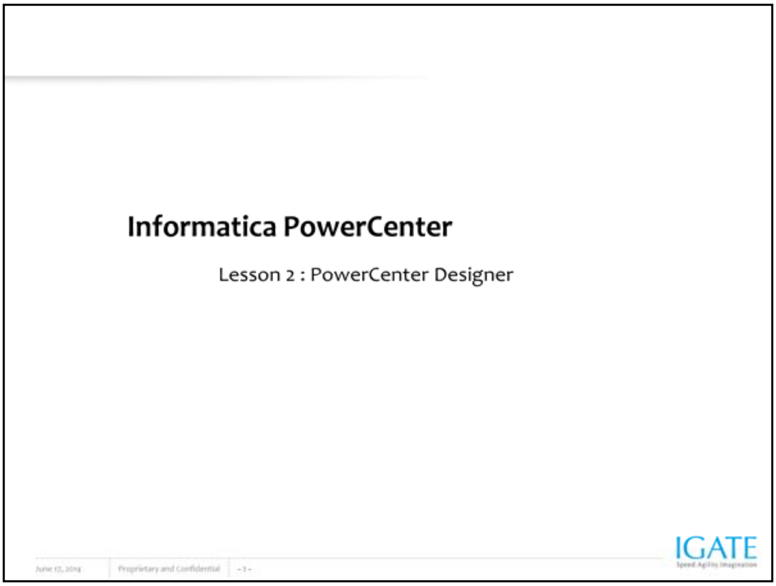
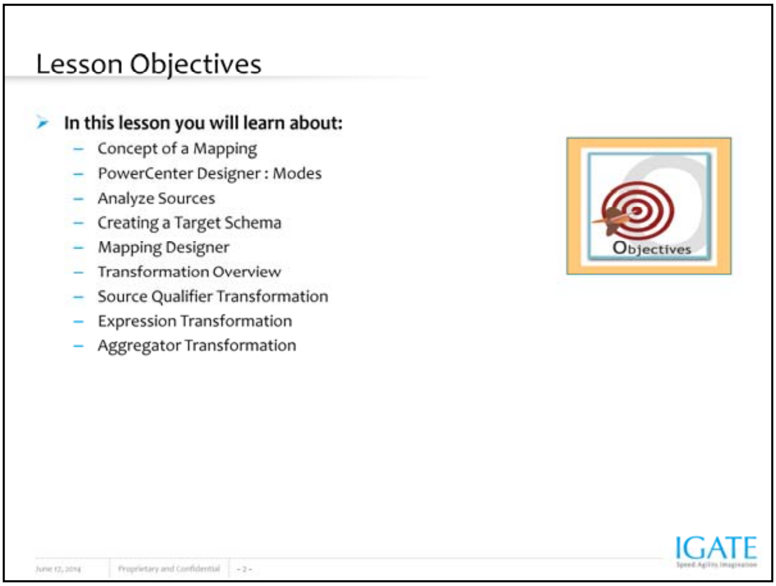


Informatica PowerCenter

Lesson 2 : PowerCenter Designer

June 17, 2014 | Proprietary and Confidential | - 1 -

IGATE
Speed Agility Imagination

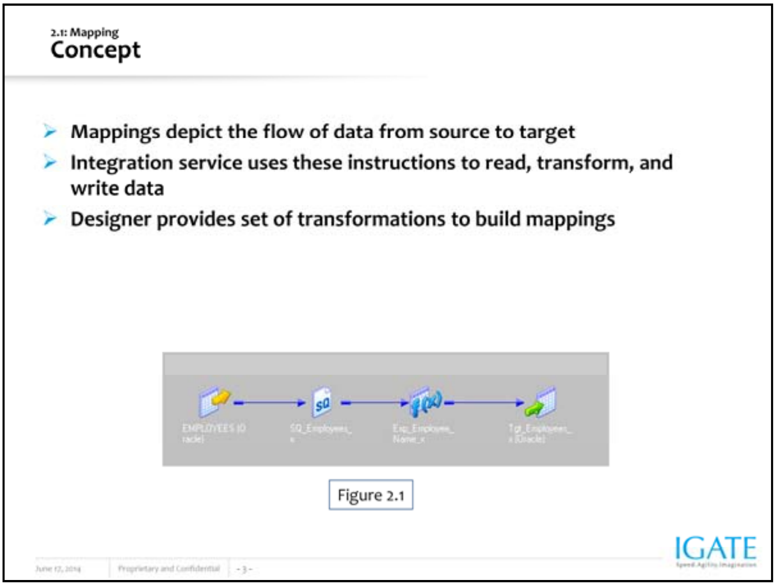


Lesson Objectives

- In this lesson you will learn about:
  - Concept of a Mapping
  - PowerCenter Designer : Modes
  - Analyze Sources
  - Creating a Target Schema
  - Mapping Designer
  - Transformation Overview
  - Source Qualifier Transformation
  - Expression Transformation
  - Aggregator Transformation

June 17, 2014 Proprietary and Confidential - 2 -

IGATE
Speed Agility Imagination



2.1: Mapping

Concept

- Mappings depict the flow of data from source to target
- Integration service uses these instructions to read, transform, and write data
- Designer provides set of transformations to build mappings

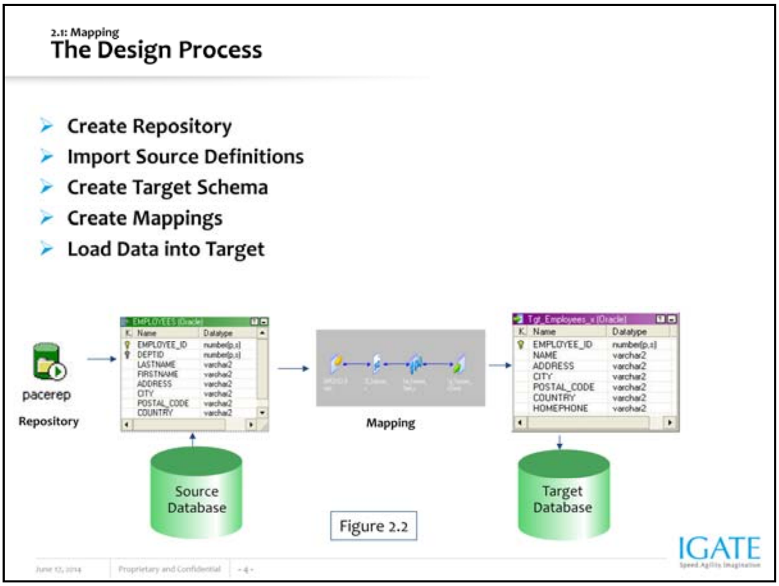
Figure 2.1

June 17, 2014 Proprietary and Confidential - 3 -

IGATE

Speed. Agility. Imagination

When the Integration service loads data, it uses the instructions configured, to read, transform, and write data.



2.1: Mapping

The Design Process

- Create Repository
- Import Source Definitions
- Create Target Schema
- Create Mappings
- Load Data into Target

EMPLOYEES (Oracle)

K	Name	Datatype
	EMPLOYEE_ID	number(p,s)
	DEPTID	number(p,s)
	LASTNAME	varchar2
	FIRSTNAME	varchar2
	ADDRESS	varchar2
	CITY	varchar2
	POSTAL_CODE	varchar2
	COUNTRY	varchar2

Tgt_Employees_x (Oracle)

K	Name	Datatype
	EMPLOYEE_ID	number(p,s)
	NAME	varchar2
	ADDRESS	varchar2
	CITY	varchar2
	POSTAL_CODE	varchar2
	COUNTRY	varchar2
	HOMEPHONE	varchar2

Figure 2.2

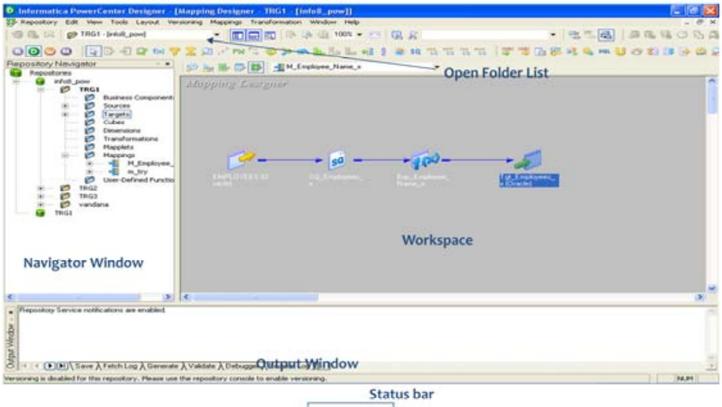
IGATE Speed. Agility. Imagination

June 17, 2014 | Proprietary and Confidential | - 4 -

- **Create Repository -** The first step is to create the Informatica repository from the Repository Server Administration Console. This repository will hold all related metadata and drive Informatica's extraction and transformation process. Once the repository and necessary folders have been created, the actual design process can begin. This is accomplished in the Designer client application, where developers will spend the majority of their time.
- **Import Source Definitions -** The next step will be to put the source definitions into the repository. The Source Analyzer within the Designer tool is used for this process
- **Create Target Schema -** After the source definitions are in the repository, the target schema should be created. This can be designed in the Target designer (also with the Designer), reverse engineered from the database, or imported through PowerPlugs
- **Create Mappings -** Mappings can be created to link the sources with the targets – the translation of 'business rules'
- **Load Data -** The final step is to load the data. The Workflow Manager is used to configure and schedule session Tasks and Workflows to run the mappings. Then, based upon all the information stored in the repository, the Integration service will extract, transform, and load the data

2.2: PowerCenter Designer Workspace

Concept



Repository Service notifications are enabled.

