

INFORMATICA Power CENTRE

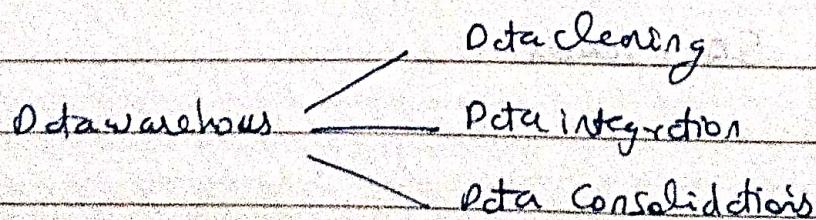
ETL Tools

- 1) Ab Initio
- 2) Informatica PowerCenter
- 3) Oracle Warehouse Builder (OWB)
- 4) SAP Data Services
- 5) IBM InfoSphere Information Server
- 6) SAS Data Management
- 7) Elixir Repertoire for Data ETL
- 8) Data Migration Studio
- 9) SQL Server Integration Services (SSIS)

Informatica Data Explorer → data discovery and profiling capability

Data warehouse

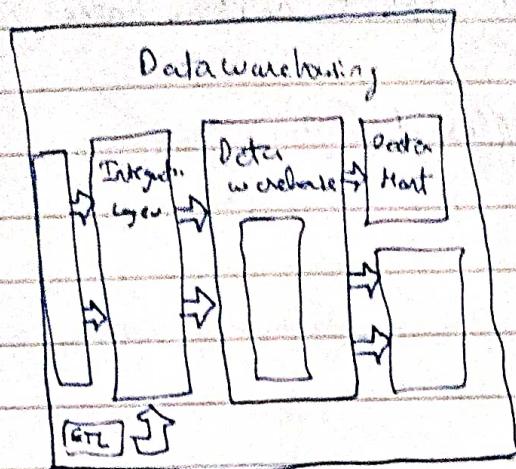
→ Data warehouse is constructed by integrating data from multiple heterogeneous sources



ETL: EXTRACT → TAKE SOURCE

TRANSFORM → APPLY LOGIC

LOAD → TAKE TARGET



flat file → txt file

Company XYZ Organization

Table: Employee, Department, Product, Finance.

① Employee.

EmployeeID	EmployeeName	EmployeeSalary
------------	--------------	----------------

② Department.

DeptID	DeptName
--------	----------

③ Product

ProductCode	ProductName	AnnualGross
-------------	-------------	-------------

④ Finance.

AccountID	AccountName	Department
-----------	-------------	------------

Datawarehouse

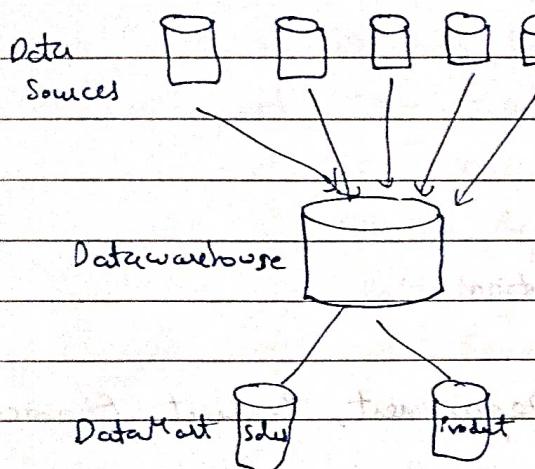
Oracle Database

Integration Layer

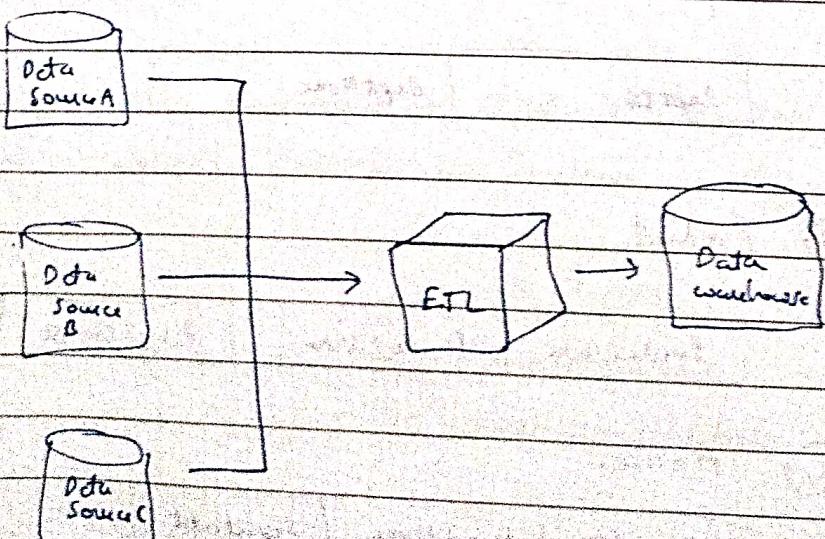
CSV and Excel files are the same

Both are different.

