Chapter 25.

January 2009

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

> How do I reference external logical and/or physical data models within the Information Server components of DataStage, and InfoSphere Data Architect?
>
> *Excellent question! This edition of (IBM) InfoSphere Information Server Developer's Notebook (IISDN) is our first to address metadata from external sources, and we'll address this topic from both the source and target perspectives.*
>
> In the examples put forth in this document, we introduce the MetaBroker and Bridges component to Information Server, build a table inside InfoSphere Data Architect, read table definitions from a live database server, read table definitions from a logical data model, and more.

Software versions

All of these solutions were *developed and tested* on (IBM) InfoSphere Information Server (IIS) version 8.1, using the Microsoft Windows XP/SP2 platform to support IIS client programs, and a RedHat Linux Advanced Server 4 (RHEL 4) FixPak U6 32 bit SMP server (Linux kernel version 2.6.9-67.EL-smp) to support the IIS server side components.

IBM InfoSphere Information Server 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 product contains the following major components;

> WebSphere Business Glossary Anywhere™, WebSphere Information Analyzer™, WebSphere Information Services Director™, WebSphere DataStage™, WebSphere QualityStage™, WebSphere Metadata Server and Metabridges™, WebSphere Metadata Workbench™, WebSphere Federation Server™, Classic Federation™, Event Publisher™, Replication Server™, InfoSphere Data Architect™, DataMirror Transformation Server™, and others.

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

25.1 Terms and core concepts

As stated above, (IBM) InfoSphere Information Server Developer's Notebook (IISDN) expects and documents a two-tier Client/Server installation of Information Server; server side components operating on RedHat Linux Advanced Server, and client side components operating on Windows/XP. With the current version 8.1 release of Information Server, the client side components are installed from a binary file entitled;

InfoSvr_Client_v8.1_Win.zip

The following is also offered;

- Wherever you unzip the file listed above, a subdirectory entitled,

  client_IA-suite

  is created with the Information Server installation programs and components.

- Underneath the client_IA-suite subdirectory, is the main installer for the Information Server client side programs, entitled;

  install.exe

  There are also two additional subdirectories in the client_IA-suite subdirectory entitled;

  BusinessGlossaryAnywhereClient

  MetaBrokersAndBridges

  These two subdirectories contain additional installation programs that may optionally be run on the client side. The MetaBrokersAndBridges subdirectory contains an installation program entitled,

  Setup.exe

  where Setup.exe is a program that should be run to complete the examples in this document.

  Figure 25-1 displays the Menu Bar of the DataStage/QualityStage Designer program. If the Menu Item entitled, Import -> Via Bridges, is greyed out, then you haven't completed the install of the MetaBrokers And Bridges component to Information Server.

Note: Again, the above step is required in order to complete the examples in this document.

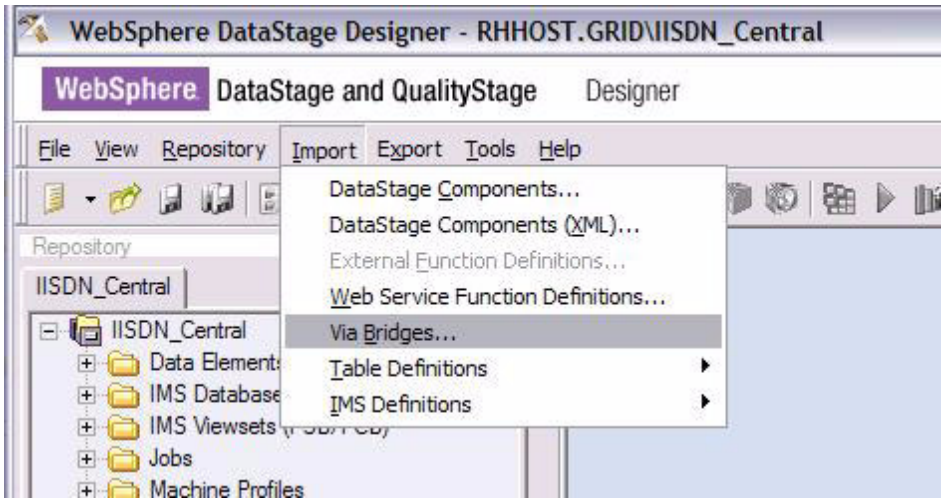

Figure 25-1 DataStage Designer menu item indicating MetaBrokers and Bridges.

The following table lists the various MetaBrokers and Bridges currently included inside InfoSphere Information Server v8.1;

Bridges for import, Bridge name and version, Notes:

Adaptive Repository & Foundation (via CWM XMI)

version 4.xPhysical schemas only

ASG Rochade (via CWM XMI)

versions 6.x to 7.xPhysical schemas only

Business Objects Data Integrator (via CWM XMI)

versions 6.x to 11.xPhysical schemas only

Business Objects Designer (File)

versions 5.x to 11.xBusiness intelligence models and physical schemas

Business Objects Designer (Repository)

versions 5.x to 11.x Business intelligence models and physical schemas

Business Objects Desktop Intelligence (File)

versions 5.x to 11.x Business intelligence reports

Business Objects Desktop Intelligence (Repository)

versions 5.x to 11.xBusiness intelligence reports

Business Objects Repository

version 11.xBusiness intelligence reports, business intelligence models, and physical schemas

Business Objects Web Intelligence

version 11.xBusiness intelligence reports

CA Erwin Data Modeler

version 4.xPhysical schemas only

CA Erwin Data Modeler (File)

version 7.xPhysical schemas only

Categories and Terms MetaBrokerBusiness categories and terms only

Cognos 8 BI reporting-Content Manager

versions RN 8.1.* and C8 v8.2 to 8.3Cognos 8 BI Reporting Content Manager functionality in Cognos ReportNet and Cognos versions 8.2 to 8.3. Business intelligence models and reports.

Cognos 8 BI reporting-Content Manager Packages

versions RN 8.1.* and C8 v8.2 to 8.3Cognos 8 BI Reporting Content Manager functionality in Cognos ReportNet and Cognos versions 8.2 to 8.3. Business intelligence models and physical schemas.

Cognos 8 BI reporting-Content Manager QueryStudio

versions RN 8.1.* and C8 v8.2 to 8.3Cognos 8 BI Reporting Content Manager functionality in Cognos ReportNet and Cognos versions 8.2 to 8.3. Business intelligence reports.

Cognos 8 BI reporting-Content Manager ReportStudio

versions RN 8.1.* and C8 v8.2 to 8.3Cognos 8 BI Reporting Content Manager functionality in Cognos ReportNet and Cognos versions 8.2 to 8.3. Business intelligence reports.

Embarcadero ER/Studio

version 1.5Physical schemas only

Embarcadero ER/Studio (via DM1)

versions 5.1.0 to 7.xPhysical schemas only

IBM DB2 Data Warehouse Edition/OLAP Acceleration (Cube Views)

version 9.xBusiness intelligence models and physical schemas

IBM DB2 Warehouse Manager (via CWM XMI)

version 7.xPhysical schemas only

IBM Rational Data Architect MetaBrokerIBM Rational Data Architect version 7.0.0.0 through 7.0.0.4, or version 6 Fix Pack 1. Physical schemas or business categories and terms.

Note: Rational Data Architect (RDA) was recently rebranded (renamed) InfoSphere Data Architect (IDA), to better reflect this software product's relationship to IBM Information Management, and the InfoSphere area within same.

The most current release of IDA is v7.5. Information Server v8.1 supports v7.5 of IDA via a PMR/Fix-Pak numbered; 42353,090,616

InfoSphere Data Architect (IDA) shares many components and abilities with Rational Software Architect, Rational Application Developer, and many of the Eclipse based IBM developer's tools. The single importance here being that documentation on any of these specific products is normally useful for IDA.

The IBM RedBook SG-247274, entitled, *Leveraging DB2 Data Warehouse Edition for Business Intelligence*, Chapter 4, details use of DB2 Data Warehouse Edition Design Studio, which offers a subset of the functionality from InfoSphere Data Architect. We reference that chapter in many of the sections that follow.

See, http://www.Redbooks.IBM.com

Microsoft SQL Server Data Source View

versions 7.0 to 9.0 (2005)Physical schemas only

MicroStrategy

versions 7.0 to 8.1Business intelligence models and physical schemas

ODBC 3.0 MetaBrokerPhysical schemas only

OMG CWM 1.x XMI 1.x

versions 1.0 to 1.1Physical schemas only

Oracle Hyperion Application Builder (via CWM XMI)Physical schemas only

Oracle Warehouse Builder (via CWM XMI)

versions 10g to 11gPhysical schemas only

SAS Data Integration Studio (via CWM XMI)

version 9.xPhysical schemas only

Sybase PowerDesigner PDM

version 6.1.xPDM files created by any version of Sybase PowerDesigner DataArchitect 6.1.

