Chapter 33. October 2009

Welcome to the October 2009 edition of IBM InfoSphere Information Server Developer's Notebook (IISDN). This month we answer the question;

> Can you show me how to perform my data mappings using FastTrack, and include a comparison to Information Analyzer and DataStage?
>
> *Excellent question! The FastTrack component to the Information Server Platform serves as the dynamic bridging between Business Analysts and IT; providing a drag and drop, instantly validated portal to discover and then communicate data flows and relationships between areas of the business.*
>
> In this edition of IISDN, we offer a primer on FastTrack including all major objects and conventions, and detail how FastTrack interacts with the adjacent components of Information Analyzer and DataStage/QualityStage.

All of these solutions were *developed and tested* on the (IBM) InfoSphere Information Server Platform (IIS) version 8.1.2, using the Microsoft Windows XP/SP3 platform to support IIS client programs, and a RedHat Enterprise Linux (RHEL 5.4) 64 bit SMP server (Linux kernel version 2.6.18-164.el5) to support the IIS server side components.

The IBM InfoSphere Information Server Platform allows for a single, consistent, and accurate view of data across the full width of the corporate enterprise, be it relational or non-relational, staged or live data. As a reminder, the IBM InfoSphere Information Server Platform software product contains the following major components;

> Business Glossary, Business Glossary Anywhere, Business Glossary Extender for Eclipse, Information Analyzer, FastTrack, Information Services Director, DataStage, QualityStage, Balanced Optimization, MetaBridges and Brokers, Metadata Workbench, Federation Server, Classic Federation, Event Publisher, Replication Server, Data Architect, DataMirror Transformation Server, the Parallel Framework, Common Connector Architecture, Validator/Discovery and more.

Obviously, the IBM InfoSphere Information Server Platform is a large and capable product, addressing many strategic needs across the enterprise, and supporting different roles and responsibilities.

33.1 Terms and core concepts

The author to this month’s edition of IBM InfoSphere Information Server Developer’s Notebook (IISDN) started in Information Technology (IT) in 1984, when the Unix operating system was still considered an exotic toy of Bell Telephone and universities, and Structured Query Language (SQL) was a data platform that would never scale. Back then, Management Information Systems (MIS: the common name of the computer department within most companies at that time) had a standard 2 year backlog of application requests; applications that would, at best, meet the narrow and core operating needs of the company. There was no concept of using IT for decision support analysis, or as a channel to produce revenue.

There has always been the challenge of gathering end user application requirements in a form that IT could record and understand, or that the end users could then read and make sense of. In 1991, this month’s author of IISDN visited what is now America’s largest retailer where they used numerous, numerous 3 foot by 8 foot sheets of paper with the corporate *data model* printed on it. This pages would hang on the outside of cube walls, where we’d flip them and hold meetings in front of them to review their hidden treasures.

Data modelling gave way to various forms of *process modelling*, wherein actions are recorded and described as; Add New Customer, Update Customer Contact Data, Add New Customer Order, etcetera. These process events sit between the corporate data model, and the graphical controls on the end user’s screen; Click this Button, and something will hopefully occur. And there are whole categories of specialized software to again, gather end user requirements in a form that IT could record and understand, and that end users could make sense of.

However, the above information is often recorded in word processing documents and spreadsheets, and become inaccurate, or out of date; no one maintains these assets, as there is diminishing value in the time spent to update these documents over the benefit they provide. These documents are isolated stove pipes of information; they don’t integrate with the rest of the operation.

> **Note:** This is the role the FastTrack component to the Information Server Platform fills; a dynamic integrated and instantly validated, accurate and live portal to your company’s data mappings and specifications. And, its designed for ease of use by Business Analysts as they communicate with IT.

A short name for the IBM InfoSphere Information Server (IIS) Platform software product name is, *Information Server Platform*, and that is a name that serves it well. IIS could be viewed as everything around a database, except the database. IIS can move, cleanse, and enrich data, while at the same time providing end to end data lineage. Serving the entire life cycle and spectrum of data creation and analysis, Information Server is a platform.

Where FastTrack sits

The FastTrack component to IBM InfoSphere Information Server Platform (IIS) can be purchased and used as a stand alone software product. In this case, FastTrack would generate its metadata from a collection of data connections and data stores, and then import and output data mappings and relationships to other areas of the business.

When used with other components to IIS, you could view FastTrack as sitting between the Information Analyzer and DataStage/QualityStage components to IIS. (With Business Glossary and MetaData Workbench not too far away.) A common deployment model might be;

– Business Analysts use Business Glossary to create a hierarchy of business terms; this is what we mean when we say, 'Break Bulk Shipping Rates', and this term is related to a "Bill of Lading", etcetera.

Note: We use the term Business Analyst to mean, a subject matter expert in an area of the business. For example, someone who knows how we operate and ship our company's merchandise, but doesn't necessarily know the first thing about how to install or configure Oracle.

Business Analysts know what tables are, and that they relate and contain columns, but that's about it from an IT perspective.

– Business Analysts use Information Analyzer to perform data discovery; I have this Customer list here from the East Coast, where else is Customer data located inside my company.

– And then Business Analysts use FastTrack to create a *mapping specification*; I have Customer data in 3 locations, I want someone or something to aggregate and clean this data, then allow me to match it with Orders along 4 measurable indices, Geography, Time, Product Line, etcetera.

  FastTrack can be used to create decision support data mappings, operational data mappings, basically data mappings for any purpose.

  The Business Analyst also uses Information Analyzer and FastTrack to associate tables and columns with Business Glossary terms, which further complete this awareness of data; its use, migration, life cycle, lineage, and so on.

An optional output of a FastTrack data mapping can be a DataStage/ QualityStage component Parallel Job, although that is not required. Just having the FastTrack tool allow the Business Analyst to deliver their needs and specifications is reward enough. In this edition of IISDN, however, we will demonstrate both capabilities.

Data connections, stores, and other dependencies

4 Prior editions of this document, IBM InfoSphere Information Server Developer’s Notebook (IISDN), detailed use and configuration of the Information Analyzer (IA) component to the IBM Information Server Platform (IIS). And another prior edition of IISDN detailed how to configure ODBC connectivity inside IIS. Comments include;

– December/2007, ODBC Configuration and Test

  This edition of IISN detailed how to configure and test ODBC connectivity when using IIS.

  Generally we configure ODBC connectivity system wide, then add this capability DataStage/QualityStage Project by Project. (Adding a desired level of control.)

  Information Analyzer has its own DataStage Project (ANALYZER PROJECT) under the covers to support the data analysis that IA needs to perform. We need to add ODBC connectivity to this (DataStage) Project as well.

– February/2009, IA Part 1

  This edition of IISDN provided an overview of IA functionality, and detailed its setup and configuration. IA requires the definition of Data Connections, which largely equate to an ODBC connection to a given/single Database.

  Under Data Connections, Data Stores are created which represent a given collection of Tables and Columns underneath said Data Connection.

  Data Connections are first defined system wide, and you may have many of these. Each Data Connection then, can have zero of more Data Stores.

  After you create a system wide (known to the entire IIS system) collection of Data Connections and Data Stores, you can then optionally add these definitions to zero or more IA Projects.

– March/2009, IA Part 2

  This edition of IISDN completed the exercises and information towards configuring IA, and understanding and using its then 5 core areas of functionality.

– August/2009, IA Part 3

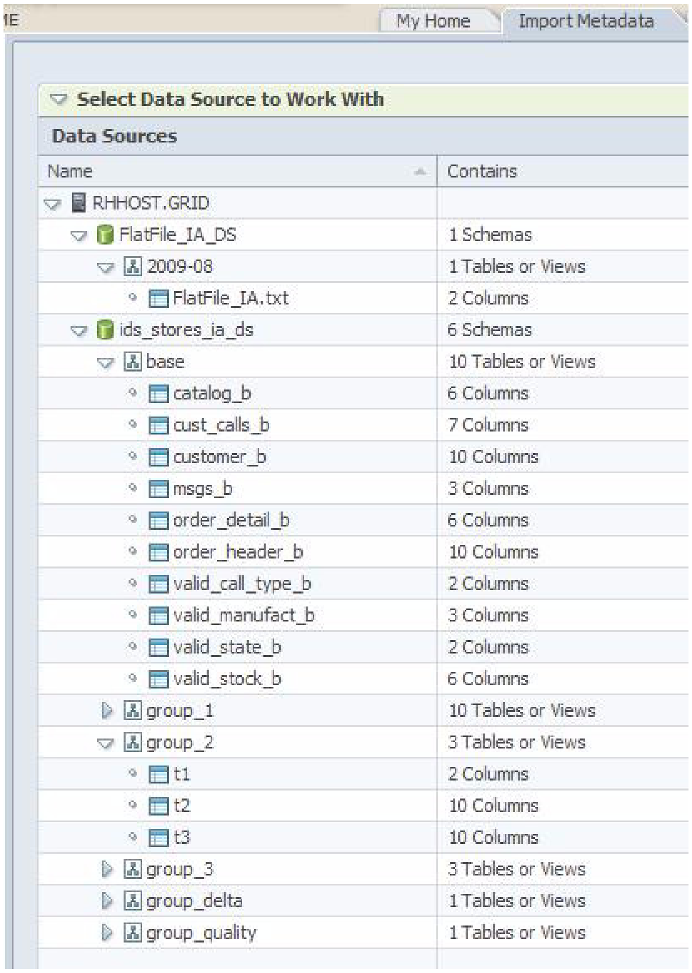
This edition of IISDN detailed areas of functionality located inside IA not previously discussed, including those delivered with version 8.1.1; a large features release to IA, and to all of IIS.

