IBM® DB2® Universal Database

# Business Intelligence Tutorial

*Version 7*

IBM® DB2® Universal Database

# Business Intelligence Tutorial

*Version 7*

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 161.

# Contents

---

# About the tutorial

This tutorial provides an end-to-end guide for typical business intelligence tasks. It has two main sections:

**Data warehousing**
> Do the lessons in this section to learn how to use the DB2 Control Center and Data Warehouse Center to create a warehouse database, move and transform source data, and write the data to the warehouse target database. Completing this section should take you about 3 hours.

**Multidimensional data analysis**
> Do the lessons in this section to learn how to use the OLAP Starter Kit to perform multidimensional analysis on relational data using Online Analytical Processing (OLAP) techniques. Completing this section should take you about an hour.

The tutorial is available in HTML or PDF format. You can view the HTML version of the tutorial from the Data Warehouse Center, OLAP Starter Kit, or the Information Center. The PDF file is available on the DB2 Publications CD-ROM.

## Tutorial business problem

You are a database administrator for a company that is called TBC: The Beverage Company. The company manufactures beverages for sale to other businesses. The financial department wants to track, analyze, and forecast the sales revenue across geographies on a periodic basis for all products sold. You have already set up standard queries of the sales data. However, these queries add to the load on your operational database. Also, users sometimes ask for additional ad-hoc queries of the data, based on the results of the standard queries.

Your company has decided to create a data warehouse for the sales data. A *data warehouse* is a database that contains data that has been cleansed and transformed into an informational format. Your task is to create this data warehouse.

You plan to use a star schema design for your warehouse. A *star schema* is a specialized design that consists of multiple dimension tables, and one fact table. *Dimension tables* describe aspects of a business. The *fact table* contains the facts or measurement about the business. In this tutorial, the star schema includes the following dimensions:

- Products
- Markets
- Scenario
- Time

The facts in the fact table include orders of the products over a period of time.

The Data warehousing part of this tutorial shows you how to define this star schema.

Your next task is to create an OLAP application to analyze your data. You first create an OLAP model and metaoutline, and then use them to create the application. The Multidimensional Analysis part of this tutorial shows you how to create an OLAP application.

## Before you begin

Before you begin, you must install the products that are covered in the sections of the tutorial that you want to use:

- For the Data warehousing section, you must install the DB2 Control Center, which includes the Data Warehouse Center administrative interface. You can install the Data Warehouse Center administrative interface on the following operating systems: Windows NT®, 95, 98, AIX®, and the Solaris Operating Environment. You must also install the DB2 server and the warehouse server, which are included in the typical install for DB2 Universal Database®. However, you must install the warehouse server on Windows NT.

  If you install the DB2 server on a different workstation from the warehouse server or the Data Warehouse Center administrative interface, you must install the DB2 client on the same workstation as the Data Warehouse Center administrative interface.

  For more information about installing DB2 Universal Database and the warehouse server, see *DB2 Universal Database Quick Beginnings* for your operating system.

  Optionally, you can install the Information Catalog Manager if you have the DB2 Warehouse Manager. If you do not have the DB2 Warehouse Manager, skip "Chapter 14. Cataloging data in the warehouse for end users" on page 85 and "Chapter 15. Working with business metadata" on page 91.

  For more information about installing the DB2 Warehouse Manager, see *DB2 Warehouse Manager Installation Guide*.

- For the Multidimensional data analysis section, you must install DB2 and the OLAP Starter Kit. The OLAP clients support Windows only.

You must also install the tutorial. In DB2 for Windows, you can install the tutorial as part of a typical install. In DB2 for AIX or the Solaris Operating Environment, you can install the tutorial with the documentation.

You need sample data to use with the tutorial. The tutorial uses the DB2 Data Warehousing sample data and the OLAP sample data.

The Data Warehousing sample data is installed on Windows NT only, when you install the tutorial. It must either be installed on the same workstation as the warehouse manager, or the remote node for the sample databases must be cataloged on the manager workstation.

You can install the OLAP sample data on Windows NT, AIX, and the Solaris Operating Environment. It must either be installed on the same workstation as the OLAP Integration Server server, or the remote node for the sample databases must be cataloged on the server workstation.

This tutorial contains several references to sample data under the X:\sqllib directory, where X is the drive under which you installed DB2. If you used the default directory structure, the data is installed under X:\Program Files\sqllib instead of X:\sqllib.

You must create the sample databases after you install the files for the sample. To create the databases:
1. Skip this step if the First Steps window is already open. Click **Start —> Programs —> IBM DB2—> First Steps**.
   The First Steps window opens.
2. Click **Create Sample Databases**. If Create Sample Databases is disabled, the sample databases have already been created.
   The Create SAMPLE Databases window opens.
3. Select the **Data Warehousing sample** check box, **OLAP sample** check box, or both, depending on which parts of the tutorial you want to do.
4. Click **OK**.
5. If you are installing the Data Warehousing sample, a window opens for the DB2 user ID and password to use to access the sample.
   a. Type the user ID and password that you want to use. Note down the user ID and password, because you will need them in a later lesson, when you define security.
   b. Click **OK**.

   DB2 starts to create the sample databases. A progress window opens. It may take a while for the databases to be created. When the database has been created, click **OK**.

If you are installing the sample on Windows NT, the databases are automatically registered with ODBC. If you are installing the sample on AIX or the Solaris Operating Environment, you must manually register the databases with ODBC. For more information about registering the databases on AIX or the Solaris Operating Environment, see *DB2 Universal Database Quick Beginnings* for your operating system.

If you selected Data Warehousing Sample, the following databases are created:

**DWCTBC**
> Contains the operational source tables that are required for the Data Warehousing section of the tutorial.

**TBC_MD**
> Contains metadata for the Data Warehouse Center objects in the sample.

If you selected the OLAP sample, the following databases are created:

**TBC**  Contains the cleansed and transformed tables that are required for the Multidimensional data analysis section of the tutorial.

**TBC_MD**
> Contains metadata for the OLAP objects in the sample.

If you select both the Data Warehousing and OLAP samples, the TBC_MD database contains metadata for both the Data Warehouse Center and OLAP objects in the sample.

Before you begin the tutorial, verify that you can connect to the sample databases:

1. Start the DB2 Control Center:
   - On Windows NT, click **Start —> Programs —> IBM DB2 —> Control Center**.
   - On AIX or the Solaris Operating Environment, type the following command:
     ```
     db2jstrt 6790
     db2cc 6790b
     ```
2. Expand the tree until you see one of the sample databases: DWCTBC, TBC, or TBC_MD.
3. Right-click the name of the database and click **Connect**.
   The Connect window opens.
4. In the **User ID** field, type the user ID that you used to create the sample.
5. In the **Password** field, type the password that you used to create the sample.
6. Click **OK**.

The DB2 Control Center connects to the database. If the DB2 Control Center is not able to establish a connection, you will see an error message.

## Conventions that are used in this tutorial

This tutorial uses typographical conventions in the text to help you distinguish between the names of controls and text that you type. For example:

- Menu items are in boldface font:

  Click **Menu —> Menu choice**.

- The names of fields, check boxes, and buttons are also in boldface font:

  Type text in the **Field** field.

- Text that you type is in example font on a new line:

  This is the text that you type.

## Related information

This tutorial covers the most common tasks that you can accomplish with the DB2 Control Center, Data Warehouse Center, and OLAP Starter Kit. For more information about related tasks, see the following documents:

**Control Center**

- The DB2 Control Center online help
- The Client Configuration Assistant online help
- The Event Monitor online help
- *DB2 Universal Database Quick Beginnings* for your operating system
- *DB2 Warehouse Manager Installation Guide*
- *DB2 Universal Database SQL Getting Started*
- *DB2 Universal Database SQL Reference*
- *DB2 Universal Database Administration Guide—Implementation*

**Data Warehouse Center**

- The Data Warehouse Center online help
- *DB2 Universal Database Data Warehouse Center Administration Guide*

**OLAP Starter Kit**

- *OLAP Setup and User's Guide*
- *OLAP Model User's Guide*
- *OLAP Metaoutline User's Guide*
- *OLAP Administrator's Guide*
- *OLAP Spreadsheet Add-in User's Guide for 1-2-3*
- *OLAP Spreadsheet Add-in User's Guide for Excel*

## Contacting IBM

If you have a technical problem, please review and carry out the actions suggested by the *Troubleshooting Guide* before contacting DB2 Customer Support. This guide suggests information that you can gather to help DB2 Customer Support to serve you better.

For information or to order any of the DB2 Universal Database products contact an IBM representative at a local branch office or contact any authorized IBM software remarketer.

If you live in the U.S.A., then you can call one of the following numbers:
- 1-800-237-5511 for customer support
- 1-888-426-4343 to learn about available service options

## Product Information

If you live in the U.S.A., then you can call one of the following numbers:
- 1-800-IBM-CALL (1-800-426-2255) or 1-800-3IBM-OS2 (1-800-342-6672) to order products or get general information.
- 1-800-879-2755 to order publications.

**http://www.ibm.com/software/data/**
> The DB2 World Wide Web pages provide current DB2 information about news, product descriptions, education schedules, and more.

**http://www.ibm.com/software/data/db2/library/**
> The DB2 Product and Service Technical Library provides access to frequently asked questions, fixes, books, and up-to-date DB2 technical information.
>
> **Note:** This information may be in English only.

**http://www.elink.ibmlink.ibm.com/pbl/pbl/**
> The International Publications ordering Web site provides information on how to order books.

**http://www.ibm.com/education/certify/**
> The Professional Certification Program from the IBM Web site provides certification test information for a variety of IBM products, including DB2.

**ftp.software.ibm.com**
> Log on as anonymous. In the directory /ps/products/db2, you can find demos, fixes, information, and tools relating to DB2 and many other products.

**comp.databases.ibm-db2, bit.listserv.db2-l**
> These Internet newsgroups are available for users to discuss their experiences with DB2 products.

**On Compuserve: GO IBMDB2**

> Enter this command to access the IBM DB2 Family forums. All DB2 products are supported through these forums.

For information on how to contact IBM outside of the United States, refer to Appendix A of the *IBM Software Support Handbook*. To access this document, go to the following Web page: http://www.ibm.com/support/, and then select the IBM Software Support Handbook link near the bottom of the page.

**Note:** In some countries, IBM-authorized dealers should contact their dealer support structure instead of the IBM Support Center.

# Part 1. Data Warehousing

**1**

# Chapter 1. About data warehousing

In this section, you will obtain an overview of data warehousing and the data warehousing tasks in this tutorial.

## What is data warehousing?

The systems that contain *operational data*—the data that runs the daily transactions of your business—contain information that is useful to business analysts. For example, analysts can use information about which products were sold in which regions at which time of year to look for anomalies or to project future sales. However, there are several problems if analysts access the operational data directly:

- They might not have the expertise to query the operational database. For example, querying IMS databases requires an application program that uses a specialized type of data manipulation language. In general, those programmers who have the expertise to query the operational database have a full-time job in maintaining the database and its applications.

- Performance is critical for many operational databases, such as databases for a bank. The system cannot handle users making ad-hoc queries.

- The operational data generally is not in the best format for use by business analysts. For example, sales data that is summarized by product, region, and season is much more useful to analysts than the raw data.

Data warehousing solves these problems. In *data warehousing*, you create stores of *informational data*—data that is extracted from the operational data and then transformed for end-user decision making. For example, a data warehousing tool might copy all the sales data from the operational database, perform calculations to summarize the data, and write the summarized data to a separate database from the operational data. End-users can query the separate database (the *warehouse*) without impacting the operational databases.

## Lesson overview

DB2 Universal Database offers the Data Warehouse Center, a DB2 component that automates warehouse processing. You can use the Data Warehouse Center to define which data to include in the warehouse. Then, you can use the Data Warehouse Center to automatically schedule refreshes of the data in the warehouse.

This tutorial covers the most common tasks that are required to set up a warehouse.

In this tutorial, you will:

- Define a *subject area* that identifies and groups the processes that you will create for the tutorial.
- Explore the source data (which is the operational data) and define warehouse sources. *Warehouse sources* identify the source data that you want to use in your warehouse
- Create a database to use as the warehouse and define *warehouse targets*, which identify the target data to include in your warehouse.
- Specify how to move and transform the source data into its format for the warehouse database. You will define a *process*, which contains the series of movement and transformation steps required to produce a target table in the warehouse from one or more source tables, views, or files. You will then divide the process into *steps*, each of which defines one operation in the movement and transformation process. Then you will test the steps that you defined and schedule them to run automatically.
- Administer the warehouse by defining security and monitoring database usage.
- Create an information catalog of the data in the warehouse if you have installed the DB2 Warehouse Manager package. An *information catalog* is a database that contains business metadata that helps users identify and locate data and information that is available to them in the organization. End users of the warehouse can search the catalog to determine which tables to query.
- Define a star schema model for the data in the warehouse. A *star schema* is a specialized design that consists of multiple *dimension tables*, which describe aspects of a business, and one *fact table*, which contains the facts about the business. For example, if you manufacture soft drinks, some dimension tables are products, markets, and time. The fact table might contain transaction information about the products that are ordered in each region by season.
- You can join the fact table and dimension tables to combine details from the dimension tables with the order information. For example, you might join the product dimension with the fact table to add information about how each product was packaged to the orders.

# Chapter 2. Creating a warehouse database

In this lesson, you will create the database for your warehouse and register the database with ODBC.

As part of DB2 First Steps, you had DB2 create the DWCTBC database, which contains the source data for this tutorial.

In this lesson, you will create the database that is to contain a version of the source data that is transformed for the warehouse. In "Chapter 3. Browsing the source data" on page 9, you learn how to view the source data. The rest of the tutorial teaches you how to transform that data and work with your warehouse database.

In this lesson, you will also learn how to register your database with Open Database Connectivity (ODBC), which allows tools like Lotus Approach and Microsoft Access to work with your warehouse.

## Creating a database

In this exercise, you will use the Create Database wizard to create the TUTWHS database for your warehouse.