Sybase PowerDesigner PDM

versions 7.5 to 12.xPDM files created by Sybase PowerDesigner versions 7.5 through 12.

User Information MetaBrokerNon-security related users for stewardship only.

Bridges for export, Bridge name and version, Notes;

Adaptive Repository & Foundation (via CWM XMI)Physical schemas only

Adaptive Repository & Foundation (via MIR XMI)Physical schemas only

Altova XMLSpyPhysical schemas only

ASG Rochade (via CWM XMI)Physical schemas only

Business Objects Data Integrator (via CWM XMI)

versions 6.x to 11.xPhysical schemas only

Business Objects Metadata Manager (via MIR XMI)Physical schemas only

Categories and Terms MetaBrokerBusiness categories and terms only

IBM DB2 Warehouse Manager (via CWM XMI)Physical schemas only

IBM Rational Data Architect MetaBrokerBusiness categories and terms only

Informatica Metadata Manager (via MIR XMI)Physical schemas only

Meta Integration Repository (MIR) XMI filePhysical schemas only

OMG CWM 1.x XMI 1.xPhysical schemas only

Oracle Hyperion Application Builder (via CWM XMI)Physical schemas only

Oracle Warehouse Builder (via CWM XMI)Physical schemas only

RedHat (MetaMatrix) Enterprise Designer (via MIR XMI)Physical schemas only

SAS Data Integration Studio (via CWM XMI)Physical schemas only

SAS Data Integration Studio (via MIR XMI)Physical schemas only

SAS Information Map Studio (via MIR XMI)Physical schemas only

SAS Management Console (via MIR XMI)Physical schemas only

SchemaLogic SchemaServer (via MIR XMI)Physical schemas only

Sybase PowerDesigner XSM

versions 10.x to 12.xPhysical schemas only

Unisys ClearPath (via MIR XMI)Physical schemas only

User Information MetaBroker

W3C XML Schema 1.0 (XSD)Physical schemas only

The above list of supported MetaBroker and Bridges is currently maintained at the following URL;

http://www-01.ibm.com/support/docview.wss?rs=14&context=SSZJPZ&dc=DA400&uid=swg27012944&loc=en_US&cs=UTF-8&lang=en&rss=ct14db2

Further configuration of Information Server

In addition to the installation of the client side MetaBrokers and Bridges component to Information Server, a small amount of configuration needs to be done on the server side to enable the examples in this document. In the December/2007 edition of this document, IISDN, we detailed configuring ODBC drivers for use as a data source to a DataStage/QualityStage Job.

Given a default server side Information Server software installation directory of,

/opt/IBM/InformationServer/

the subdirectory,

Server/DSEngine/

contains the ASCII text configuration files,

.odbc.ini

uvodbc.config

You must edit these two files per the instructions contained in the December/2007 edition of IISDN.

Additionally, under the directory entitled,

/opt/IBM/InformationServer/Server/Projects/

will be an additional directory for each DataStage/QualityStage Project, Information Analyzer Project, and more. These subdirectory names will equal the DataStage/QualityStage, etcetera, Project name. In order for certain program wizards and menu items to work as you might expect, *you must also edit the* uvodbc.config *file under the Project subdirectory.*

Note: This is actually a feature, with the higher level directory uvodbc.config being a parent type file, and the same named file under the Project directory then further qualifying what external resources are allowed for a given, single Project.

These configuration files are edited in the same manner, syntax, other.

What is a metabroker or bridge anyway?

So far we've talked about metabrokers and bridges, metadata, and related, without really defining what these things are. The InfoSphere Information Server platform has a single, shared MetaData Repository; any insight, observation, or knowledge you gather about a given data source, data table, data process, cleansing or data standardization routine, or related, is shared across the entire componentry of Information Server. Third party (non-IBM) business applications, software packages, etcetera, generally offer limited (or even zero) API or means to share their intelligence (their metadata) with external sources. The Metabrokers and Bridges component to Information Server, works to repair that shortcoming, bringing third party metadata into a usable/re-usable and extensible server and tool set.

In many cases, the entity you wish to retrieve metadata from is not a server proper, having no process agent that could even respond to service requests. While IBM is working to improve the best case condition that exists today, many of the Metabrokers and Bridges are on demand, or unidirectional, basically push/pull technology. Generally you will see better cooperation within the IBM software portfolio as a whole, as this is a stated direction of IBM. However, many third party vendors have little or no economic motivation to open their application, and share metadata and related. Standards bodies help, and pressure from customers help too.

25.2 Complete the following examples

Assuming the configuration(s) listed above have been completed, the following examples are offered in this section;

- Making a data table definition in InfoSphere Data Architect.

  This definition is used later to demonstrate importing a data model (table definitions) into the InfoSphere Information Server, DataStage/QualityStage Designer program.
- Import the above table definition into the Information Server MetaData Repository using the DataStage/QualityStage Designer program.
- Import an existing SQL table definition into a DataStage/QualityStage Project, allowing it to variably enter the Information Server MetaData Repository.
- Import an existing SQL table definition into InfoSphere Data Architect.

Using InfoSphere Data Architect, make a (SQL) Table definition

Complete the following;

1. Launch the InfoSphere (Rational) Data Architect graphical design program.

   If prompted to select a Workspace, select the default or another directory.

Note: Remember this Workspace (directory) value, you will need it later.

A Workspace is an Eclipse concept, and as InfoSphere Data Architect is based on Eclipse, it inherits this and many other terms and ideas. A Workspace indicates the parent directory where your work will be located (saved). This parent directory will contain a subdirectory named, .metadata, with all of your personal settings (preferences and related), and an additional subdirectory for each Project you create.

Note: When introducing concepts that InfoSphere Data Architect inherits from Eclipse, we will only mention Eclipse, and not Eclipse and InfoSphere Data Architect both.

2. Create a Project.

   Everything you do in Eclipse must be contained inside a Project. A Project is a logical term, and contains assets, source files, etcetera, *and the instructions* as to how one compiles, deploys, etcetera, the contents of the Project.

   a. From the Menu Bar, select, File -> New -> Project -> Data -> Data Design Project.

   b. Click Next.

   c. In the text entry field entitled, [Project name], enter a Project name.

   d. Click Finish.

   e. This action will create the Project, and prompt to change the Perspective.

      Click Yes.

      Eclipse is best be termed (identified as), a developer's workbench. The various distinct visual areas of the Workbench are called Views. (By example: HTML calls these same entities, Frames.) There are so many Views in Eclipse, that they are organized into collections called, Perspectives. Generally a Perspective is associated with a given task or purpose, and offers Views with the program capabilities to accomplish same.

You can change Perspectives at any time, or make your own custom Perspectives, just as you can open, close, move, or resize Views as you desire.

3. Create a new Physical Data Model.

   Our end goal is to produce a Table definition, that we may import into the InfoSphere Information Server MetaData Repository. To do that, we must first create a Physical Data Model, and a few more (SQL) related items.

   a. From the Data Project Explorer View, Right-click the entry entitled, Data Diagrams, and select, New -> Physical Data Model.

      Example as displayed in Figure 25-2.

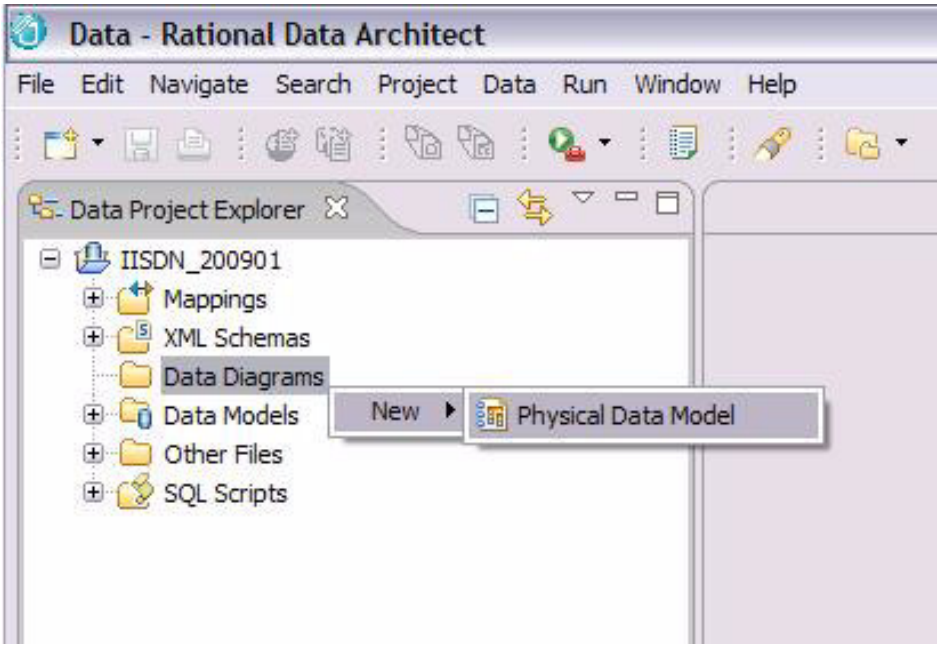

Figure 25-2 Creating a new Physical Data Model.

   b. This action will produce the New Physical Data Model dialog box, as displayed in Figure 25-3.

