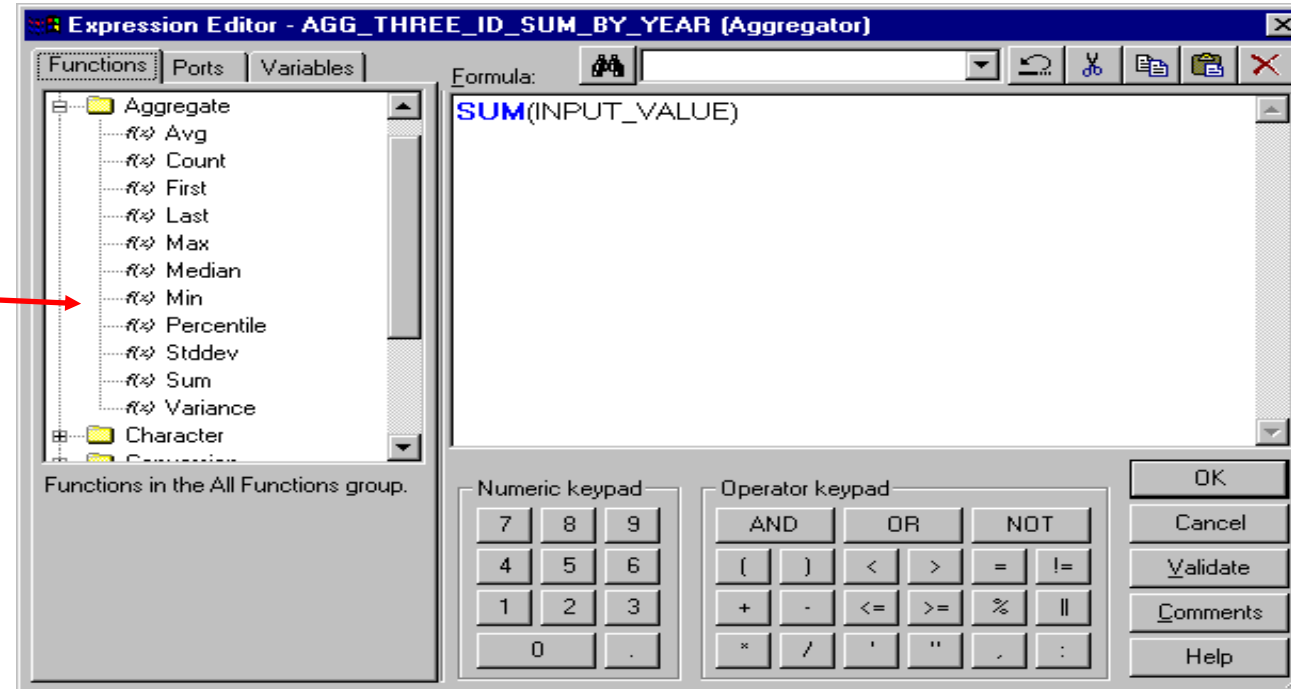
Conditional Aggregate expressions are supported

Conditional SUM format: SUM(value, condition)

Aggregator T/F - Properties

Sorted Input Property

Instructs the Aggregator to expect the data to be sorted

Set Aggregator cache sizes (on Informatica Server machine)

Sorted Data – Aggregator T/F

- The Aggregator can handle sorted or unsorted data

  - Sorted data can be aggregated more efficiently, decreasing total processing time

  - The Server will cache data from each group and release the cached data -- upon reaching the first record of the next group

  - Data must be sorted according to the order of the Aggregator "Group By" ports

  - Performance gain will depend upon varying factors
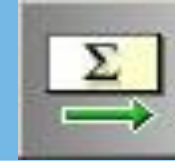
Aggregating Unsorted Data
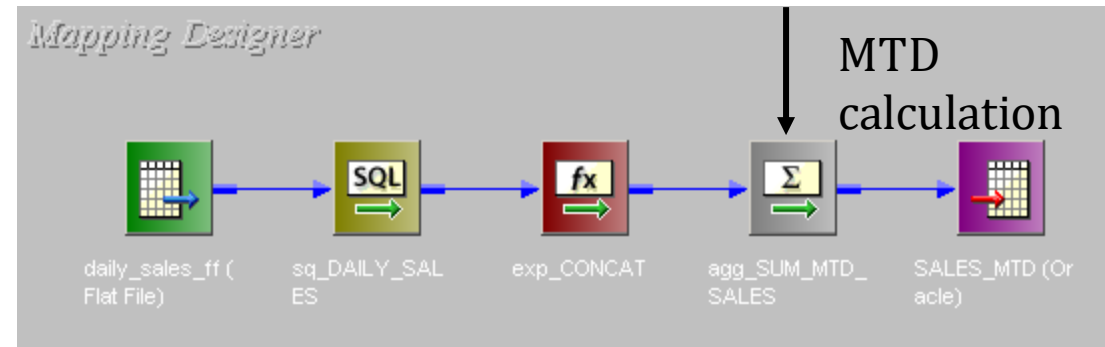
Unsorted data