To create the database:

1. Start the DB2 Control Center:
   - On Windows NT, click **Start —> Programs —> IBM DB2 —> Control Center**.
   - On AIX or the Solaris Operating Environment, type the following command:
     ```
     db2jstrt 6790
     db2cc 6790b
     ```
2. Expand the **Systems** folder tree until you see the Databases folder.
3. Right-click the **Databases** folder, and select **Create —> Database Using Wizard**. The Create Database wizard opens.
4. In the **Database name** field, type the name of the database:
   ```
   TUTWHS
   ```
5. From the **Default drive** list, select a drive for the database.
6. In the **Comment** field, type a description of the database:
   ```
   Tutorial warehouse database
   ```
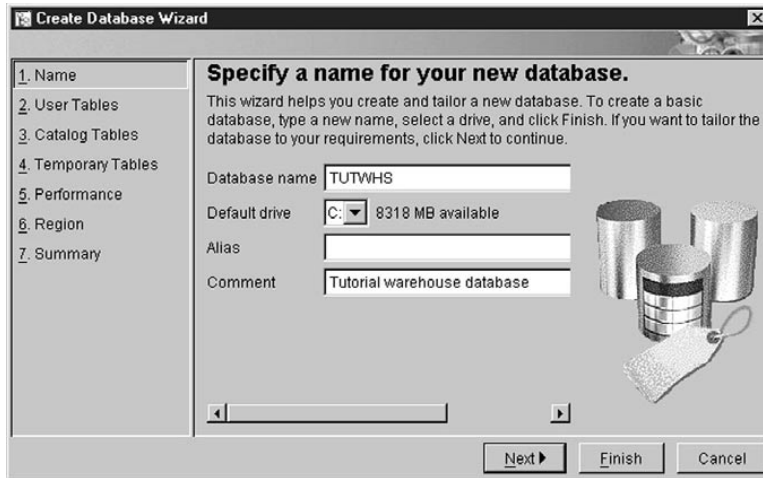
7. Click **Finish**. All other fields and pages in this wizard are optional. The TUTWHS database is created and is listed in the DB2 Control Center.

## Registering a database with ODBC

There are several ways that you can register a database with ODBC. You can use the Client Configuration Assistant on Windows NT, the Command Line Processor, or the ODBC32 Data Source Administrator on Windows NT. In this exercise, you will use the Client Configuration Assistant.

For more information about the Command Line Processor, see the *DB2 Universal Database Command Reference*. For more information about the ODBC32 Data Source Administrator, see the online help in the Administrator.

To register the TUTWHS database with ODBC:

1. Start the Client Configuration Assistant by clicking **Start —> Programs —> IBM DB2 —> Client Configuration Assistant**. The Client Configuration Assistant window opens.

2. Select **TUTWHS** from the list of databases.
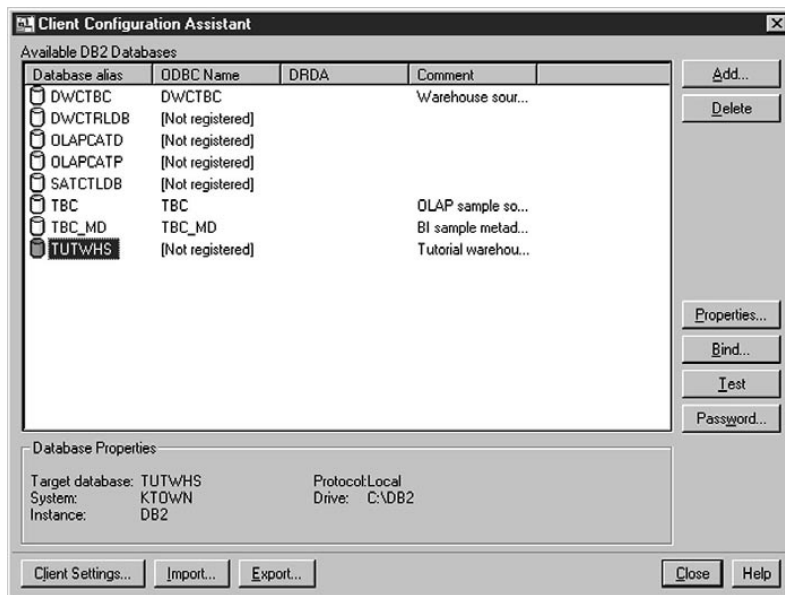


3. Click **Properties**. The Database Properties window opens.
4. Select **Register this database for ODBC**. Use the default selection of **As a system data source**, which means that the data is available to all users on the system.
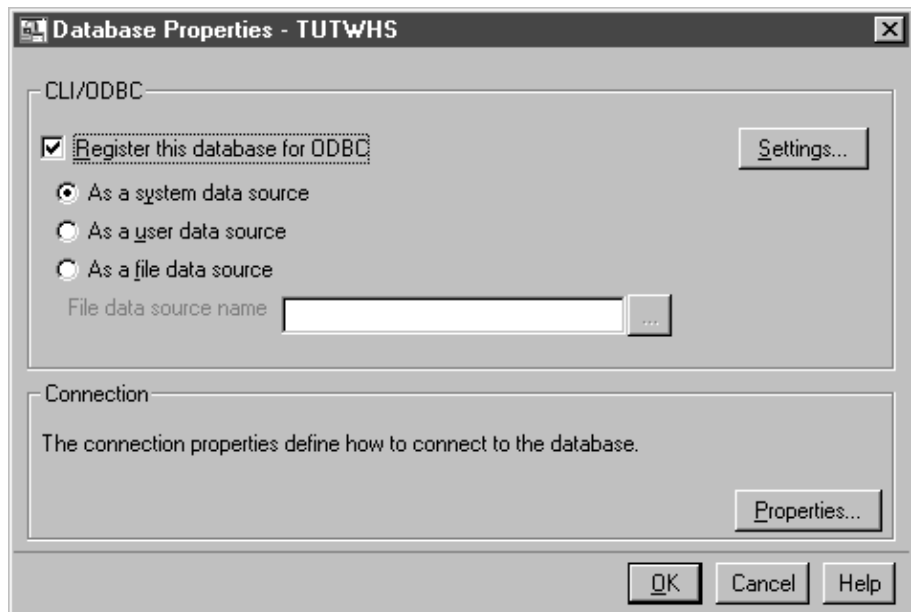
5. Click **OK**. All other fields are optional. The TUTWHS database is registered with ODBC.

   The **Properties** and **Settings** push buttons in the Client Configuration Assistant window are used to optimize your ODBC connections and configuration. You do not need to adjust these properties or settings for the tutorial, but there is online help available if you need to work with them in your daily environment.
6. Click **OK** to close the DB2 Message window.
7. Close the Client Configuration Assistant.

## Connecting to the target database

Before you use the database that you defined, you must verify that you can connect to the database.

To connect to the database:
1. From the DB2 Control Center, expand the tree until you see the TUTWHS database.
2. Right-click the name of the database and click **Connect**.

   The Connect window opens.
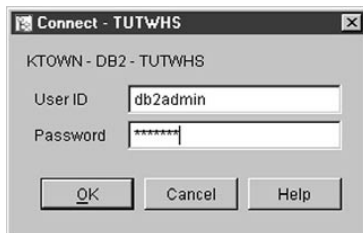


3. Type the user ID and password that you used to log on to the DB2 Control Center.
4. Click **OK**.

   The DB2 Control Center connects to the database.

## What you just did

In this lesson, you created the TUTWHS database to contain the data for the warehouse. Then, you registered the database with ODBC. Finally, you verified that you can connect to the database. In the next lesson, you will view the source data that you will later transform and store in the database that you just created.

# Chapter 3. Browsing the source data

In this lesson, you will browse the source data that is available to you in the sample. You will investigate ways that you can transform this data into the star schema for the warehouse.

Source data is not always well structured for analysis and might need to be transformed to be more usable. The source data that you will be using consists of DB2 Universal Database tables and a text file. Some other typical types of source data are non-DB2 relational tables, MVS™ data sets, and Microsoft Excel spreadsheets. As you browse the data, look for relationships among the data and consider what information might be of the most interest to users.

In general, when you design a warehouse, you gather information about the operational data to use as input to the warehouse and the requirements for the warehouse data. The database administrator who is responsible for the operational data is a good source for information about the operational data. The business users who will be making business decisions based on the data in the warehouse are a good source for information about the requirements of the warehouse.

## Viewing table data

In this exercise, you will use the DB2 Control Center to view the first 200 rows of a table.

To view the table:

1. Expand the objects in the DWCTBC database until you see the **Tables** folder.
2. Click the folder. In the right panel, you see all the tables for the database.
3. Find the GEOGRAPHIES table. Right-click it, and click **Sample Contents**.

Up to 200 rows of the table are displayed. The column names are displayed at the top of the window. You might need to scroll to the right to see all the columns and scroll down to see all the rows.

4. Click **Close**.
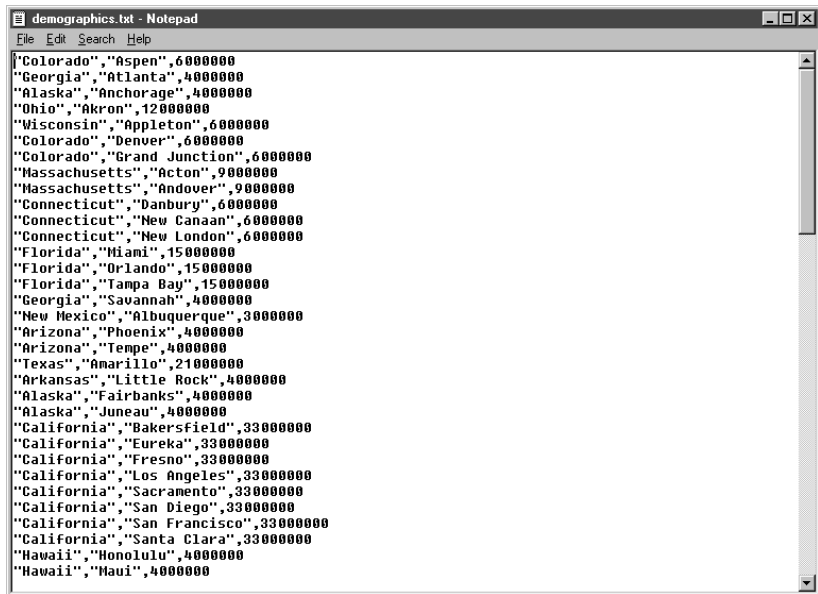
## Viewing file data

In this exercise, you will use Microsoft® Notepad to view the contents of the demographics.txt file.

To view the file:

1. Click **Start —> Programs —> Accessories —> Notepad** to open Microsoft Notepad.
2. Click **File —> Open**.

3. Use the Open window to locate the file. For example, it might be located in *X*:\program files\sqllib\samples\db2sampl\dwc\demographics.txt, where *X* is the drive on which you installed the sample.

4. Select the demographics.txt file and click **Open** to view its contents.



Note that the file is comma-delimited. You will need to supply this information in a later lesson.

5. Close Notepad.

## What you just did

In this lesson, you viewed the GEOGRAPHIES source table and the demographics.txt file, which are provided in the Data Warehousing sample. In the next lesson, you will open the Data Warehouse Center and start defining your warehouse.

# Chapter 4. Defining warehouse security

In this lesson, you will define security for your warehouse.

The first level of security is the logon user ID that is in use when you open the Data Warehouse Center. Although you log on to the DB2 Control Center, the Data Warehouse Center verifies that you are authorized to open the Data Warehouse Center administrative interface by comparing your user ID to entries in the warehouse control database. The *warehouse control database* contains the control tables that are required to store Data Warehouse Center metadata. You initialize the control tables for this database when you install the warehouse server as part of DB2 Universal Database or use the Data Warehouse Center Control Database Management window. During initialization, you specify the ODBC name of the warehouse control database, a valid DB2 user ID, and a password. The Data Warehouse Center authorizes this user ID and password to update the warehouse control database. In the Data Warehouse Center, this user ID is defined as the *default warehouse user*.

**Tip:** The default warehouse user requires a different type of database and operating system authorization for each operating system that the warehouse control database supports. For more information, see *DB2 Warehouse Manager Installation Guide*.

The default warehouse user is authorized to access all Data Warehouse Center objects and perform all Data Warehouse Center functions. However, you probably want to restrict access to certain objects within the Data Warehouse Center and the tasks that users can perform on the objects. For example, warehouse sources and warehouse targets contain the user IDs and passwords for their corresponding databases. You might want to restrict access to those warehouse sources and warehouse targets that contain sensitive data, such as personnel data.

To provide this level of security, the Data Warehouse Center provides a security system that is separate from the database and operating system security. To implement Data Warehouse Center security, you define warehouse users and warehouse groups. A *warehouse group* is a named grouping of warehouse users and their authorization to perform functions. Warehouse users and warehouse groups do not have to match the DB users and DB groups that are defined for the warehouse control database.

For example, you might define a warehouse user that corresponds to someone who uses the Data Warehouse Center. You might then define a warehouse group that is authorized to access certain warehouse sources, and add the

new user to the new warehouse group. The new user is authorized to access the warehouse sources that are included in the group.

There are various types of authorization that you can give users. You can include any of the different types of authorization in a warehouse group. You can also include a warehouse user in more than one warehouse group. The combination of the groups to which a user belongs is the user's overall authorization.

In this lesson, you will log on to the Data Warehouse Center as the default warehouse user, define a new warehouse user, and define a new warehouse group.

## Specifying the warehouse control database

When you install the Data Warehouse Center as part of the default DB2 installation, the installation process registers the default warehouse control database as the active warehouse control database. However, you must use the TBC_MD database in the sample as the warehouse control database so that you can use the sample metadata. To make TBC_MD the active database, you must reinitialize it.

To reinitialize TBC_MD:

1. Click **Start —> Programs —> IBM DB2 —> Warehouse Control Database Management.**

   The Data Warehouse Center - Control Database Management window opens.