Data Mart



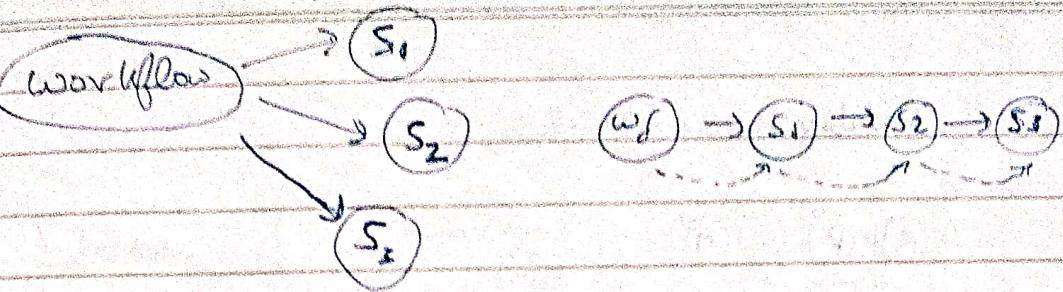
ETL Diagram



Powercenter Repository:

PAGE NO.

DATE / /



Task runs parallelly or Sequentially.

Powercenter Repository:

→ Metadata

Server:

- 1) Repository Services
- 2) Integration Services
- 3) Domain
- 4) Node

Client:

- 1) Repository
- 2) Designer
- 3) Workflow
- 4) Monitor

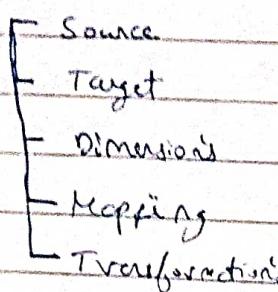
Informatica Installation

- 1) Java JRE
- 2) Oracle Server
- 3) Oracle Client
- 4) Informatica Client
- 5) Informatica Server

Database Type: Oracle

(R) (D) (W) (M)

Repository → MetaData



→ Everything we do in Designer Store is in Repository

Repository → Folder To Store all the data Created in Designer

Folder → Create folder.

(R) → (D) → (W) → (M)

List of Transformation

- 1) Expression Transformation
- 2) Router Transformation
- 3) Filter Transformation
- 4) Aggregator Transformation
- 5) Joiner Transformation
- 6) Rank Transformation & Sequence Generator Transformation
- 7) Lookup Transformation

Active Transformation: Aggregator, Filter, Joiner, Rank, Router, Sorter, Union, Update Strategy, Now

Passive Transformation: Expression, Sequence Generator, Stored Procedure, External Procedure, XML Parser, XML Generator and Source Qualifier, Lookup TX.

$f(x)$

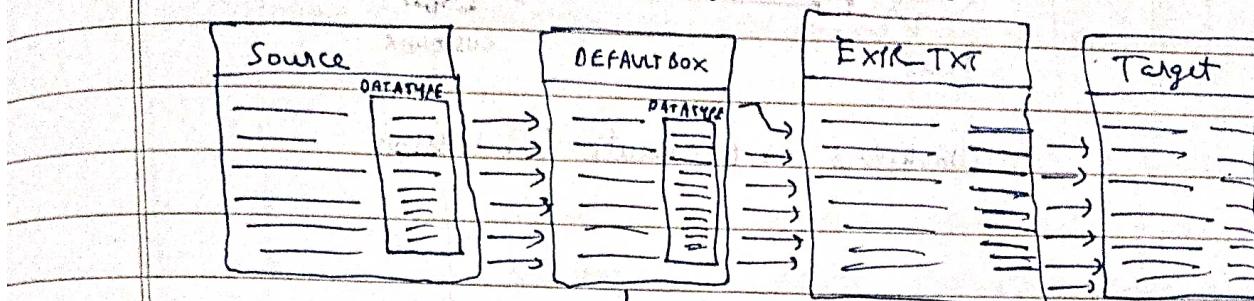
PAGE NO.

DATE / /

Expression Transformation: \rightarrow Passive Transformation

Designer

Mapping (Source to Target)

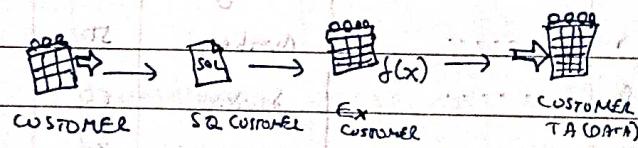


Default Box Converts

one datatype to another

(SQL datatype to Informatica datatype)

Right Click \rightarrow Arrange all icons



CUSTOMER TABLE

CUSTOMER_ID

CUST_FULL_NAME

CUST_LAST_NAME

CUST_PHONE

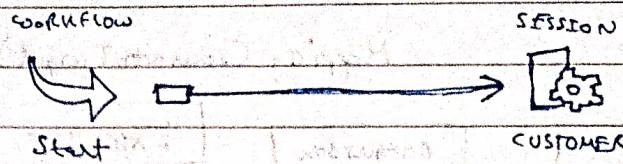
CUST_EMAIL_ID

ORDER_ID

Connect Stream: xe

To find?