Versioning is disabled for this repository. Please use the repository console to enable versioning.

Figure 2.3

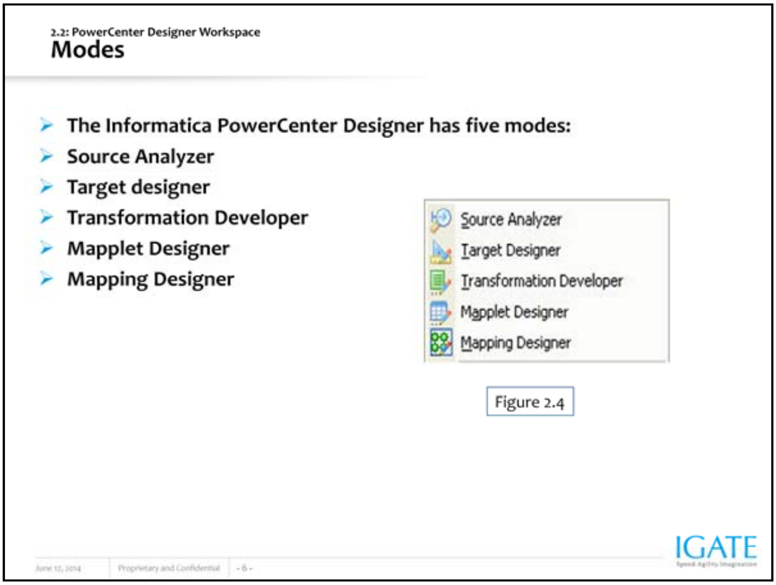
IGATE

June 17, 2014 Proprietary and Confidential - 5 -

The PowerCenter Designer has five tools which helps to build mappings and mapplets so that one can specify how to move and transform data between sources and targets. The Designer helps to create source definitions, target definitions, and transformations to build the mappings.

The Designer consists of the following :

- **Navigator window -** Use to connect to and work in multiple repositories and folders. Copy and delete objects options are available and shortcuts can be created using the Navigator
- **Workspace window-** Use to view or edit sources, targets, mapplets, transformations, and mappings. Work with a single tool at a time in the workspace. One can use the workspace in default or workbook format
- **Status bar -** Displays the status of the operation being performed
- **Output window-** Provides details while performing certain Tasks, such as saving the work or validating a mapping. Right-click the Output window to access window options, such as printing output text, saving text to file, and changing the font size



2.2: PowerCenter Designer Workspace

Modes

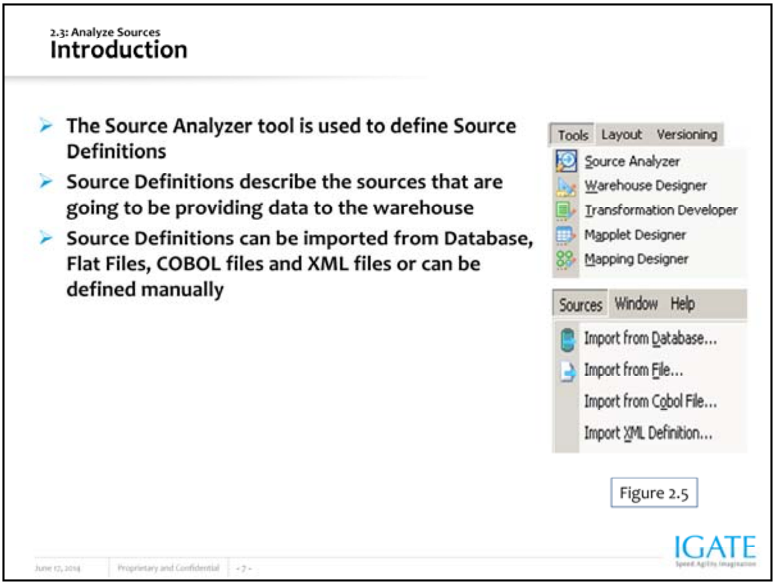
- The Informatica PowerCenter Designer has five modes:
- Source Analyzer
- Target designer
- Transformation Developer
- Mapplet Designer
- Mapping Designer

Figure 2.4

June 17, 2014 | Proprietary and Confidential | - 6 -

The Informatica Designer has five modes:

- The Source Analyzer is used to create or reverse-engineer source definitions
- The Target designer is used to create or reverse-engineer target schema
- The Transformation Developer is used to create reusable transformations
- The Mapplet Designer is used to create and edit mapplets. Mapplets are reusable objects that represent a set of transformations
- The Mapping Designer can be used to create the source to target mappings that contain the business rules for the server during the extract, transform, and load process



2.3: Analyze Sources

Introduction

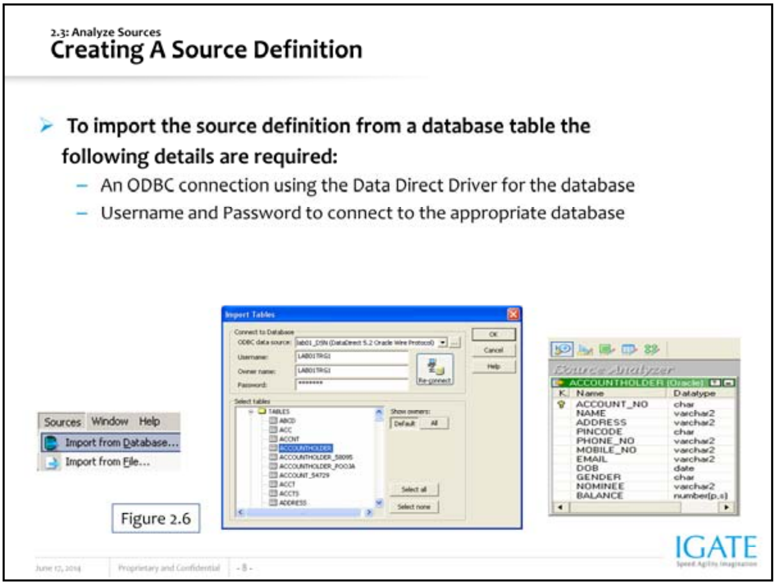
- The Source Analyzer tool is used to define Source Definitions
- Source Definitions describe the sources that are going to be providing data to the warehouse
- Source Definitions can be imported from Database, Flat Files, COBOL files and XML files or can be defined manually

Figure 2.5

June 17, 2014 | Proprietary and Confidential | - 7 -

The first tool, the Source Analyzer, is used to identify the sources that will be used to build the data mart or warehouse and create repository definitions for those sources. Source definitions describe the sources that are going to be providing data to the warehouse. The different ways to create source definitions in the Repository.

- Import from Database
- Import from File
- Import from COBOL File
- Import from XML…
- Create manually
- Import from SAP
- Import from PeopleSoft
- Import from Siebel



2.3: Analyze Sources

Creating A Source Definition

- To import the source definition from a database table the following details are required:
  - An ODBC connection using the Data Direct Driver for the database
  - Username and Password to connect to the appropriate database

Import Tables

Connect to Database

ODBC data source: lab01_DSN (DataDirect 5.2 Oracle Wire Protocol)

Username: LAB01TRG1

Owner name: LAB01TRG1

Password: ********

ACCOUNTHOLDER (Oracle)

K	Name	Datatype
🔑	ACCOUNT_NO	char
	NAME	varchar2
	ADDRESS	varchar2
	PINCODE	char
	PHONE_NO	varchar2
	MOBILE_NO	varchar2
	EMAIL	varchar2
	DOB	date
	GENDER	char
	NOMINEE	varchar2
	BALANCE	number(p,s)

Figure 2.6

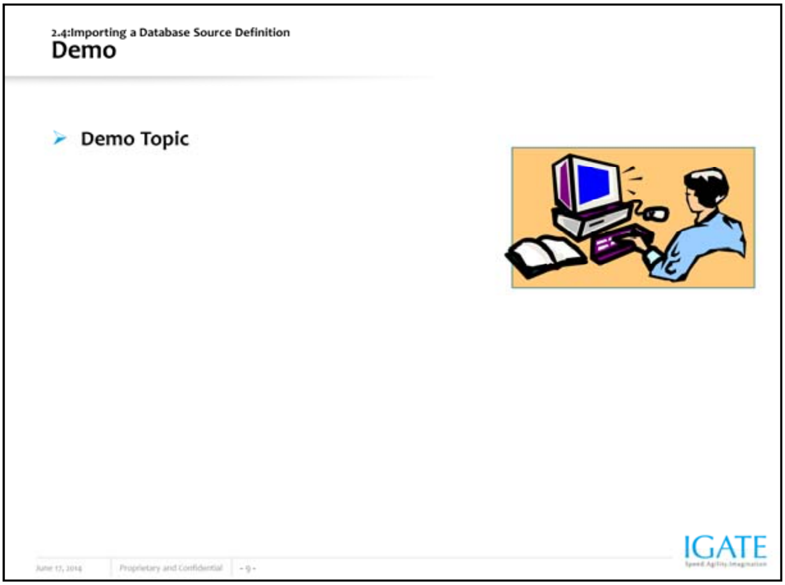
June 17, 2014 | Proprietary and Confidential | - 8 -

Relational source definitions can be imported from database tables, views, and synonyms. When importing a source definition, import the following source metadata:

- Source name
- Database location
- Column names
- Datatypes
- Key constraints

To import a source definition, first connect to the source database from the client machine using a properly configured ODBC data source or gateway. Read permission is required on the database object.