2. In the **New control database** field, type the name of the new control database that you want to use.

   TBC_MD

3. In the **Schema** field, use the default schema of IWH.

4. In the **User ID** field, type the name of the user ID that is required to access the database.

5. In the **Password** field, type the name of the password for the user ID.

6. In the **Verify Password** field, type the password again.

7. Click **OK**.

   The window remains open. The Messages field displays messages that indicate the status of the creation and migration process.

8. After the process is complete, close the window. TBC_MD is now the active warehouse control database.

## Starting the Data Warehouse Center

In this exercise, you will start the Data Warehouse Center from the DB2 Control Center and log on as the default warehouse user. When you log on, you will use the TBC_MD warehouse control database. The default warehouse user for TBC_MD is the user ID that you specified when you created the Data Warehousing sample databases.

TBC_MD must be a local or a cataloged remote database on the workstation that contains the warehouse server. It must also be a local or cataloged remote database on the workstation that contains the Data Warehouse Center administrative client.

To start the Data Warehouse Center:

1. From the DB2 Control Center window, click **Tools —> Data Warehouse Center**. The Data Warehouse Center Logon window opens.
2. Click the **Advanced** push button.

   The Advanced window opens.
3. In the **Control database** field, type TBC_MD, the name of the warehouse control database that is included in the sample.
4. In the **Server host name** field, type the TCP/IP host name for the workstation where the warehouse manager is installed.
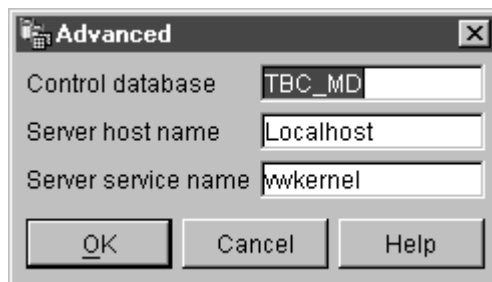


5. Click **OK**.

   The Advanced Logon window closes.

   The next time that you log on, the Data Warehouse Center will use the settings that you specified in the Advanced Logon window.
6. In the **User ID** field of the Data Warehouse Center Logon window, type the default warehouse user ID.
7. In the **Password** field, type the password for the user ID.

Data Warehouse Center Logon

| User ID | db2admin |
| Password | ******** |

[OK] [Cancel] [Advanced...] [Help]

8. Click **OK**.

The Data Warehouse Center Logon window closes.

9. Close the Data Warehouse Center Launchpad window.

## Defining a warehouse user

In this exercise, you will define a new user to the Data Warehouse Center.

The Data Warehouse Center controls access with user IDs. When a user logs on, the user ID is compared to the warehouse users that are defined in the Data Warehouse Center to determine whether the user is authorized to access the Data Warehouse Center. You can authorize additional users to access the Data Warehouse Center by defining new warehouse users.

The user ID for the new user does not require authorization to the operating system or the warehouse control database. The user ID exists only within the Data Warehouse Center.

To define a warehouse user:

1. In the left panel of the main Data Warehouse Center window, expand the **Administration** folder.
2. Expand the **Warehouse Users and Groups** tree.
3. Right-click the **Warehouse Users** folder and click **Define**.

   The Define Warehouse User notebook opens.
4. In the **Name** field, type the business name of the user:

   Tutorial User

   The name identifies the user ID within the Data Warehouse Center. This name can be up to 80 characters-including spaces.
5. In the **Administrator** field, type your name as the contact for this user.
6. In the **Description** field, type a short description of the user:

   This is a user that I created for the tutorial.

   **Tip:** You can use the **Description** and **Notes** fields to provide metadata about the definitions for your warehouse. You can then publish this metadata in an information catalog for the warehouse. Users of the

warehouse can search the metadata to find the warehouse that contains the information they need to query.

7. In the **User ID** field, type the new user ID:

   `tutuser`

   The user ID must be no longer than 60 characters and cannot contain spaces, dashes, or special characters (such as @, #, $, %,>, +, =). It can contain the underscore character.

   **Specifying a unique user ID:**

   To determine if a user ID and password is unique:

   a. From the main Data Warehouse Center window, expand the **Administration** tree.

   b. Click on the **Warehouse Users** folder. All of the user IDs for the data warehouse appear in the right panel. Any ID that doesn't appear in the right panel is a unique ID.

8. In the **Password** field, type the password:

   `password`

   Passwords must be a minimum of six characters and cannot contain spaces, dashes, or special characters.

   **Tip:** You can change your password on this page of the user notebook.

9. In the **Verify Password** field, type your password again.

10. Verify that the **Active User** check box is selected.

> **Tip:** You can clear this check box to temporarily revoke a user's access to the Data Warehouse Center, without deleting the user definition.



11. Click **OK** to save the warehouse user and close the notebook.

## Defining the warehouse group

In this exercise, you will define a warehouse group that will authorize the Tutorial User that you just created to perform tasks.

To define the warehouse group:

1. From the main Data Warehouse Center window, right-click the **Warehouse Groups** folder and click **Define**.

The Define Warehouse Group notebook opens.
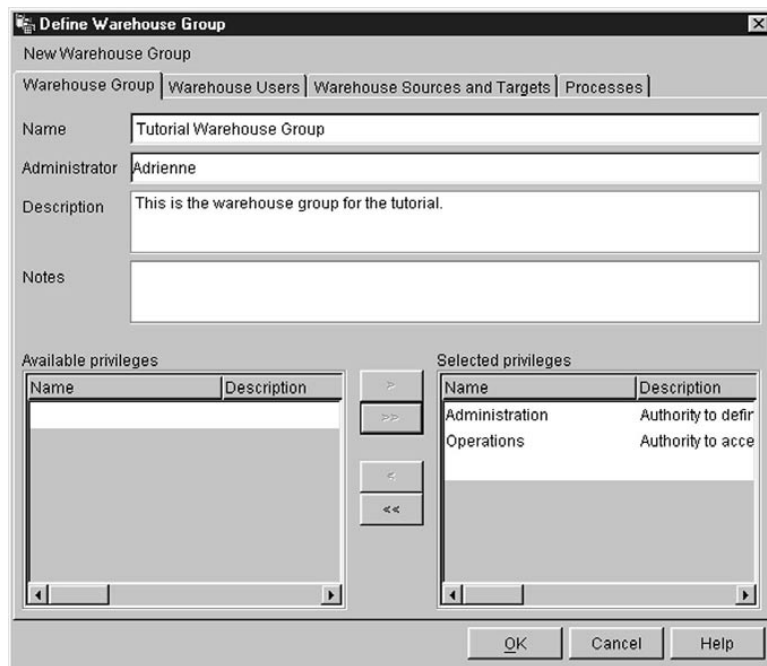


2. In the **Name** field, type the name for the new group:

   Tutorial Warehouse Group

3. In the **Administrator** field, type your name as the contact for this new group.

4. In the **Description** field, type a short description of the new group:

   This is the warehouse group for the tutorial.

5. From the **Available privileges** list, click **>>** to select all privileges for your group.

   The Administration and Operations privileges move to the **Selected privileges** list. Your group now has the following privileges.

   **Administration**
   Users in the warehouse group can define and change warehouse users and warehouse groups, change Data Warehouse Center properties, import metadata, and define which warehouse groups have access to objects when they are created.

   **Operations**
   Users in the warehouse group can monitor the status of scheduled processing.

6. Click the **Warehouse Users** tab.

7. From the **Available warehouse users** list, select the **Tutorial User**.

8. Click **>**.

The Tutorial User moves to the **Selected warehouse users** list.
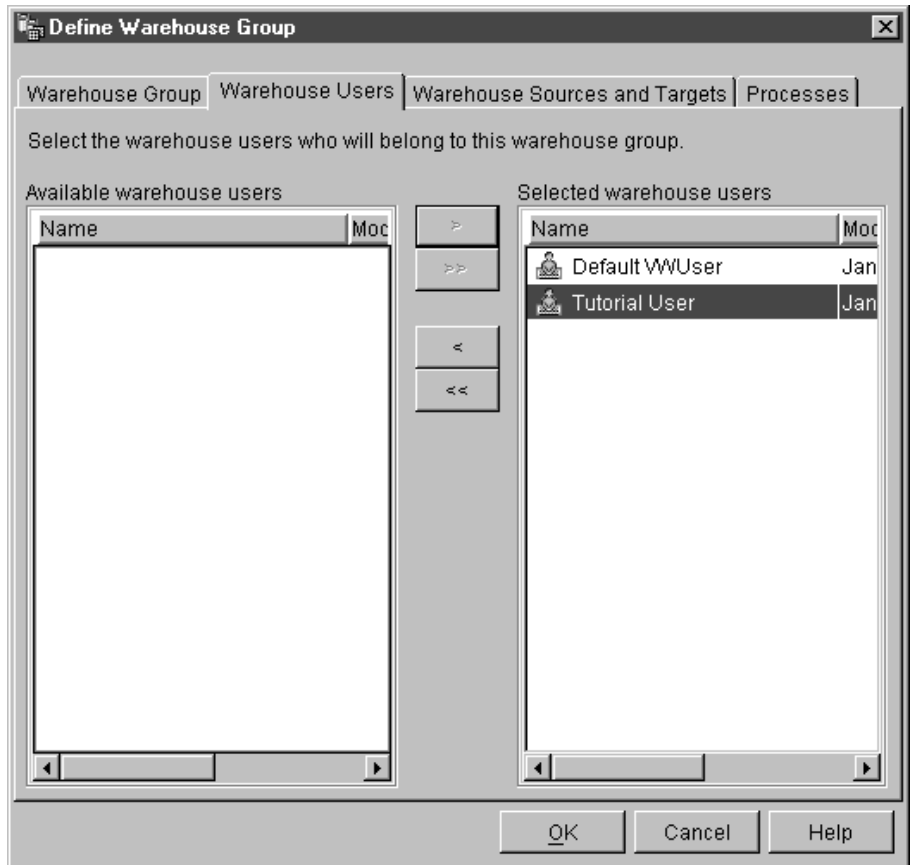


The user is now part of the warehouse group.

Skip the Warehouse sources and targets page and the Processes page. You will create these objects in subsequent lessons. You will authorize the warehouse group to access objects as you create the objects.

9. Click **OK** to save the warehouse user group and close the notebook.

## What you just did

In this lesson, you logged on to the Data Warehouse Center, created a new user, and defined a warehouse group. In subsequent lessons, you will authorize the warehouse group to access the objects that you will define.

# Chapter 5. Defining a subject area

In this lesson, you will use the Data Warehouse Center to define a subject area. A *subject area* identifies and groups processes that relate to a logical area of the business.

For example, if you are building a warehouse of sales and marketing data, you define a Sales subject area and a Marketing subject area. You then add the processes that relate to sales underneath the Sales subject area. Similarly, you add the definitions that relate to the marketing data underneath the Marketing subject area.

For this tutorial, you will define a TBC Tutorial subject area to contain the definitions for the tutorial.

Any user can define a subject area, so you do not need to change the authorizations for the Tutorial Warehouse Group.
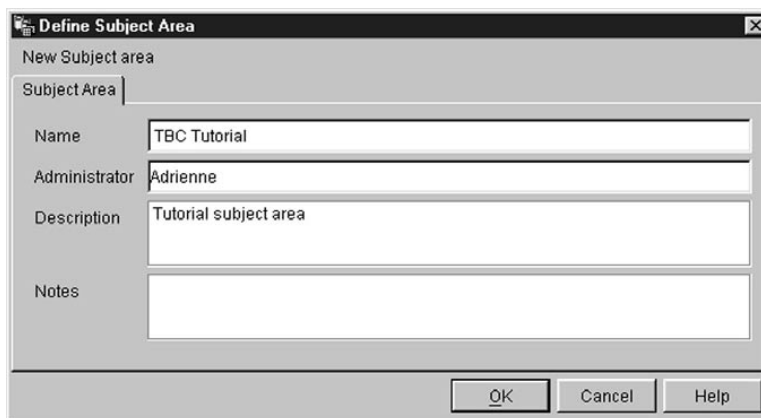
## Defining the TBC Tutorial subject area

To define the subject area:

1. From the Data Warehouse Center tree, right-click on the **Subject Areas** folder, and click **Define**.

   The Subject Area Properties notebook opens.

   | Define Subject Area | ✕ |
   |---|---|
   | New Subject area | |

   **Subject Area**

   | | |
   |---|---|
   | Name | TBC Tutorial |
   | Administrator | Adrienne |
   | Description | Tutorial subject area |
   | Notes | |

   OK    Cancel    Help

2. In the **Name** field, type the business name of the subject area for this tutorial:

   TBC Tutorial

The name can be 80 characters-including spaces.

3. In the **Administrator** field, type your name as the contact for this new subject.

4. In the **Description** field, type a short description of the subject area:

   Tutorial subject area

   You can also use the **Notes** field to provide additional information about the subject area.

5. Click **OK** to create the subject area in the Data Warehouse Center tree.

## What you just did

In this lesson, you defined the TBC Tutorial subject area. In "Chapter 8. Defining data transformation and movement" on page 35, you will define processes under this subject area.

# Chapter 6. Defining warehouse sources

In the next few lessons, you will focus on defining the Market dimension table that was introduced in "Tutorial business problem" on page vii. In this lesson, you will define *warehouse sources*, which are logical definitions of the tables and files that will provide data to the Market dimension table. The Data Warehouse Center uses the specifications in the warehouse sources to access and select the data. You will define two warehouse sources that correspond to the source data that you viewed in "Chapter 3. Browsing the source data" on page 9:

**Tutorial Relational Source**
> Corresponds to the GEOGRAPHIES source table in the DWCTBC database.

**Tutorial File Source**
> Corresponds to the demographics file, which you will load into the warehouse database in a later lesson.

If you are using source databases that are remote to the warehouse server, you must register the databases on the workstation that contains the warehouse server.

## Updating the TBC sample sources

The sample warehouse sources do not have a user ID and password associated with them. You need to add a user ID and password before you can work with these sources. In this exercise, you will add a user ID and password for the TBC Sample Sources.