In this edition of IISDN, we also detailed a new Flat File Wizard, improving the capability to analyze and perform other operations on flat files of data; not requiring that they be hosted inside a SQL database.

– September/2009, IA Part 4

And this edition of IISDN detailed Data Rules, a new 6th core area of functionality inside IA.

In those prior editions of IISDN, we introduced a data model we used throughout each of those documents. Figure 33-1 displays the Home -> Metadata Management -> Import Metadata menu item from within Information Analyzer.



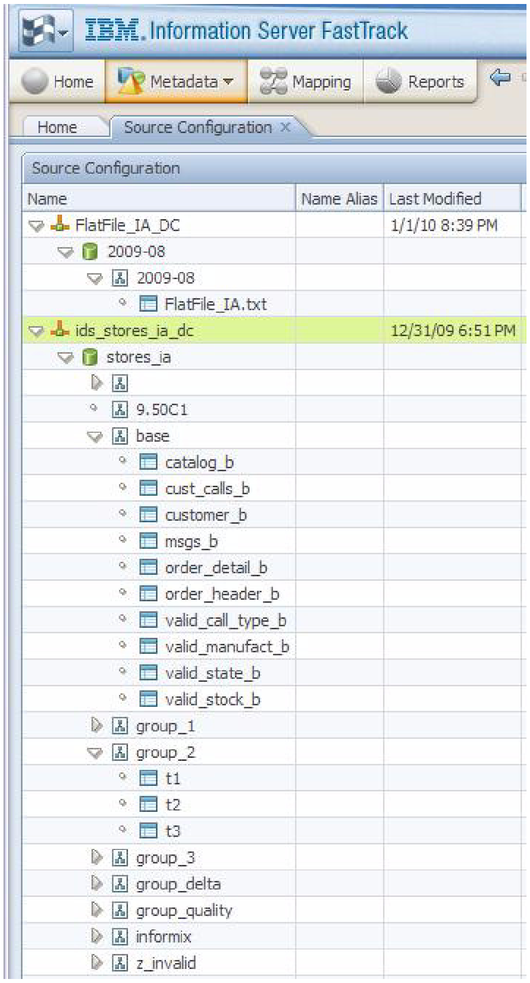
Select Data Source to Work With

Data Sources

Name	Contains
RHHOST.GRID	
FlatFile_IA_DS	1 Schemas
2009-08	1 Tables or Views
FlatFile_IA.txt	2 Columns
ids_stores_ia_ds	6 Schemas
base	10 Tables or Views
catalog_b	6 Columns
cust_calls_b	7 Columns
customer_b	10 Columns
msgs_b	3 Columns
order_detail_b	6 Columns
order_header_b	10 Columns
valid_call_type_b	2 Columns
valid_manufact_b	3 Columns
valid_state_b	2 Columns
valid_stock_b	6 Columns
group_1	10 Tables or Views
group_2	3 Tables or Views
t1	2 Columns
t2	10 Columns
t3	10 Columns
group_3	3 Tables or Views
group_delta	1 Tables or Views
group_quality	1 Tables or Views

Figure 33-1 Connections and Stores object hierarchy inside IA.

And figure Figure 33-2 displays Metadata -> Source Configuration menu items from within FastTrack.



Source Configuration

Name	Name Alias	Last Modified
FlatFile_IA_DC		1/1/10 8:39 PM
2009-08		
2009-08		
FlatFile_IA.txt		
ids_stores_ia_dc		12/31/09 6:51 PM
stores_ia		
9.50C1		
base		
catalog_b		
cust_calls_b		
customer_b		
msgs_b		
order_detail_b		
order_header_b		
valid_call_type_b		
valid_manufact_b		
valid_state_b		
valid_stock_b		
group_1		
group_2		
t1		
t2		
t3		
group_3		
group_delta		
group_quality		
informix		
z_invalid		

Figure 33-2 Connections and Stores object hierarchy inside FastTrack.

The Information Analyzer and FastTrack components to IIS share the very same Data Connections, Data Stores, and then the resulting SQL Schemas, Tables and Column objects. If you imported these objects inside IA, you can fully access and manage them in FastTrack, and vice versa.

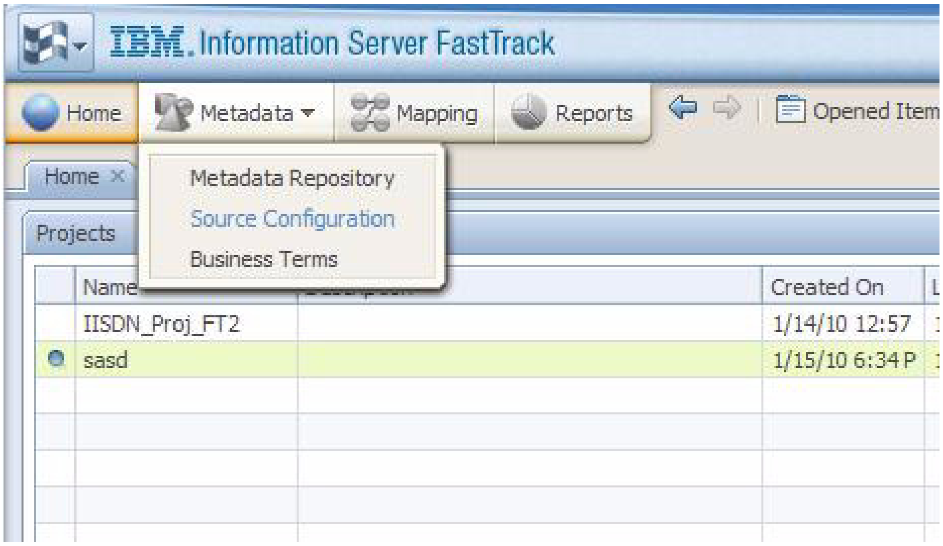
How one configures Data Connections and then Data Stores inside Information Analyzer or FastTrack are very similar. We think the FastTrack user interface is a little cleaner, but either one works.

> **Note:** Because IA and FastTrack are so similar in this area, we are not going to document how to create Data Connections or Data Stores, or then import Table and Column metadata; literally, you could read the past editions of IISDN written for IA, and figure it out for FastTrack.
>
> Further, while our examples below will use the Tables displayed in Figure 33-1 and Figure 33-2, we are going to keep our examples simple. In most cases, you only need any 2 or 3 simple tables with a small number of columns.

FastTrack menus, and objects

Figure 33-3 displays the FastTrack main menu. From the FastTrack Home menu, you can Add, Delete and Open FastTrack Projects. Unlike Information Analyzer (IA) Projects, FastTrack Projects are lighter, and have access to the entire defined collection of Data Connections, Data Stores, etcetera.

Also, for whatever reason, there are no Folders inside FastTrack Projects; no means to create a hierarchy of same typed objects. (It wasn't viewed as a necessary feature.) As a result, you may then choose to work with more Projects to further organize your activities.



Name		Created On	L
IISDN_Proj_FT2		1/14/10 12:57	1
sasd		1/15/10 6:34 P	1

Figure 33-3 FastTrack main menus.

The FastTrack Metadata menu is where you define Data Connections and Data Stores, and import Schemas, Table and Column definitions. You can also use the Business Terms menu item and other areas of FastTrack to do some amount of work related to your creation of management of a hierarchy of Terms, your Business Glossary.

It is the FastTrack Mapping menu item where you do the bulk of your activities inside FastTrack.

> **Note:** We're also not going to cover the FastTrack Report menu item in this edition of IISDN. While there are Reports that are specific to FastTrack certainly, defining, running and opening the results of a given Report is a topic we have covered before.

As a means to offer an overview of FastTrack related objects and their relation to one another, the following is offered;

- Under the FastTrack Metadata menu item, you see 2 primary objects; Mapping Specifications, and Mapping Components.
- Related to Mapping Specifications, the following is offered;
    - A Mapping Specification could be as simple as listing (n) columns from 1 Source Table, and writing the same number of columns in 1 Target Table.
    - Additional complexities and modifiers to the above could include,

      Adding a Filter (for example, ShipWeight > 5), to reduce the number of rows that flow from the Source to the Target.

      Adding a Switch (for example, ShipWeight < 5 goes to table Target table T1, ShipWeight >= 5 goes to Target table T2) to direct output variably to two or more Target tables.

> **Note:** All complete Mapping Specifications requires at least one Source Table, and one Target Table.
>
> To use a Switch, you must have at least two Target tables.
>
> A Switch that would only output to one Target Table is by definition, a Filter.

Adding an Aggregate (for example, Sum(ShipWeight)), to collapse detail records and provide summary values; Sum(), Max, Min(), Count(), etcetera.

- All of the above (Filter, Switch, Aggregate) work with a simple Specification Mappings. You can also perform Lookups and Joins.

Functionally, Lookups and Joins are very similar.