SELECT * FROM SYS.GLOBAL_NAME;



Workflow → to execute the session

CSV (Comma delimited) File

Character and Data Type

Structure of CSV file

	Column Name	datatype	Prec	Scale	Not	Format
1	CUST....	Number	38	0		
2	String	30	0		
3	String	30	0		
4	Number	50	0		
5	String	50	0		
6	Number	38	0		
7	String	50	0		

Precision (Prec) → Length of the word ("Informatica") 11 (character)
 Scale →

(Ex: 12345.67)

Precision (Prec) → Total Digits (5) → 12345

Scale → Digits after Decimal(2) → 67

Source file :

CUSTOMER-ID	CUST-FIRST-NAME	CUST-LAST-NAME	CUST-PH	CUST-EMAIL	ORDER-ID
1	POOJA	BORDE	78451443	abc@gmail	123
2	ASHWINI	CHAUHAN	9245859	abc@gmail	124
3	STEVEN	KING	563748	abc@gmail	125
4	DAVID	AUSTIN	3256378	abc@gmail	125
5	AMAR	SINGH	64252123	abc@gmail	168

Target file :

CUSTOMER-ID	CUST-FULL-NAME	CUST-PH-NO	CUST-EMAIL-ID	ORDER-ID
1	POOJA BORDE	78451443	abc@gmail	123
2	ASHWINI CHAUHAN	9245859	abc@gmail	124
3	STEVEN KING	5637469	abc@gmail	125
4	DAVID AUSTIN	3256378	abc@gmail	125
5	AMAR SINGH	642523654	abc@gmail	168

→ Expression f(x) Transformation to combine two names.

To Concat both the columns write function

CONCAT (CUST-FIRST-NAME, CUST-LAST-NAME) → without space

CONCAT (CUST-FIRST-NAME, CONCAT (' ', CUST-LAST-NAME)) → with space

CUST-FIRST-NAME || '-' || CUST-LAST-NAME → with hyphen

CUST-FIRST-NAME || ' ' || CUST-LAST-NAME → with space

Source → Source Qualifier → Expression (ff(x)) → Target Table

To find or insert the file →

Source file:



Session

→ Double click Session → Go to Mapping → Properties → Go to

Source file directory

Target file:



→ Double click session → Go to Mapping → Properties → Go to

Reject file directory

Grattt chart

Task details

Source/Target statistics

Partition details

Performance

□ → If the job has been indicated by Green color then it is successful

□ → If the Job has been indicated by Orange this it is failed



→ Mapping → Properties → Truncate previous Table

Session

(or)

ORACLE SQL → TRUNCATE TABLE CUSTOMER DETAILS;

-- → Comment in Informatica

I O V

V-CUST-FULL-NAME

□ □ □

V-CUST-FULL-NAME

→ There is no Input or Output port for Variable port.

V-CUST-FULL-NAME → CONCAT(CONCAT(CUST-FIRST-NAME, ','),
CUST-LAST-NAME)