      From the Database (type) Drop Down List Box, select a database type.

      We selected our favorite database type, Informix.

   c. Then Click Finish.

      This action will close the New Physical Data Model dialog box, and produce a new Database entry in the Data Project Explorer View.

Much like Microsoft Windows Explorer, or Internet Explorer, Data Project Explorer allows you to browse through, manage, etcetera, the various components your Project now contains, only faster and without crashing.

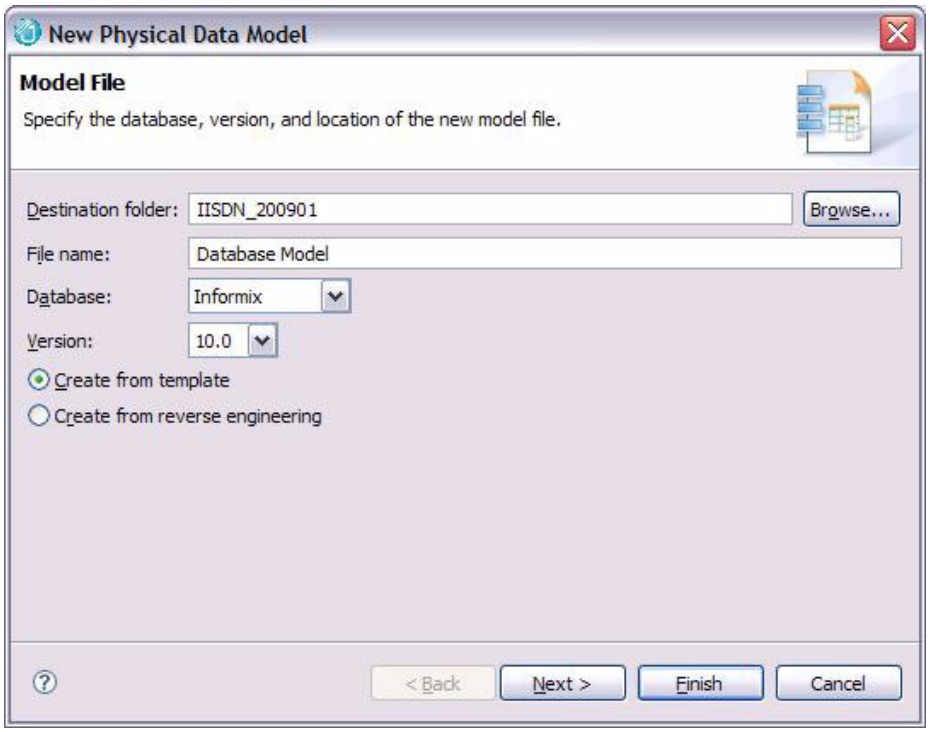

Figure 25-3 New Physical Data Model dialog box.

4. Rename the Database, create a Schema.
   a. In the Data Project Explorer View, Single-click the newly create database entry entitled, New Database.

      This action gives this object focus, making it the currently displayed object in the Properties View below.

   b. In the Properties View, rename this Database. We called ours, MyDatabase.

      Simply type in the Properties View, text entry field entitled, Name. Then TAB out of this control.

   c. In the Data Project Explorer View, Right-click the Database object entry entitled, MyDatabase, and select, Add Data Object -> Schema.

      This action will make a new Schema in this Database entitled, Schema1.

d. In the Data Project Explorer View, Single-click Schema1 to make it current.

e. In the Properties View, rename Schema1 to equal MySchema.

5. Make a new Table definition.

a. In the Data Project Explorer View, Right-click MySchema, and select, Add Data Object -> Table.

b. In the Properties View, use the General TAB, and the Columns TAB to make a Table definition of your choosing.

Our new Table definition is displayed in Figure 25-4.

(You'll need to use the New button to add columns, as shown with balloon help in Figure 25-4.)

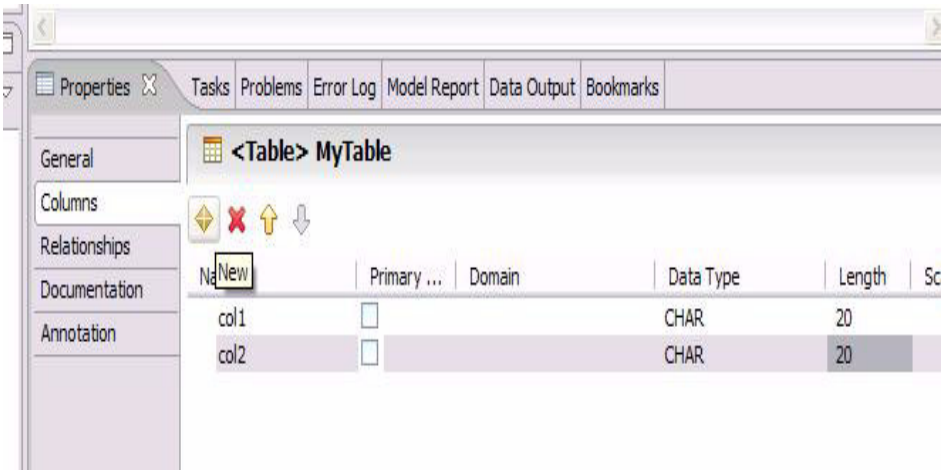

Figure 25-4 Properties View displaying our new Table definition.

Note: Tables and most other objects can be defined graphically in the Palette View, or less graphically via the Properties View. We are using the Properties View because it is faster and easier to describe for our simple task at hand.

6. Save your work.

From the Menu Bar, select, File -> Save All.

Congratulations, you just created a logical data model including one Schema and one Table definition. This Table has not been instantiated anywhere; it

exists only as a model ready for deployment to database servers of various types and conditions.

Note: Want to learn how to do more inside InfoSphere Data Architect?

We used pages 57-61 of the IBM RedBook SG-247274, entitled, *Leveraging DB2 Data Warehouse Edition for Business Intelligence*.

These pages detail use of DB2 Data Warehouse Edition Design Studio, which offers a subset of the functionality from InfoSphere Data Architect. See, http://www.Redbooks.IBM.com

Many other books and Redbooks cover this topic, this was just the first free (no cost) book we found.

Import the Table definition into Information Server (DataStage)

Now we call to import the Table definition made above in InfoSphere Data Architect, into InfoSphere Information Server. Complete the following;

7. Launch the DataStage/QualityStage graphical Designer program.
   a. From the Menu Bar, select, Import -> Via Bridges.

      Example as displayed in Figure 25-1.

      This action produces the (MetaBrokers and) Bridge(s) Selection dialog box, as displayed inFigure 25-5.

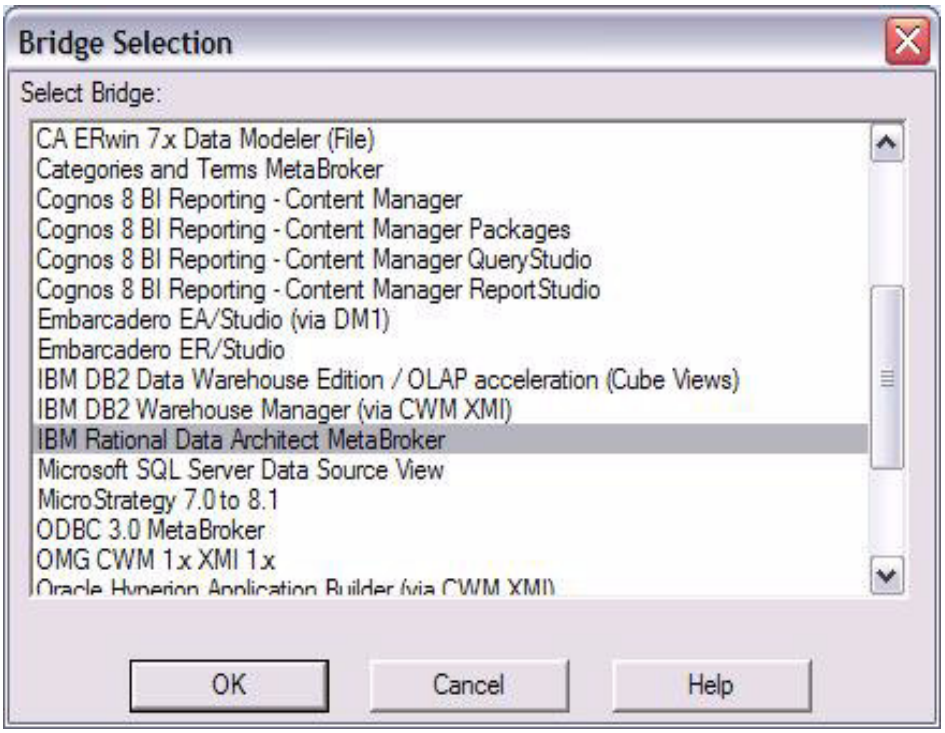

Figure 25-5 (MetaBrokers and) Bridges Selection dialog box.

b. Highlight the entry entitled, IBM Rational Data Architect MetaBroker, and Click OK.