To update the TBC sample sources:

1. Expand the **Warehouse Sources** tree.
2. Right-click on **TBC Sample Sources**, and click **Properties**.
   The Properties — TBC Sample Sources window opens.
3. Click the **Database** tab.
4. In the **User ID field**, type the user ID that you specified when you created the sample database in "Chapter 2. Creating a warehouse database" on page 5.
5. In the **Password** field, type the password for the user ID.
6. In the **Verify Password** field, type the password again.
7. Click **OK**.

## Defining a relational warehouse source

In this exercise, you will define a relational warehouse source called the Tutorial Relational Source. It corresponds to the GEOGRAPHIES relational table that is provided in the DWCTBC database.

To define the Tutorial Relational Source:

1. Right-click the **Warehouse Sources** folder.
2. Click **Define —> DB2 Family —> DB2 UDB for Windows NT**.

   The Define Warehouse Source notebook opens.
3. In the **Name** field, type the business name (a descriptive name that users will understand) for the warehouse source:

   Tutorial Relational Source

   You will use this name to refer to your warehouse source throughout the Data Warehouse Center.
4. In the **Administrator** field, type your name as the contact for the warehouse source.
5. In the **Description** field, type a short description of the data:

   Relational data for the TBC company



6. Click the **Database** tab.
7. In the **Database name** field, select or type DWCTBC as the name of the physical database.
8. In the **User ID** field, type a user ID that has access to the database.

   Use the user ID that you specified when you created the sample database in "Chapter 2. Creating a warehouse database" on page 5.
9. In the **Password** field, type the password for the user ID.

10. In the **Verify password** field, type the password again.



11. Click the **Tables and views** tab.

    Because the tables are in a DB2 database, you can import the table definitions from DB2 rather than defining them manually.

12. Expand the **Tables** folder.

    The Filter window opens.

13. Click **OK**.

    The Data Warehouse Center displays a progress window. The import might take a while.

    After the import finishes, the Data Warehouse Center lists the imported objects in the **Available tables and views** list.

14. From the **Available tables and views** list, select the **SAMPLTBC.GEOGRAPHIES** table.

15. Click > to move the SAMPLTBC.GEOGRAPHIES table to the **Selected tables and views** list.
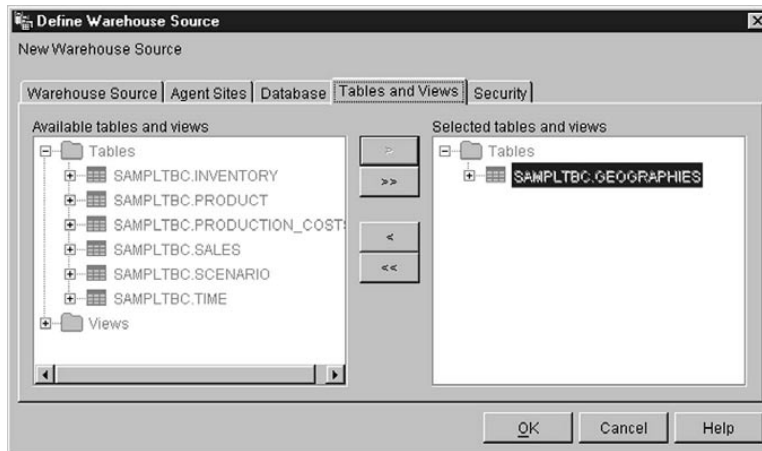
16. Click the **Security** tab.

17. Click the **Tutorial Warehouse Group** (which you created in "Defining the warehouse group" on page 18) to grant your user ID the ability to create steps that use this warehouse source.

18. Click >

    Adding the source to the Selected warehouse groups list authorizes the users in the group (in this case, you) to define tables and views for the source.

19. Click **OK** to save your changes and close the Define Warehouse Sources notebook.

## Defining a file source

In this exercise, you will define a file warehouse source called the Tutorial File Source. It corresponds to the Demographics file that is provided with the Data Warehousing sample. For this tutorial, you will define only one file in the warehouse source, but you can define multiple files in a warehouse source.

To define the Tutorial File Source:

1. Right-click the **Warehouse Sources** folder.

2. Click **Define —> Flat File —> Local files**.

   The source type is Local files because the file that will be used in this exercise was installed on your workstation along with the tutorial.

   The Define Warehouse Source notebook opens.

3. In the **Name** field, type the business name for the warehouse source:

   Tutorial file source

4. In the **Administrator** field, type your name as the contact for the warehouse source.

5. In the **Description** field, type a short description of the data:

   File data for the TBC company

6. Click the **Files** tab.

7. Right-click in the blank area of the **Files** list, and click **Define**.

   The Define Warehouse Source File notebook opens.

8. In the **File name** field, type the following name:

   *X*:\Program Files\sqllib\samples\db2sampl\dwc\demographics.txt

   where:

   - *X* is the drive on which you installed the sample. This entry is the path and file name for the demographics file.
   - *sqllib* is the directory under which you installed DB2 Universal Database.

   On a UNIX® system, file names are case-sensitive.

9. In the **Description** field, type a short description of the file:

   Demographics data for sales regions.

10. In the **Business name** field, type:

    Demographics Data

11. Click the **Parameters** tab.
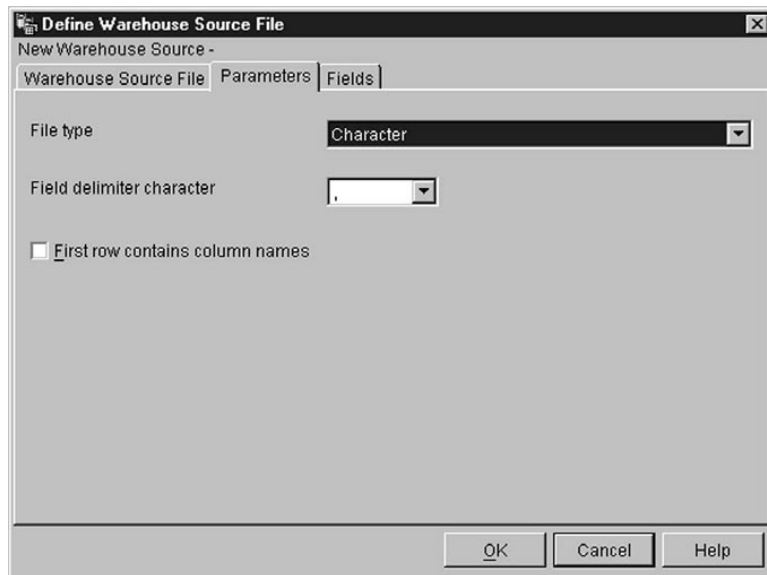


12. Verify that **Character** is selected in the **File type** list.

13. Verify that the comma is selected in the **Field delimiter character** field.

As you saw in the lesson "Chapter 3. Browsing the source data" on page 9 , the file is comma-delimited.

14. Verify that the **First row contains column names** check box is cleared.

    The file does not contain column names.

15. Click the **Fields** tab.

    The Data Warehouse Center reads the file that you specified on the Warehouse Source File page. It defines columns based on the fields in the file, and displays the column definitions in the **Fields** list. It displays sample data in the **File** preview area. Up to 10 rows of sample data are displayed. You can scroll to see all the sample data.

16. Click the **COL001** column name to change the column name.

17. Type the new name for the column:

    STATE

18. Repeat steps 16 and 17 to rename the rest of the columns. Rename **COL002** as CITY and **COL003** as POPULATION.
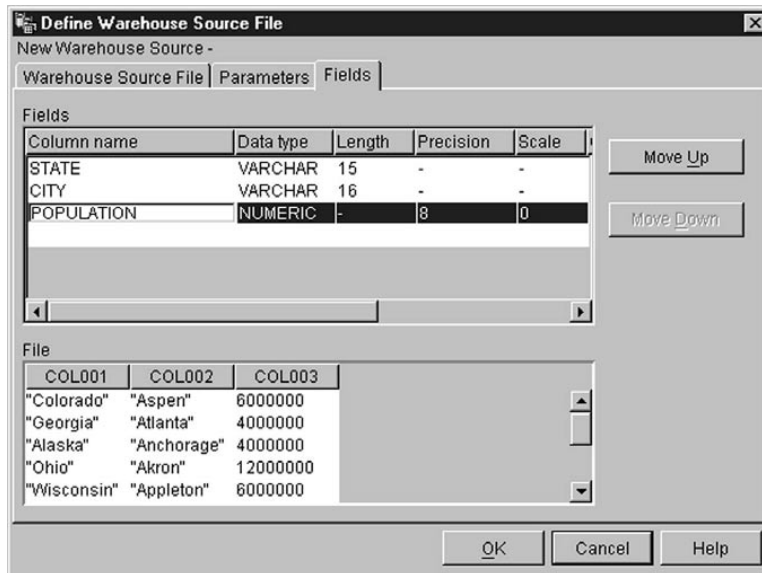


19. Click **OK**.

    The Define Warehouse Source File notebook closes.

20. In the Define Warehouse Source notebook, click the **Security** tab.

21. Select the **Tutorial Warehouse Group** to grant your user ID the ability to create steps that use this warehouse source.

22. Click > to move the Tutorial Warehouse Group to the **Selected Warehouse Groups** list.

23. Click **OK** to save your changes and close the Define Warehouse Source notebook.

## What you just did

In this lesson, you defined a relational warehouse source and a file warehouse source. You will use these sources in "Chapter 8. Defining data transformation and movement" on page 35 to indicate that these sources provide data to the LOOKUP_MARKET dimension table for the star schema that you are defining for the TBC Company.

# Chapter 7. Defining warehouse targets

In this lesson, you will define warehouse targets. *Warehouse targets* identify the database and tables that the Data Warehouse Center is to use for your data warehouse. Generally, the target tables that are defined in the warehouse target are also used as the dimension and fact tables in the star schema. However, the warehouse target might also include interim target tables that are used for data transformation.

In this lesson, you will define the Tutorial Targets warehouse target, which is a logical definition for the warehouse database that you created in "Chapter 2. Creating a warehouse database" on page 5. Within the warehouse target, you will define the DEMOGRAPHICS_TARGET target table.

You can also specify that the Data Warehouse Center is to generate the target table. This step will be covered in the next lesson.

## Defining a warehouse target

In this exercise, you will define the Tutorial Targets warehouse target, which is a logical definition for the TUTWHS database that you created in "Chapter 2. Creating a warehouse database" on page 5.

To define the warehouse target:
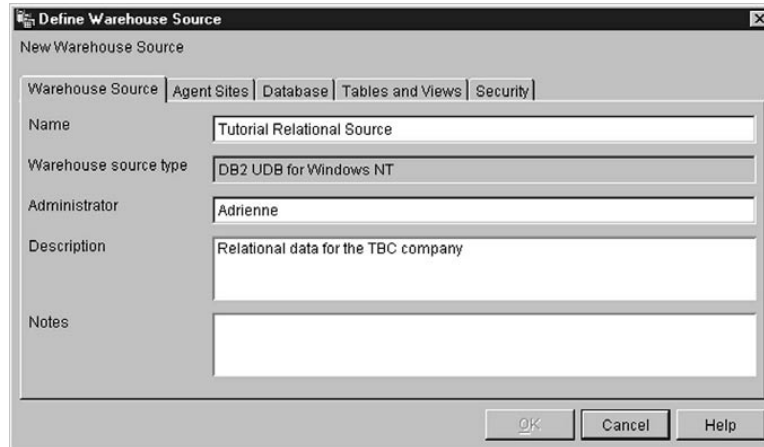1. From the Data Warehouse Center window, right-click the **Warehouse Targets** folder.
2. Click **Define —> DB2 Family —> DB2 UDB for Windows NT**.
   The Define Warehouse Target notebook opens.
3. In the **Name** field, type the business name for the warehouse target:
   ` Tutorial Targets`
4. In the **Administrator** field, type your name as the contact for the warehouse target.
5. In the **Description** field, type a short description of the data:
   `Warehouse for the TBC company`
6. Click the **Database** tab.
7. In the **Database name** field, click or type the name of the database:
   `TUTWHS`

   You are creating the target database in the default system, so you can skip the **System name** field.

8. In the **User ID** field, type the user ID for the Default DWC User.
9. In the **Password** field, type the password for the user ID.
10. In the **Verify password** field, type the password again.

    Use the default values for the rest of the controls on the page.
11. Click the **Security** tab.
12. Select the **Tutorial Warehouse Group**.
13. Click **>** to move the Tutorial Warehouse Group to the **Selected warehouse groups** list.

    Adding the target to the warehouse group authorizes the users in the group (in this case, you) to create steps that use this warehouse target.
14. Click **OK** to save your changes and close the Define Warehouse Target notebook.

## Defining a target table

In this exercise you will define the DEMOGRAPHICS_TARGET table within the Tutorial Targets warehouse target.

To define a warehouse target table:

1. Expand the **Warehouse Targets** tree until you see the **Tables** folder under the **Tutorial Targets** warehouse target.
2. Right-click the **Tables** folder, and click **Define**.

   The Define Warehouse Target Table notebook opens, and you can define the DEMOGRAPHICS_TARGET table.



3. In the **Table schema** list, click or type the user ID under which you created the warehouse database in "Chapter 2. Creating a warehouse database" on page 5.

4. In the **Table name** field, type the name of the target table:

   DEMOGRAPHICS_TARGET

   Because you are creating the tables in the default table space, you can skip the **Table space** and **Index table space** lists.

5. In the **Description** field, type the description of the table:

   Demographics data for sales regions

6. In the **Business name** field, type the business name for the table:

   Demographics Target

7. Verify that the **Data Warehouse Center created table** check box is selected.

   The Data Warehouse Center will create this table when the step that loads the Demographic data runs.

   You use this option when you want the Data Warehouse Center to create the target table. Otherwise, you can use a target table that is already defined.

8. Verify that the **Grant to public** check box is selected.

   This check box specifies that anyone who has access to the database has access to the table.

9. Clear the **Part of an OLAP schema** check box.

   The **Part of an OLAP schema** check box indicates that the table is a dimension table or fact table that is to be exported to OLAP Integration Server. For more information about exporting to OLAP Integration Server, see "Chapter 16. Creating a star schema from within the Data Warehouse Center" on page 103.

   Skip the rest of the controls on the page. For more information about them, see "Defining a warehouse target table" in the online help.