call the Variable Port

Router Transformation

1. we can specify multiple filter condition

Active Transformation

Filter Transformation

1. we can specify only one filter condition

Active Transformation

CUST-NAME

Amount

NoOfOrders

Tanish

45000

5

Harpreeet

20000

2

Viray

20000

1

Ashay

5000

4

Avinash

2000

7

Teet

80000

15

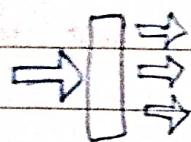


Router Transformation

Active Transformation

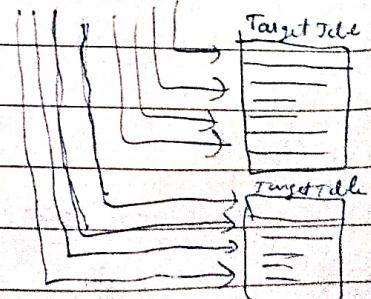
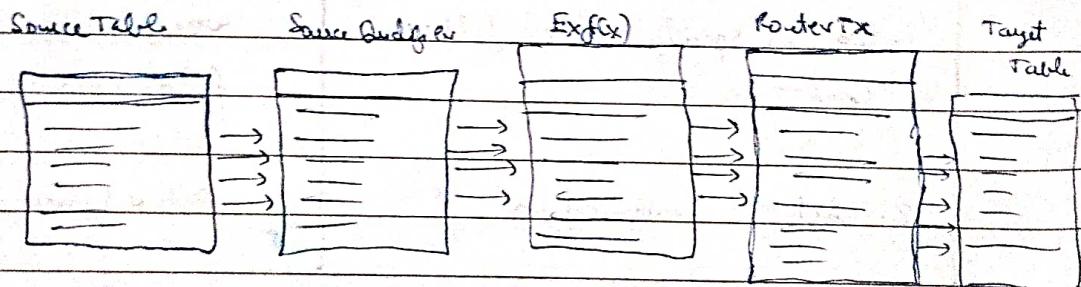
Using Extract transformation we can achieve 2 results

1. Customers who have given more than 5 orders
2. Customers who have spent less than 10,000.
3. Default group: which fetches the data which do not satisfy any of the above filter condition



* Single Input Multiple Output

* Single Source file Multiple Target file



Filter Transformation

→ Filter Transformation is an active transformation as it changes the no of records.

Active Transformation → Changing no of rows and records

Sources

<u>cust_name</u>	<u>Amount</u>	<u>No of orders</u>
Jasli	45000	5
Harpreeet	70000	2
Vinay	20000	1
Abhay	5000	4
Avinash	2000	7
Teet	80000	15

→ Customer Name who has spent more than 40,000....

Target:

<u>Cust_name</u>	<u>Amount</u>	<u>No of orders</u>
Indi	45000	5
Harpreeet	70,000	2
Teet	80,000	15

Rowset Transformation

- 1) We can specify multiple filter condition
- 2) Active Transformation

Filter Transformation

- 1) We can specify only one filter condition
- 2) Active Transformation

- 1) Employee Salary should be less than or equal to 10,000 and Department ID should be 10

Formula: $\text{EMP_SALARY} \leq 10000 \text{ AND } \text{DEPT_ID} = 10$

Rowset Transformation

- (*) Single input \rightarrow Multioutput

Filter transformation

- (*) Single input \rightarrow Single Output

- (*) Specify more than one filter condition

- (*) Specify only one filter condition

\rightarrow Rowset Transformation is more efficient than the filter transformation

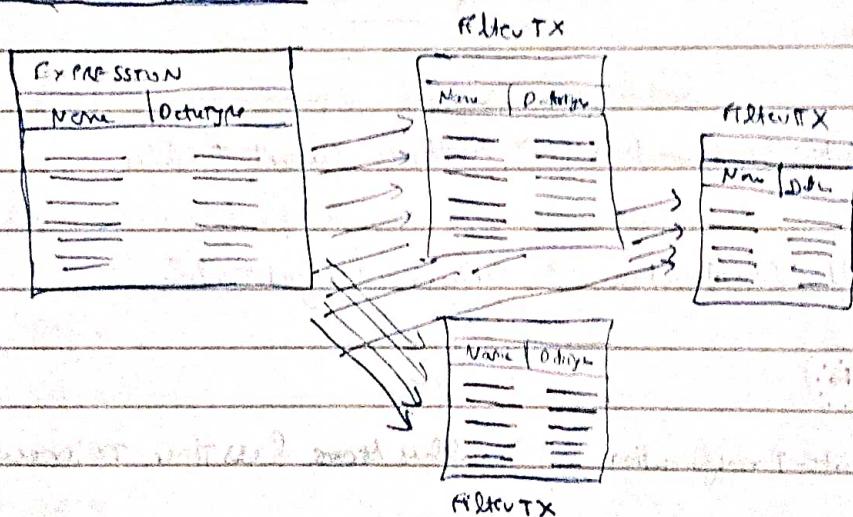
Concatenation:

`CONCAT(EMP-FIRST-NAME, ','), EMP-LAST-NAME)`

variable cont has no Input and output.

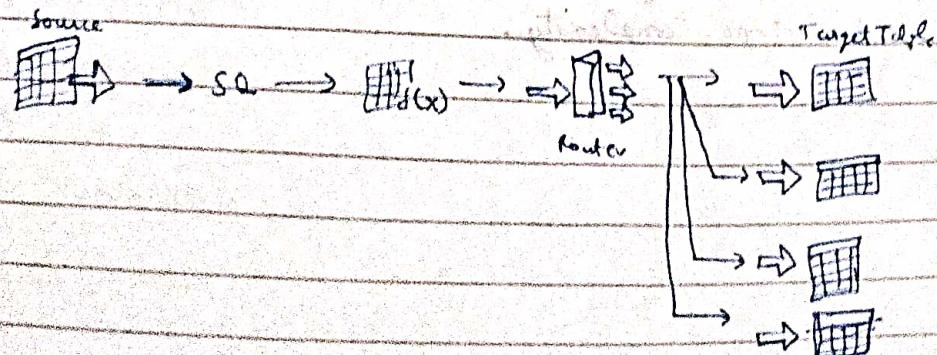
I O V
 V-EMP-FULL-NAME □ □ □' V-PM-FULL-NAME[2]