Generally a Lookup is done to a symbol table; for example, I have the Department_Code, lookup the Department_Name. Generally a lookup table is one that is static (non-changing) and small.

A Join, however, generally is done to combine two Source data sets; for example, read the Customer table and the Orders table, combine them, and let me continue processing downstream.

FastTrack Mapping Specifications support both of these; Lookup and Join.

- And FastTrack supports a logical operation this document will call Source Merge; for example, I have Customer records in three distinct Source tables, read all 3 and continue processing downstream.

Note: If you know much about the DataStage/QualityStage components to the Information Server Platform, a Source Merge is done via a Funnel Stage. To use a Funnel Stage, the upstream data sources have to match in their column definitions. Huh?

Each of the data sources to the Funnel Stage have to send the same number, name, and compatible column data types to the source side of the Funnel.

While described simply above, a Source Merge can become quite complex; perhaps only one of the three data Sources should have a given data validation performed, perhaps only Source-2 should first default out missing or Null column values.

The FastTrack component to the Information Server Platform uses a unique and easy technique to create a (combined) Mapping Specification to perform a Source Merge. In effect, you create multiple Mapping Specifications writing to the same Target table, and then Generate a Job of all participating Mapping Specifications.

Note: We have an example of the Source Merge technique below.

- And the Rules and Functions.

While FastTrack already has Filters (described above, which is a form of Rule, and Lookups, which can be used for validation, also a rule), FastTrack also has objects called Rules and Functions.

Because Business Analysts are not expected to be able to write procedural programming code, a FastTrack Rule specification gives the Business Analyst a free form entry area to enter things like, If the ShipWeight is less than 20 pounds, send via FedEx Air, otherwise send via FedEx Ground.

This specification is then added to the metadata and sent to the (presumably) DataStage/QualityStage developer who implements this code.

In the event that the Business Analyst does know some (more exact) programming, they can enter a Function, in place of a Rule. Examples of Functions would include, SetNull(), UpCase(), or Col1 : Col2, which concatenates 2 Source columns into 1 Target column.

As a general guideline, we will say that Rules fully and adequately act as a replacement for Functions, and require a less technical skill set for the Business Analyst.

Note: Which is probably good, currently there is no Wizard, Picker, Content Assist or similar for the embedded Function set inside FastTrack.

FastTrack supports all Functions found to exist inside a DataStage/QualityStage component Parallel Job, Parallel Transformer Stage.

While there is no graphical (selector) for Functions, the FastTrack user interface will give feedback as to whether the entered Function is valid or invalid.

– And then Mapping Components.

Inside FastTrack, the Business Analyst creates Mapping Specifications, which alone or grouped together (remember Source Merge), define data flows, data migrations, data validations, etcetera.

As a product of this activity, it would be expected that something or someone is going to use a tool or write a computer program to deliver the above.

The DataStage/QualityStage components to the Information Server Platforms are one such tool. To aid in code re-usability, DataStage/ QualityStage have a number of objects themselves to promote this behavior. Such code re-use objects include;

- Parallel or Server Shared Containers.
- Parallel or Server Routines.
- Java, XSLT and Web Service Transformers.
- And many more.

Note: Several past editions of this document discuss code re-use inside the DataStage/QualityStage component to the Information Server Platform. If you were to read only one of these documents, the May/2008 edition would be a good choice.

Mapping Components inside FastTrack currently support Parallel Shared Containers only; not Server Shared Containers, Routines of any type, nor Java/XSLT/Web-Service Transformers.

To support those objects, you must place them inside a Parallel Shared Container.

A Parallel Shared Container may provide just output, accept just input, or provide both input and output.

A FastTrack Mapping Component allows the Business Analyst to look for, and then include, Parallel Shared containers into their Mapping Specifications. A single Mapping Specification can include a given Mapping Component for just input or output, not both. To make use of a Parallel Shared Container that accepts input and provides output, you need to create 2 Mapping Specifications, in a manner similar to the Source Merge discussion above.

What do Parallel Shared Containers do?

Lots of stuff. A Parallel Shared Container could represent a real time change data capture data source (or target), or a complex and re-usable piece of application logic. and more.

Note: In this edition of IISDN we are not going to provide an example of using Mapping Components (access to Parallel Shared Containers).

Look for us to cover this topic in a future edition of IISDN, likely to be addressing the more power user type functions of FastTrack.

33.2 Complete the following examples

In this section of this document we are going to create a number of simpler FastTrack Mapping Specifications. Ultimately when you are done, you could combine these singleton examples into 1 or more, more powerful FastTrack Mapping Specifications.

The following are considered pre-requisites to these examples;

– You have already defined an ODBC connection to a SQL database with 3 or more tables.

  We are going to use the sample tables from the February/2009 edition of this document, but any database with a Customer, Valid_State, and some sort of Target table will work.

  (Make a Target table with some number of VarChar columns and you will be fine there.)

– You have already defined your Data Connections, Data Stores, and imported the Schemas, Tables and Column definitions related to the above.

  As mentioned above, the February/2009 edition of this document details how to perform this work within the Information Analyzer component to the Information Server Platform. With an understanding of that material, you can easily perform these steps inside FastTrack.

– If you wish to use the Discover (More) feature within FastTrack, where a given single or set of Source/Target columns automatically present you with a list of appropriate/matching columns, you will need to complete Column Analysis inside the Information Analyzer component to the Information Server Platform, *and you will need* to Publish Results.

> **Note:** The above is another powerful example of the benefits of the Shared Metadata Repository to the Information Server Platform.
>
> Knowledge gained during the Column Analysis phase of using the Information Analyzer component aids this area of functionality inside FastTrack. Without this activity having been completed and published, you are left having to manually search for matching columns inside FastTrack.
>
> Not hard, but not as nice as having the tool do it for you automatically.

– Optionally, you may choose below to Generate a DataStage/QualityStage component Parallel Job. You need access to the Designer Client to

DataStage/QualityStage, and need to log on to an existing DataStage/QualityStage Project.

– And lastly, you have to logon to the FastTrack graphical client, and make a new FastTrack Project. (Don’t worry about any of the Properties settings for this new Project at this time.)

Example-1, simple Mapping Specification and Job Generation

1. Create a simple 1 to 1 table Mapping Specification inside FastTrack.

   a. From the Mapping main menu item, highlight the Mapping Specifications Folder under your FastTrack Project.

      This action will enable the New Mapping Specification menu item, in the Tasks pane to the right of the display.

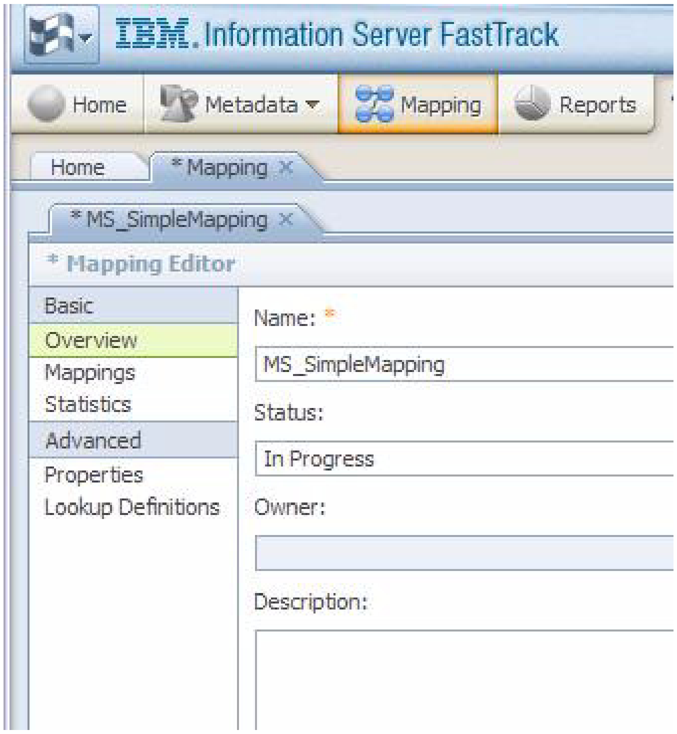
      Click the New Mapping Specification menu item.

   b. The menu within (New) Mapping Specification has at least 5 panes, 3 under Basic (Overview, Mappings, and Statistics), and 2 under Advanced (Properties and Lookup Definitions).

      Initially all of our real activity is done under, Basic -> Mappings.

      Under Basic -> Overview, set this Mapping Specification Name to, MS_SimpleMapping.

      Example as shown in Figure 33-4.



Name: *

MS_SimpleMapping

Status:

In Progress

Owner:

Description:

Figure 33-4 Menu within (New) Mapping Specification.

c. As a first time user, its is possible that the many Repository (pick lists) are not currently displayed.

From the main menu, Select, View -> Database Metadata.