   Leave the Define Warehouse Target Table window open and go on to the next exercise.

## Adding columns to the target table

To add columns to the target table:

1. Click the **Columns** tab.
2. Right-click the empty space in the list.
3. Click **Add**.

   A row is added to the list, and you can define the **State** column, which is one of the main values for the table.

4. Click the **Column name** column, and type STATE.
5. In the **Data type** column, verify that **CHAR** is selected.
6. Click the **Length** column, and type 50.

   Skip the **Precision** and **Scale** columns because they apply to decimal data only.

7. Verify that the **Nullable** check box is selected.

8. In the **Business name** field, type State.

9. Repeat steps 2 on page 33 through 8 using the values in the following table to define the rest of the columns in the table:

| Name | Data type | Length | Allow nulls | Business name |
|---|---|---|---|---|
| CITY | CHAR | 50 | No | City |
| POPULATION | INTEGER | N/A | No | Population |
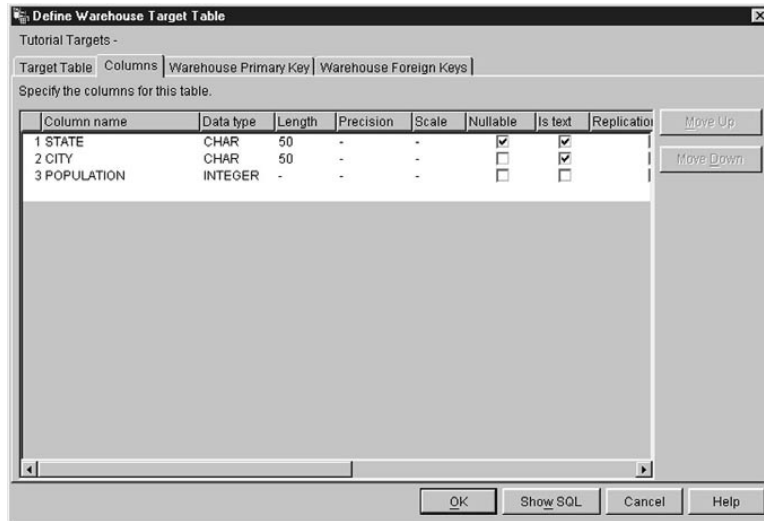
Skip the rest of the notebook. You will use the DB2 Control Center and Data Warehouse Center to define primary and foreign keys on the physical target tables. You can use primary and foreign keys for joins.

10. Click **OK**.

The Define Warehouse Target Table notebook closes. The Demographics table is displayed under the **Tables** folder.

## What you just did

In this lesson, you defined a warehouse target with one target table, DEMOGRAPHICS_TARGET. In "Chapter 8. Defining data transformation and movement" on page 35, you will generate target tables for this warehouse target, including the LOOKUP_MARKET dimension table for the star schema.

# Chapter 8. Defining data transformation and movement

In this lesson, you will define how the Data Warehouse Center is to move and transform data into a format for the data warehouse. First, you will define a *process*, which contains a series of steps in the data transformation and movement process. Next, you will specify the source tables that are to be transformed for the warehouse. Finally, you will define data transformation steps that use two different methods of transformation:

- Load data into the warehouse database with a program
- Select source data and join tables with SQL statements

Specifically, you will define the Build Tutorial Market Dimension process, which performs the following processing:

1. Loading the Demographics file into the warehouse database.
2. Selecting data from the GEOGRAPHIES table and creating a target table.
3. Joining the data in the Demographics table and in the GEOGRAPHIES target table.

The end result is the LOOKUP_MARKET target table.

## Defining a process

In this exercise, you will define the process object for the Build Tutorial Market Dimension process.

To define the process object:

1. From the Data Warehouse Center window, expand the **Subject Areas** tree.
2. Expand the **TBC Tutorial** subject area, which you defined in "Chapter 5. Defining a subject area" on page 21.
3. Right-click the **Processes** folder, and click **Define**.

   The Define Process notebook opens.
4. In the **Name** field, type the name of the process:

   ```
   Build Tutorial Market Dimension
   ```

   The name can be up to 80 characters long and is case-sensitive. The first character of the name must be alphanumeric. You cannot use & as the first character in English. This field is required.
5. In the **Administrator** field, type your name as the contact for the process definition.

6. In the **Description** field, type the description of the process:

   Process to create the LOOKUP_MARKET table

7. Click the **Security** tab.

8. In the **Available warehouse groups** list, click the **Tutorial Warehouse Group**, which you defined in "Defining the warehouse group" on page 18.

9. Click > to move the Tutorial Warehouse Group to the **Selected warehouse groups** list.

   Adding the process to the warehouse group authorizes the users in the group (in this case, you) to open and add objects to the process.

10. Click **OK**.

   The Define Process notebook closes.

## Opening the process

In this exercise, you will open the process so that you can graphically define the flow of data.

To open the Build Tutorial Market Dimension process:

1. Expand the **TBC Tutorial** tree until you see the **Build Tutorial Market Dimension** process.

2. Right-click the **Build Tutorial Market Dimension** process.

3. Click **Open**.

## Adding tables to a process

To define the flow of data, you need to add each source that the steps transform and the target tables that result from the transformation.

In the Build Tutorial Market Dimension process, you will load the Demographics file into the target database. You need to add to the process the source file and the DEMOGRAPHICS_TARGET table for the step. The Demographics source file is part of the Tutorial File Source warehouse source, which you defined in "Chapter 6. Defining warehouse sources" on page 23. The DEMOGRAPHICS_TARGET table is part of the Tutorial Targets warehouse target, which you defined in "Chapter 7. Defining warehouse targets" on page 31.

To add the Demographics file:

1. Click the **Add Data** icon.

2. Click the canvas (the area on the right side of the window) at the spot where you want to place the table. The Add Data window opens.

3. In the **Available source and target tables** list, expand the **Warehouse Sources** tree.

   A list of the available warehouse sources is displayed.

4. Expand the tree for the Tutorial File Source warehouse source.

5. Expand the **Files** tree.

   In the tree, you should see the following file:

   `X:\program files\sqllib\samples\db2sampl\dwc\demographics.txt`, where *X* is the drive on which you installed the sample.

6. Select the Demographics file.

7. Click **>** to add the Demographics file to the **Selected source and target tables** list.

To add the DEMOGRAPHICS_TARGET table:

1. In the **Available source and target tables** list, expand the **Warehouse Targets** tree.

   A list of the available warehouse targets is displayed.

2. Expand the **Tutorial Targets** warehouse target tree.

3. Expand the **Tables** tree.

   The DEMOGRAPHICS_TARGET table is displayed in the list.

4. Select the DEMOGRAPHICS_TARGET table.

5. Click **>** to add the DEMOGRAPHICS_TARGET table to the **Selected source and target tables** list.

## Adding the SAMPLETBC.GEOGRAPHIES table to the process

In the next part of this exercise, you need to add the source table. When you define a step that selects data from the SAMPLTBC.GEOGRAPHIES table, you can specify that the Data Warehouse Center automatically generate a target table, so you do not need to add a target table.

To add the SAMPLTBC.GEOGRAPHIES source table:

1. Expand the **Warehouse Sources** tree until you see the **Tutorial Relational Source** warehouse source.

2. Expand the **Tutorial Relational Source** tree until you see the SAMPLTBC.GEOGRAPHIES table.

3. Select the **SAMPLTBC.GEOGRAPHIES** table.

4. Click **>** to add the SAMPLTBC.GEOGRAPHIES table to the **Selected source and target tables** list.

5. Click **OK** to add the file and tables that you selected to the process.

The file and tables that you selected are displayed in the Process Model window.

6. Click the demographics.txt icon and move it up until you see the DEMOGRAPHICS_TARGET table.

7. Arrange the icons in the following order from top to bottom: demographics.txt, DEMOGRAPHICS_TARGET. Leave room for one icon between demographics.txt and DEMOGRAPHICS_TARGET. Place the Geographies icon to the right of the demographics.txt icon.
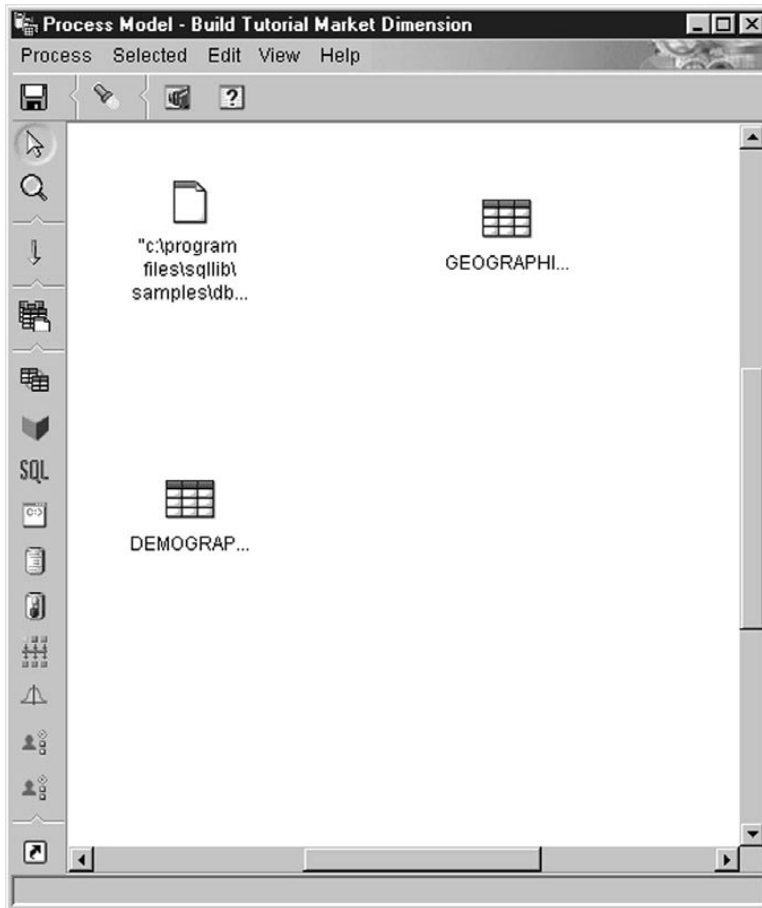


The last step will use the DEMOGRAPHICS_TARGET table and the SAMPLTBC.GEOGRAPHIES table as sources, so you do not need to specify sources for the step. You can specify that the Data Warehouse Center automatically generate the resulting target table, LOOKUP_MARKET, so you do not need to specify a target table for the step.

## Adding steps to the process

Now, you need to add the steps that define how the source data is to be transformed into the target data. You must define three steps:

**Load Demographics Data**

A DB2 program step that loads data from the Demographics file into a table in the TBC Warehouse database, which you created in "Chapter 2. Creating a warehouse database" on page 5.

**Select Geographies Data**

An SQL step that selects columns from the SAMPLTBC.GEOGRAPHIES source table.

**Join Market Data**

An SQL step that joins the SAMPLTBC.GEOGRAPHIES and Demographics tables and writes the joined table into the Tutorial Warehouse database.

## Defining the Load Demographics Data step

To define the Load Demographics Data step:

1. From the palette (the toolbar on the left side of the window), click the **DB2 Programs** icon

   

   Each program option in the menu represents a *program group*, which is a grouping of similar programs.

2. Click **DB2 UDB —> Load**.

3. Click the spot on the canvas between the demographics.txt file and the DEMOGRAPHICS_TARGET table.

   An icon for the step is added to the window between the demographics.txt file and the DEMOGRAPHICS_TARGET table.

4. Right-click the new step.

5. Click **Properties**.

   The Properties notebook for the step opens.

6. In the **Name** field, type the name of the step:

   Load Demographics Data

7. In the **Administrator** field, type your name as the contact for this step.

8. In the **Description** field, type the description of the step:

   Loads demographics data into the warehouse.

9. Click **OK**.

   The Properties notebook for the step closes.

10. Click the **Task Flow** icon

11. Click the **Data Link** icon

    The Data Link icon defines the flow of data from the source file, through transformation by a step, to the target table.

12. Click the middle of the demographics.txt file, hold down the mouse button, and drag to the Load Demographics Data step.

    The Data Warehouse Center draws a line between the file and the step. The line indicates that the demographics.txt file contains the source data for the step.

13. Click the middle of the Load Demographics Data step, hold down the mouse button, and drag the pointer to the DEMOGRAPHICS_TARGET table.
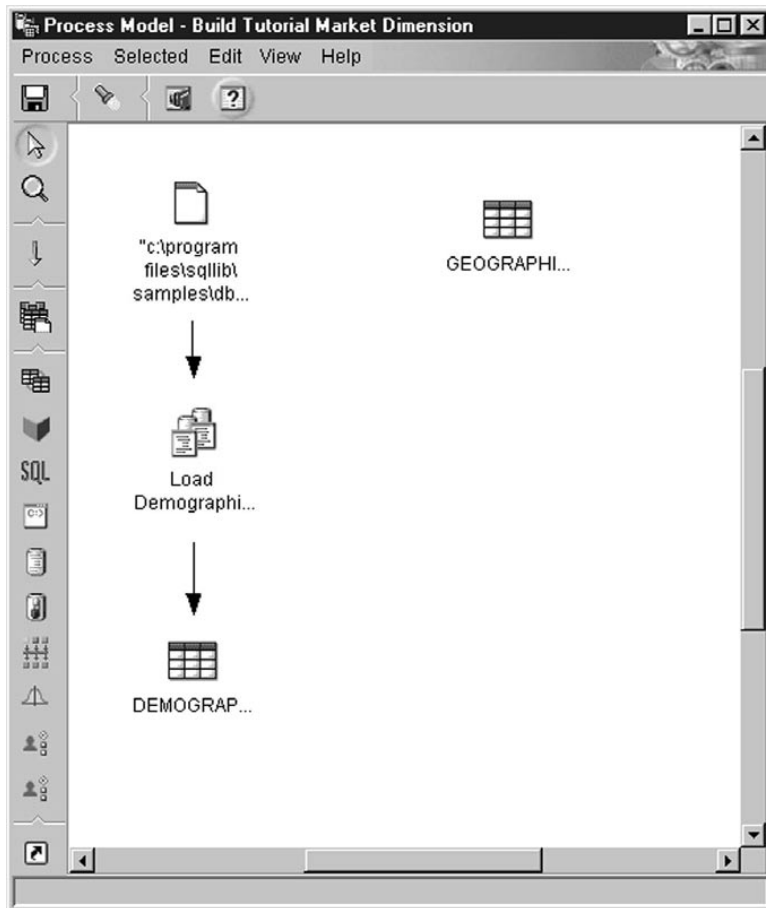
This line indicates that the DEMOGRAPHICS_TARGET table contains the target data for the step.