After importing a relational source definition, optionally enter business names for the table and columns. Also manually define key relationships, which can be logical relationships created in the repository that do not exist in the database.



2.4:Importing a Database Source Definition

Demo

- Demo Topic

June 17, 2014 | Proprietary and Confidential | - 9 -

IGATE

Speed. Agility. Imagination

Steps to import a source definition:

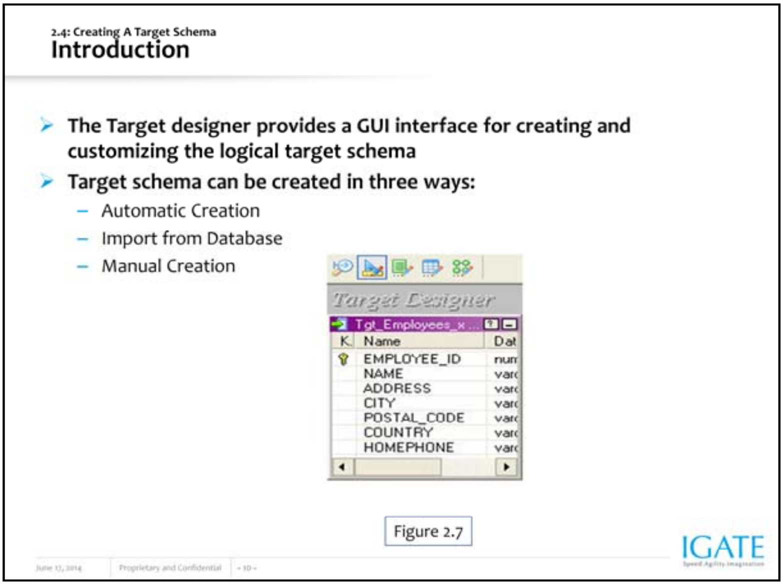
1. In the Source Analyzer, choose ***Sources* | *Import from Database***.
2. Select the ODBC data source used to connect to the source database. To create or modify an ODBC data source, click the ***Browse*** button to open the ODBC Administrator. Create the appropriate data source and click ***OK***. Select the new ODBC data source.
3. Enter a database username and password to connect to the database.

Note: The username must have the appropriate database permissions to view the object. The owner name for database objects will have to be specified in order to use them as sources.

4. Click ***Connect***.

If no table names appear or if the table to be imported does not appear, click ***All***.

5. Scroll down through the list of sources to find the source to be imported. Select the relational object or objects to be imported. Hold down the Shift key to select a block of sources within one folder, or hold down the Ctrl key to make non-consecutive selections within a folder.
6. Click ***OK***.



2.4: Creating A Target Schema

Introduction

- **The Target designer provides a GUI interface for creating and customizing the logical target schema**
- **Target schema can be created in three ways:**
  - Automatic Creation
  - Import from Database
  - Manual Creation

K.	Name	Dat
🔑	EMPLOYEE_ID	nurr
	NAME	varc
	ADDRESS	varc
	CITY	varc
	POSTAL_CODE	varc
	COUNTRY	varc
	HOMEPHONE	varc

Figure 2.7

June 12, 2014 | Proprietary and Confidential | - 10 -

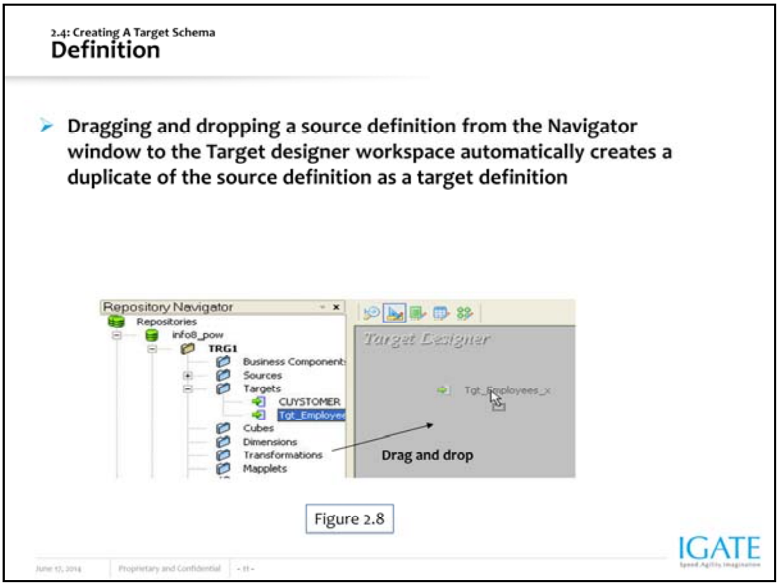
The next step in the design process is to create the target schema. This is accomplished in the second component of the Designer, the Target designer.

The Target designer provides a GUI interface for creating and customizing the logical target schema. Target definitions can be customized as needed.

When the customization is complete, one can create the physical targets in the database from SQL scripts executed from the Target designer.

The methods of creating target schema are:

- Automatic Creation
- Import from Database
- Manual Creation



2.4: Creating A Target Schema

Definition

- Dragging and dropping a source definition from the Navigator window to the Target designer workspace automatically creates a duplicate of the source definition as a target definition

Figure 2.8

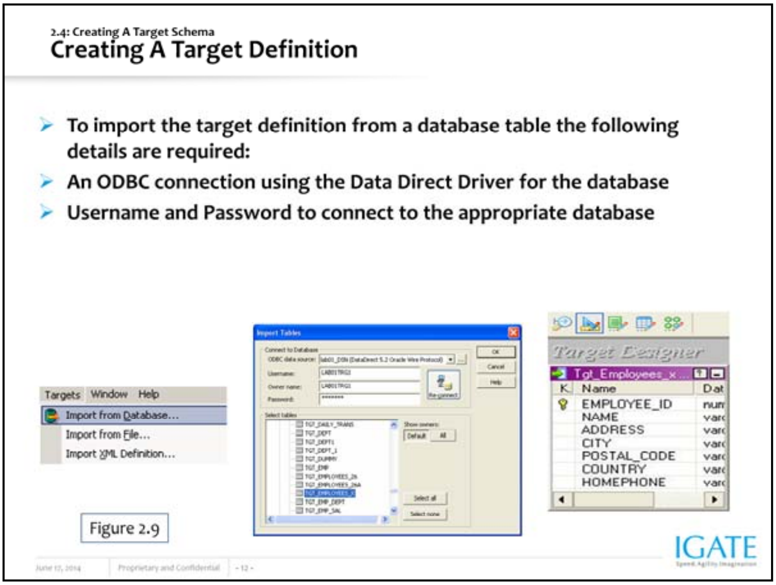
June 17, 2014 | Proprietary and Confidential | - 11 -

IGATE

Target definitions are stored in a separate section of the repository.

Automatic target creation is useful for creating:

- Staging tables, such as those needed in Dynamic Data Store
- Target tables that will mirror much of the source definition
- Target tables that will be used to migrate data from different databases, e.g., Sybase to Oracle



2.4: Creating A Target Schema

Creating A Target Definition

- To import the target definition from a database table the following details are required:
- An ODBC connection using the Data Direct Driver for the database
- Username and Password to connect to the appropriate database

Import Tables

Connect to Database

ODBC data source: lab01_DSN (DataDirect 5.2 Oracle Wire Protocol)

Username: LAB01TRG1

Owner name: LAB01TRG1

Password: ********

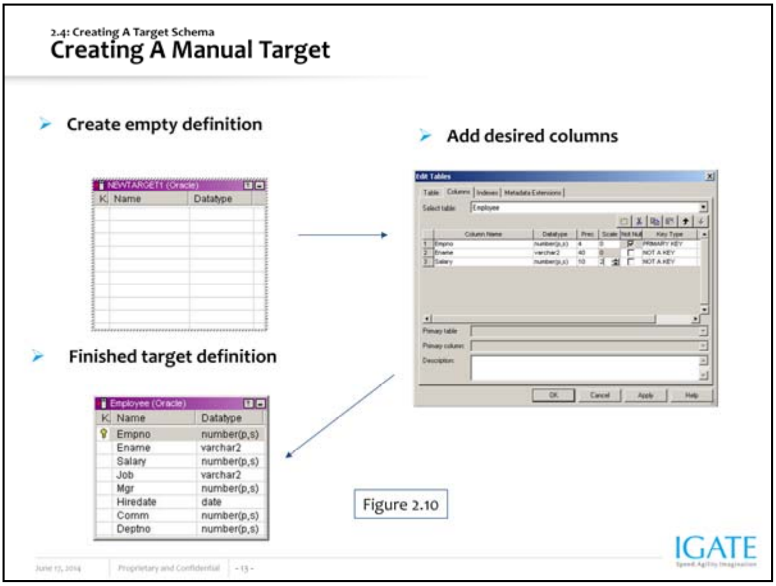
Select tables

Show owners: Default All

Figure 2.9

When a target definition is imported from a relational table, the Designer imports the following target details:

- **Target name -** The name of the target
- **Database location -** Specify the database location of a relational source is to be imported. Specify a different location while editing the target definition in the Target designer and while configuring a session
- **Column names -** The names of the columns
- **Data types -** The Designer imports the native datatype for each column
- **Key constraints -** The constraints in the target definition can be critical, since they may prevent data from moving into the target if the Integration service violates a constraint during a Workflow. For example, if a column contains the NOT NULL constraint and one fail to map data to this column, the Integration service cannot insert new records into the target table
- **Key Relationships -** The Target designer can be customized to automatically create primary-foreign key relationships. Choose ***Tools | Options*** and select the Format tab. Check Import Primary and Foreign Keys. Also logical relationships can be created in the repository. Key relationships do not have to exist in the database



2.4: Creating A Target Schema

Creating A Manual Target

- Create empty definition

NEWTARGET1 (Oracle)

K	Name	Datatype

- Add desired columns

Select table: Employee

	Column Name	Datatype	Prec	Scale	Not Null	Key Type
1	Empno	number(p,s)	4	0	☑	PRIMARY KEY
2	Ename	varchar2	40	0	□	NOT A KEY
3	Salary	number(p,s)	10	2	□	NOT A KEY

Primary table:

Primary column:

Description:

- Finished target definition

Employee (Oracle)

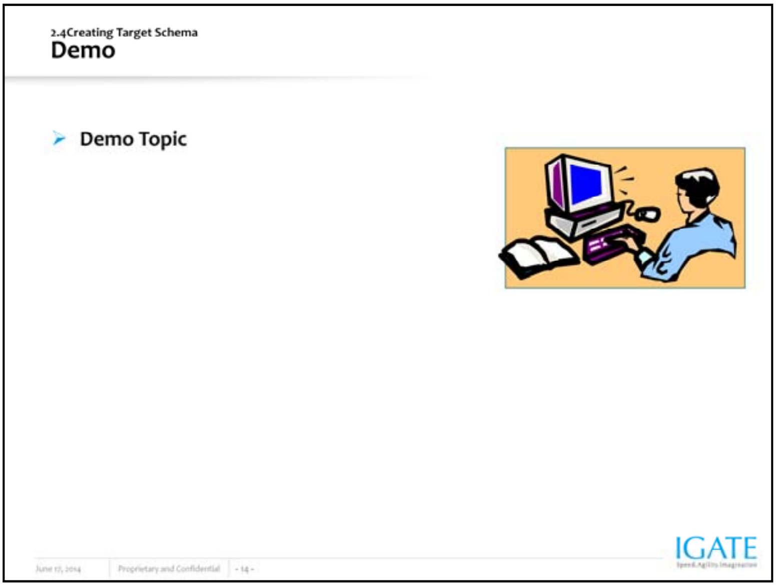
K	Name	Datatype
🔑	Empno	number(p,s)
	Ename	varchar2
	Salary	number(p,s)
	Job	varchar2
	Mgr	number(p,s)
	Hiredate	date
	Comm	number(p,s)
	Deptno	number(p,s)

Figure 2.10

June 17, 2014 | Proprietary and Confidential | - 13 -

Target Definitions can also be created manually via a menu option or the toolbar. An empty definition is created, you can enter the table name, columns, datatypes, precision, and key information.

Manual target creation is useful for creating tables whose columnar information cannot be copied from source definitions, such as time dimension tables or aggregate tables.



2.4Creating Target Schema

Demo

- Demo Topic

June 17, 2014 | Proprietary and Confidential | - 14 -

Steps to create a Target schema:

I Automatic Target Schema:

1. Drag a relational source definition into the Target designer workspace, the Designer will create a relational or flat file target definition that matches the source definition.

II From the Database:

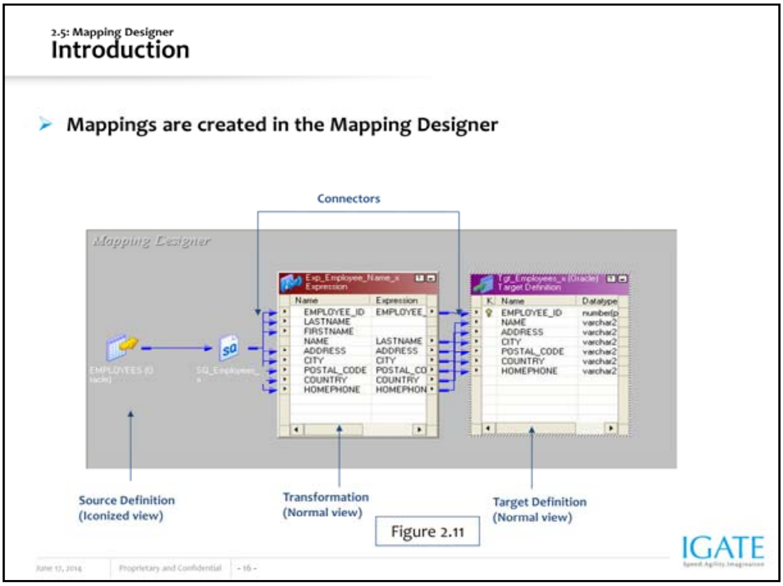
To import a relational target definition:

1. In the Target designer, choose ***Targets | Import from Database***.
2. Select the ODBC data source used to connect to the target database. To create or modify an ODBC data source first, click the ***Browse*** button to open the ODBC Administrator. After creating or modifying the ODBC source, continue with the following steps.
3. Enter the username and password needed to open a connection to the database, and click ***Connect***. If the user is not the owner of the table to be used as a target, specify the owner name.
4. Drill down through the list of database objects to view the available tables which can be used as targets.

5. Select the relational table or tables to import the definitions into the repository. Hold down the Shift key to select a block of tables, or hold down the Ctrl key to make non-contiguous selections. Also the ***Select All*** and ***Select None*** buttons can be used to select or clear all available targets.
6. Click ***OK***. The selected target definitions now appear in the Navigator under the Targets icon.
7. Choose ***Repository*** | ***Save***.

III Manual:

1. In the Target designer, choose ***Targets*** | ***Create***.
2. Enter a name for the target and select the target type. One can create a flat file target definition by choosing Flat File for the target type.
3. The target name entered is the name of the new table in the database when a relational target definition is created. Follow any database-specific naming conventions.
4. Click ***Create***.
5. An empty table structure appears in the workspace. (It may be covered by the dialog box.) The new target definition also appears within the Navigator window.
6. If another target definition is to be created, enter a new target name and target type and click ***Create***. Repeat this step for each target to be created.
7. Click ***Done*** when all target definitions are created.
8. Configure the target definition i.e., add the required columns.
9. Choose ***Repository*** | ***Save***.



2.5: Mapping Designer

Introduction

- Mappings are created in the Mapping Designer

Connectors
Mapping Designer
Source Definition (Iconized view)
Transformation (Normal view)
Target Definition (Normal view)

Figure 2.11

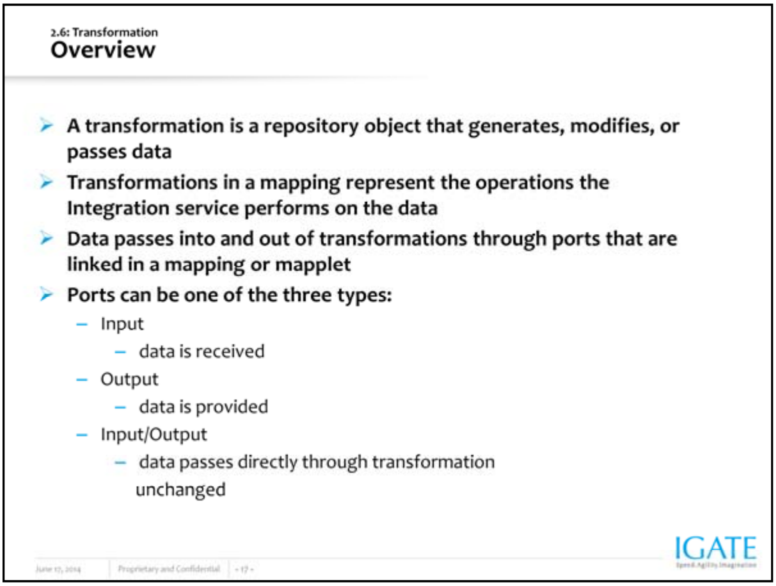
June 17, 2014 Proprietary and Confidential - 16 -

The above diagram shows how a mapping looks like. It has the various transformations which the source data has to go through. Every mapping must contain the following components:

- **Source definition -** Describes the characteristics of a source table or file
- **Transformation -** Modifies data before writing it to targets. Use different transformation objects to perform different functions
- **Target Definition -** Defines the target table or flat file
- **Connectors -** Connect sources, targets, and transformations so the Integration service can move the data as it transforms it

The Mapping Designer displays objects in three different views:

- **Iconized -** Shows an icon of the object with the object name
- **Normal -** Shows the columns in the ports tab and the input and output port indicators. One can connect objects that are in the normal view
- **Edit -** Shows the object properties. Switch between the different tabs and configure the object in this view