Group By: store, department, date
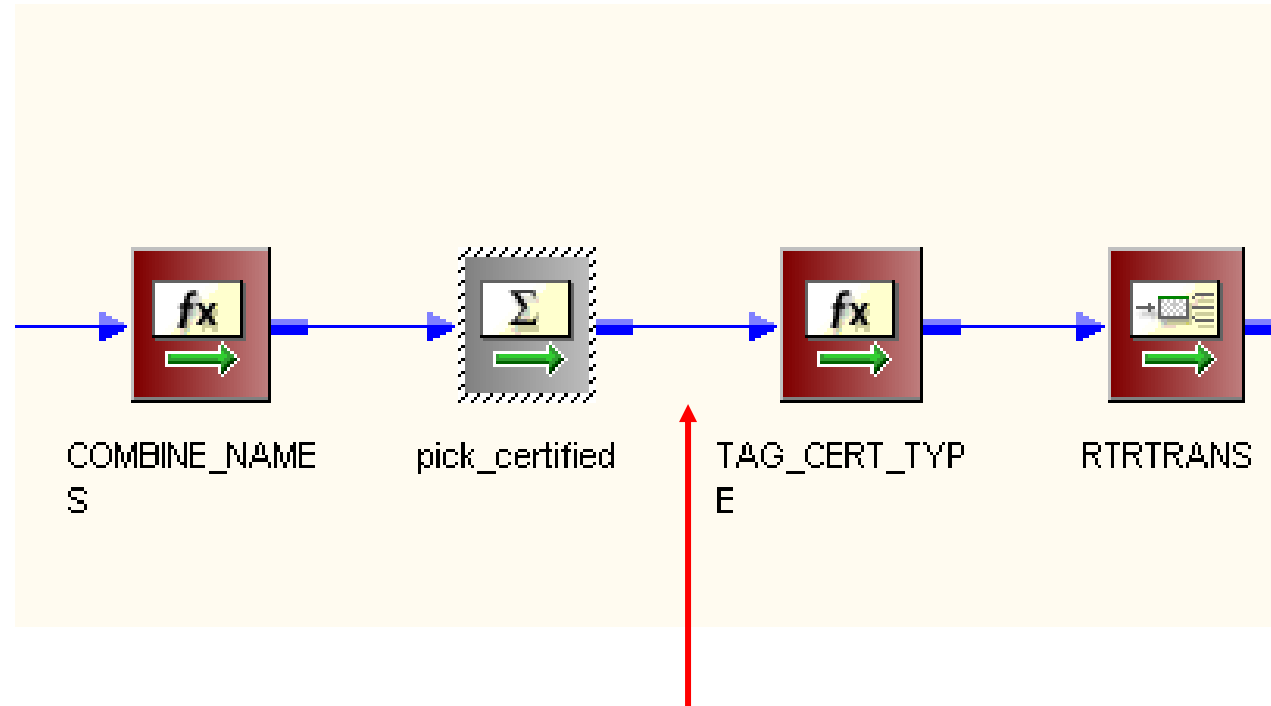
No rows are released from Aggregator until all rows are aggregated

Incremental Aggregation

Trigger in   Session
Properties,
Performance Tab

MTD
calculation

- Cache is saved into $PMCacheDir: *aggregatorname.DAT*
                                                 aggregatorname.IDX

- Upon next run, files are overwritten with new cache information

    Example: When triggered, PowerCenter Server will save new
    MTD totals. Upon next run (new totals), Server will subtract
    old totals; difference will be passed forward

Best Practice is to copy these files in case a rerun of data is ever required. Reinitialize
when no longer needed, e.g. – at the beginning new month processing

Sorter Transformation

Sorter Transformation

- Can sort data from relational tables or flat files

- Sort takes place on the Informatica Server machine

- Multiple sort keys are supported

- The Sorter transformation is often more efficient than a sort performed on a database with an ORDER BY clause

- Can use sort transformation to give output row as DISTINCT Values.