   This action produces the Parameter Selection dialog box, as displayed in Figure 25-6.

   Use the Browse icon to the right of the visual control entitled, DBM or NDM file. You will need to Browse under the Workspace Directory you specified in Step-1 above. (Under the Workspace directory, then under Project Name, then a file entitled, Database Model.dbm.)

   Click OK.

.

Note: Our default entry for the visual control entitled, Eclipse Path, was accurate and worked fine.

But we were testing at version 7.0.0.5 of InfoSphere Data Architect (IDA). Other folks have reported grief with this setting and its behavior on earlier versions of IDA. See the following URL for more details,

http://www-01.ibm.com/support/docview.wss?rs=3045&context=SSSRXA&dc=DB560&dc=DB520&uid=swg21318217&loc=en_US&cs=UTF-8&lang=en&rss=ct3045db2

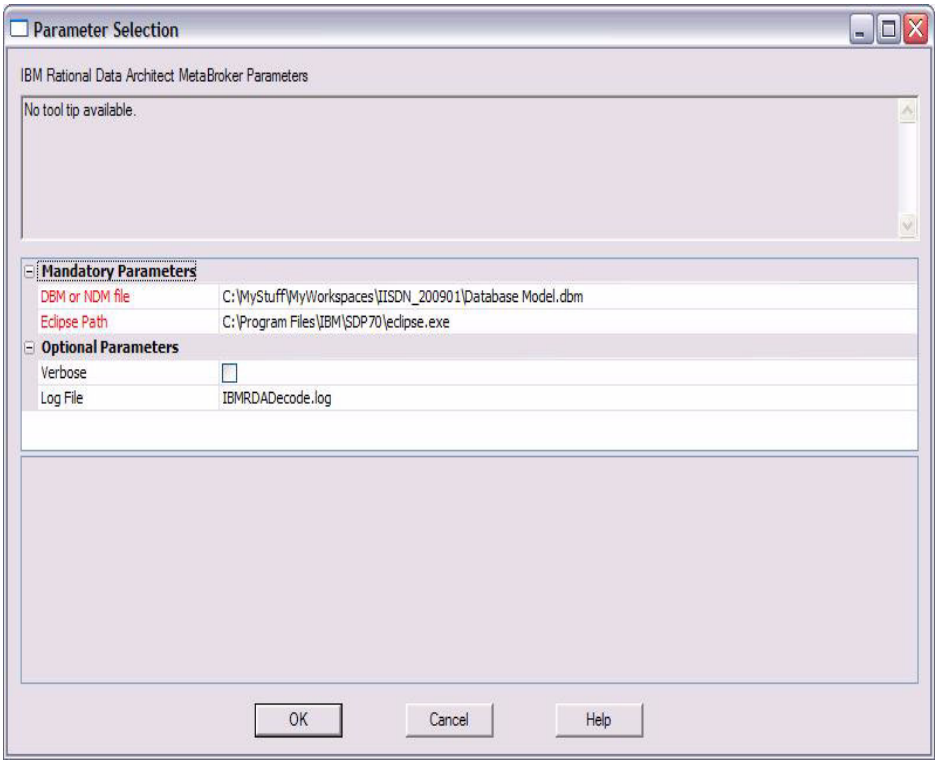

Figure 25-6 Parameter Selection dialog box.

c. This action produces the Status dialog box as displayed in Figure 25-7. Click [Select All] to continue.

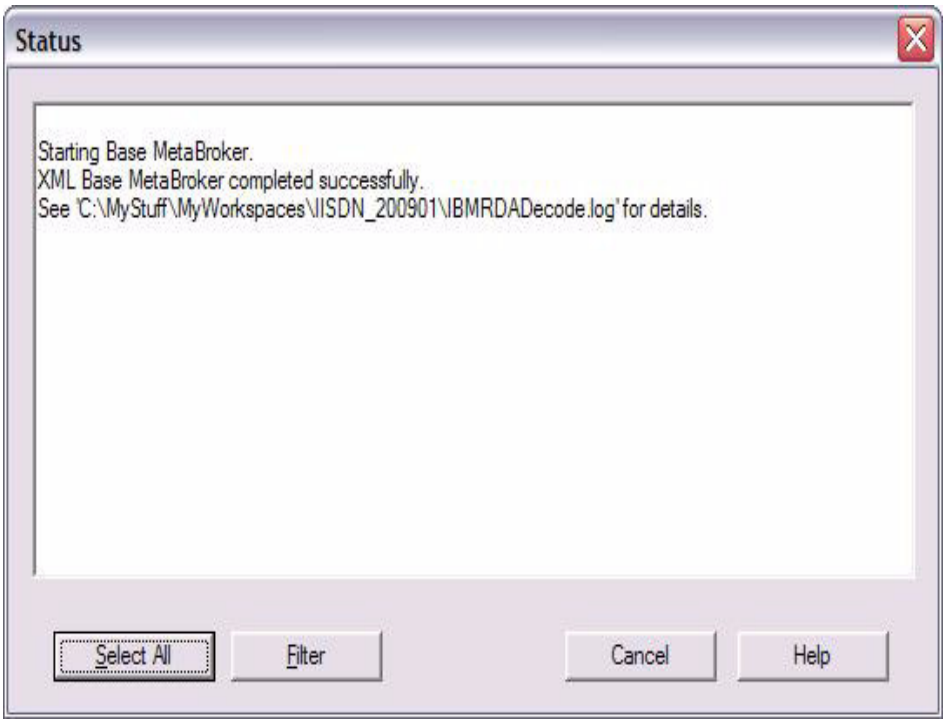

Figure 25-7 Status dialog box.

d. This action produces a further Status dialog box, as displayed in Figure 25-8.

   Click Finish.

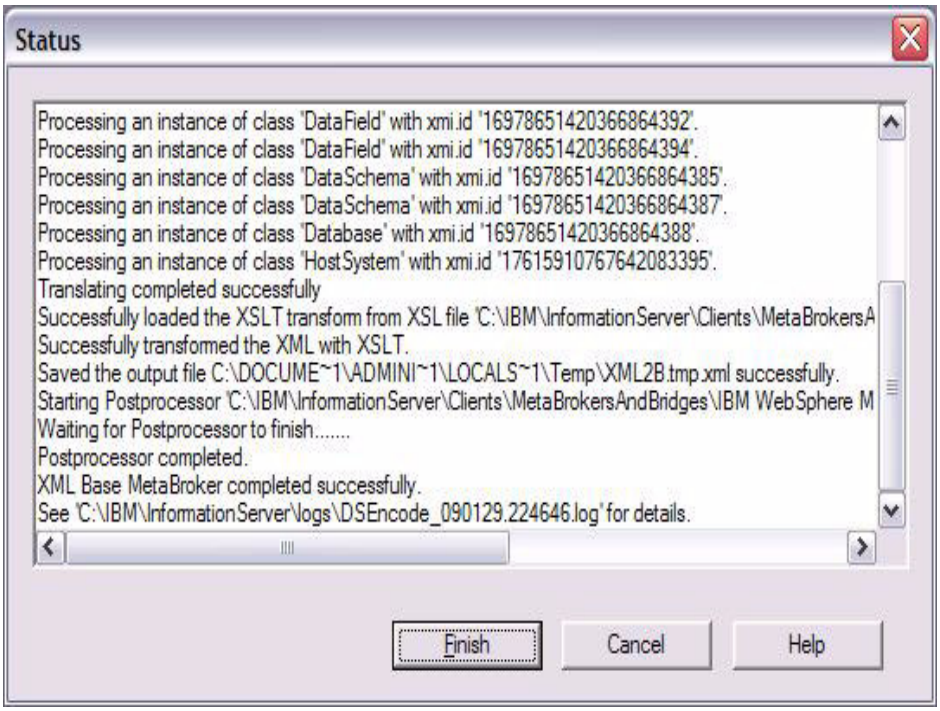

Figure 25-8 Status dialog, part II.

e. The above action prompts you for a location to save your Table definition. Example as displayed in Figure 25-9.