2.6: Transformation

Overview

- A transformation is a repository object that generates, modifies, or passes data
- Transformations in a mapping represent the operations the Integration service performs on the data
- Data passes into and out of transformations through ports that are linked in a mapping or mapplet
- Ports can be one of the three types:
  - Input
    - data is received
  - Output
    - data is provided
  - Input/Output
    - data passes directly through transformation unchanged

June 17, 2014 | Proprietary and Confidential | - 17 -

IGATE

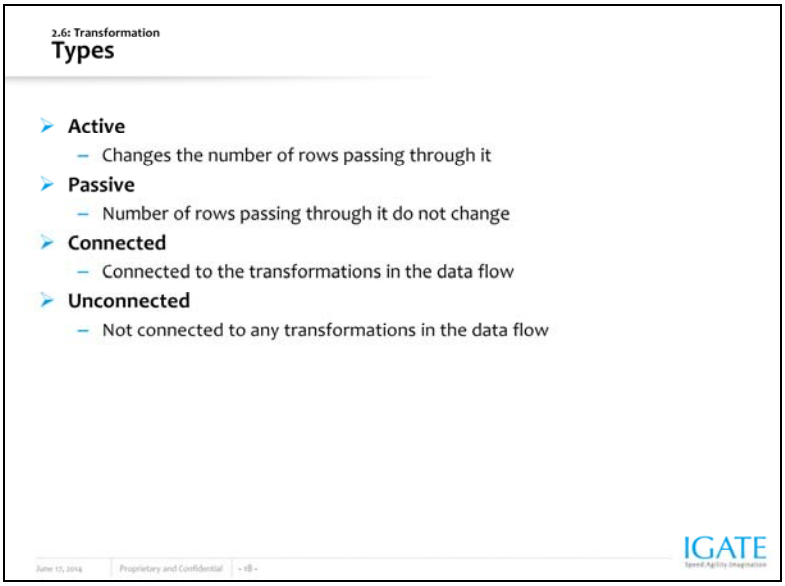
Speed. Agility. Imagination

The Designer provides a set of transformations that perform specific functions. For example, an Aggregator transformation performs calculations on groups of data. Perform the following Tasks to incorporate a transformation into a mapping:

- **Create the transformation -** Create it in the Mapping Designer as part of a mapping, in the Mapplet Designer as part of a mapplet, or in the Transformation Developer as a reusable transformation

- **Configure the transformation -** Each type of transformation has a unique set of options that can be configured

- **Link the transformation to other transformations and target definitions -** Drag one port to another to link them in the mapping or mapplet



2.6: Transformation

Types

- **Active**
  - Changes the number of rows passing through it
- **Passive**
  - Number of rows passing through it do not change
- **Connected**
  - Connected to the transformations in the data flow
- **Unconnected**
  - Not connected to any transformations in the data flow

June 17, 2014 | Proprietary and Confidential | - 18 -

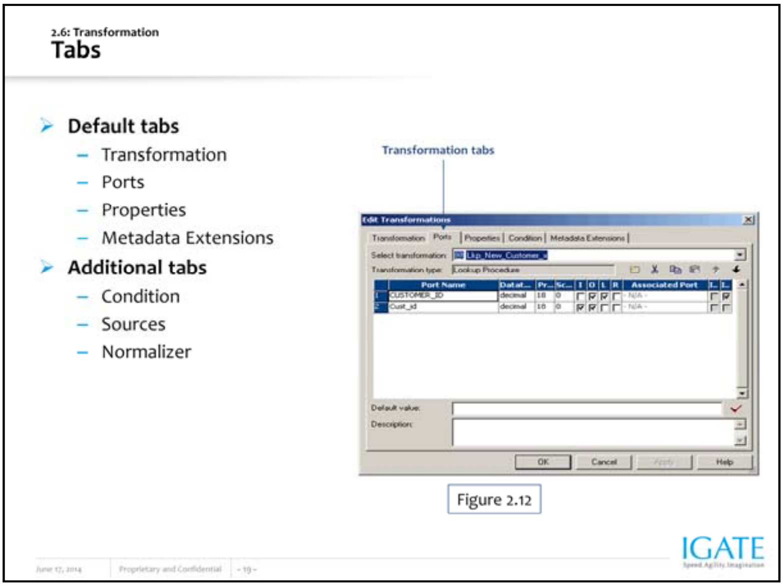
IGATE
Speed.Agility.Imagination

Active - An *active transformation* can change the number of rows that pass through it, such as a Filter transformation that removes rows that do not meet the filter condition.

Passive - A *passive transformation* does not change the number of rows that pass through it, such as an Expression transformation that performs a calculation on data and passes all rows through the transformation.

Connected - *Connected transformations* are connected to the data flow.

Unconnected - An *unconnected transformation* is not connected to other transformations in the mapping. It is called within another transformation, and returns a value to that transformation.



2.6: Transformation

Tabs

- Default tabs
  - Transformation
  - Ports
  - Properties
  - Metadata Extensions
- Additional tabs
  - Condition
  - Sources
  - Normalizer

Select transformation: Lkp_New_Customer_s

Transformation type: Lookup Procedure

	Port Name	Datat...	Pr...	Sc...	I	O	L	R	Associated Port	I...	L...
1	CUSTOMER_ID	decimal	18	0	☐	☑	☑	☐	- N/A -	☐	☑
2	Cust_id	decimal	18	0	☑	☑	☐	☐	- N/A -	☐	☐

Default value:

Description:

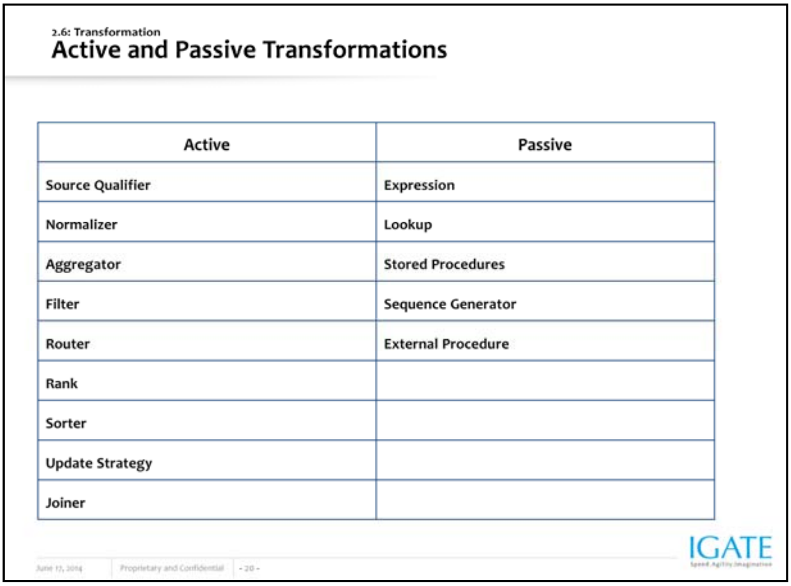
Figure 2.12

Each transformation has a minimum of four tabs:

- **Transformation** – allows to rename a transformation, switch between transformations, enter transformation comments, and make a transformation reusable
- **Ports** – allows to specify level attributes such as port name, datatype, precision, scale, primary/foreign keys, nullability
- **Properties** – allows to specify the amount of detail in the session log, and other properties specific to each transformation
- **Metadata Extensions** – allows to extend the data stored in the repository by associating information with individual repository objects

The additional tabs are :

- **Condition** – allows to enter conditions for e.g. Joiner or Filter Transformation
- **Sources** – allows to specify additional source definitions e.g. Source Qualifier Transformation
- **Normalizer** – allows to enter new ports for the Normalizer Transformation



2.6: Transformation

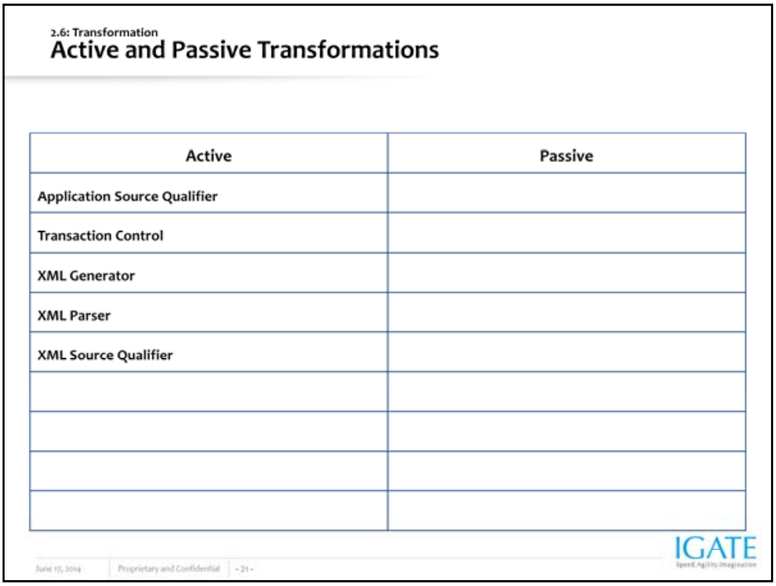
Active and Passive Transformations

Active	Passive
Source Qualifier	Expression
Normalizer	Lookup
Aggregator	Stored Procedures
Filter	Sequence Generator
Router	External Procedure
Rank	
Sorter	
Update Strategy	
Joiner	

June 17, 2014 | Proprietary and Confidential | - 20 -

Active Transformations:

- **Source Qualifier** - represents all data queries from the source
- **Normalizer** - normalizes records from VSAM or relational sources
- **Aggregator** - performs aggregate calculations
- **Filter** - serves as a conditional filter
- **Router** - serves as a group conditional filter
- **Rank** - limits records to top or bottom range
- **Sorter** - sorts the input rows in ascending or descending order
- **Update Strategy** - allows for logic to insert, update, delete, or reject data
- **Joiner** – allows extraction of data from heterogeneous sources
- **Application Source Qualifier -** Represents the rows that the Integration Service reads from an application, such as an ERP source, when it runs a session.
- **Transaction Control** - Defines commit and rollback transactions.
- **XML Generator** - Reads data from one or more input ports and outputs XML through a single output port.
- **XML Parser** - Reads XML from one input port and outputs data to one or more output ports.
- **XML Source Qualifier** - Represents the rows that the Integration Service reads from an XML source when it runs a session.