Sorter Transformation
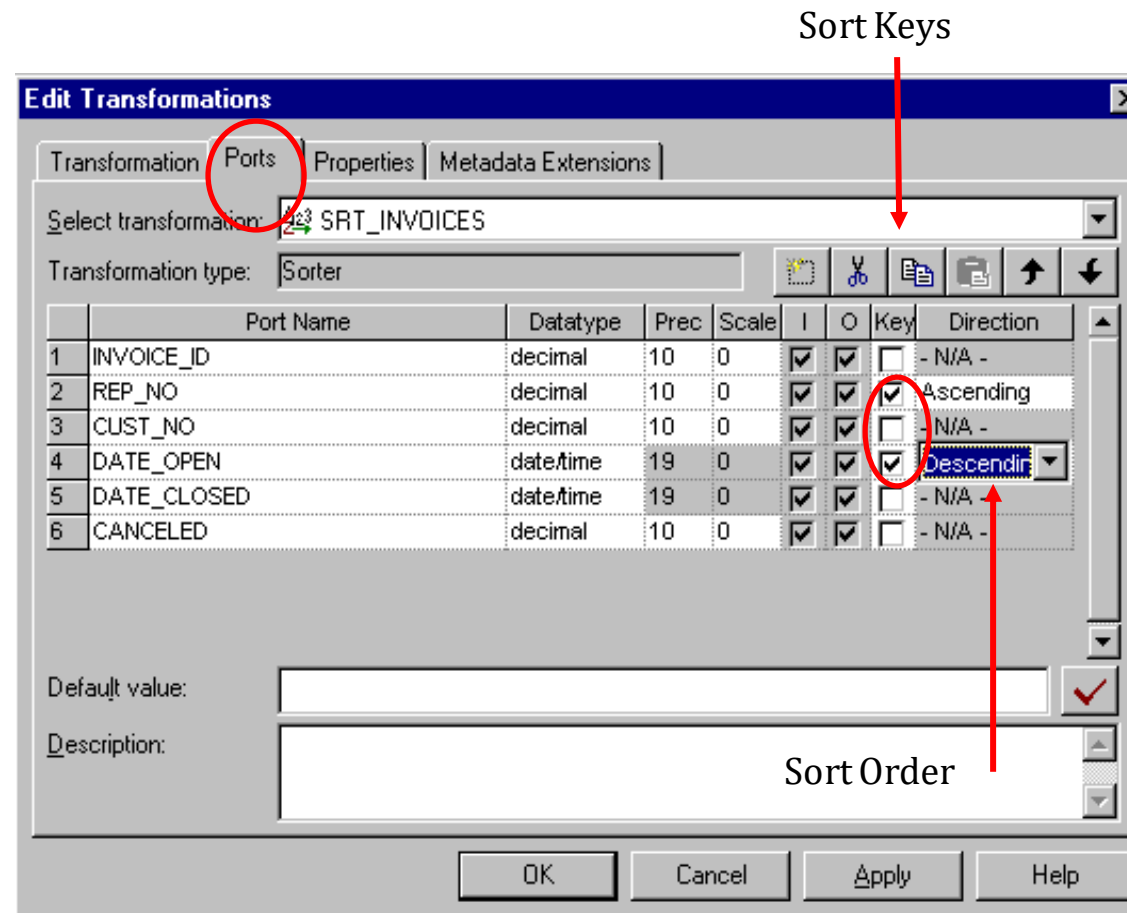
Sorts data from any source, at any