14. Right-click the **Load Demographics Data** step.
15. Click **Properties**.

The Properties notebook for the step opens.

16. Click the **Parameters** tab.
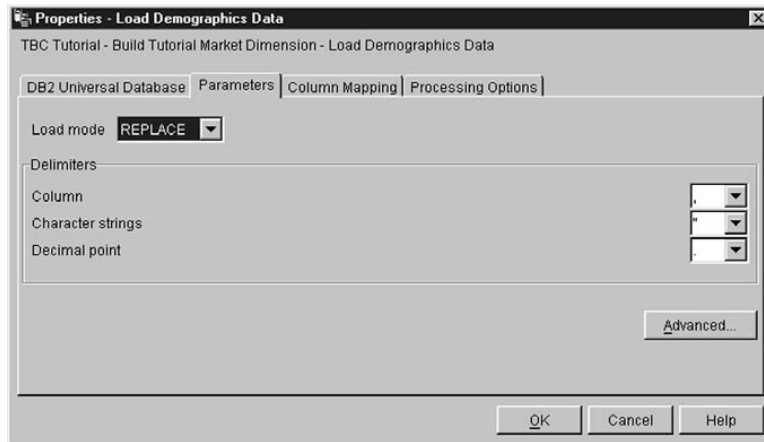


17. From the **Load mode** list, click **REPLACE**.

   For this exercise, you will use the default values on the page.

   You can skip the Column Mapping page because the Load program does not use the column mapping to generate the target table. It generates the target table based on the source file.

18. Click **OK**.

   The Properties notebook for the step closes.

## Defining the Select Geographies step

In this exercise, you will define the Select Geographies step:

1. From the palette, click the **SQL** icon

   ## SQL

2. Click a spot on the canvas below the GEOGRAPHIES table to add a step icon to the window.

3. Right-click the step.

4. Click **Properties**.

   The Properties notebook for the step opens.

5. In the **Name** field, type the name of the step:

   Select Geographies Data

6. In the **Administrator** field, type your name as the name of the contact for the step.

7. In the **Description** field, type the description of the step:

   Selects Geographies data from the warehouse source

8. Click **OK**.

The Properties notebook for the step closes.

9. Click the **Task Flow** icon

10. Click the **Data Link** icon

11. Click the middle of the Geographies source table, hold down the mouse button, and drag it to the middle of the Select Geographies Data step.

    The Data Warehouse Center draws a line that indicates that the Geographies source table contains the source data for the step.

    You do not need to link a target table to the step because you will use the Data Warehouse Center to create a target table.

## Selecting columns from the Geographies source table

To select columns from the Geographies source table:

1. Right-click the Select Geographies Data step.
2. Click **Properties**.

   The Properties notebook for the step opens.
3. Click the **SQL Statement** tab.
4. Click the **Build SQL** push button.

   The Build SQL wizard opens. The **Tables** page is displayed.

5. Verify that SAMPLTBC.GEOGRAPHIES is listed.
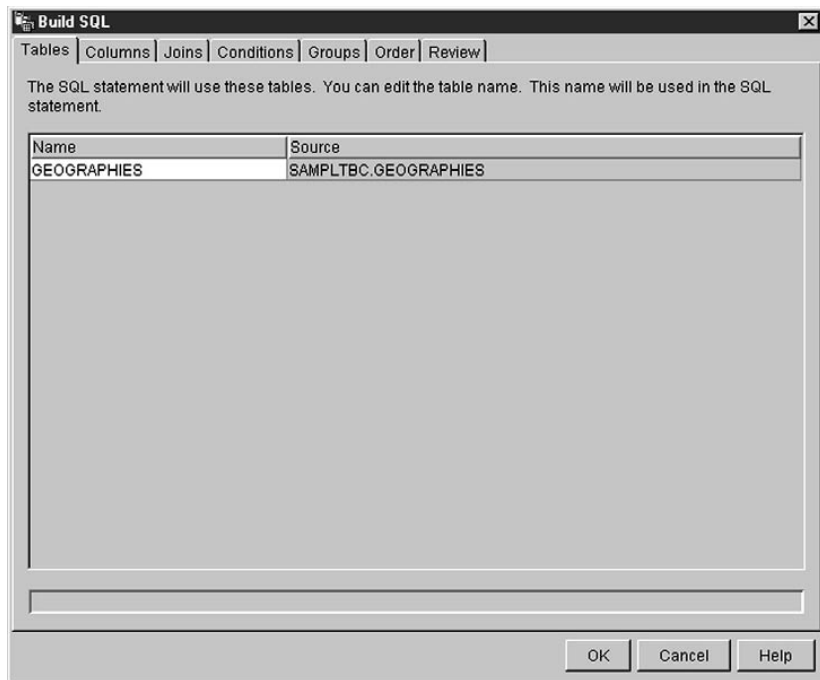


6. Click the **Columns** tab.

7. Click **>>** to add all the columns from the SAMPLTBC.GEOGRAPHIES table to the **Selected columns** list.



8. Click the **Review** tab to look at the SQL statement that you just created.
9. Click **OK**.

The Build SQL wizard closes. The SQL statement that you created is displayed on the SQL Statement page.

```
Properties - Select Geographies Data                                        [X]

TBC Tutorial - Build Tutorial Market Dimension - Select Geographies Data

SQL | SQL Statement | Column Mapping | Processing Options |

SQL statement

 ┌──────────────────────────────────────────────────────────┐   ┌──────────┐
 │ SELECT                                                    │   │Build SQL │
 │   SAMPLTBC.GEOGRAPHIES.REGION AS REGION,                  │   └──────────┘
 │   SAMPLTBC.GEOGRAPHIES.REGION_TYPE_ID AS REGION_TYPE_ID,  │   ┌──────────┐
 │   SAMPLTBC.GEOGRAPHIES.STATE AS STATE,                    │   │   Edit   │
 │   SAMPLTBC.GEOGRAPHIES.STATE_TYPE_ID AS STATE_TYPE_ID,    │   └──────────┘
 │   SAMPLTBC.GEOGRAPHIES.CITY_ID AS CITY_ID,                │   ┌──────────┐
 │   SAMPLTBC.GEOGRAPHIES.CITY AS CITY                       │   │   Test   │
 │ FROM                                                      │   └──────────┘
 │   SAMPLTBC.GEOGRAPHIES                                    │   ┌──────────┐
 │                                                          │   │  Clear   │
 │                                                          │   └──────────┘
 └──────────────────────────────────────────────────────────┘

                                          OK  |  Cancel  |  Help
```

10. Click **Test** to test the SQL that you just created.

   The Data Warehouse Center returns sample results of your SELECT statement. Compare your results to the results that you got in the sample shown in the topic, "Viewing table data" on page 9.

11. Click **Close** to close the window.

   Now that you have specified which columns the SQL step will access in the source table, you are ready to create the target table. Leave the Properties notebook for the step open and go to the exercise, "Creating the GEOGRAPHIES_TARGET table".

## Creating the GEOGRAPHIES_TARGET table

In this exercise, you will create the GEOGRAPHIES_TARGET table using the Column Mapping page in the Select Geographies Data step.

To create the GEOGRAPHIES_TARGET table:

1. Click the **Column Mapping** tab.

   The Source columns are displayed on the left side of the page, and the target columns list is on the right side of the page. There are no target columns in the list because you did not specify the parameters for creating the target table.

2. Click the **Generate Default Table** push button.

   The Generate Default Table window opens.

3.  In the **Warehouse target** list, click **Tutorial Targets**.

    The warehouse target is the database or file system in which the target table is created.

4.  In the **Table schema** list, type or click the table schema for the target table. Use the same schema under which you created the other tables.

    Because you are creating the table in the default table space, you can skip the **Table space** list.

5.  In the **Table name** field, type:

    GEOGRAPHIES_TARGET

6.  Click **OK** to close the Generate Default Table window.

    The columns of the GEOGRAPHIES_TARGET table are displayed in the target columns list on the right side of the Column Mapping page.

7.  Verify that the source columns map to the correct target columns.
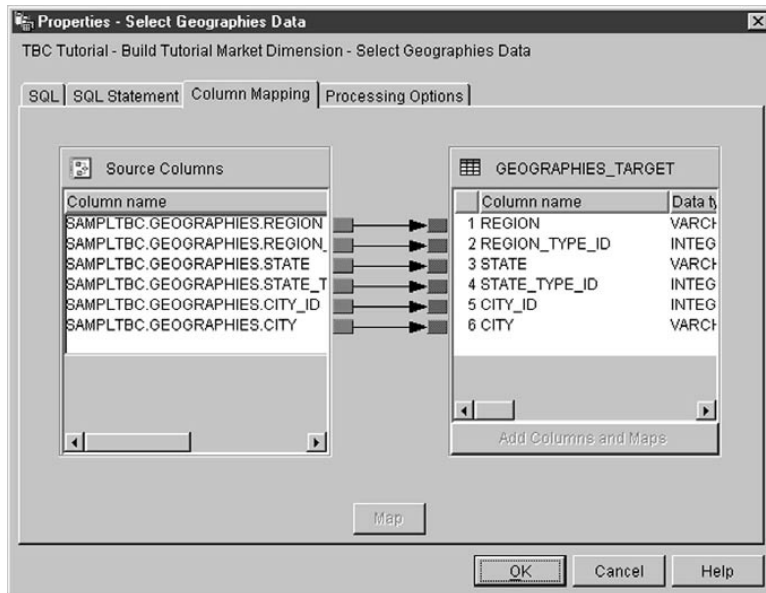


8.  Click the **Processing Options** tab.
9.  Verify that **Replace** is selected in the **Population type** list.
10. Verify that the **Run on demand** check box is selected.

    For this exercise, you will use the default values on this page.

11. Click **OK**.

    The Properties notebook for the step closes. The Data Warehouse Center creates a target table called GEOGRAPHIES_TARGET and links it to the step.

## Specifying properties for the GEOGRAPHIES_TARGET table

In this exercise, you will specify the properties for the GEOGRAPHIES_TARGET table that you created in the exercise, "Creating the GEOGRAPHIES_TARGET table" on page 46.

To specify the properties of the GEOGRAPHIES_TARGET table:

1. In the Process Model window, right-click the GEOGRAPHIES_TARGET table, and click **Properties**.
2. In the **Business name** field, type a descriptive name for the table:
   Geographies Target
3. Clear the **Part of an OLAP schema** check box.
4. Click **OK**. The Properties notebook for the table closes.

## Defining the Join Market Data step

To define the Join Market Data step:

1. From the palette, click the **SQL** icon.
2. Click the spot on the canvas below the two columns of steps and tables.
   An icon for the step is added to the window.
3. Right-click the new step.
4. Click **Properties**.
   The Properties notebook for the step opens.
5. In the **Name** field, type the name of the step:
   Join Market Data
6. In the **Administrator** field, type your name as the contact for the step.
7. In the **Description** field, type the description of the step:
   Joins the Geographies table with the Demographics table
8. Click **OK**.
   The Properties notebook for the step closes.
9. Click the **Task Flow** icon



10. Click the **Data Link** icon



11. Click the middle of the GEOGRAPHIES_TARGET table, hold the mouse button down, and drag it to the Join Market Data step.
    The Data Warehouse Center draws a line that indicates that the GEOGRAPHIES_TARGET table contains source data for the step.

Because you will specify that the Data Warehouse Center is to create the target table, you do not need to link a target table to the step.

12. Repeat steps 9 on page 48 to 11 on page 48 with the DEMOGRAPHICS_TARGET table and the Join Market Data step.

   The Data Warehouse Center draws a line that indicates that the DEMOGRAPHICS_TARGET table contains source data for the step.

## Joining the DEMOGRAPHICS_TARGET and GEOGRAPHIES_TARGET tables

To join the DEMOGRAPHICS_TARGET and GEOGRAPHIES_TARGET tables:

1. Right-click the Join Market Data step.
2. Click **Properties**.

   The Properties notebook for the step opens.
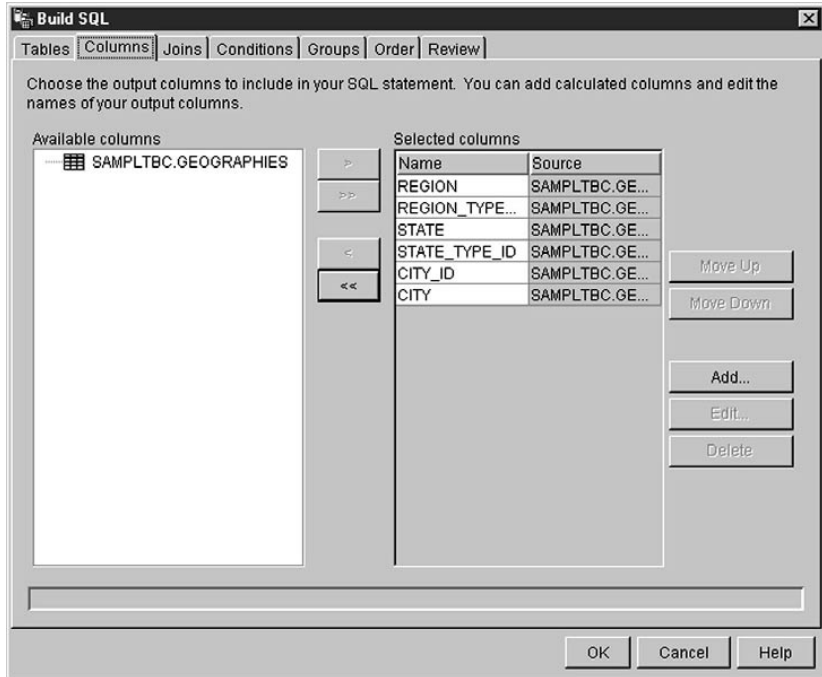3. Click the **SQL Statement** tab.
4. Click the **Build SQL** push button to have the Data Warehouse Center create SQL. (Otherwise, you can create your own SQL.)