Router Transformation:



Router Transformation vs Alter Transformation

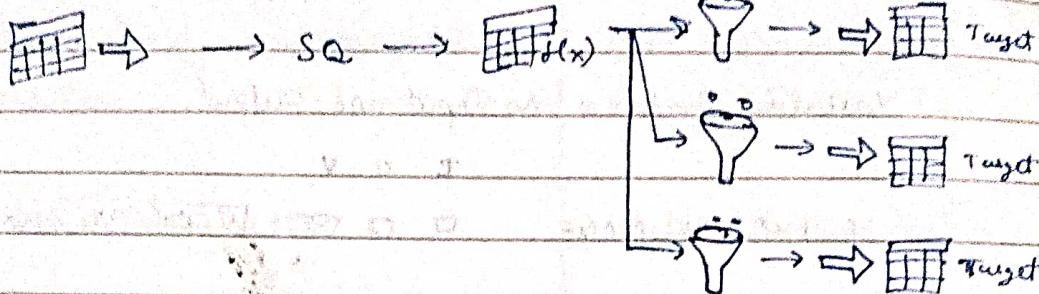
RouterTX



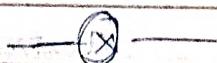
Single Input - Multi Output

Filter Transformation

Source Table



Single Input Single Output.



Router Transformation → Multiple Target Tables

Filter Transformation → Only One Target Table



Router Transformation → Takes less time to execute

Filter Transformation → Takes more time to execute

When lots of Data come's in Router transformation is best for Time Complexity.

Aggregator Transformation



AGGREGATOR TX → Active Transformation

→ Used to perform calc like sum, average etc.

V-OUT-PORT → ERROR

Aggregator Transformation → Does not allow variable port

Agg TX → Group by checkbox

Function used in Aggregator transformation

- > AVG
- > COUNT
- > FIRST
- > LAST
- > MAX
- > MEDIAN
- > MIN
- > PERCENTILE
- > STDEV
- > SUM
- > VARIANCE.

Session → Mapping → Properties

Source file type: Direct

Source file Directory: M:\2010\Information and Oracle\Source files

Source file Name: org-employee-data.csv

Source file type: Indirect [V]

Source file directory: M:\2019\Informatica And Oracle\Source files

Source file name: ORCL-EMPLOYEE DATA.txt

→ Inside txt file path is there. [Indirect]

Aggregator TX → Properties → Tracing level → Normal

Aggregator TX → Properties → Tracing level → verbose data.

Tunitory → Right click → Show session log.

Joiner Transformation

Joiner Transformation → Active Transformation

Joins can be created for heterogeneous system.

Type of Joins

Informatica:

- 1) Master Outer Join
- 2) Detail Outer Join
- 3) Full Outer Join
- 4) Normal Join

Database:

- 1) Inner Join
- 2) Left Outer Join
- 3) Right Outer Join
- 4) Full Outer Join

Informatica: First creating table then loading into SQL

→ : Open and create a table and then click on

→ Target → Generate / Execute SQL

→ It will generate a SQL Query based on Table.

Normal:

ORACLE SQL → → Table (Informatica)

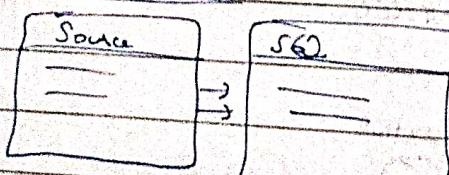
Advanced:

(Informatica) Table → → oracleSQL

Joiner TX:



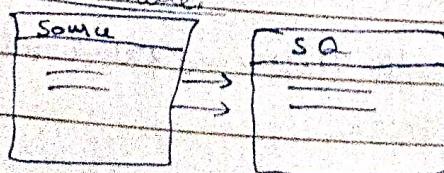
MaterSource:



Master - JOIN

Mater
Mater
Detail
Detail

Retail Source:



SOURCE

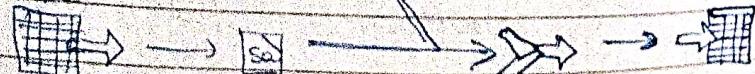


SQL

Joiner



TARGET

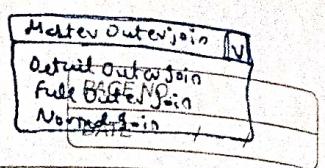


SQL



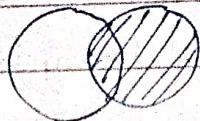
JOINER

TX Double Click \rightarrow Properties \rightarrow Join Type \rightarrow



Master Detail

1) MASTER OUTER JOIN



- * It will give all records from detail source and only matching records from master source \rightarrow Right Outer Join