point in a data flow
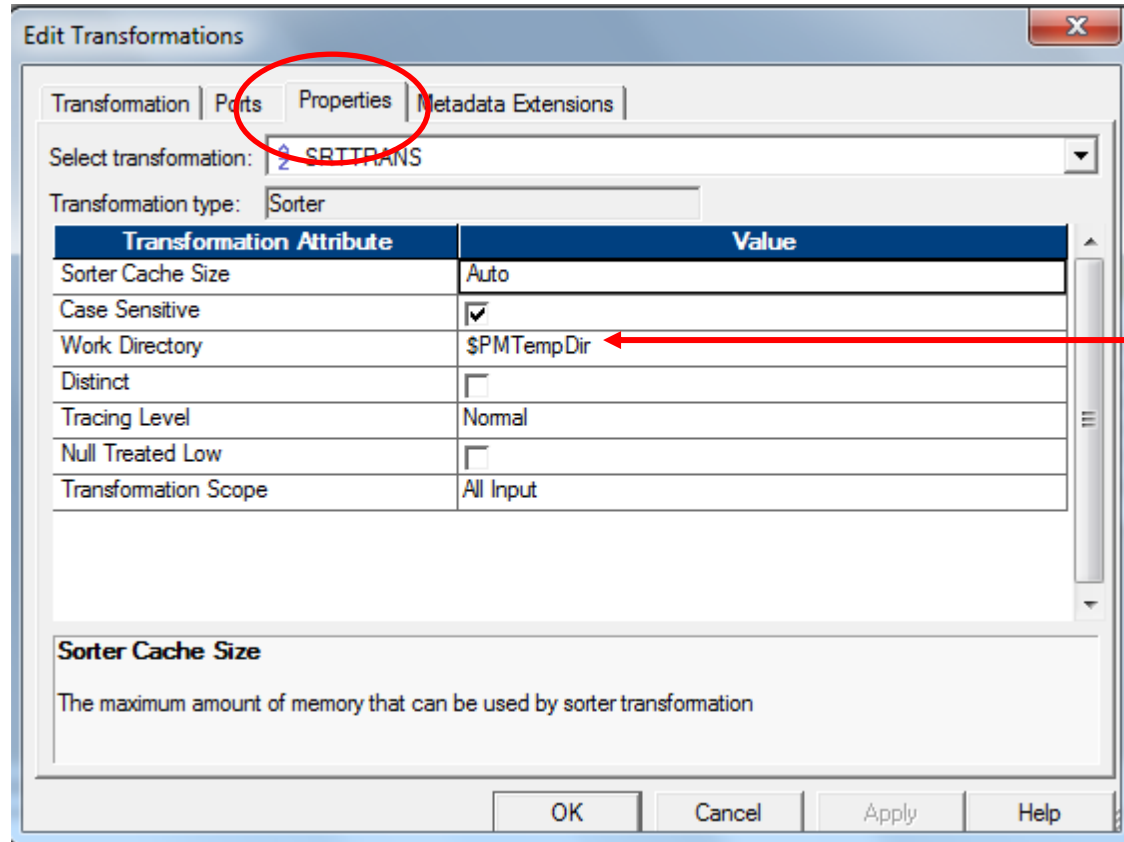
Active Transformation
Connected