   Select a destination folder, and Click OK.

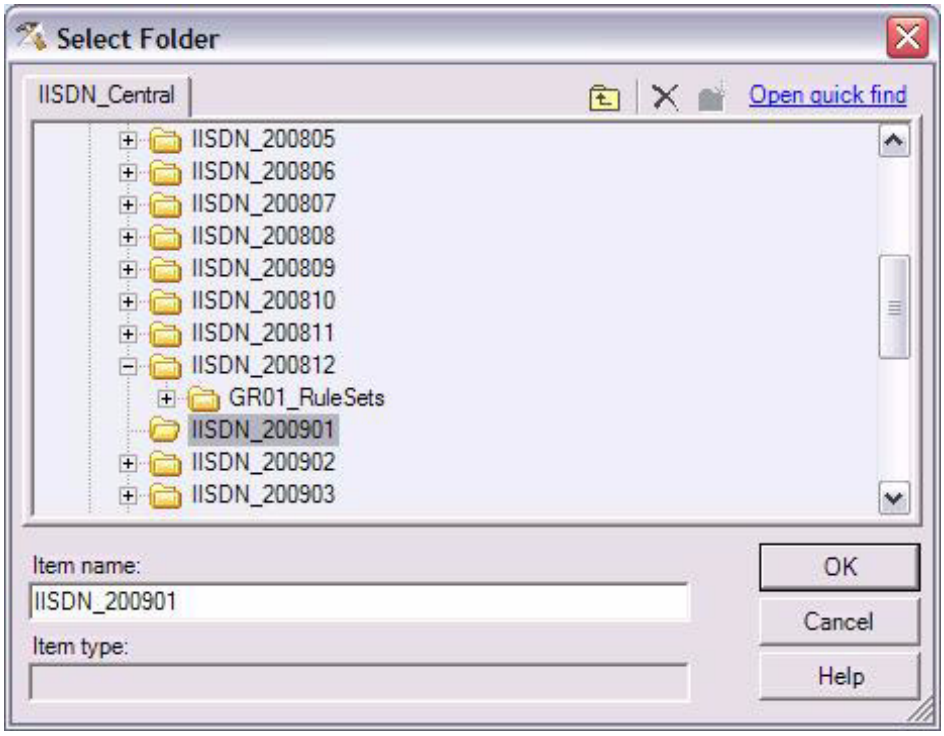

Figure 25-9 Select Folder dialog box.

f. The above action creates your new (Shared) Table definition. Example as displayed in Figure 25-10.

Note: DataStage/QualityStage offer two primary types of Table definitions; (Local) or Shared.

Local table definitions exist only within the scope of the current DataStage/QualityStage Project. Shared table definitions exist in the Information Server MetaData Repository, and are available to all components of Information Server.

Local Tables may be converted to Shared Tables in the Repository View of the Designer program, via a Right-click, Shared Table Creation Wizard.

Shared Tables may be converted to Local Tables for editing, and then returned as Shared Tables.

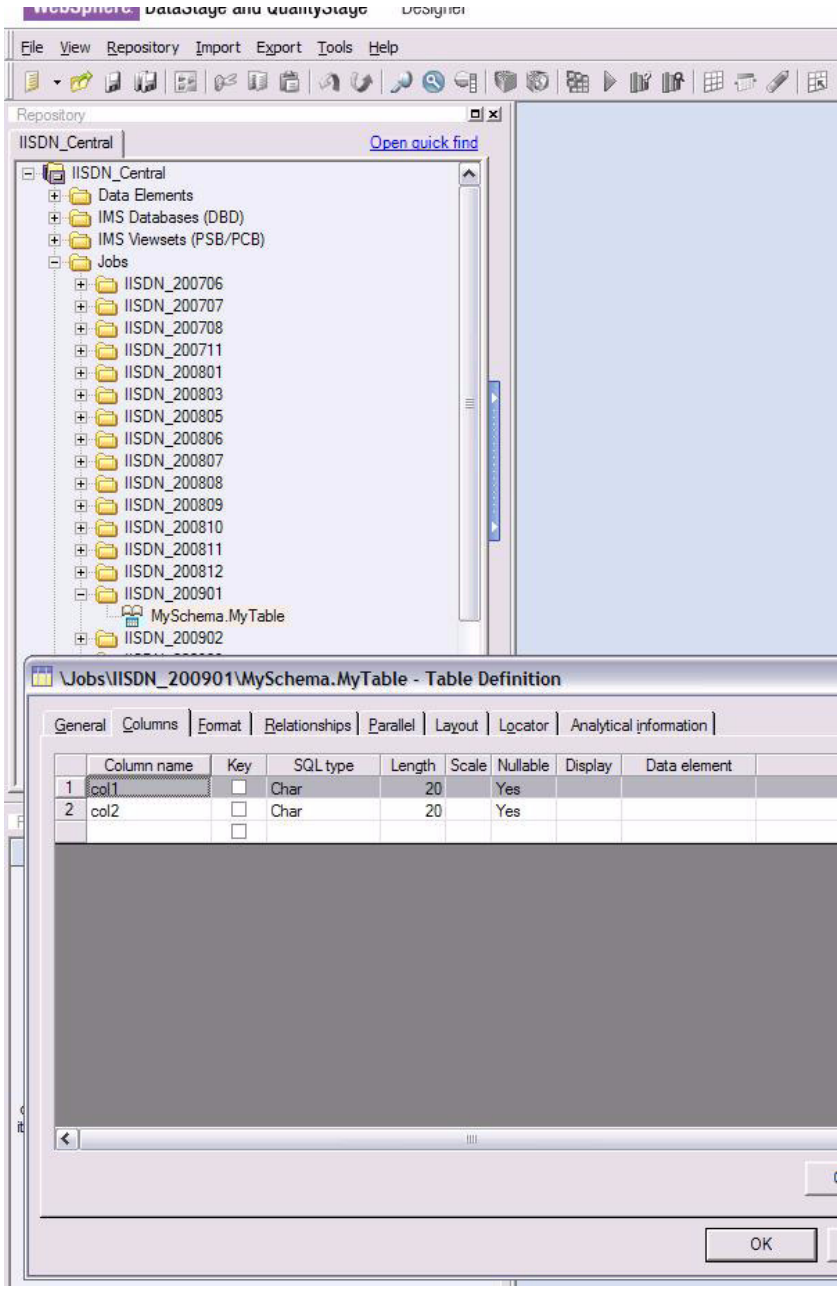

Figure 25-10 DataStage/QualityStage Repository View of new Shared Table definition.

Figure 25-10 also displays the Table Definition dialog box.

A Double-click on the numeric/left column (see red arrow) allows access to further and deep column level properties which novice users may not be aware of. Example as displayed in Figure 25-11.