2.6: Transformation

Active and Passive Transformations

Active	Passive
Application Source Qualifier	
Transaction Control	
XML Generator	
XML Parser	
XML Source Qualifier	

June 17, 2014 | Proprietary and Confidential | - 21 -

IGATE
Speed. Agility. Imagination

Passive Transformations:

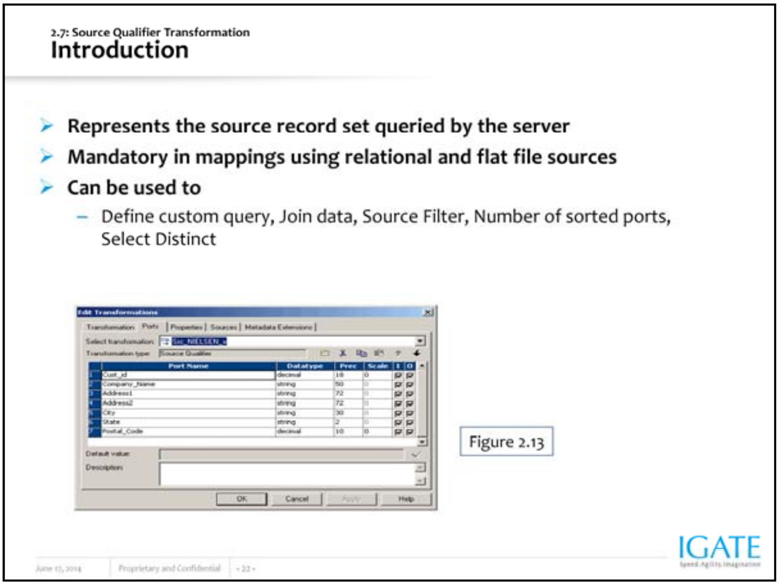
- **Expression** - performs simple calculations
- **Lookup** - looks up values and passes to other objects
- **Stored Procedures** - calls a stored procedure and captures return values
- **Sequence Generator** - generates unique ID values

- **External Procedure -** Calls a procedure in a shared library

Some transformation can be both **active or passive** which are as follows:

- **Java -** Executes user logic coded in Java.
- **Complex** - Transforms data in unstructured and semi-structured formats.
- **Custom** - Calls a procedure in a shared library or DLL
- **SQL** - Executes SQL queries against a database.

These transformations will be discussed in the further lessons.



2.7: Source Qualifier Transformation

Introduction

- Represents the source record set queried by the server
- Mandatory in mappings using relational and flat file sources
- Can be used to
  - Define custom query, Join data, Source Filter, Number of sorted ports, Select Distinct

Select transformation: Src_NIELSEN_s

Transformation type: Source Qualifier

	Port Name	Datatype	Prec	Scale	I	O
1	Cust_id	decimal	18	0	☑	☑
2	Company_Name	string	50	0	☑	☑
3	Address1	string	72	0	☑	☑
4	Address2	string	72	0	☑	☑
5	City	string	30	0	☑	☑
6	State	string	2	0	☑	☑
7	Postal_Code	decimal	10	0	☑	☑

Default value:

Description:

Figure 2.13

June 17, 2014 | Proprietary and Confidential | - 22 -

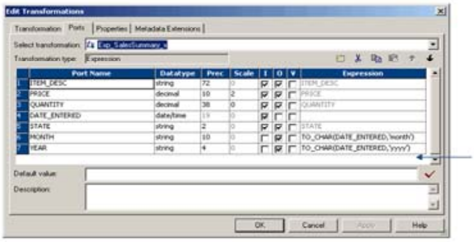
The Source Qualifier displays the transformation datatypes. The transformation datatypes in the Source Qualifier determine how the source database binds data when the Integration service reads it. Do not alter the datatypes in the Source Qualifier. If the datatypes in the source definition and Source Qualifier do not match, the Designer marks the mapping invalid when saved. Source Qualifier can be used to perform following tasks:

- **Join data originating from the same source database -** One can join two or more tables with primary-foreign key relationships by linking the sources to one Source Qualifier
- **Filter records when the Integration service reads source data -** If a filter condition is included, the Integration service adds a *WHERE* clause to the default query
- **Specify an outer join rather than the default inner join -** If a user-defined join is included, the Integration service replaces the join information specified by the metadata in the SQL query
- **Specify sorted ports -** If a number for sorted ports are specified, the Integration service adds an *ORDER BY* clause to the default SQL query
- **Select only distinct values from the source -** If Select Distinct is chosen, the Integration service adds a *SELECT DISTINCT* statement to the default SQL query
- Create a custom query to issue a special *SELECT* statement for the Integration service to read source data

2.8: Expression Transformation

Introduction

- Used to perform non-aggregate calculations
- Expression statement is entered in an output port
- Can enter only one expression for each output port



Select transformation: Exp_SalesSummary_x

Transformation type: Expression

	Port Name	Datatype	Prec	Scale	I	O	V	Expression
1	ITEM_DESC	string	72	0	☑	☑	□	ITEM_DESC
2	PRICE	decimal	10	2	☑	☑	□	PRICE
3	QUANTITY	decimal	38	0	☑	☑	□	QUANTITY
4	DATE_ENTERED	date/time	19	0	☑	□	□	
5	STATE	string	2	0	☑	☑	□	STATE
6	MONTH	string	10	0	□	☑	□	TO_CHAR(DATE_ENTERED,'month')
7	YEAR	string	4	0	□	☑	□	TO_CHAR(DATE_ENTERED,'yyyy')

Default value:

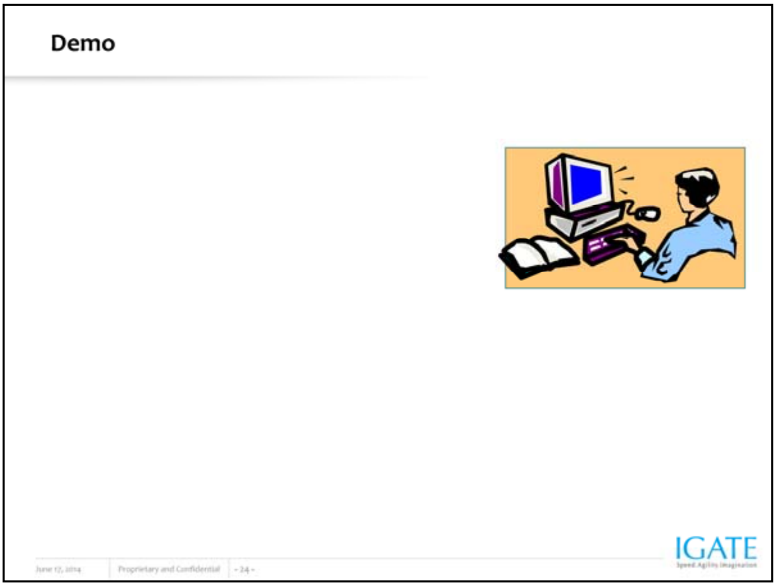
Description:

Expression Statement

Figure 2.14

IGATE
Speed. Agility. Imagination

The ***Expression Transformation*** is used to calculate values in a single row. For example, in order to adjust employee salaries, concatenate first and last names, or convert strings to numbers. Use the Expression transformation to perform any non-aggregate calculations.



Demo

June 17, 2014 | Proprietary and Confidential | - 24 -

The first step in the process of moving data between sources and targets is to create a mapping in the Mapping Designer.

Steps to create a mapping:

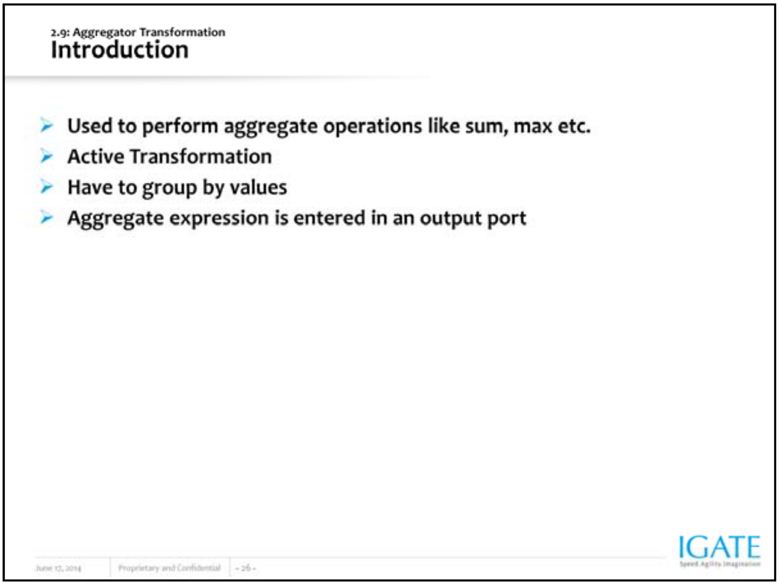
1. Open the Mapping Designer.
2. Choose ***Mappings* | *Create*** menu option, or drag a repository object into the workspace.
3. Enter a name for the new mapping and click ***OK***. The naming convention for mappings is m_*MappingName*, such as m_demo

Steps to create a Source Qualifier Transformation:

Drag the required source, which can be a flat file or relational table in the mapping area, by default the Source Qualifier Transformation will be created for the source definition.