Master outer join is equivalent to right outer join

Source Table: Target Table: Transformation:

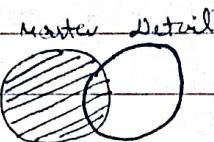
Subject

STUDENT_REC

Joiner Transformation

Student

2) Detail Outer Join



- * All the records from the Master table and only matching records from the Detail Table \rightarrow Left Outer Join

Source Table:

Subjects

STUDENT_REC

Joiner Transformation

Students

3) Full Outer Join



- * All the records from both Master and Detail tables are returned \rightarrow Full Outer Join

Full outer join is equivalent to full outer join

Source Table:

Subjects

Target Table:

STUDENT_REC

Transformation

Students

Joiner Transformation

Master Detail



4) NORMAL JOIN

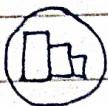
- ④ It gives only the matching records from both MASTER and DETAIL Table \rightarrow INNER JOIN

Normal join is equivalent to Inner join

Source Table:
Subjects
Students

Target Table:
STUDENT REC

Transformation:
Joiner transformation

Rank Transformation

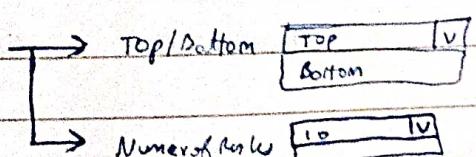
- ⑤ Rank Transformation \rightarrow Active Transformation

- ⑥ Select Top or Bottom rank of Data

- ⑦ Select Smallest or Largest Numeric/String Value

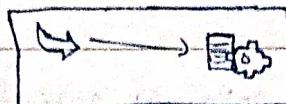
- ⑧ Integration Service \rightarrow  \rightarrow Input Data \rightarrow Rank Calculation

RankTX (Double click) \rightarrow Properties



④ If we do any change in Design

Existing Mapping → Refresh → Mapping



Rank Transformation	
RANKINDEX	→ Default
Employee-ID	
Department-ID	
Salary	
Rank	

Informatica Rank Transformation → the RankIndex Column is automatically generated

RankIndex: →

when multiple rows have the same rank, Informatica needs a way to uniquely identify and order them.

RankIndex Column → assigns a unique number to each row.

RankIndex	Employee-ID	Dept-ID	Salary	Rank
1	4	30	50000	1
2	1	10	25000	2
3	2	10	15000	3

RANK TRANSFORMATION

Source:

Id, Salary

1, 2000

2, 20

3, 50

4, 5

5, 250

Source:

EMPLOYEE DATA

Transformation:

RANK TRANSFORMATION (Bottom 2 employees who has min salary)

Target:

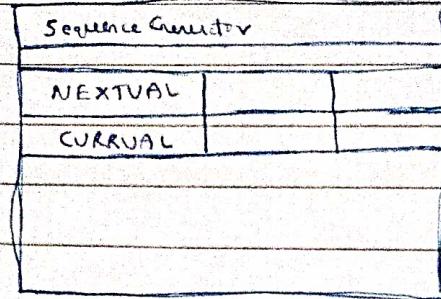
EMPLOYEE_RANK

 (X)

(X)

Sequence Transformation

1 z^3 Sequence Generator Transformation \rightarrow passive transformationActive Transformation \rightarrow Number of Input is not same as No of Output.
Rows (6 \neq 5) \neq RowsPassive Transformation \rightarrow Number of Input is equal to No of Output.
Rows (6 = 6) = RowsResultant value : NEXTVAL +
CURRENT



Sequence Generator → It is used to generate a unique number for each of the row.

Ex:

SNo	Emp_ID	DEPT_ID	SALARY
1	1	10	25000
2	2	10	15000
3	3	20	2000
4	4	30	50000
5	5	30	1000



Lookup Transformation



(*) Lookup Transformation → passive transformation

(*) It used to lookup a Source, Source Andifier or Target to get the relevant data.

(*) It is a kind of Join operation in which one of the joining tables is the source data and the other joining table is the lookup table.