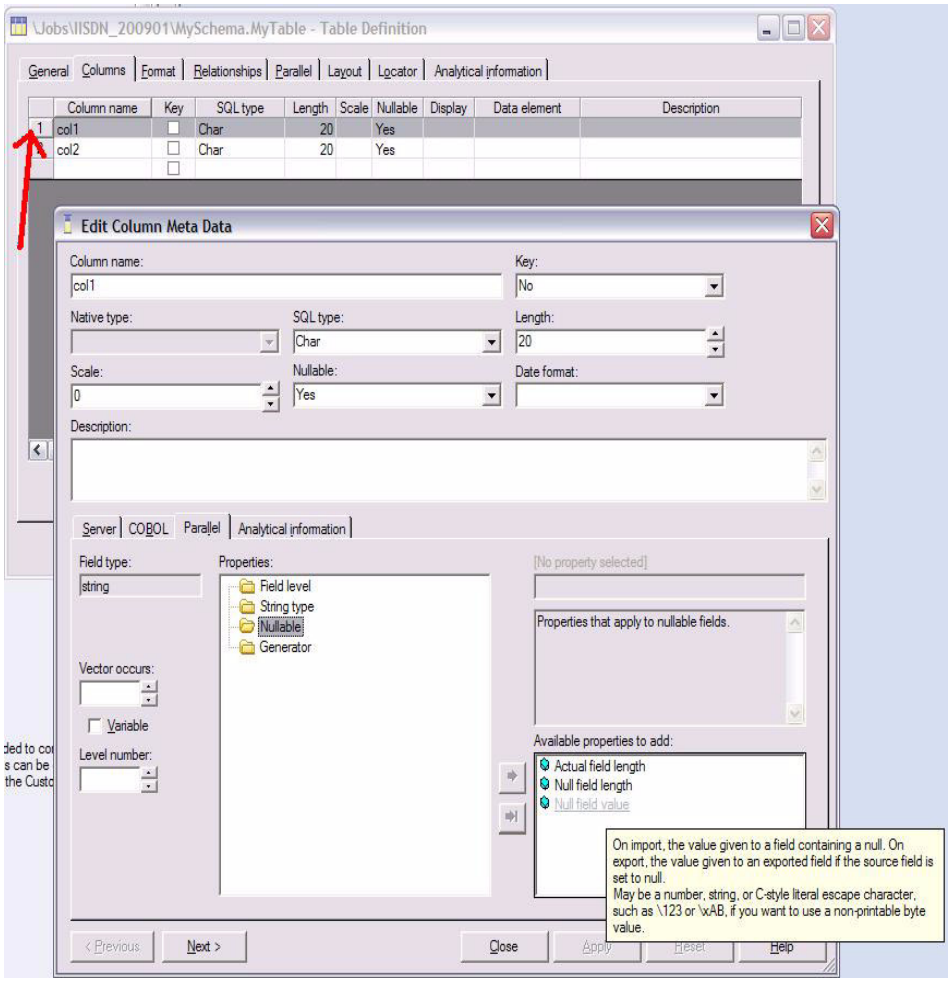

Figure 25-11 Deep column level properties, Table Definition dialog box.

g. As a further set of tests or instructions, look at the example as displayed in Figure 25-12.

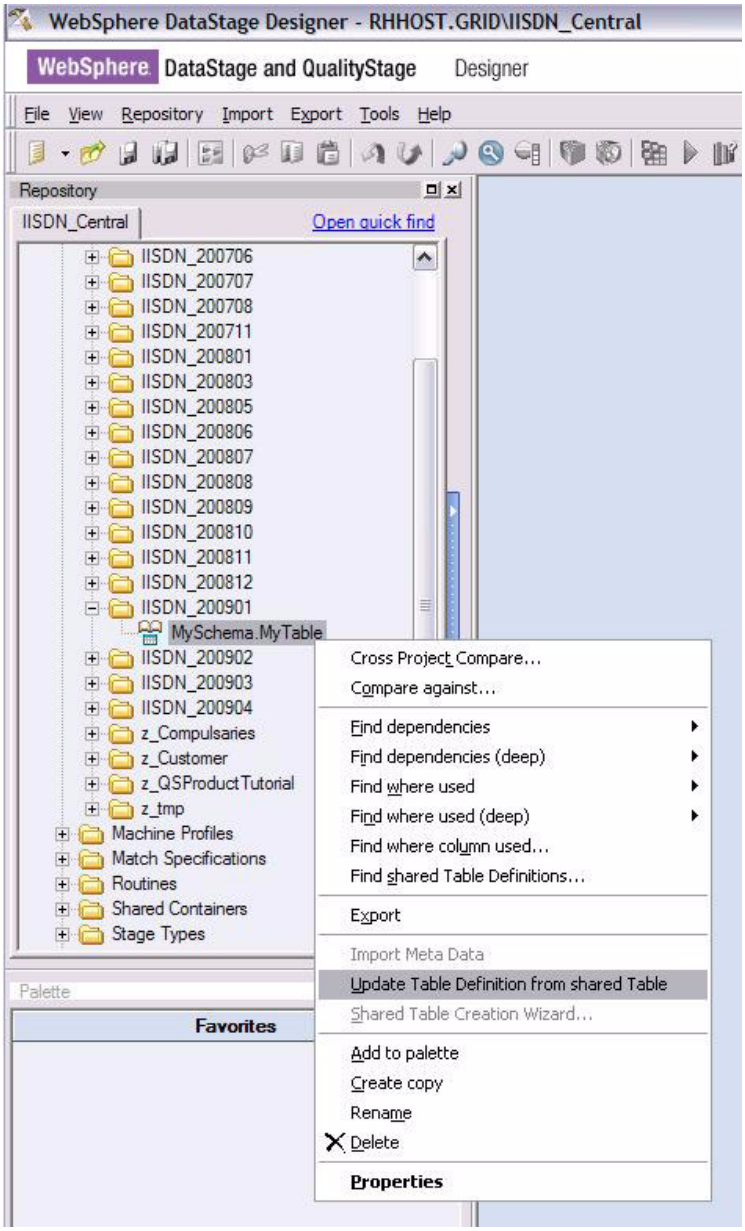

Figure 25-12 Updating Shared Table definition from Information Server Repository.

The menu option displayed in Figure 25-12, updates the Shared Table definition from the Information Server MetaData Repository. For example, if a Business Glossary user made comments or corrections, etcetera.

If you wished to update the Shared Table definition from any work done inside InfoSphere Data Architect (IDA), currently you'd have to re-run Steps-7,a-e above. And you will receive a prompt ensuring whether you wish to over-write your previously saved definition.

Import an existing SQL Table into (DataStage)

Above we imported a Table definition from a modelling tool into the Information Server MetaData Repository. While we used IBM InfoSphere Data Architect (IDA), we could just as easily used CA Er/Win, or pretty much anything else.

In this section, we import a table definition directly from a SQL database, a (SQL) Table that has already been instantiated. Using the DataStage/QualityStage Designer program, we import this definition first into our DataStage/QualityStage Project as a local Table definition, giving us the option to promote this Table definition into the Information Server MetaData Repository or not.

Complete the following;

8. Launch the DataStage/QualityStage graphical Designer program.
   a. From the Menu Bar, select, Import -> Table Definitions -> ODBC Table Definitions.

      This action will produce the Import Meta Data (ODBC) dialog box as displayed in Figure 25-13.

      Manage the display in Figure 25-13 to point to any existing and configured ODBC data source you may have, and Click OK.

Figure 25-13 Import Meta Data (ODBC) dialog box.

Note: What entries are available in the DSN Drop Down List Box displayed in Figure 25-13?

Whatever entries you have in the uvodbc.config file in the Project directory for this given Project; not the parent directory, but the given Project subdirectory.

b. The above action produces the Import Meta Data (ODBC) dialog box as displayed in Figure 25-14.

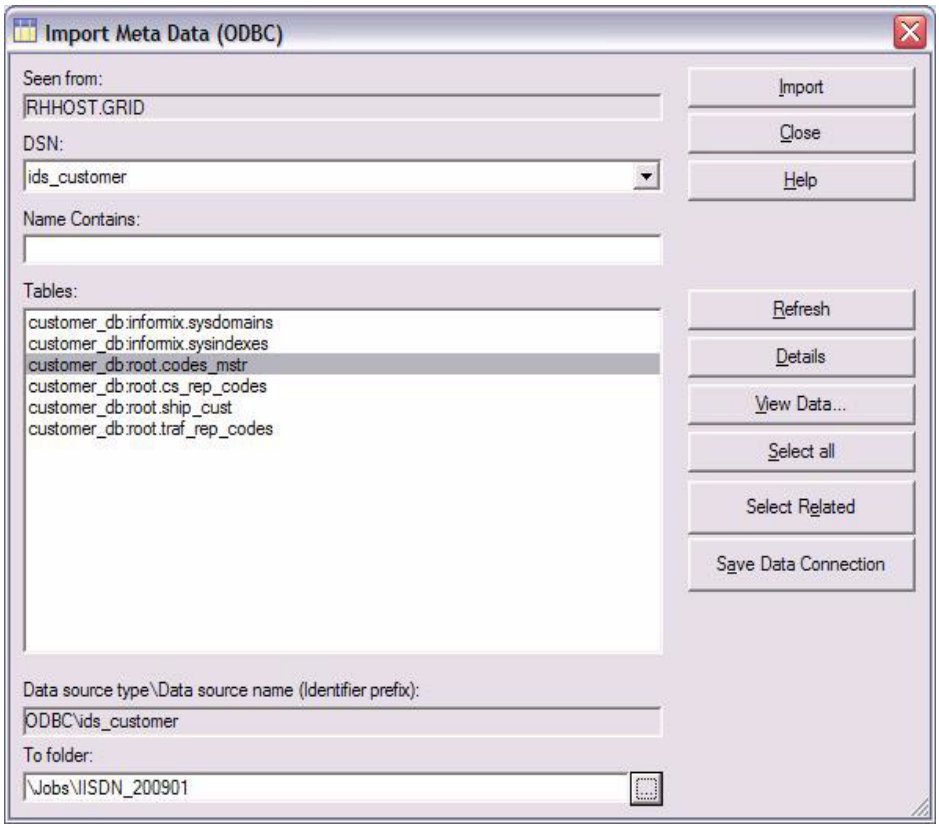

Figure 25-14 Import Meta Data (ODBC) dialog box II.