Steps to create an Expression Transformation:

1. In the Mapping Designer, choose ***Transformation* | *Create***. Select the Expression transformation. Enter a name for it (the convention is EXP_*TransformationName*) and click ***OK***.
2. Create the input ports. If the input transformation is already available, select Link Columns from the Layout menu and then click and drag each port used in the calculation into the Expression transformation. With this method, the Designer copies the port into the new transformation and creates a connection between the two ports. Or, open the Edit dialog box and create each port manually.
   Note: In order to make this transformation reusable, create each port manually within the transformation.
3. Repeat the previous step for each input port which has to be added to the expression.
4. Create the output ports (O) needed, making sure to assign a port datatype that matches the expression return value.
5. Click the small button that appears in the Expression section of the dialog box and enter the expression in the Expression Editor. To prevent typographic errors, where possible, use the listed port names and functions.
6. Port names used as part of an expression in an Expression transformation follow stricter rules than port names in other types of transformations:
   - A port name must begin with a single- or double-byte letter or single- or double-byte underscore (_)
   - It can contain any of the following single- or double-byte characters: a letter, number, underscore (_), $, #, or @
7. Check the expression syntax by clicking ***Validate***. If necessary, make corrections to the expression and check the syntax again. Then save the expression and exit the Expression Editor.
8. Connect the output ports to the next transformation or target.
9. Choose ***Repository* | *Save.***



2.9: Aggregator Transformation

Introduction

- Used to perform aggregate operations like sum, max etc.
- Active Transformation
- Have to group by values
- Aggregate expression is entered in an output port

June 17, 2014 | Proprietary and Confidential | - 26 -

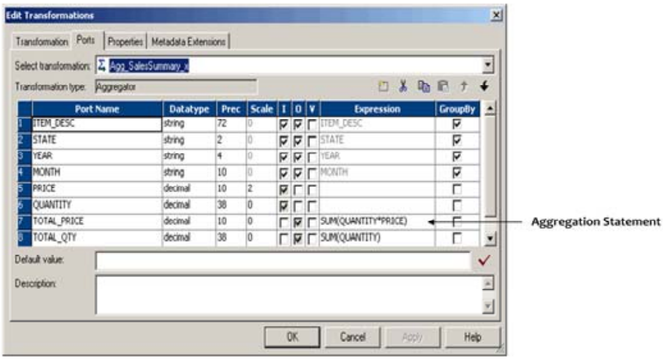
IGATE
Speed Agility Imagination

The *Aggregator Transformation* allows to perform aggregate calculations, such as average and sum.

To configure ports in the Aggregator Transformation we can:

- Enter an aggregate expression in any output port, using conditional clauses or non-aggregate functions in the port
- Create multiple aggregate output ports
- Configure any input, input/output, output, or variable port as a group by port, and use non-aggregate expressions in the port
- Improve performance by connecting only the necessary input/output ports to subsequent transformations, reducing the size of the data cache
- Use variable ports for local variables

Aggregator Transformation



Edit Transformations

Transformation | Ports | Properties | Metadata Extensions

Select transformation: Agg_SalesSummary_x

Transformation type: Aggregator

	Port Name	Datatype	Prec	Scale	I	O	V	Expression	GroupBy
1	ITEM_DESC	string	72	0	☑	☑	☐	ITEM_DESC	☑
2	STATE	string	2	0	☑	☑	☐	STATE	☑
3	YEAR	string	4	0	☑	☑	☐	YEAR	☑
4	MONTH	string	10	0	☑	☑	☐	MONTH	☑
5	PRICE	decimal	10	2	☑	☐	☐		☐
6	QUANTITY	decimal	38	0	☑	☐	☐		☐
7	TOTAL_PRICE	decimal	10	0	☐	☑	☐	SUM(QUANTITY*PRICE)	☐
8	TOTAL_QTY	decimal	38	0	☐	☑	☐	SUM(QUANTITY)	☐

Default value:

Description:

OK | Cancel | Apply | Help

Aggregation Statement

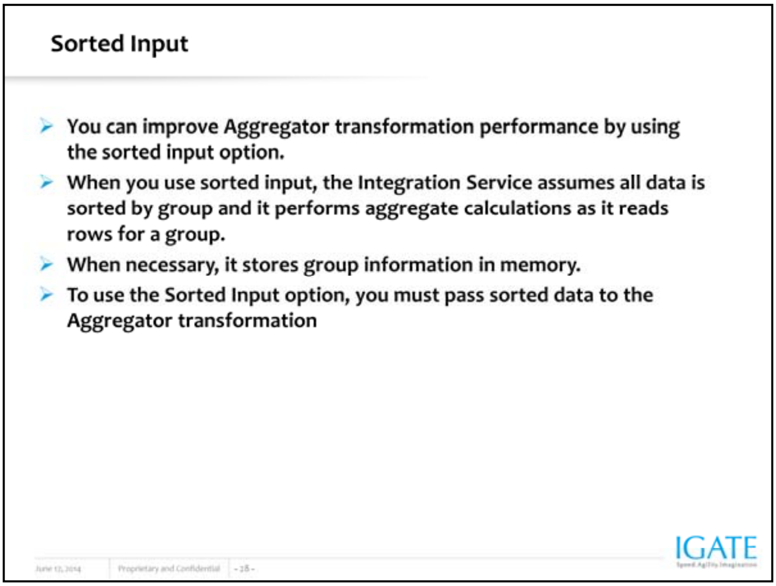
Figure 2.15

June 12, 2014 | Proprietary and Confidential | - 27 -

Components of the Aggregator Transformation

The Aggregator is an active Transformation, changing the number of rows in the data flow. It must be connected to the data flow. The Aggregator Transformation has the following components and options:

- **Aggregate expression -** Entered in an output port. Can include non-aggregate expressions and conditional clauses
- **Group by port -** Indicates how to create groups.
- **Sorted input -** Is used to improve session performance. To use this option, the data passed to the Aggregator Transformation sorted by Group by port in ascending or descending order
- **Aggregate Cache -** The Integration service stores data in the aggregate cache until it completes aggregate calculations. It stores group values in an index cache and row data in the data cache.

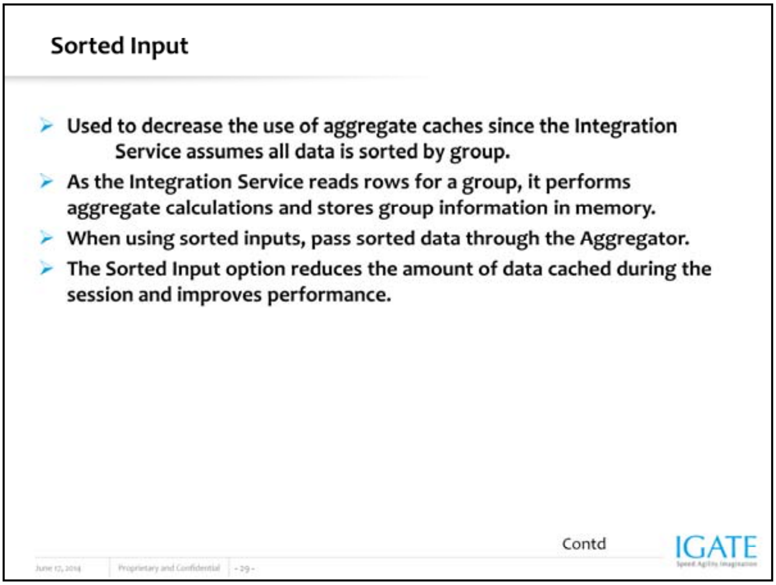


Sorted Input

- You can improve Aggregator transformation performance by using the sorted input option.
- When you use sorted input, the Integration Service assumes all data is sorted by group and it performs aggregate calculations as it reads rows for a group.
- When necessary, it stores group information in memory.
- To use the Sorted Input option, you must pass sorted data to the Aggregator transformation

June 17, 2014 | Proprietary and Confidential | - 28 -

IGATE
Speed. Agility. Imagination



Sorted Input

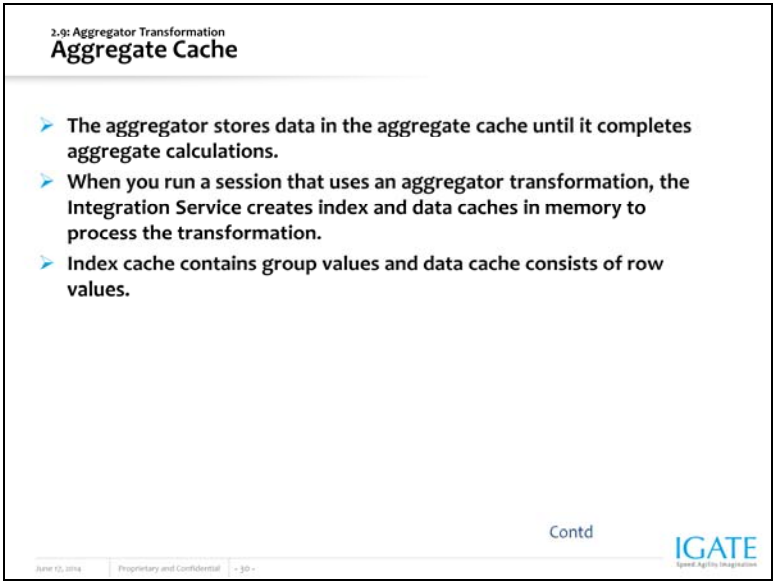
- Used to decrease the use of aggregate caches since the Integration Service assumes all data is sorted by group.
- As the Integration Service reads rows for a group, it performs aggregate calculations and stores group information in memory.
- When using sorted inputs, pass sorted data through the Aggregator.
- The Sorted Input option reduces the amount of data cached during the session and improves performance.