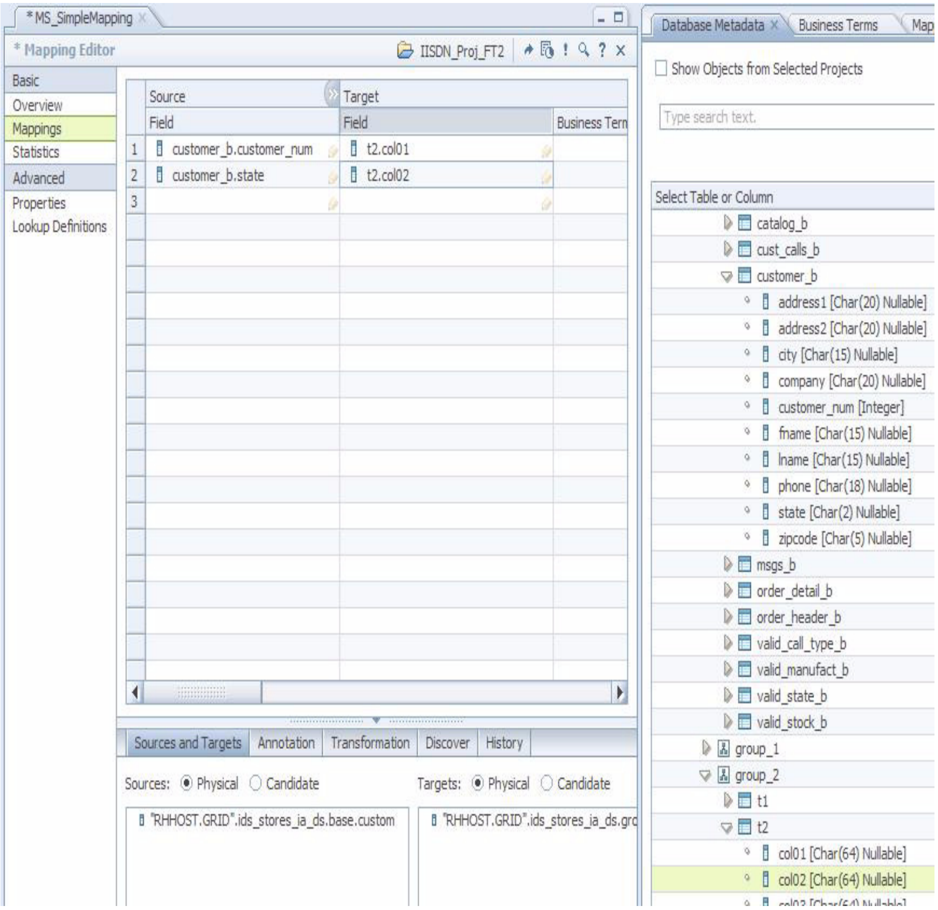
Also, Click the Basic -> Mappings pane from the left side of the display.

d. Here we wish to specify a mapping with 2 Source columns (Fields) and 2 Target columns (Fields). The Source should read from only 1 table, and the Target should write to only 1 table.

You can Drag and Drop Columns from the Database Metadata pane to the Mappings pane.

Drag and Drop 2 Columns to Source, and 2 to Target.

Example as shown in Figure 33-5.



	Source	Target	
	Field	Field	Business Tern
1	customer_b.customer_num	t2.col01	
2	customer_b.state	t2.col02	
3			

Sources: ● Physical ○ Candidate

"RHHOST.GRID".ids_stores_ia_ds.base.custom

Targets: ● Physical ○ Candidate

"RHHOST.GRID".ids_stores_ia_ds.grc

Figure 33-5 (New) Mapping Specification, Basic -> Mappings pane.

Believe it or not, this simple Mapping Specification is done.

Optionally we are going to call to Generate a DataStage/QualityStage Parallel Job, as a means to further understand what we created above.

e. Click Save, then Click Generate Job.

Generating a (DataStage/QualityStage Parallel) Job is a 4 step process.

Step-1, Composition Type, is used when combining 2 or more Mapping Specifications to create more complex, more capable Mappings and Jobs.

We don't need to perform an activity on Step-1 at this time, Click Next.

Step-2 is where we specify the output location of our generated Job, and if/where we want any Shared Tables to be created.

Browse to a location for the Job we are about to Generate under the Tree entitled, DataStage and QualityStage Project and Folder.

Name this Job.

And Un-Check the control entitled, Use Shared Table Definitions.

Then Click, Next.

Note: Table Definitions inside the Information Server Platform are either Local or Shared.

A Shared Table Definition is a primary means that column metadata, domain analysis, comments, etcetera, are transmitted/communicated between the various components of the Information Server Platform.

We are not going to demonstrate any Shared Table Definition use or behavior at this time.

Step-3 is where you can set variable connection parameters for all Source and Target tables. And you can create and manage a number of re-usable Connection objects.

We already inherited connection parameters with our Source and Target tables, and don't wish to use these feature at this time.

Click Next.

Note: FastTrack Connection objects are not related to the similarly named DataStage/QualityStage Connection objects.

That is probably good. You wouldn't automatically want Business Analysts overwriting the Connection objects you create and manage inside DataStage/QualityStage.

Once the Job arrives in DataStage/QualityStage, you can modify the Parallel Job, including saving the connection criteria as a stand alone object.

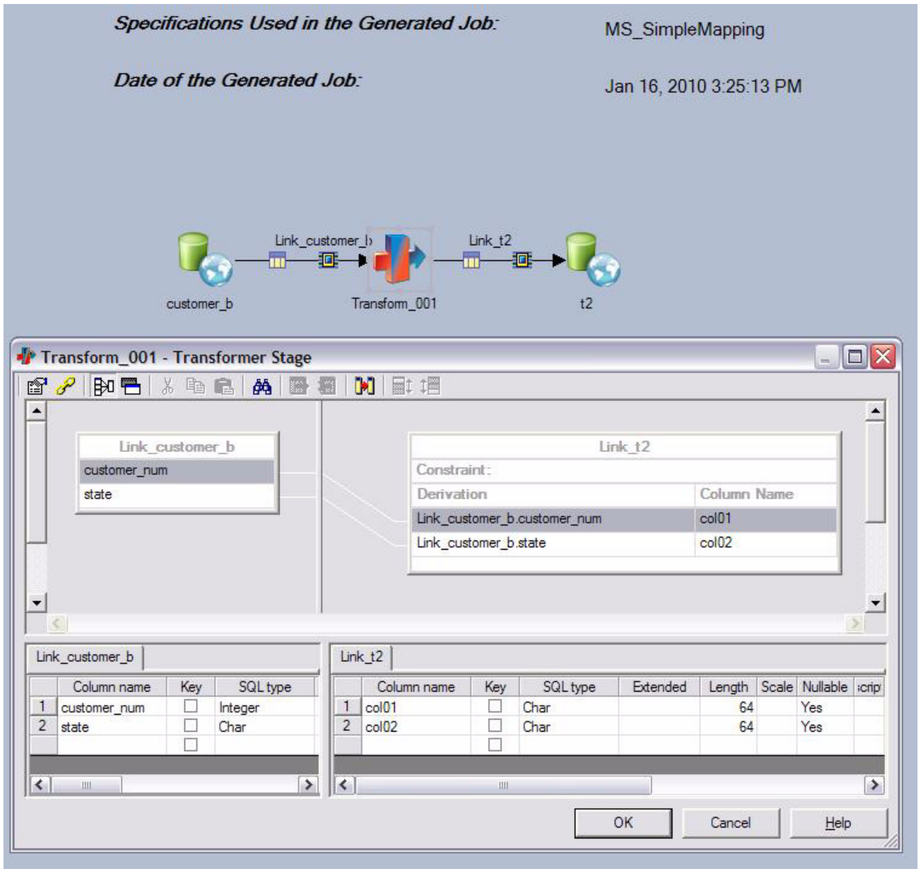
And Step-4 is where you can over ride naming standards for the Stages that are about to be generated.

Click Finish.

f. Inside the DataStage/QualityStage Designer program, open the newly Generated Parallel Job.

You will regularly have to use the main menu item, Repository -> Refresh, as the Designer client does not automatically poll the shared Metadata Repository for newly arrived objects.

Example as shown in Figure 33-6. A code review follows.



Specifications Used in the Generated Job: MS_SimpleMapping

Date of the Generated Job: Jan 16, 2010 3:25:13 PM

Link_customer_b
customer_num
state

Link_t2

Constraint:

Derivation	Column Name
Link_customer_b.customer_num	col01
Link_customer_b.state	col02

Link_customer_b

	Column name	Key	SQL type
1	customer_num	☐	Integer
2	state	☐	Char
		☐	

Link_t2

	Column name	Key	SQL type	Extended	Length	Scale	Nullable	:crip
1	col01	☐	Char		64		Yes	
2	col02	☐	Char		64		Yes	
		☐						

Figure 33-6 Generated Parallel Job.

A code review related to Figure 33-6 includes;

– The generated Parallel Job has our Source and Target ODBC Connector Stages, as expected.

  The ODBC Connector Stages are pre-configured with the DSN Name, and User Name from the data sources that these column definitions came from.

  Minus passwords, this Parallel job is ready to Run.

- The Transformer Stage is supplied to re-cast the column data types as expected, from Source to Target. Again, inherited from the metadata from the actual Source and Target columns.

Note: Allowing for a more robust generated Parallel Job, you will see a number of Transformer Stages are generated for your ease in modifying these Parallel Jobs.

g. If you wish and if you can, enter passwords to the Source and Target Stages and Run this Parallel Job.