LOOKUP TRANSFORMATION:

Employee : (SourceTable)

EMP_ID

EMPLOYEE_NAME

DEPARTMENT_ID

Departments (LookupTable)

DEPARTMENT_ID

DEPARTMENT_NAME

Employee_Dept: (TargetTable)

EMP_ID

EMPLOYEE_NAME

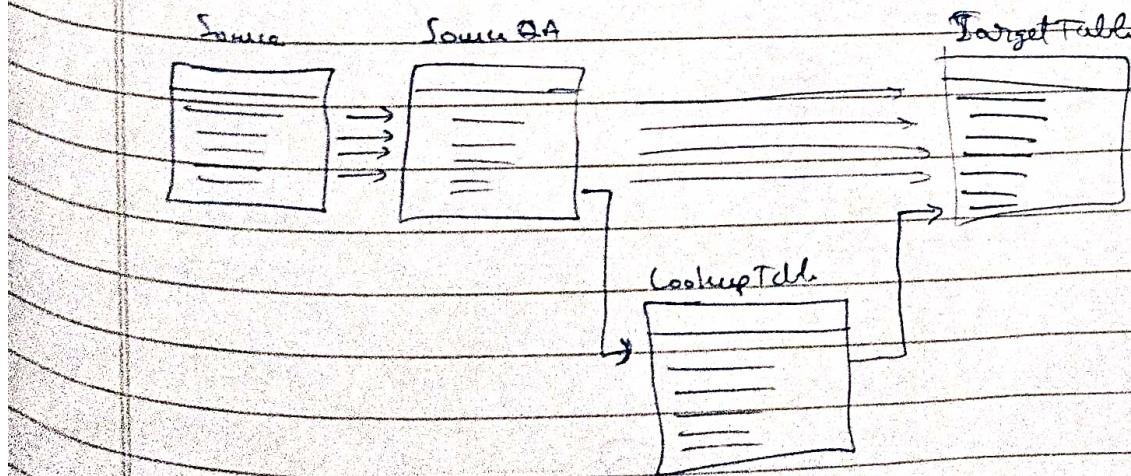
DEPARTMENT_ID

DEPARTMENT_NAME

Types of Lookup

- Relational or Flatfile
- Catched or uncatched
- Connected or Unconnected

LookupTable → Flatfile Lookup



Lookup table \rightarrow Connected lookup

Lookup table \rightarrow Unconnected lookup

$$\text{Department_m} \quad = \quad n - \text{Department_pp}$$

lookup \rightarrow Search particular record is there or not

Informatica Powercenter Fundamentals

Union Transformation →  after transformation

① The Union Transformation is used to extract data from multiple sources like flat files, sql tables or produce one output file in the target file.

② Union Transformation allows multiple input groups but produce single output.

① Transformation Tab

② Properties tab

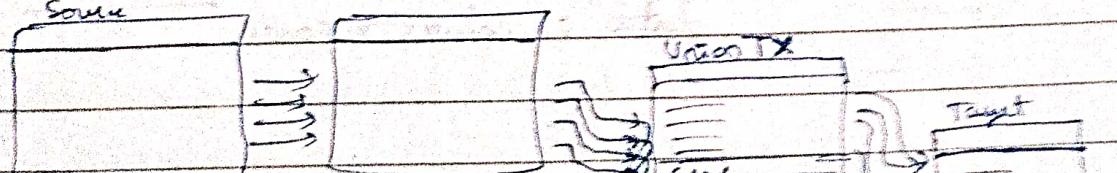
③ Group tab

④ Group parts tab

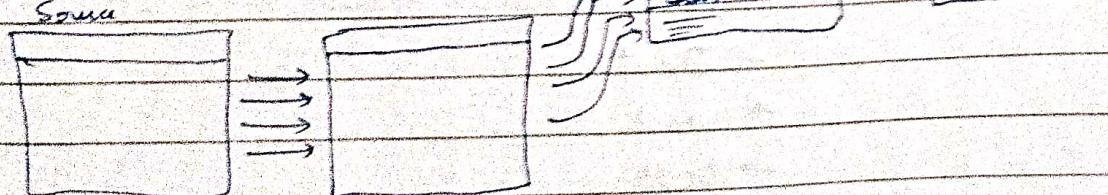
Tab → Transformation, Ports, Properties, Groups, Group parts

Union TX

Source



Source



Union TX

Source Table

1
2
3
4
5

Source Table

1
2
3
4
5
6
7
8
9
10

Union TX

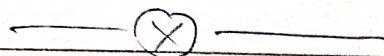
UNION

Target Table

1
2
3
4
5
6
7
8
9
10

⊕ Union Transformation is too Simple To Understand

✗ Combine Two Tables and Show in a Single Table



Normalized Transformation.