   The Build SQL window opens.
5. On the Tables page, verify that the DEMOGRAPHICS_TARGET and GEOGRAPHIES_TARGET tables are listed.
6. Click the **Columns** tab.
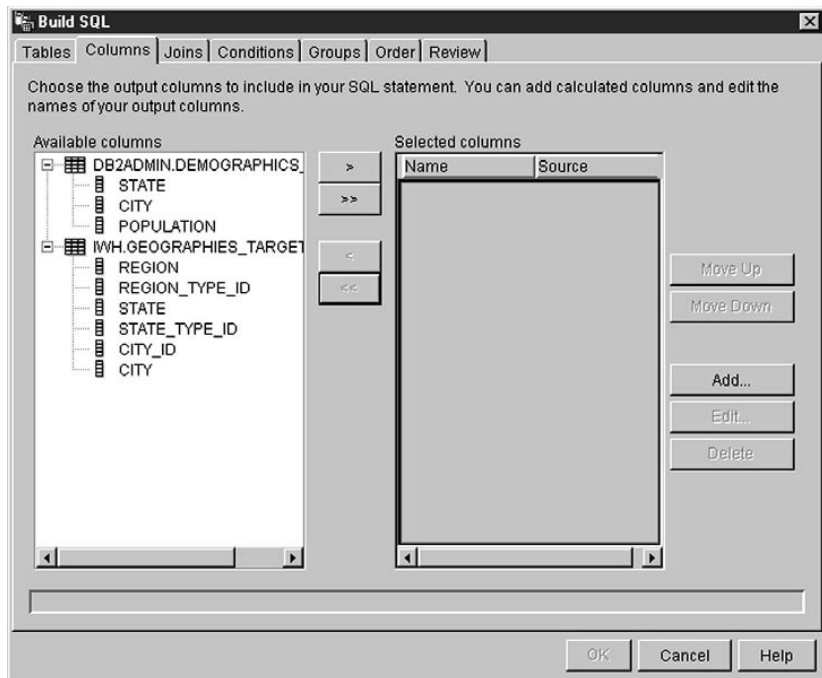
7. Click **>>** to add all the columns from the GEOGRAPHIES_TARGET table and the DEMOGRAPHICS_TARGET table to the **Selected columns** list.

8. From the **Selected columns** list, click **DEMOGRAPHICS_TARGET.STATE**.

9. Click **<** to move the DEMOGRAPHICS_TARGET.STATE column to the **Available columns** list.

10. Click **DEMOGRAPHICS_TARGET.CITY**.

11. Click **<** to move the DEMOGRAPHICS_TARGET.CITY column to the **Available columns** list.

    You do not need the DEMOGRAPHICS_TARGET.STATE and DEMOGRAPHICS_TARGET.CITY columns because they are already defined in the GEOGRAPHIES_TARGET table. You will join the tables using the CITY column because it is unique to both tables.

12. Click the **Add** push button.

    The Expression Builder window opens.

13. In the **Expression** field, type the following CASE statement:

```
case
when POPULATION < 9000000   then 1
when POPULATION < 15000000  then 2
Else  3
end
```
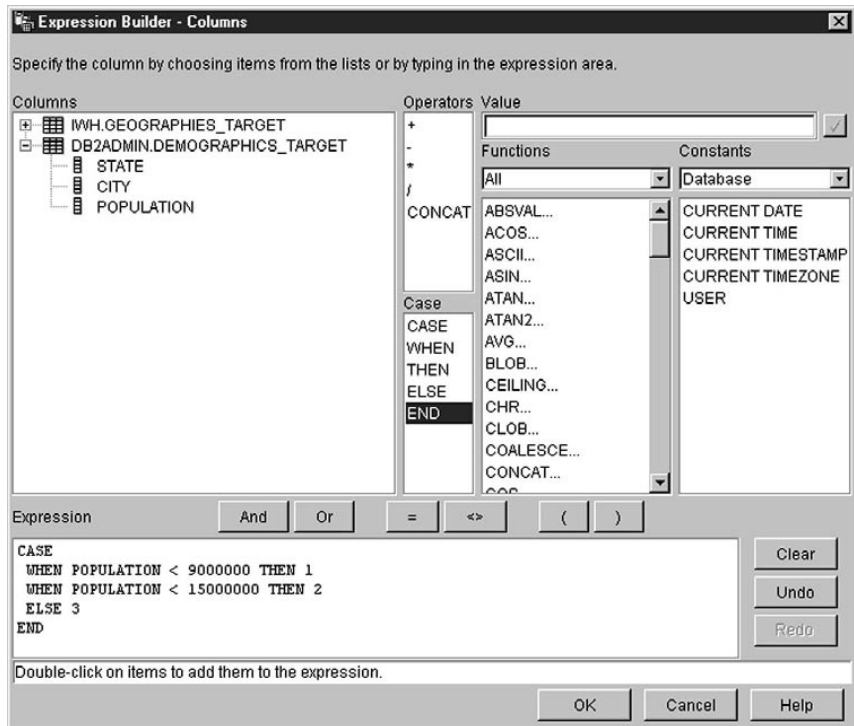
14. Click **OK**.

   A new column is added to the **Selected columns** list.

15. Click the **Name** field of the new column, and type the name of the column:

   SIZE_ID

16. Press Enter.

17. Click **Move Up** until the SIZE_ID column is above the POPULATION column.

18. Click the **Joins** tab.

19. In the GEOGRAPHIES_TARGET table, click the CITY column.

20. In the DEMOGRAPHICS_TARGET table, click the CITY column.

21. Click **Join**.

   The Build SQL wizard draws a line between the CITY columns, which indicates that the tables are joined on that column.

```
Build SQL                                                              [X]
Tables | Columns | Joins | Conditions | Groups | Order | Review |
Specify the join conditions that will be used to join tables.
                                                              [  Join  ]
 IWH.GEOGRAPHIES_TARGET                                       [ < ][ > ]
 REGION                                                       [ Unjoin ]
 REGION_TYPE_ID              DB2ADMIN.DEMOGRAPHICS_TARGET
 STATE                        STATE                           [Join Type...]
 STATE_TYPE_ID                CITY
 CITY_ID                      POPULATION
 CITY

 Inner join: Includes only rows from DB2ADMIN.DEMOGRAPHICS_TARGET and IWH.GEOGRAPHIES_TARGET
 where the joined columns are equal.

                                               [ OK ] [ Cancel ] [ Help ]
```

22. Click the **Review** tab to see the SQL statement that you just created.
23. Click **OK**.

    The Build SQL wizard closes.

### Creating the LOOKUP_MARKET table

To create the LOOKUP_MARKET table:

1. Click the **Column Mapping** tab.
2. Click the **Generate Default Table** push button.
   The Generate Default Table opens.
3. In the **Warehouse target** list, click **Tutorial Targets**.
4. In the **Table schema** list, type or click **IWH**. Skip the **Table space** list.
5. In the **Table name** field, type:
   LOOKUP_MARKET
6. Click **OK**.

   The Generate Default Table window closes. The target columns are displayed in the target columns list on the right side of the Column Mapping page.
7. Click the **Processing Options** tab.
8. In the **Population type** list, verify that **Replace** is selected.

9. Select the **Run on demand** check box to run the step on demand.

   For this exercise, you will use the default values for the rest of this page.

10. Click **OK**.

    The Properties notebook for the step closes, and the LOOKUP_MARKET table is displayed in the Process Model window.

## Defining the properties of the LOOKUP_MARKET table

To define the properties of the LOOKUP_MARKET table:

1. Right-click the LOOKUP_MARKET table, and click **Properties**.
2. Click **Properties**.

   The Properties notebook for the table opens.
3. In the **Description** field, type a description of the table:

   Market dimension data
4. Verify that the **Part of an OLAP schema** check box and **Dimension table** radio button are selected.

   The LOOKUP_MARKET table is one of the dimension tables that you will include in a star schema in "Chapter 16. Creating a star schema from within the Data Warehouse Center" on page 103.
5. Click the **Columns** tab.
6. Clear the **Nullable** check box for the CITY_ID column. You will define this column as the primary key for the table in "Chapter 11. Defining keys on target tables" on page 71.
7. Click the **Warehouse Primary Key** tab.
8. Click **CITY_ID** in the **Available columns** list.
9. Click **>** to move the CITY_ID column to the **Warehouse primary key columns** list.
10. In the **Constraint name** field, type:

    "Whse Market PK"
11. Click **OK**. The Properties notebook for the table closes.

    Your process now looks like this:
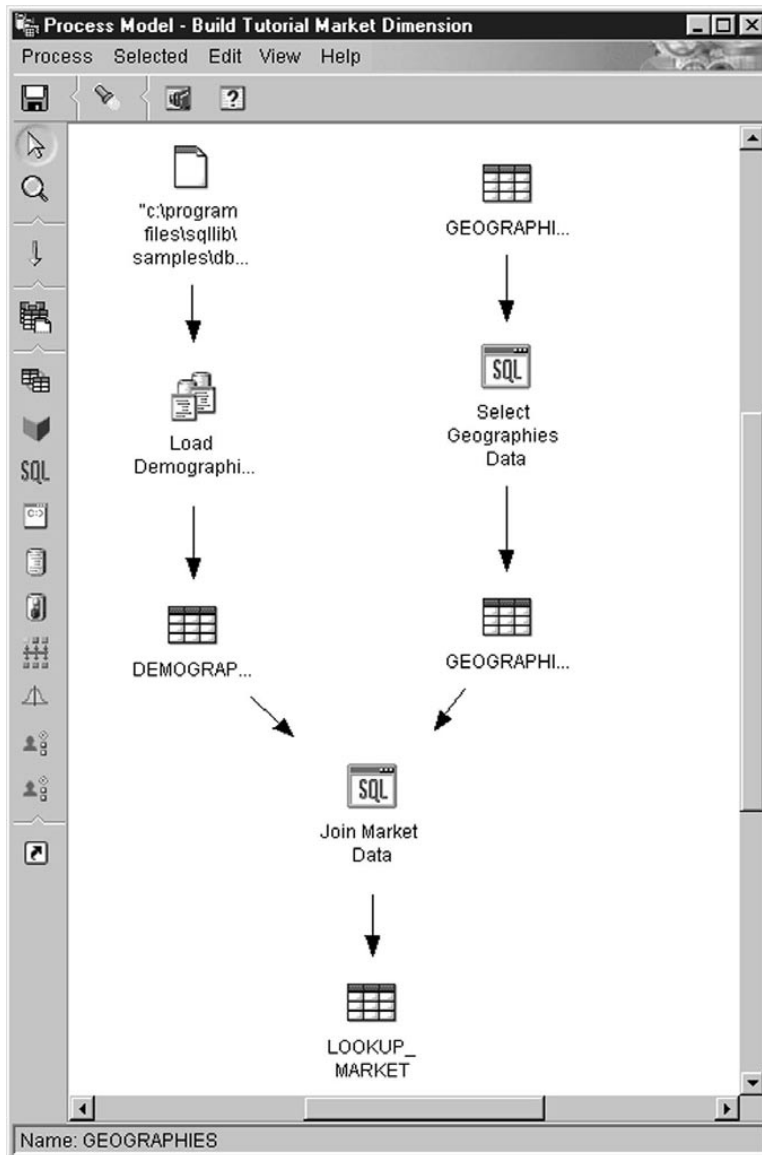
12. Click the Save icon in the toolbar to save the process.



13. Close the Process Model window.

### What you just did

You defined the LOOKUP_MARKET dimension table in the Build Tutorial Market Dimension process, which contains three steps:

- Load Demographics Data
- Select Geographies Data
- Join Market Data

The sources and targets for each of the steps are shown in the following table:

| Step | Sources | Target |
|------|---------|--------|
| Load Demographics Data | Demographics file | DEMOGRAPHICS_TARGET table |
| Select Geographies Data | Geographies table | GEOGRAPHIES_TARGET table |
| Join Market Data | Demographics Target table and Geographies Target table | LOOKUP_MARKET table |

You added the data links for each step when you defined the properties of each step. Another way you can accomplish this task is to add all the steps in the process at one time, link the steps to their sources and targets, and then define the properties of each step. The Data Warehouse Center assigns default names to the steps, which you can change in the Properties notebook for the step.

## Defining the rest of the tables for the star schema (optional)

The table that you created in the previous exercise, LOOKUP_MARKET, is one of the dimension tables described in the example in "Tutorial business problem" on page vii. This exercise covers the steps for building the rest of the dimension tables and the fact table in the star schema.

You need to complete the steps in this exercise before you can do the following lessons:
- "Chapter 11. Defining keys on target tables" on page 71
- "Chapter 14. Cataloging data in the warehouse for end users" on page 85
- "Chapter 15. Working with business metadata" on page 91
- "Chapter 16. Creating a star schema from within the Data Warehouse Center" on page 103

If you want to skip this exercise, go to "Chapter 9. Testing warehouse steps" on page 61.

When you define each table, you must define a new process for the table. Instead of defining a step for the process, you will copy the step that is

defined in the sample. When you copy the step, the Data Warehouse Center copies the sources that the step uses and generates a target table.

To define the Product dimension table:

1. Define a new process by following the steps in "Defining a process" on page 35. Name the process:

   Build Tutorial Product Dimension

2. Under the **Subject Areas** folder in the Data Warehouse Center window, expand the **TBC Sample** tree until you see the **Build Sample Product Dimension** process.

3. Right-click the **Build Sample Product Dimension** process.

4. Click **Open**.

   The Process Model window opens.

5. Right-click the **Select Product** step.

6. Click **Copy**.

   The Copy Step window opens.