When you are done, be certain to close this Parallel Job. We are about to edit this Mapping Specification, and unless we Generate to a Parallel Job with a new/different name, having this Parallel Job open will cause conflict.

Example-2, Filters

2. Add a Filter to the above Mapping Specification.

a. Return to the FastTrack graphical client, and navigate to the Advanced -> Properties pane of the Mapping Specification we created above.

b. Highlight the Filter TAB, and enter the following in the text entry field entitled, (Filter) Expression,

```
customer_b.state <> "CO"
```

Basically, table_name.column_name (expression) value

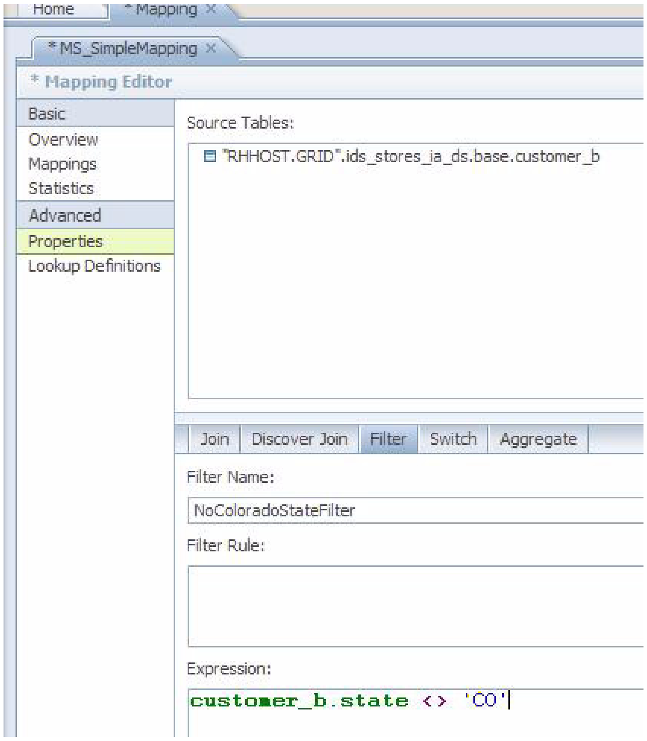
Example as shown in Figure 33-7.

If you see your Expression displayed in Green, it is correct. Red means there is an error.

Note: Sadly, there is no Wizard, Picker or Code Assist to be found here currently.

If you don't know the expression you wish to specify, you can always enter un-formatted text in the text entry field entitled, Filter Rule

As free form text, this action will not generate specific code, but your Mapping Specification Filter detail is recorded.



Source Tables:

"RHHOST.GRID".ids_stores_ia_ds.base.customer_b

Filter Name: NoColoradoStateFilter

Filter Rule:

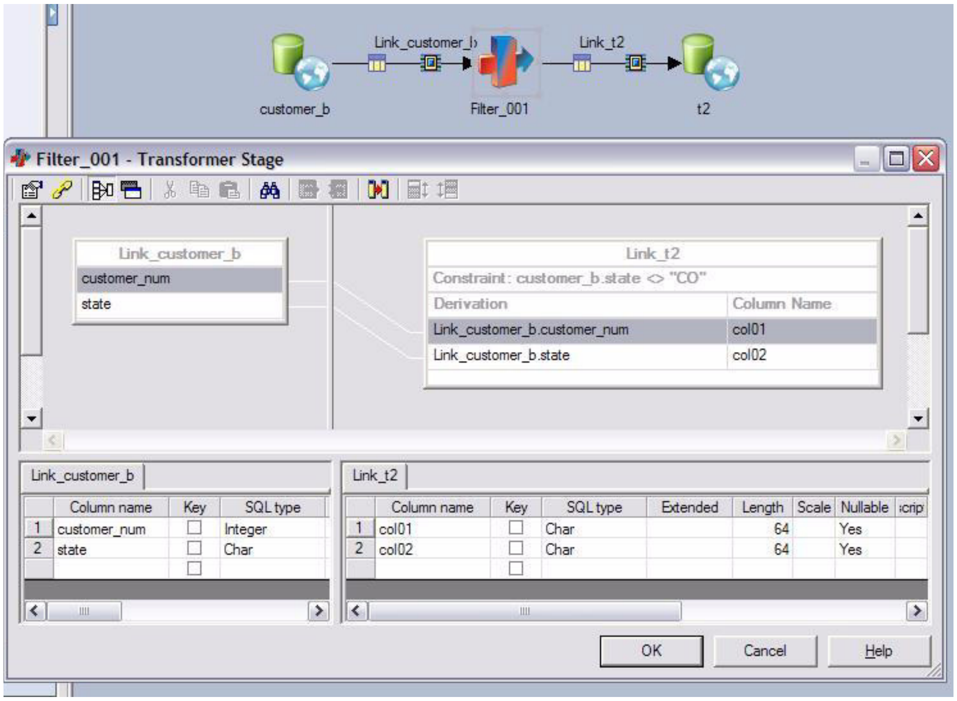
Expression:

```
customer_b.state <> 'CO'
```

Figure 33-7 Example of Filter Expression.

c. Complete the steps to Generate this Parallel Job.

Example output is displayed in Figure 33-8.



Link_customer_b

- customer_num
- state

Link_t2

Constraint: customer_b.state <> "CO"

Derivation	Column Name
Link_customer_b.customer_num	col01
Link_customer_b.state	col02

Link_customer_b

	Column name	Key	SQL type
1	customer_num	□	Integer
2	state	□	Char
		□	

Link_t2

	Column name	Key	SQL type	Extended	Length	Scale	Nullable	:crip
1	col01	□	Char		64		Yes	
2	col02	□	Char		64		Yes	
		□						

Figure 33-8 Example of generated output with Filter.

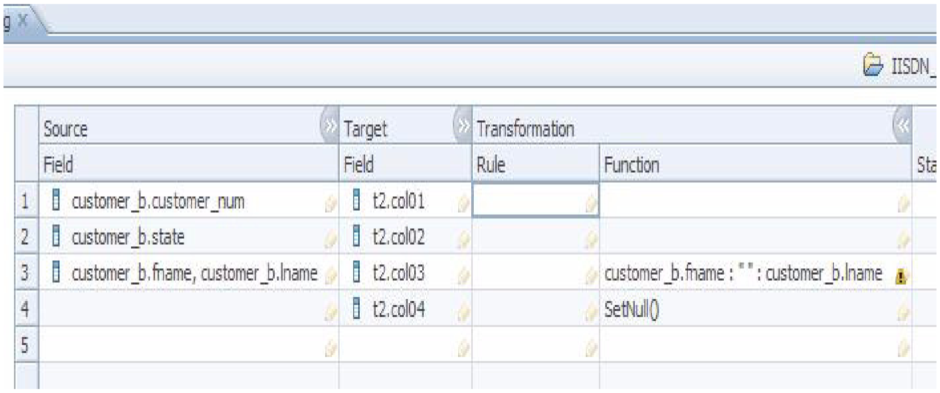
Example-3, Function Expressions

3. Add Function Expressions to given Target Columns.

   a. Return to the FastTrack graphical client, and navigate to the Basic -> Mappings pane of the Mapping Specification we created above.

   b. From the Database Metadata pane, we are going to Drag and Drop a number of new Columns from the Database Metadata pane onto our Source and Target Columns.

   Example as shown in Figure 33-9.



	Source	Target	Transformation		
	Field	Field	Rule	Function	Sta
1	customer_b.customer_num	t2.col01			
2	customer_b.state	t2.col02			
3	customer_b.fname, customer_b.lname	t2.col03		customer_b.fname : " " : customer_b.lname	
4		t2.col04		SetNull()	
5					

Figure 33-9 New Source and Target Columns, and new Functions.

In Figure 33-9, we Dragged and Dropped 2 Columns from the Database Metadata pane into the same Source Column (Field), on Line-3.

Just because we can place 2 or more Columns there, doesn’t mean that FastTrack knows how we want to map these multiple objects.

The Function Expression, also on Line-3 solves that problem for us. That’s the Function Expression for concatenate two Source columns with a space in between.

The Function Expression on Line-4 says we don’t have a Source column, and set this Target Column to Null.

Note: Source Tables and Columns (Fields) exist before we arrive; we can’t change they or their contents.

As a result, Functions and Rules are always applied to Target Tables and Columns.

A Mapping Specification may use/list a given Source Column multiple times. A Target column may be used only once. (You can’t write 2 things into 1 Target column.)