In Figure 25-14, select one or more Table definitions, and/or Browse to a new, [To Folder], and then Click Import.

This action will create the newly selected Table definitions in the folder of your choosing in the Repository View.

Example as shown in Figure 25-15.

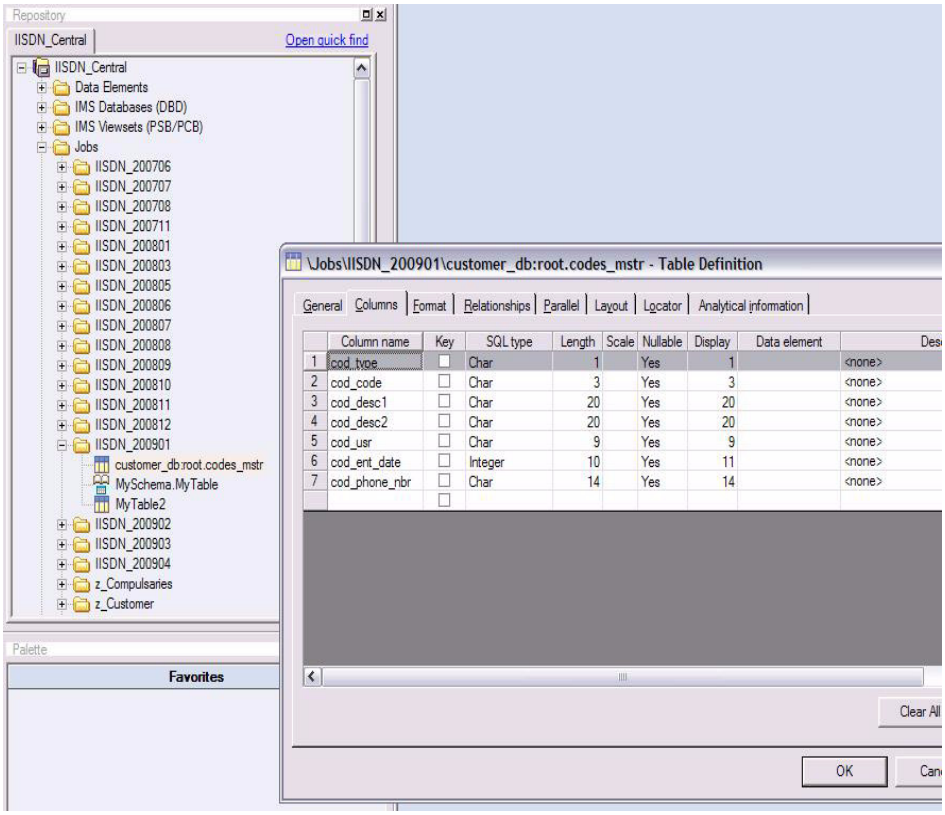

Figure 25-15 Completed result.

Import an existing SQL Table into InfoSphere Data Architect

As this customer question was originally asked, we wanted to read SQL Table definitions from tables resident in Microsoft/Access, into InfoSphere Data Architect. Microsoft/Access is a nice tool, but its not a database server; there is no program entity listening for new service requests. Just as the Microsoft Excel spreadsheet program operates against XLS files, Microsoft Access Forms and Reports read Microsoft/Access tables. Its not a database server pre se.

In order to read Microsoft/Access tables in this manner, some entity needs to acts as a server process, as a listening (new connection) daemon. We've heard of third-party (non-Microsoft) software of this type, but haven't used it. We are

also vaguely aware that Microsoft has some sort of SQL/Server to MS/Access bridge, but haven't used that either.

Lastly, be aware the InfoSphere Data Architect is a Java based tool; you need a JDBC driver to talk to any relational database source.

So, what we are left with that fits within the scope of this document, is to import (SQL) Table definitions from a SQL DDL file; the task we complete next.

Note: The IBM Redbook we've mentioned, SG-247274, entitled, *Leveraging DB2 Data Warehouse Edition for Business Intelligence,* covers this subtopic on pages 61-65 in great detail.

Complete the following;

9. Using your favorite editor, create a SQL DDL Command file. The example we use moving forward is displayed in Figure 25-16.

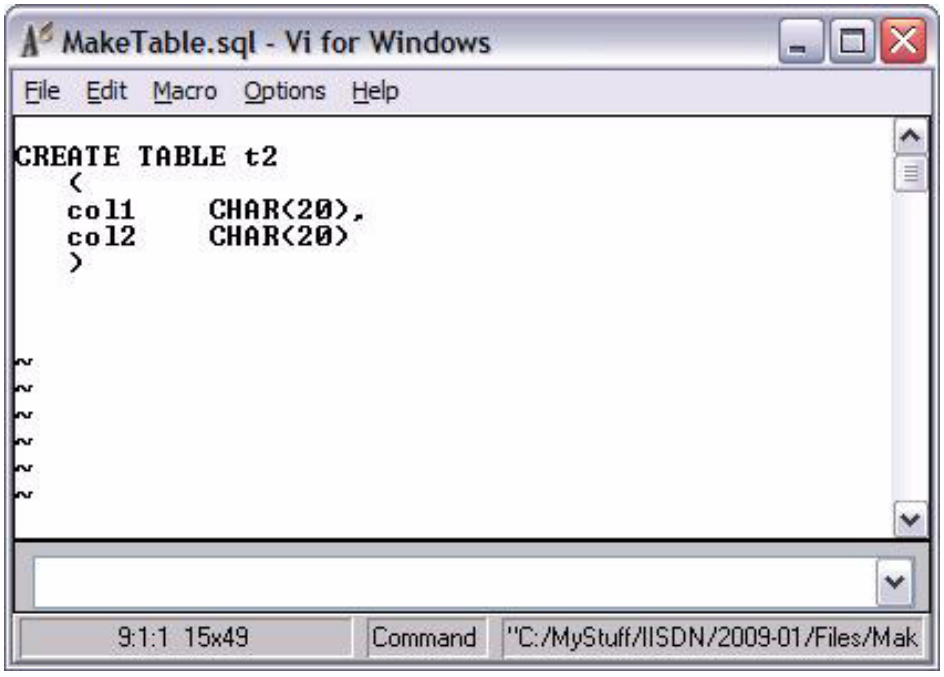

Figure 25-16 Sample SQL DDL Command file.

10. Inside the InfoSphere Data Architect program, complete the following;
    a. From the Menu Bar, select, File -> New -> Physical Data Model.

b. In the New Physical Data Model dialog box that is produced, press the Radio Button entitled, [Create from reverse engineering].

   Click Next.

   Example as shown in Figure 25-17.

   .

Note: In Figure 25-17, you have to choose a database type as DB2/UDB. No other option allows use the choices we need on the *next* dialog box.

Don't freak out. All the product is saying is that it can't promise it will read any non-ANSI Standard SQL DDL script. (DB2 is very ANSI compatible/standard.)

As the customer question was asked, we wish to read Microsoft variant SQL DDL. As long as you stay away from ANSI extensions, you should be fine.

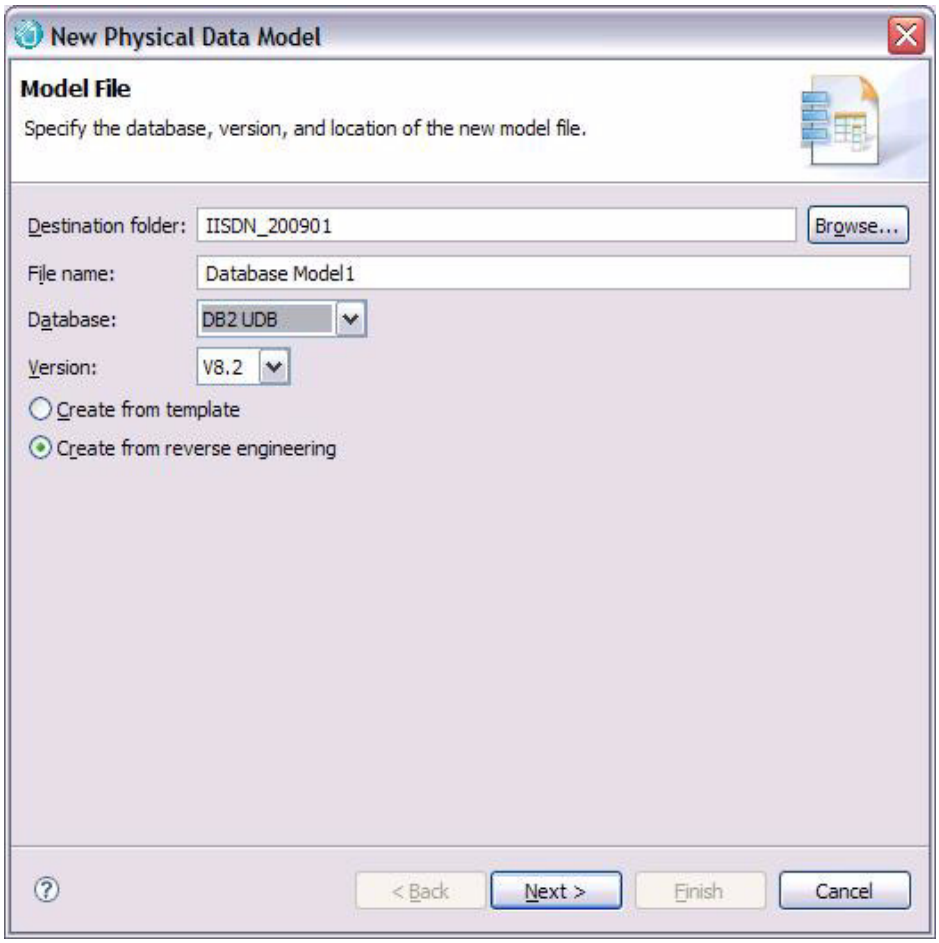

Figure 25-17 New Physical Data Model dialog box.