Ports
- Input/Output
- Define one or more sort keys
- Define sort order for each key

Usage
- Sort data from Flat Files
- Re-sort data in a mapping

Sort Keys

Edit Transformations

Transformation | Ports | Properties | Metadata Extensions

Select transformation: SRT_INVOICES

Transformation type: Sorter

	Port Name	Datatype	Prec	Scale	I	O	Key	Direction
1	INVOICE_ID	decimal	10	0	☑	☑	☐	- N/A -
2	REP_NO	decimal	10	0	☑	☑	☑	Ascending
3	CUST_NO	decimal	10	0	☑	☑	☐	- N/A -
4	DATE_OPEN	date/time	19	0	☑	☑	☑	Descending
5	DATE_CLOSED	date/time	19	0	☑	☑	☐	- N/A -
6	CANCELED	decimal	10	0	☑	☑	☐	- N/A -

Default value:

Description:

Sort Order

OK | Cancel | Apply | Help

Sorter Transformation

Cache size can not be adjusted. It will increase Cache memory by size.

Default is 16 Mb

Server uses twice the cache listed

Sequence Generator Transformation

Sequence Generator Transformation

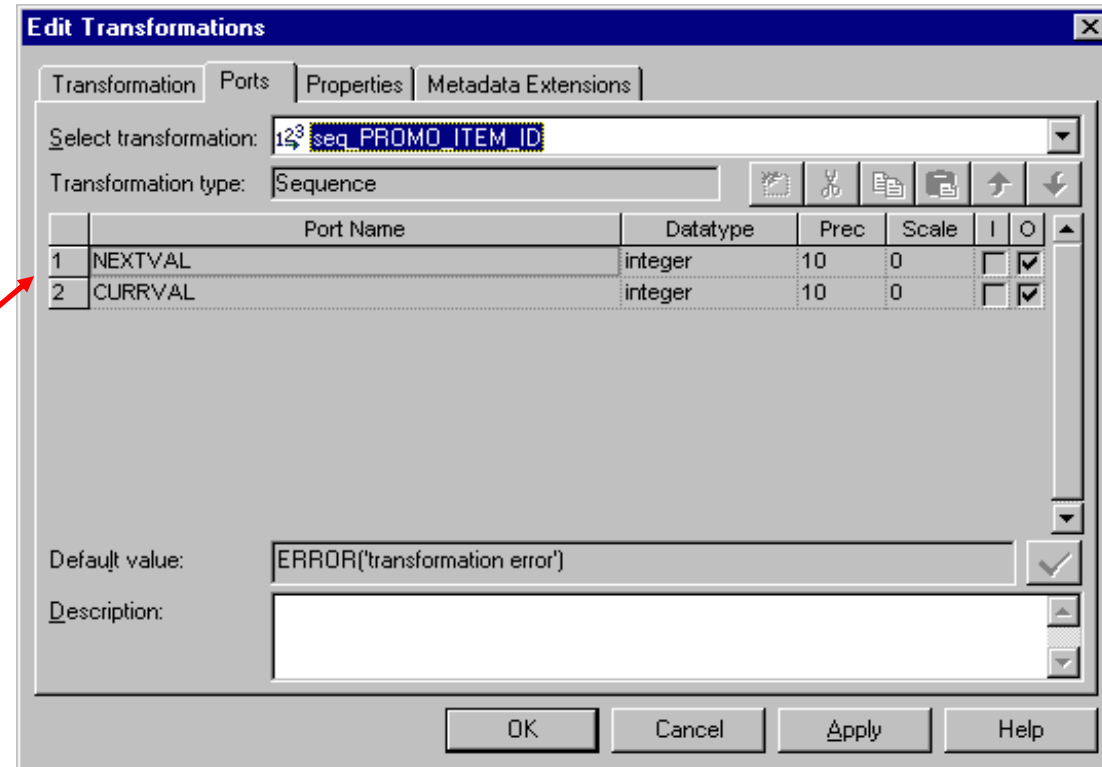Generates unique keys for any port on a row

Passive Transformation
Connected

Ports
- Two predefined output ports, NEXTVAL and CURRVAL
- No input ports allowed

Usage
- Generate sequence numbers
- Shareable across mappings
- Cycle through a sequential range of numbers.

Sequence Generator Transformation

Start Value
Increment By

Number of
Cached
Values

Edit Transformations

Transformation | Ports | Properties | Metadata Extensions

Select transformation: SEQ_EMP_NO

Transformation type: Sequence

Transformation Attribute	Value
Start Value	0
Increment By	1
End Value	9223372036854775807
Current Value	29
Cycle	☐
Number of Cached Values	30
Reset	☐
Tracing Level	Normal

Start Value

First value generated for the sequence

OK   Cancel   Apply   Help

Stored Procedure Transformation

Stored Procedure Transformation

- A Stored Procedure transformation is
    - an important tool for populating and maintaining databases
    - used to call a stored procedure

- It's **Passive** and can be **Connected** or **Unconnected** Transformation.

- The stored procedure must exist in the database before creating a Stored Procedure transformation

- One of the most useful features of stored procedures is the ability to send data to the stored procedure, and receive data from the stored procedure

- There are three types of data that pass between the Integration Service and the stored procedure:
    - **Input / Output parameters** - For many stored procedures, you provide a value and receive a value in return
    - **Return values** - Most databases provide a return value after running a stored procedure
    - **Status codes** - The stored procedure issues a status code that notifies whether or not it completed successfully. Status codes provide error handling for the Integration Service during a session.

Properties

- **Stored Procedure Name** – Name of the procedure/function as in the database
- **Connection Information** – Database containing the stored procedure. Default value is $Target
- **Call Text** – This is used only when transformation type is not Normal. Input parameters passed to SP can be included within the call text.
- **Execution Order** – If the mapping calls more than one source or target pre or post load stored procedure, they are executed in the specified order.
- **Stored Procedure Type** - Specifies when the stored procedure runs.

Edit Transformations

Transformation | Ports | Properties | Metadata Extensions

Select transformation: SP_FUNT

Transformation type: Stored Procedure

Transformation Attribute	Value
Stored Procedure Name	FN_SALGRADE
Connection Information	$Target
Call Text	
Stored Procedure Type	Normal
Execution Order	1
Tracing Level	Normal
Subsecond Precision	6
Output is Repeatable	Based On Input Order
Output is Deterministic	

Output is Repeatable

Specifies whether the output data is ordered: Based On Input Order. This property is Based On Input Order for

OK | Cancel | Apply | Help

The following list describes the options for running a Stored Procedure transformation:

- **Normal** - During a session, the stored procedure runs where the transformation exists in the mapping on a row-by-row basis
- **Source Pre-load** - Before the session retrieves data from the source, the stored procedure runs
- **Source Post-load** - After the session retrieves data from the source, the stored procedure runs
- **Target Pre-load** - Before the session sends data to the target, the stored procedure runs
- **Target Post-load** - After the session sends data to the target, the stored procedure runs
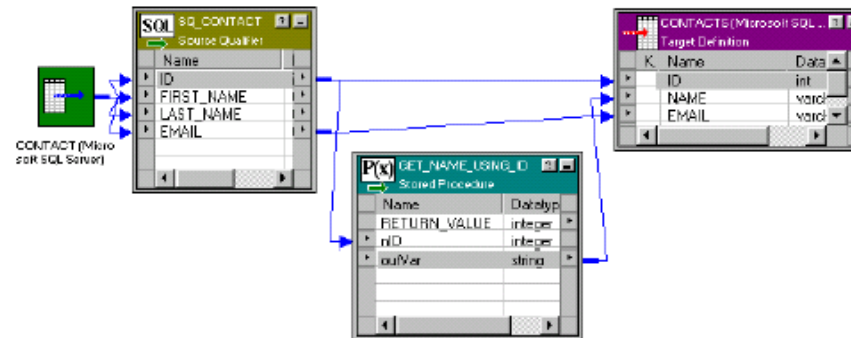
- A Connected Stored Procedure transformation is linked with the Input port from the preceding transformation and the Output port can be linked ahead.
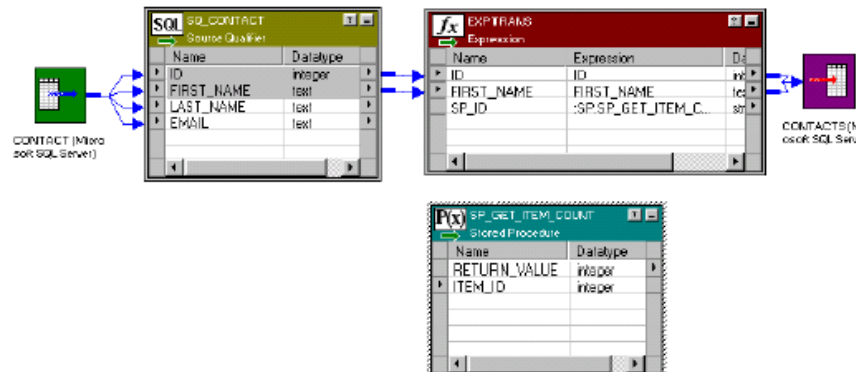
- A Unconnected Stored Procedure transformation can be called by passing the parameters of Input through an expression in the following format,

  :SP.<Proc_Name>(Input, Output)

Stored Procedure T/F – Guidelines

- Can set up the Stored Procedure transformation in one of two modes, either connected or unconnected.

- The flow of data through a mapping in connected mode also passes through the Stored Procedure transformation.

- Cannot run the same instance of a Stored Procedure transformation in both connected and unconnected mode in a mapping. You must create different instances of the transformation.

- The unconnected Stored Procedure transformation is not connected directly to the flow of the mapping.

- It either runs before or after the session or is called by an expression in another transformation in the mapping.

- Informatica supports following databases for Stored Procedure Transformation:
  - Oracle
  - Informix
  - Sybase SQL Server
  - Microsoft SQL Server
  - IBM DB2
  - Teradata

Rank Transformation

Rank Transformation

Filters the top or bottom range of records

Active Transformation
Connected

Ports
- Mixed
- One pre-defined output port RANKINDEX
- Variables allowed
- Group By allowed
- RANK Port is used to column that needs to be designated as rank

Properties

- Select top/bottom

- Number of records

- Cache Directory

- Case Sensitive

Edit Transformations

Transformation | Ports | Properties | Metadata Extensions

Select transformation: RNK_SAL_BY_DEPT

Transformation type: Rank

Transformation Attribute	Value
Cache Directory	$PMCacheDir
Top/Bottom	Top
Number of Ranks	10
Case Sensitive String Comparison	☑
Tracing Level	Normal
Rank Data Cache Size	Auto
Rank Index Cache Size	Auto
Transformation Scope	All Input

Cache Directory

Directory for cached rank values

OK | Cancel | Apply | Help

HCL

Demo and Q&A

- Demo
  - Simple mappings consists of the following transformations
    - Aggregator
    - Sorter
    - Sequence Generator
    - Stored Procedure
    - Rank

Thank You