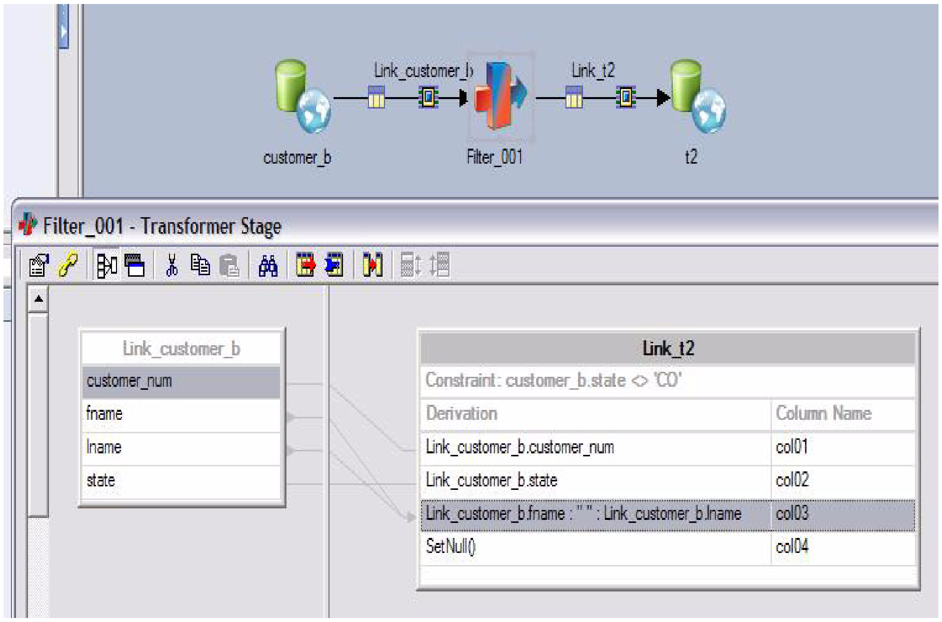
Note: You will occasionally see a Warning or Error marker on a given Mapping line or control until you Save and/or Generate the Job. This is normal.

If you have a blank line on your Mapping that you wish to delete, Right-Click the numeric line marker on the left side of the display, and Select, Delete.

In effect that Right-Click on the number designator selects the whole row.

c. Complete the steps to Generate this Parallel Job.

Example output is displayed in Figure 33-10.



Filter_001 - Transformer Stage

Link_customer_b
customer_num
fname
lname
state

Link_t2

Constraint: customer_b.state <> 'CO'

Derivation	Column Name
Link_customer_b.customer_num	col01
Link_customer_b.state	col02
Link_customer_b.fname : " " : Link_customer_b.lname	col03
SetNull()	col04

Figure 33-10   Sample output with new Functions, and our previous Filter.

Example-4, Lookups

4. Add a Lookup of State Name from associated State Abbreviation.

a. Return to the FastTrack graphical client, and navigate to the Advanced -> Lookup Definition pane of the Mapping Specification we created above.

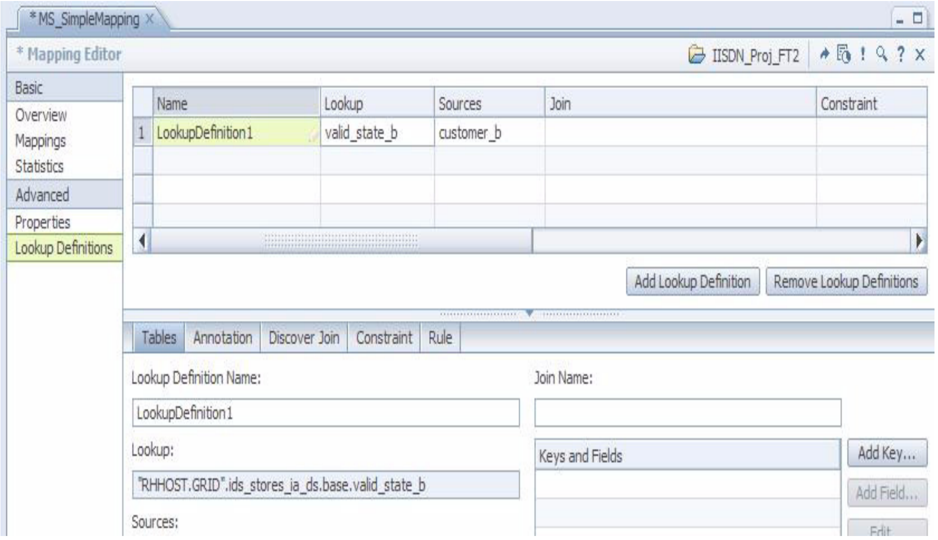
b. Click, Add Lookup Definition.

c. Give this Lookup Definition a Name, and Click OK.

d. You have to Drag and Drop 2 Tables Name into the upper portion of the display, entitled, Lookup and Sources.

   Example as displayed in Figure 33-11.

   You drag Table names here, not Column names.



	Name	Lookup	Sources	Join	Constraint
1	LookupDefinition1	valid_state_b	customer_b		

Lookup Definition Name: LookupDefinition1

Join Name:

Lookup: "RHHOST.GRID".ids_stores_ia_ds.base.valid_state_b

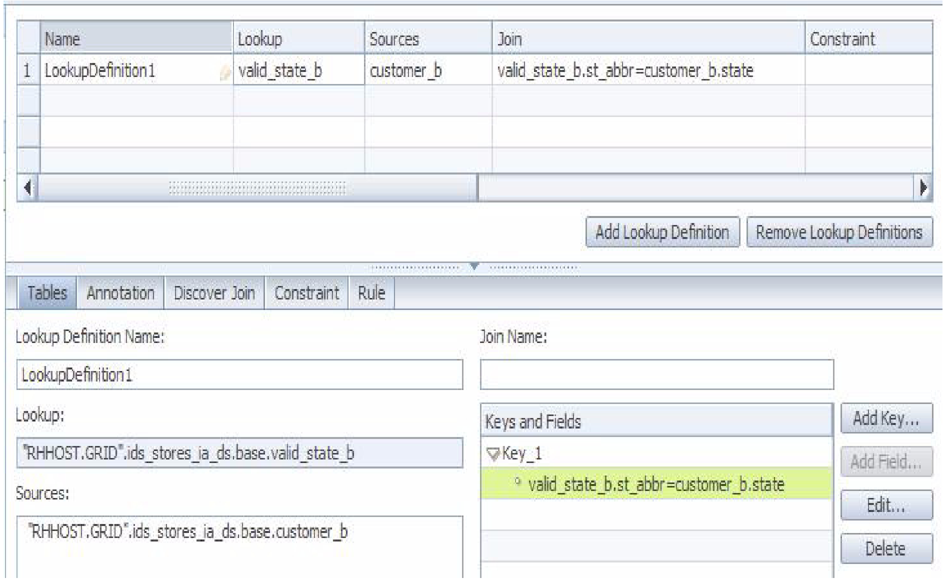
Sources:

Figure 33-11 Creating a Lookup Definition.

e. After you have completed the steps in Figure 33-11, then you can Click, Add Key.

   Complete specifying the Join Keys between the two Tables in our Lookup Definition.

   Example as shown in Figure 33-12.



	Name	Lookup	Sources	Join	Constraint
1	LookupDefinition1	valid_state_b	customer_b	valid_state_b.st_abbr=customer_b.state	

Lookup Definition Name: LookupDefinition1

Join Name:

Lookup: "RHHOST.GRID".ids_stores_ia_ds.base.valid_state_b

Sources: "RHHOST.GRID".ids_stores_ia_ds.base.customer_b

Keys and Fields

- Key_1
  - valid_state_b.st_abbr=customer_b.state

Figure 33-12 Completing the Join Key specification.

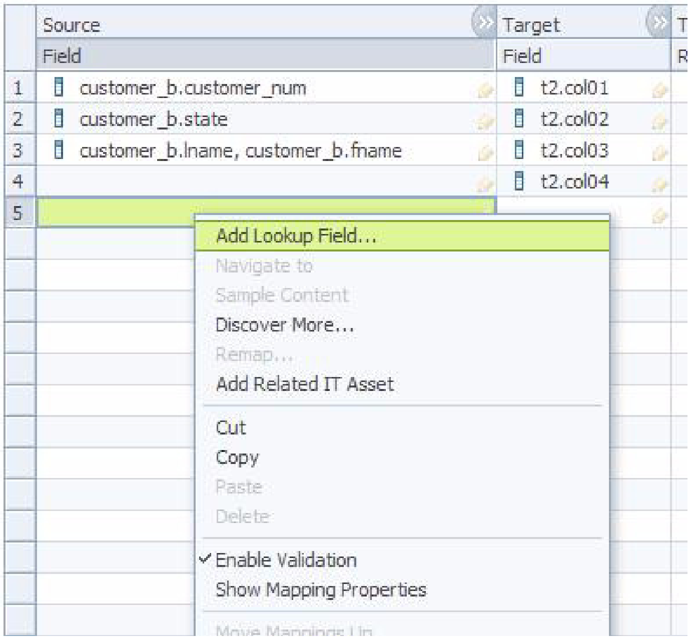
f. Now we add this Lookup Column to the Mappings panel of our Mapping Specification.

Return to the Basic -> Mappings panel.

As a new column we are reading (Sourcing), we add this to our Source Columns.

Right-Click on a new Source column, and Select, Add Lookup Field.

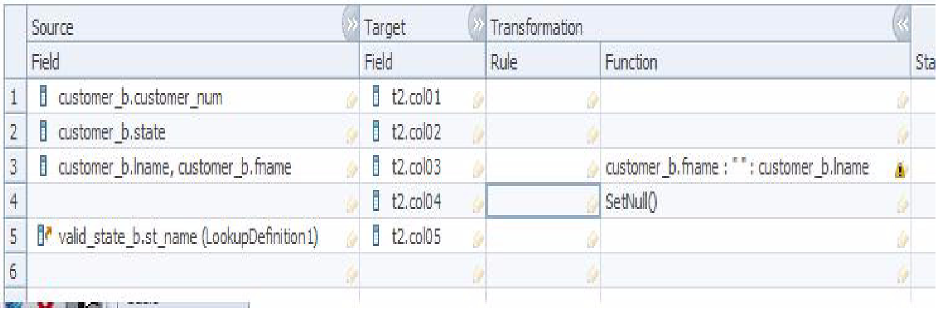
Example as shown in Figure 33-13.