c. As displayed in Figure 25-18, Press the Radio Button entitled, (create from) DDL script.

   Click Next.

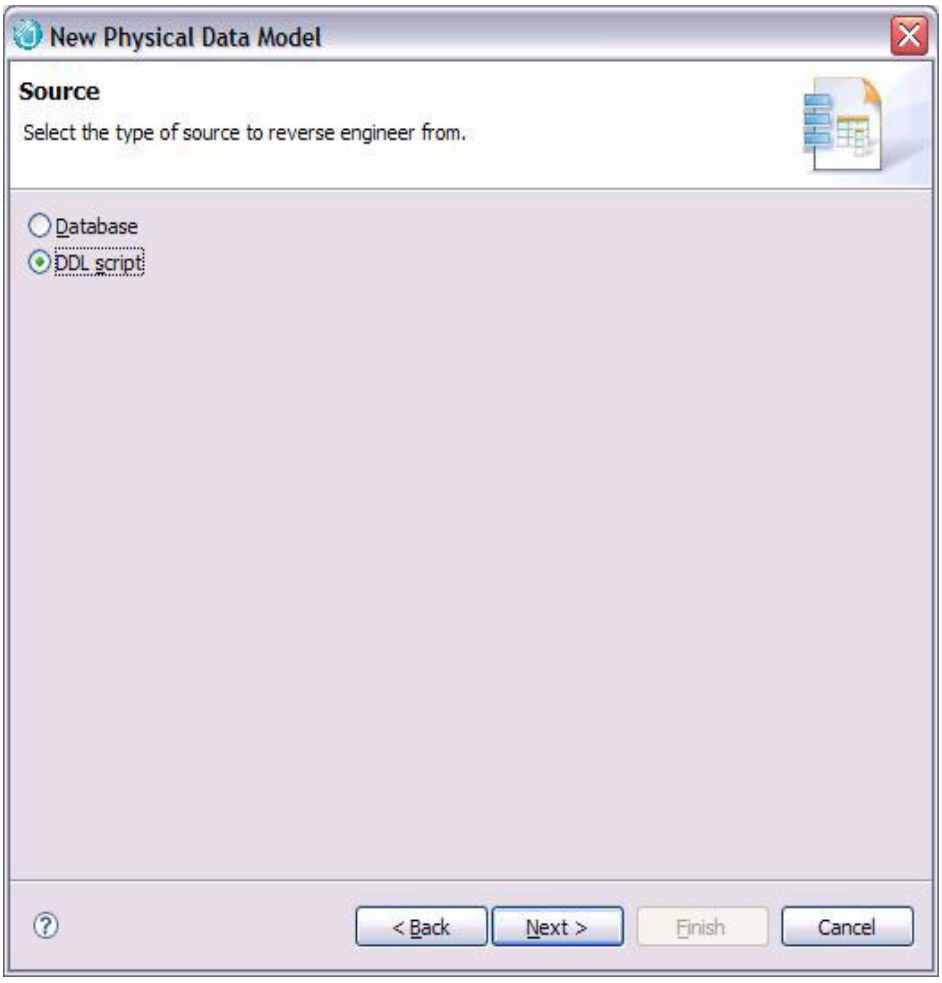

Figure 25-18 New Physical Data Model dialog box II.

d. Browse to the SQL DDL file you created in Step-9 above.

   This process is not displayed in this document.

   Click through to completion.

   A successful result is displayed in Figure 25-19.

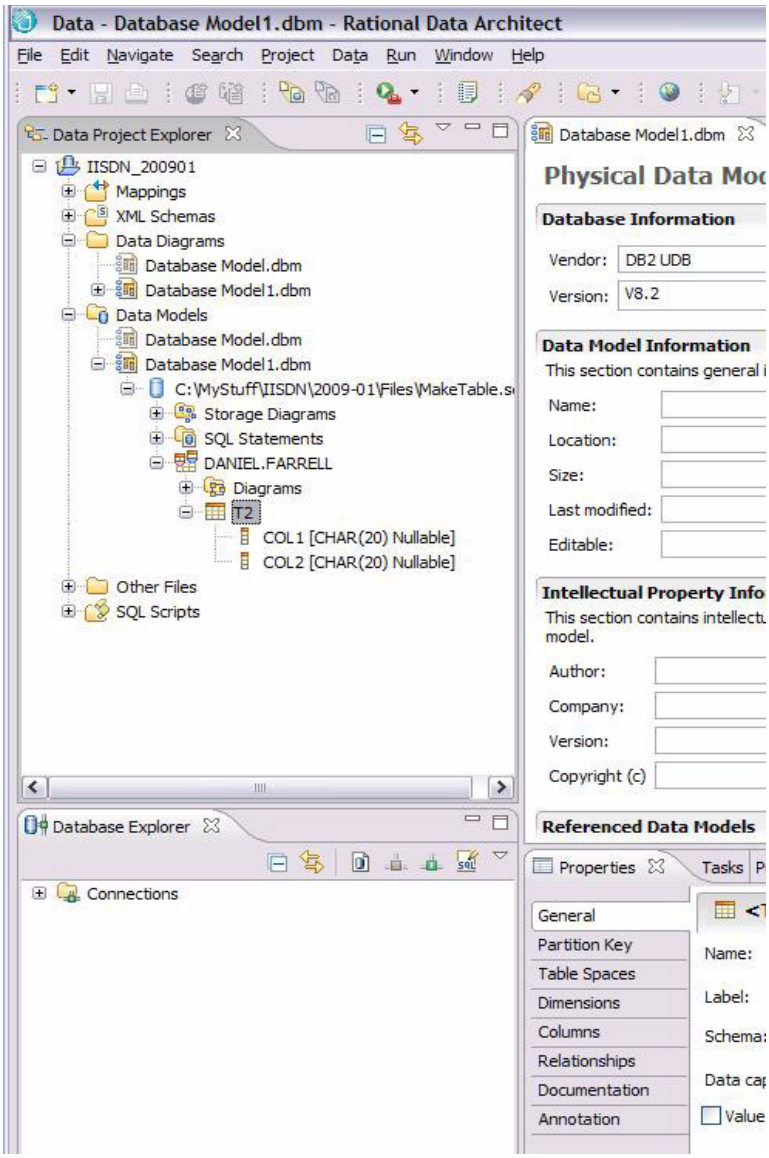

Figure 25-19 Successful result.

25.3 In this document, we reviewed or created:

We examined the (IBM) InfoSphere Information Server (IIS) MetaBrokers and Bridges component, and importing logical and physical Table definitions from InfoSphere Data Architect and other sources. We also reviewed (Local) and Shared Table definitions inside Information Server, and basic use of the Information Server MetaData Repository as a whole.

Persons who help this month.

Riccardo Tani, John Whyman, Denise Gosline, Walter Crockett, and Steve Totman just a tiny bit.

Additional resources:

As listed, the IBM Redbook SG-247274, entitled, *Leveraging DB2 Data Warehouse Edition for Business Intelligence.*

The following support URL's,

http://www-01.ibm.com/support/docview.wss?rs=3045&context=SSSRXA&dc=DB560&dc=DB520&uid=swg21318217&loc=en_US&cs=UTF-8&lang=en&rss=ct3045db2

http://www-01.ibm.com/support/docview.wss?rs=14&context=SSZJPZ&dc=DA400&uid=swg27012944&loc=en_US&cs=UTF-8&lang=en&rss=ct14db2

Legal statements:

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.