⊕ Normalization TX → Active Transformation

⊕ Normalizer can be used to Transpose the data in columns to rows

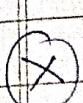
$$AT = BT$$

Properties:

- Input
- Output
- GCD
- GK_VALUE

Example:

Using Normal-TX



Name	Store1	Store2	Store3	Store4	Store5
------	--------	--------	--------	--------	--------

None	Puma	Puma	Puma	Puma	Puma
store	1	2	3	4	5



Name	Store	Value
Puma	1	457000
Puma	2	4500
Puma	3	12000
Puma	4	785420
Puma	5	60000

GCID and GK Value :-

ID	Name	Store	Value
1	Puma	1	
2	Puma	2	GCID
3	Puma	3	
4	Puma	4	
5	Nick	1	
6	Nick	2	GCID
7	Nick	3	
8	Nick	4	

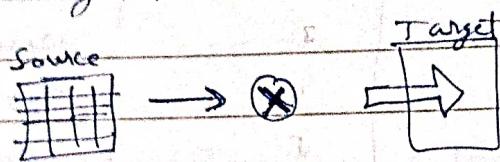
GK-Value

Versioning → checkin / checkout

Strategy Transformation



- ④ Update Strategy transformation is used to insert, update and delete records in the target table.
- ⑤ It can also reject the records without records in the target table.



Stored Procedure Transformation

- ④ It is used to execute stored procedures through Informatica ETL.

- ④ The stored procedure that are to be executed to be pre-built on the database which can be connected through Informatica.

- ④ The stored procedure must exist in the database before creating a stored procedure transformation.

Connected Stored procedure Transformation: The stored procedure transformation is connected to other transformations.

Unconnected stored procedure Transformation: The stored procedure transformation is not connected to other transformations.

- Normal
- Pre-load of the base
- Post load of the source
- Pre load of the target
- Post load of the target

Source Table:

Emp ID	Salary	Commission
6		

Stored Procedure:

Add Salary and commission and this result will be in
column 'TOTAL'

TRANSFORMATION:

Normal Procedure Transformation

Target Table:

Emp ID	Salary	Commission	Total
6			

How to take stored procedure!

Transformation → Import stored Procedure



SCD → Slowly Changing Dimensions

(X)

→ Data changes slowly, rather than changing regularly on a time basis

→ In Dd warehouse there is a need to track changes in dimension attributes in order to report historical data

Source:

DEPT-ID	DEPT-NAME	DEPT-LOC
1	FINANCE	INDIA
2	MARKETING	UK
3	RETAIL	UK
4	BFS	US
5	BFS	INDIA

REQUIREMENT:

- 1) As the data comes insert the record
- 2) As the data comes, update the record
- 3) Update the record directly, or mark the previous record as inactive and create a new record for update

Target: SCD type 1

DEPT-ID	DEPT-NAME	DEPT-LOC
1	FINANCE	INDIA
2	MARKETING	UK
3	RETAIL	UK
4	BFS	US
5	BFS	INDIA

DEPT-ID	DEPT-NAME	DEPT-LOC	VERSION
1	FINANCE	INDIA	1
2	MARKETING	UK	1
3	RETAIL	US	1
4	BFS	US	1
5	BFS	INDIA	1
3	RETAIL	UK	2

Types of SCD

1) SCD Type 1

2) SCD Type 2

3) SCD Type 3

Formulasif-clue:

$\text{IF}((\text{DEPT-ID} = \text{LKP-DEPT-ID}) \text{ AND } (\text{DEPT-NAME} \neq \text{LKP-DEPT-NAME}) \text{ OR } (\text{DEPT-LOC} \neq \text{LKP-DEPT-LOC}) \text{ OR } (\text{DEPT-HEAD} \neq \text{LKP-DEPT-HEAD}), \text{TRUE}, \text{FALSE})$.

$\text{IF}(\text{ISNULL}(\text{LKP-DEPT-ID}), \text{TRUE}, \text{FALSE})$

SCD Type 2

1) Versioning

2) Flagging

Flagging : SCD type 2

SN.	DEPT_ID	DEPT_NAME	DEPT_LOC	FLAG
1	1	FINANCE	INDIA	0
2	2	MARKETING	UK	1
3	3	RETAIL	INDIA	1
4	4	BFS	USA	1
5	5	BFS	INDIA	1
6	6	RETAIL	UK	1
7	1	FINANCE	UK	1

④ For older records it turns ~~to~~ to 0. (Flag → 0 only)

SCD: type 3

⑤ Only the current status and previous status of the row is maintained in the table.

⑥ type 3 Method will have limited history.

~~Some~~ ⑦ historical data is stored as it is we can see previous history as well.

Source:

EMP-ID	EMP-NAME	EMP-SALARY	EMP-OFC-BRANCH
1	MAYESH	450000	INDIA
2	MAYURI	78000	USA
3	RUCHITA	80000	UK
4	SNEHA	90000	UK

Target:

EMP-ID	EMP-NAME	EMP-SALARY	EMP-OFC-AREA	EMP-REL
1	MAYESH	450000	INDIA	UK
2	MAYURI	78000	USA	USA
3	RUCHITA	80000	UK	UK
4	SNEHA	90000	UK	UK

(*) History remains as it is.

SCD Type 1 → Overwrites Old data (No History is Maintained)

SCD Type 2 → Maintains full historical records (old data along with New data)

SCD Type 3 → keep limited history with additional Columns