	Source Field	Target Field
1	customer_b.customer_num	t2.col01
2	customer_b.state	t2.col02
3	customer_b.lname, customer_b.fname	t2.col03
4		t2.col04
5		

Figure 33-13 Adding a Lookup Source Column.

g. Complete the example as shown in Figure 33-14.



	Source Field	Target Field	Transformation Rule	 Function
1	customer_b.customer_num	t2.col01		
2	customer_b.state	t2.col02		
3	customer_b.lname, customer_b.fname	t2.col03		customer_b.fname : " " : customer_b.lname
4		t2.col04		SetNull()
5	valid_state_b.st_name (LookupDefinition1)	t2.col05		
6				

Figure 33-14 Completed example with Lookup Source Column.

h. Complete the steps to Generate this Parallel Job.

Example output is displayed in Figure 33-15.

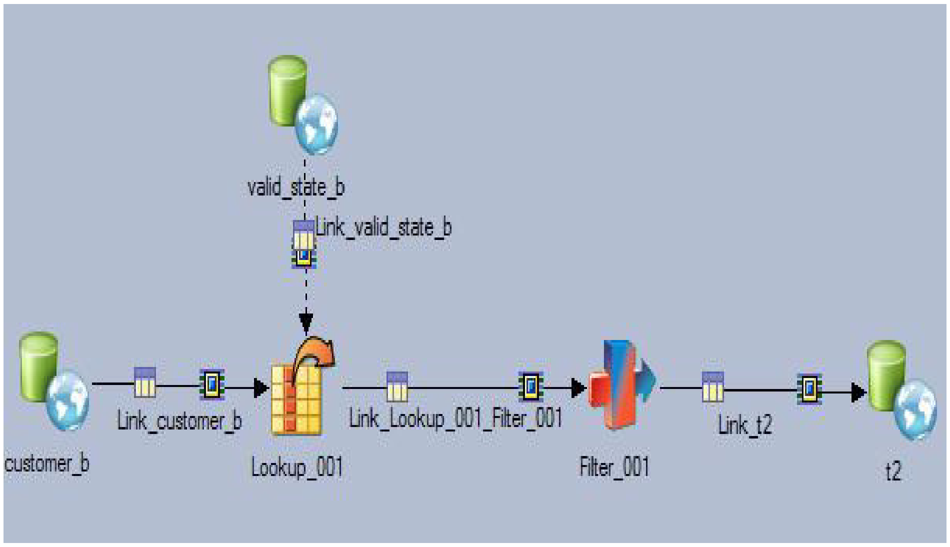


Figure 33-15 Generated Job with Lookup, and remainder of mappings.

Note: Initially the generated Job above will contain the Lookup portion of the Mapping Specification inside a Local Container; trying to generate best practice style code, isolating a possible re-usable unit of work.

If you wish to view the job as displayed above, Right-Click the Local Container and Select, Deconstruct.

Example-5, Switches, writing to multiple Target tables

5. Add a Switch (write to multiple Target Tables)

   A use case for writing to multiple Target Tables might include; write data from the last 90 days to the 'live' table, and data older than that to the 'history' table. Or, you may just need to write different columns to different tables.

   Here we are going to use FastTrack in a manner most similar to the Spreadsheet which it replaces.

   a. Return to the FastTrack graphical client, and navigate to the Basic -> Mappings pane of the Mapping Specification we created above.

   b. Given that you created 5 Source Columns, Click the Number-1, and then Shift-Click the Number-5 row numbers.

      Then Right-Click and Select Copy.

c. Click anywhere on Row-6, Right-Click and Select Paste.

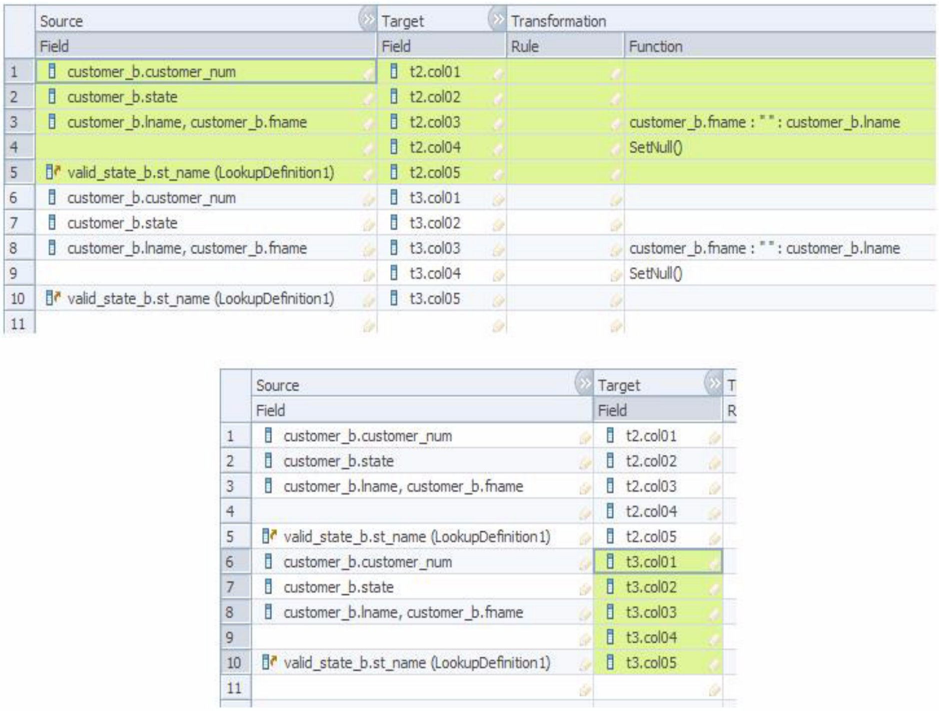
Initially the FastTrack user interface is going to complain that you duplicated 5 Target columns, which is a no no.

d. Click the Target Column in Row-6, then Shift-Click the Target Column in Row-10. Then Right-Click and Select, Delete.

e. Drag and Drop 5 new Target columns into these now empty Rows 6-10 Target Column Cells.

Example as shown in Figure 33-16.

In order for this example (creating a Switch) to work, these 5 new Target Columns need to be from a second, never before seen table to this Mapping Specification Target.



	Source	Target	Transformation	
	Field	Field	Rule	Function
1	customer_b.customer_num	t2.col01		
2	customer_b.state	t2.col02		
3	customer_b.lname, customer_b.fname	t2.col03		customer_b.fname : " " : customer_b.lname
4		t2.col04		SetNull()
5	valid_state_b.st_name (LookupDefinition1)	t2.col05		
6	customer_b.customer_num	t3.col01		
7	customer_b.state	t3.col02		
8	customer_b.lname, customer_b.fname	t3.col03		customer_b.fname : " " : customer_b.lname
9		t3.col04		SetNull()
10	valid_state_b.st_name (LookupDefinition1)	t3.col05		
11				

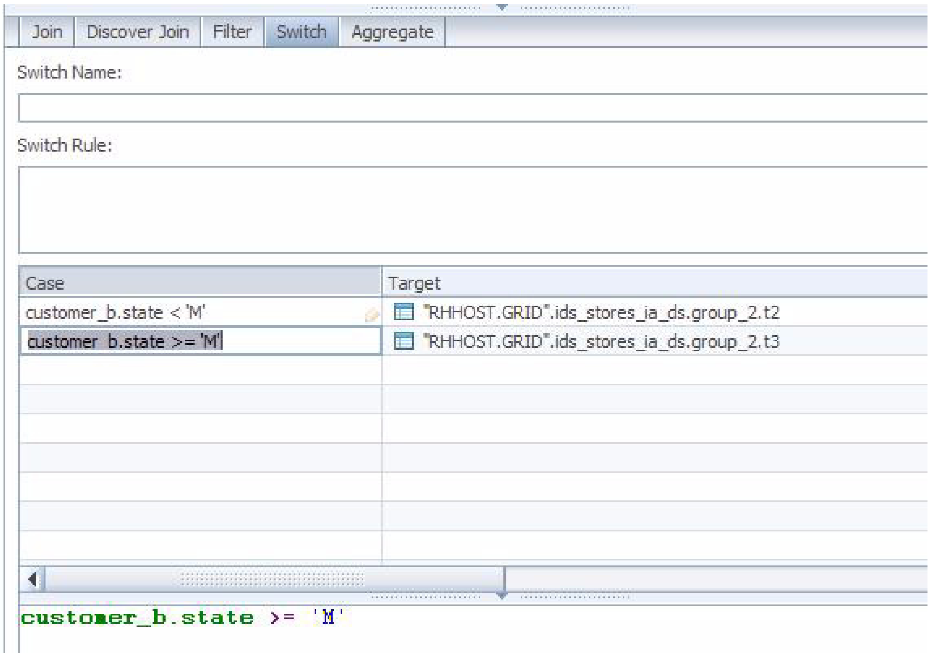
	Source	Target	T
	Field	Field	R
1	customer_b.customer_num	t2.col01	
2	customer_b.state	t2.col02	
3	customer_b.lname, customer_b.fname	t2.col03	
4		t2.col04	
5	valid_state_b.st_name (LookupDefinition1)	t2.col05	
6	customer_b.customer_num	t3.col01	
7	customer_b.state	t3.col02	
8	customer_b.lname, customer_b.fname	t3.col03	
9		t3.col04	
10	valid_state_b.st_name (LookupDefinition1)	t3.col05	
11			

Figure 33-16 Mapping Specification -> Basic -> Mappings panel.

f. Move to the Advanced -> Properties panel, Switch TAB.