Contd

June 17, 2014 | Proprietary and Confidential | - 29 -

Ex: for calculating sum of salaries department wise, if data comes in random order, Integration service needs to use cache for holding data until all the entries for a particular department is calculated (deptno 10,20....).

By using sorted input, Integration service does not use cache and just performs calculations assuming the data comes in sorter order thus improvement in performance.



2.9: Aggregator Transformation

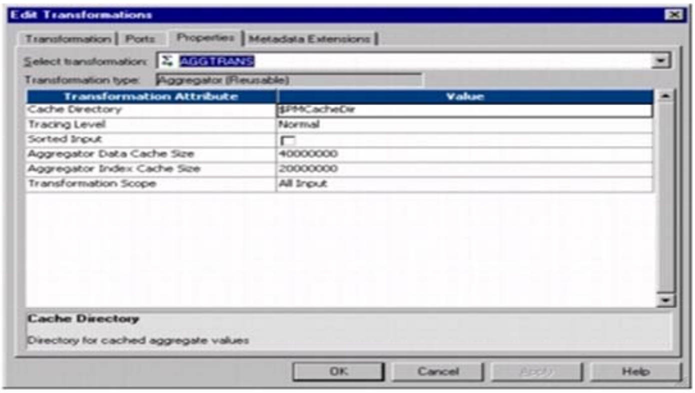
Aggregate Cache

- The aggregator stores data in the aggregate cache until it completes aggregate calculations.
- When you run a session that uses an aggregator transformation, the Integration Service creates index and data caches in memory to process the transformation.
- Index cache contains group values and data cache consists of row values.

Contd

2.9: Aggregator Transformation

Aggregate Cache



Select transformation: AGGTRANS

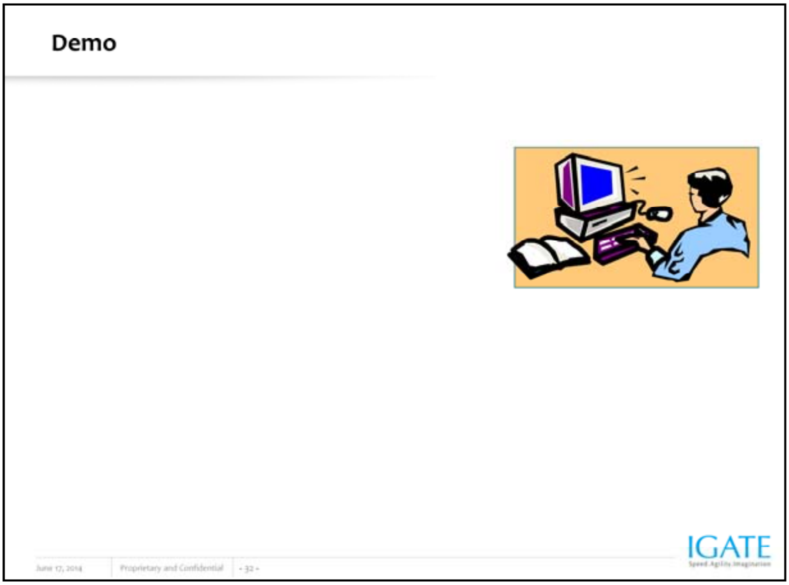
Transformation type: Aggregator (Reusable)

Transformation Attribute	Value
Cache Directory	$PMCacheDir
Tracing Level	Normal
Sorted Input	□
Aggregator Data Cache Size	40000000
Aggregator Index Cache Size	20000000
Transformation Scope	All Input

Cache Directory

Directory for cached aggregate values

Figure 2.16



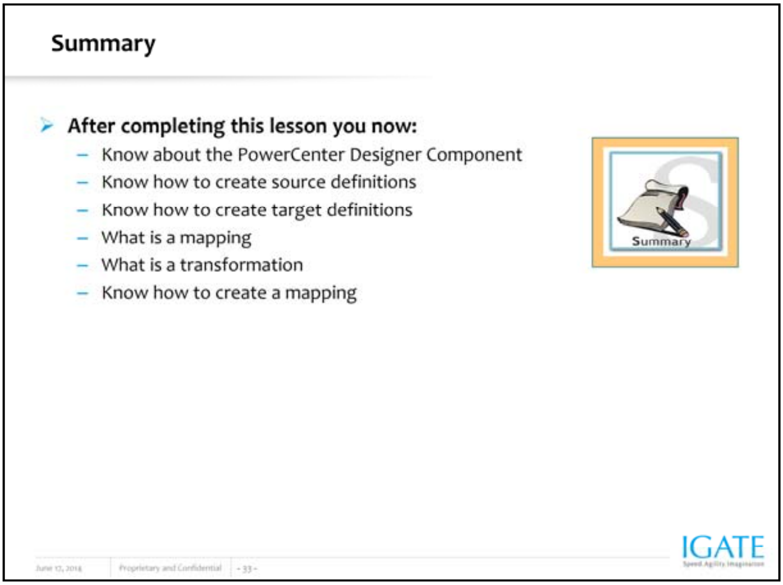
Demo

June 17, 2014 | Proprietary and Confidential | - 32 -

IGATE
Speed.Agility.Imagination

To create an Aggregator Transformation:

1. In the Mapping Designer, choose ***Transformation* | *Create*** menu option. Select the Aggregator Transformation. The naming convention for Aggregator Transformations is AGG_*TransformationName*.
2. Enter a description for the transformation.
3. Enter a name for the Aggregator, click ***Create***. Then click ***Done***. The Designer creates the Aggregator Transformation.
4. Drag the desired ports to the Aggregator Transformation. The Designer creates input/output ports for each port included.
5. Double-click the title bar of the transformation to open the Edit Transformations dialog box.
6. Select the ***Ports*** tab.
7. Click the group by option for each column the Aggregator should use to create groups.
8. If a non-aggregate expression to modify groups has to be used, click the ***Add*** button and enter a name and data type for the port. Make the port an output port by clearing Input (I).

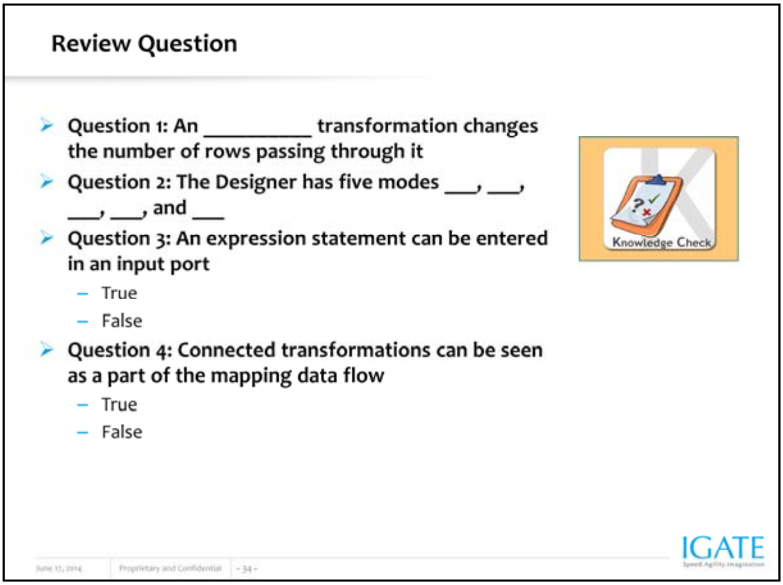


Summary

- **After completing this lesson you now:**
  - Know about the PowerCenter Designer Component
  - Know how to create source definitions
  - Know how to create target definitions
  - What is a mapping
  - What is a transformation
  - Know how to create a mapping

Summary

IGATE
Speed.Agility.Imagination



Review Question

- Question 1: An __________ transformation changes the number of rows passing through it
- Question 2: The Designer has five modes ___, ___, ___, ___, and ___
- Question 3: An expression statement can be entered in an input port
  - True
  - False
- Question 4: Connected transformations can be seen as a part of the mapping data flow
  - True
  - False

Knowledge Check

June 17, 2014 | Proprietary and Confidential | - 34 -

IGATE
Speed.Agility.Imagination