Complete the user interface as displayed in Figure 33-17.

Here we create a condition where output rows (Target Rows) go to 1 of 2 output tables. While we call for 2 tables, you could have several, and with different output columns.



Switch Name:

Switch Rule:

Case	Target
customer_b.state < 'M'	"RHHOST.GRID".ids_stores_ia_ds.group_2.t2
customer_b.state >= 'M'	"RHHOST.GRID".ids_stores_ia_ds.group_2.t3

```
customer_b.state >= 'M'
```

Figure 33-17 Specifying the Switch Criteria.

g. Complete the steps to Generate this Parallel Job.

Example output is displayed in Figure 33-18.

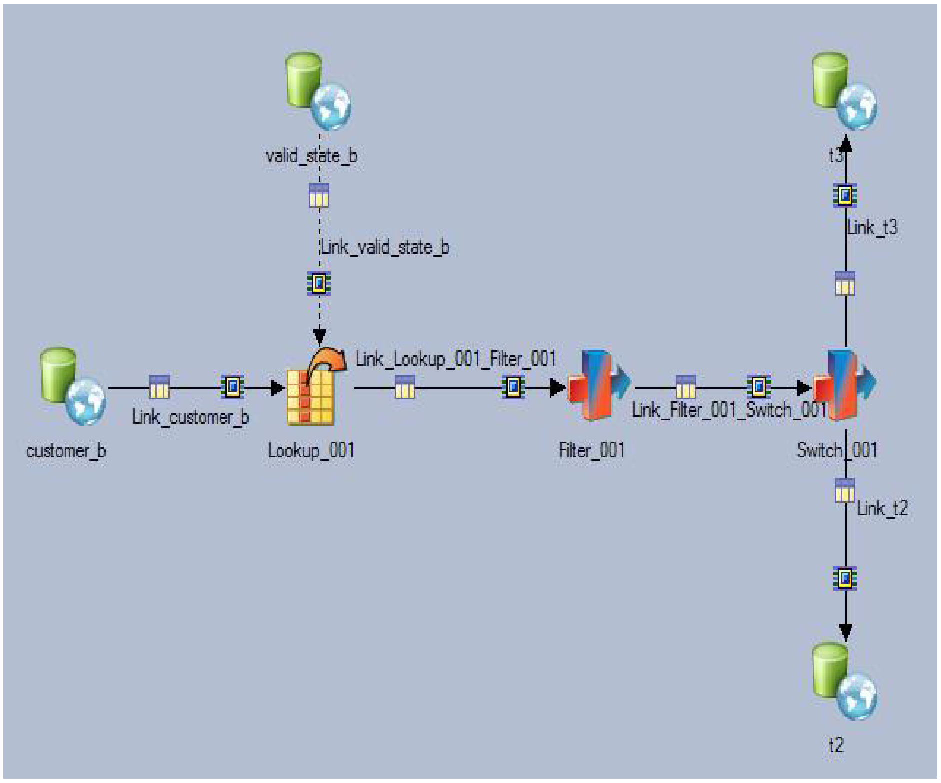


Figure 33-18   Generated Job with Switch included.

Examples-6, Source Merge (our word)

Thus far we have created a new Mapping Specification that contains;

- Filters
- Lookups
- Functions Expressions, creating 2 new Target (output) Columns
- And a Switch, writing to numerous Target tables

Now we wish to create a Source Merge (our word); a use case where we read, for example, Customer data from 2 or more Source tables and write to a single, consolidated Target table.

At this point we are going to start a new 2 new Mapping Specifications; Source Merge doesn't require this, however, we have enough happening in our prior example that it may be too wordy for a beginning user.

When you complete the Source Merge example below, go back and apply it to the example we have been working on above.

6. Create the Source Merge example.
    a. Create a New Mapping Specification.
    b. Read 2 Source Columns from a given table and write to a given Target Table. (Much like the first Mapping Specification we created above.)
    c. Save and Close this Mapping Specification.
    d. Create another new Mapping Specification.
        i. Have this Mapping Specification read (Source) from a table that is different than the Mapping Specification above.
        ii. Have it write to the same 2 Target Columns in the same Target Table as above.
        iii. Save this Mapping Specification, do not exit.
    e. Still in the second Mapping Specification, Click, Generate Job.

       Generating a Job has 4 Steps, and this is the first time we are using/configuring Step-1.
    f. In the table entitled, Mapping Specification in Job, Click the Add Button.

       Select the Mapping Specification we defined in Steps 6.a-c above.

       And Click OK.
    g. For the Radio Button entitled, Composition Type, Select, Parallel Composition.
    h. Click Next, and complete the Job Generation as before.

       Example output is displayed in Figure 33-19.

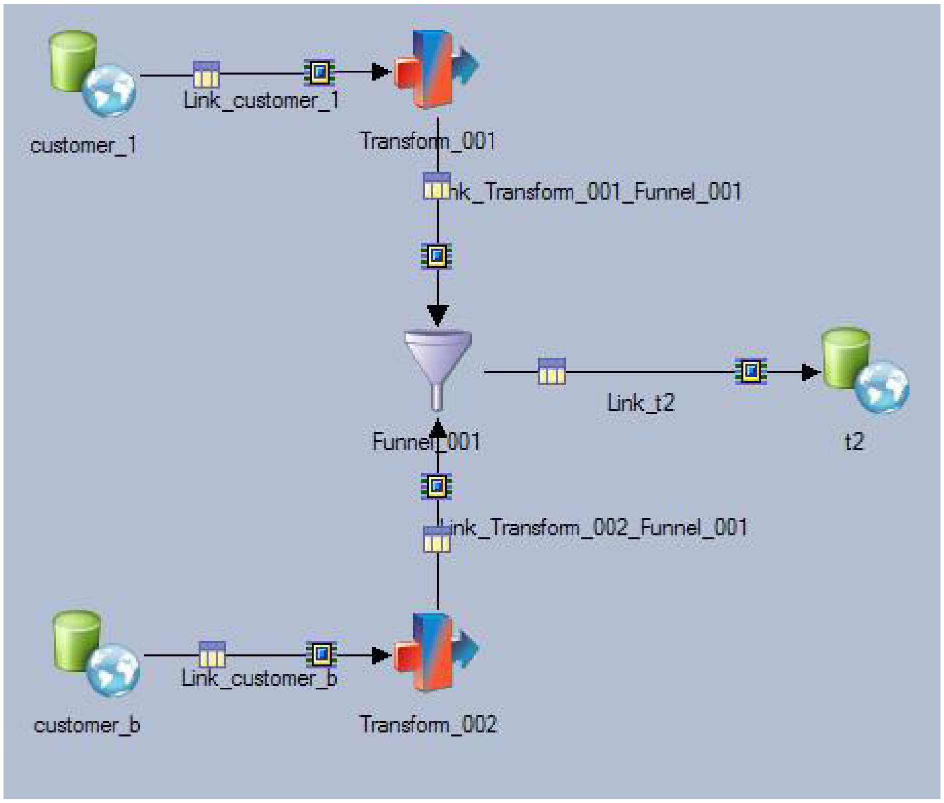


Figure 33-19 Combining 2 Mapping Specifications in Job Generation; Source Merge

At this point you have created Mapping Specifications for many standard data mapping requirements; Filter, Lookup, Function Expressions, Switches, Source Merge, and probably more.

If you wish, go back and start one new example, with all of the techniques displayed, against a data set that matches those you might find in your work.

33.3 In this document, we reviewed or created:

In this document we detailed configuration and use of the FastTrack component to the Information Server Platform. For configuration we relied on many prior editions of this IISDN document, including those on ODBC configuration, and then Data Connection and Data Store definition as is largely similar with the Information Analyzer component.

We spent the bulk of our time on Mapping Specifications, including; Filters, Lookups, Function Expressions, Switches, and Source Merge.

For all of the features of FastTrack we did detail, we left about 4 times as many unexplained. In a future edition of IISDN, we will seek to repair that, as well as cover greater integration with other component to the Information Server Platform.

Persons who help this month.

Martin Klumpp, Harald Smith, and Stefan Eberl.

Additional resources:

Legal statements:

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating trademarks that were owned by IBM at the time this information was published. A complete and current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

Other company, product or service names may be trademarks or service marks of others.

Special attributions:

The listed trademarks of the following companies require marking and attribution:

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Microsoft trademark guidelines

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Intel trademark information

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

Other company, product, or service names may be trademarks or service marks of others.