7. In the **Name** field, type the name for the copy of the step:

   Tutorial Select Product

8. In the **New process** field, click the name of the process to which you want to copy the step:

   Build Tutorial Product Dimension

9. Verify that the **Copy Target Table** check box is selected.

10. In the **Warehouse target** field, click the **Tutorial Targets** warehouse target in the list. The warehouse target will be the same for the fact table and each dimension table that you define in this tutorial.
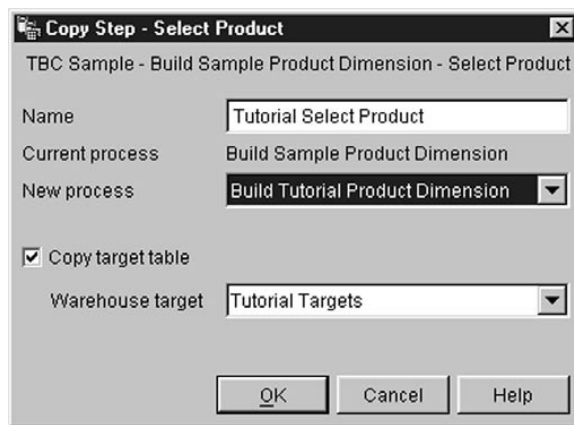


11. Click **OK**.

The step and its sources are copied to the Build Tutorial Product Dimension process. The Data Warehouse Center creates the corresponding target table.

12. Close the Process Model window.
13. Open the Build Tutorial Product Dimension process by following the steps in "Opening the process" on page 36.
14. Verify that the process includes the following objects:
    - The PRODUCT source tables
    - The Tutorial Select Product step
    - The "SelectProd_T" target table
15. Rename the "SelectProd_T" target table to:
    ```
    LOOKUP_PRODUCT
    ```

    To rename the target table:
    a. In the Process Model window, right-click the "SelectProd_T" target table.
    b. Click **Properties**.

       The Properties notebook for the table opens.
    c. In the **Table name** field, type:
       ```
       LOOKUP_PRODUCT
       ```
    d. Verify that the **Part of an OLAP schema** check box and **Dimension table** radio button are selected.
    e. Click **OK** to save your changes and close the Properties notebook for the table.

Repeat this procedure for the Time and Scenario dimension tables, substituting the values in the following table where appropriate.

| Dimension | Tutorial process | Sample process | Sample step to copy | New tutorial step name | Source tables | Target table | New target table name |
|---|---|---|---|---|---|---|---|
| Time | Build Tutorial Time Dimension | Build Sample Time Dimension | Select Time | Tutorial Select Time | TIME | "SelectTime _T" | LOOKUP_ TIME |
| Scenario | Build Tutorial Scenario Dimension | Build Sample Scenario Dimension | Select Scenario | Tutorial Select Scenario | SCENARIO | "SelectScen _T" | LOOKUP_ SCENARIO |

In the previous exercise, you defined the dimension tables in the star schema. In this exercise, you will define the fact table in the star schema.

To define the fact table:

1. Define a new process by following the steps in "Defining a process" on page 35. Name the process:

   ```
   Build Tutorial Fact Table
   ```

2. Under the **Subject Areas** folder in the Data Warehouse Center window, expand the **TBC Sample** tree until you see the **Build Sample Fact Table** process.

3. Right-click the **Build Sample Fact Table** process.

4. Click **Open**.

   The Process Model window opens.

5. Right-click the **Fact Table Join** step.

6. Click **Copy**.

   The Copy Step window opens.

7. In the **Name** field, type the name for the copy of the step:

   ```
   Tutorial Fact Table Join
   ```

8. In the **New process** field, select the name of the process to which you want to copy the step:

   ```
   Build Tutorial Fact Table
   ```

9. Verify that the **Copy Target Table** check box is selected.

10. In the **Warehouse target** field, select the **Tutorial Targets** warehouse target from the list.

11. Click **OK**.

    The step and its sources are copied to the Build Tutorial Fact Table process. The Data Warehouse Center generates the corresponding target table.

12. Close the Process Model window.

13. Open the Build Tutorial Fact Table process by following the steps in "Opening the process" on page 36. If it is already open, close it, then open it again.

14. Verify that the process includes the following objects:

    - The SALES, INVENTORY, and PRODUCTION_COSTS source tables
    - The Tutorial Fact Table Join step
    - The "FactTable_T" target table

15. Rename the "FactTable_T" target table:

    ```
    FACT_TABLE
    ```

    To rename the target table:

    a. In the Process Model window, right-click the "FactTable_T" target table.

    b. Click **Properties**.

The Properties notebook for the table opens.

c. In the **Table name** field, type:

FACT_TABLE

d. Verify that the Part of an **OLAP schema** check box and the **Fact table** radio button are selected.

e. Click **OK** to save your changes and close the Properties notebook for the table.

## What you just did

You now have five processes, with their associated sources, targets, and steps, defined in the Data Warehouse Center:

- Build Tutorial Market Dimension
- Build Tutorial Product Dimension
- Build Tutorial Scenario Dimension
- Build Tutorial Time Dimension
- Build Tutorial Fact Table

# Chapter 9. Testing warehouse steps

In this lesson, you will populate the LOOKUP_MARKET table by running the steps that you created in "Chapter 8. Defining data transformation and movement" on page 35:

- Load Demographics Data
- Select Geographies Data
- Join Market Data

You will then verify the results of running them.

Before you run the steps, you must promote them to test mode. Up to this point, the steps you created were in development mode. In development mode, you can change any of the specifications for the step. When you promote the step to test mode, the Data Warehouse Center creates the target table for the step. Therefore, after you promote a step to test mode, you can make only those changes that are not destructive to the target table. For example, you can add columns to a target table when its associated step is in test mode, but you cannot remove columns from the target table.

After you promote the steps to test mode, you will run each step individually. In a later lesson, you will specify that the steps run in sequence.

## Testing the Load Demographics Data step

In this exercise, you will promote and run the Load Demographics Data step. Then you will promote the rest of the steps in the Build Tutorial Market Dimension process.

To promote the Load Demographics Data step:

1. From the Process Model window for the Build Tutorial Market Dimension process, right-click the Load Demographics Data step.
2. Click **Mode–> Test**.

   The Data Warehouse Center starts to create the target table, and displays a progress window. Wait until the Data Warehouse Center finishes processing before you start the next procedure.

   A lock appears on the step icon to indicate that only non-destructive changes can be made to the step.

To verify that the Demographics target table was created:

1. If the Control Center is not open, then from the main Data Warehouse Center window, click **Tools –> Control Center** to open the DB2 Control Center.
2. Expand the object tree until you see TUTWHS, the warehouse database that you created in "Chapter 2. Creating a warehouse database" on page 5.
3. Expand the **TUTWHS** database tree.
4. Click the **Tables** folder.

   A list of tables displays in the right panel of the window.
5. Verify that the DEMOGRAPHICS_TARGET table is there.

   If you are looking in the Control Center, right-click the **Tables** folder and click **Refresh** to refresh the list of tables in the right panel.

To test the Load Demographics Data step:
1. From the Process Model window for the Build Tutorial Market Dimension, right-click the Load Demographics Data step.
2. Click **Test**.

To verify the results of the step's processing:
1. From the DB2 Control Center, right-click the DEMOGRAPHICS_TARGET table.
2. Click **Sample Contents**.

   The DB2 Control Center displays a subset of the data in the table.
3. Click **Close**.

Repeat the steps in this lesson for the Select Geographies Data step and the Join Market Data step. The target table for the Select Geographies Data step is GEOGRAPHIES_TARGET. The target table for the Join Market Data step is LOOKUP_MARKET.

## Promoting the rest of the steps in the star schema (optional)

To create the rest of the tables in the star schema, you must promote the steps that you created in "Defining the rest of the tables for the star schema (optional)" on page 55. This section is optional, but if you do not complete the steps in this section, you will not be able to do the following lessons:
- "Chapter 11. Defining keys on target tables" on page 71
- "Chapter 14. Cataloging data in the warehouse for end users" on page 85
- "Chapter 15. Working with business metadata" on page 91
- "Chapter 16. Creating a star schema from within the Data Warehouse Center" on page 103

If you want to skip this section, go to "Chapter 10. Scheduling warehouse processes" on page 65.

To promote the steps, open the process that contains the steps, and follow the procedure in step 1 on page 61 through 5 on page 62. You are not required to test the remaining steps.

Promote the following steps:

| Tutorial Process | Tutorial Step | Warehouse Database | Target Table |
|---|---|---|---|
| Build Tutorial Product Dimension | Tutorial Select Product | TUTWHS | LOOKUP_PRODUCT |
| Build Tutorial Time Dimension | Tutorial Select Time | TUTWHS | LOOKUP_TIME |
| Build Tutorial Scenario Dimension | Tutorial Select Scenario | TUTWHS | LOOKUP_SCENARIO |
| Build Tutorial Fact Table | Tutorial Fact Table Join | TUTWHS | FACT_TABLE |

## What you just did

In this lesson, you promoted the Load Demographics Data, Select Geographies Data, and Join Market Data steps to test mode. Then you ran them to test them. In "Chapter 10. Scheduling warehouse processes" on page 65, you will schedule these steps to run automatically.

# Chapter 10. Scheduling warehouse processes

In this lesson, you will specify that the steps in the Tutorial Market process are to run in the following sequence:

1. Load Demographics Data
2. Select Geographies Data
3. Join Market Data

Then you will specify that the Load Demographics Data step is to run at a scheduled time. You will activate the schedule by promoting the steps in the process to production mode.

## Running steps in sequence

To specify that the steps are to run in sequence:

1. From the Data Warehouse window, right-click on the **Build Tutorial Market Dimension** process and click **Open**.
2. From the Process Model window, click the **Task Flow** icon:

    

3. Click the **On Success** icon (the green arrow).

    **On Success** indicates that a step is to be started only if the step before it ran successfully. You can also select the following states:

    **On Completion**
    Indicates that a step is to be started when the step before it runs successfully or fails. (The blue arrow.)

    **On Failure**
    Indicates that a step is to be started only if the step before it failed. (The red arrow.)

    For more information, see "Scheduling a step" in the online help.

4. Click the Load Demographics Data step, which is the step that is to run first.
5. Hold down the mouse button and drag the mouse to the Select Geographies Data step, which is to run after the Load Demographics step.
6. Release the mouse button.

    A green arrow that represents the task flow is displayed on the canvas between the two steps.

7. Repeat steps 4 to 6 with the Select Geographies Data step and the Join Market Data step. This specifies that the Select Geographies Data step will run before the Join Market Data step.
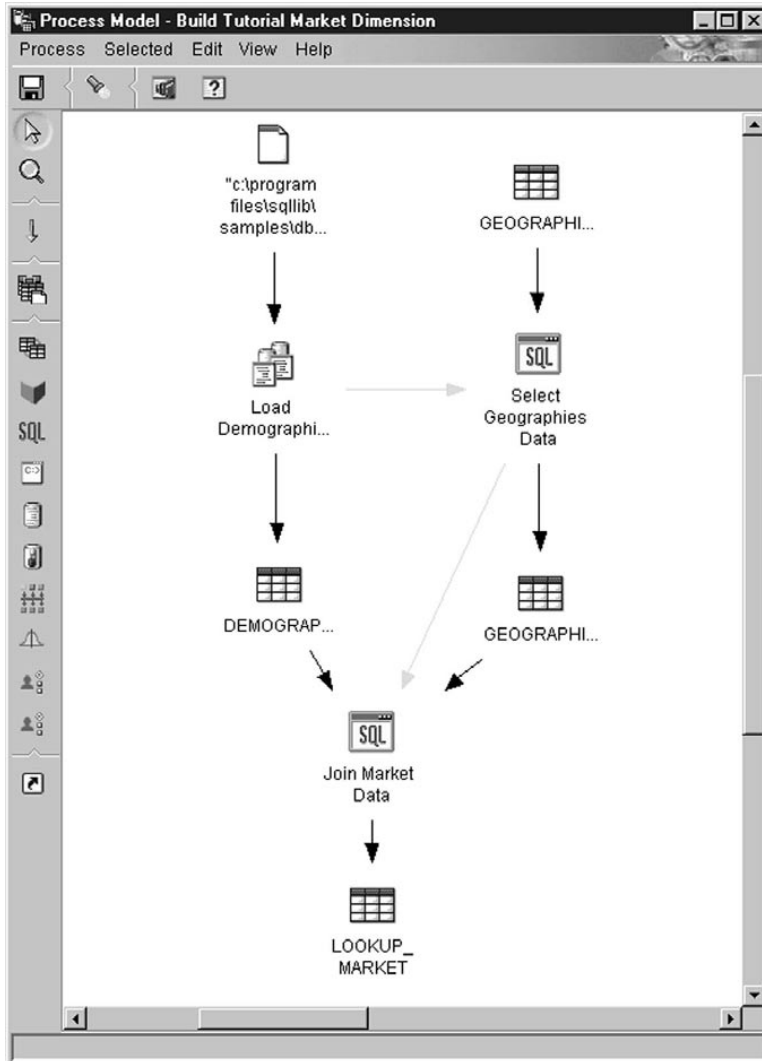


The steps will now run in the order that is listed in the introduction to this lesson.

## Scheduling the first step

Now, you will schedule the Load Demographics Data step, to start at a specified date and time. The Data Warehouse Center will start the step at the time you specify. When the Load Demographics Data step finishes running, the Data Warehouse Center starts the next step in the sequence that you defined in the previous section.

When you schedule a step, you can specify one or more dates and time on which the step is to run. You can also specify the step is to run once or is to run at a specified interval, such as every Saturday.

To schedule the Load Demographics Data step:

1. Right-click the Load Demographics Data step, and click **Schedule**.
   The Schedule notebook opens.
2. From the **Interval** list, click **Yearly**.
3. In the **Start date and time** field, accept the default of the current date, and specify a time that is a few minutes from the current time.
   The step will run at the time you select.
4. In the **End** field, accept the default that the schedule is to run indefinitely.
5. Click **Add**.

The schedule is added to the **Schedule list**.
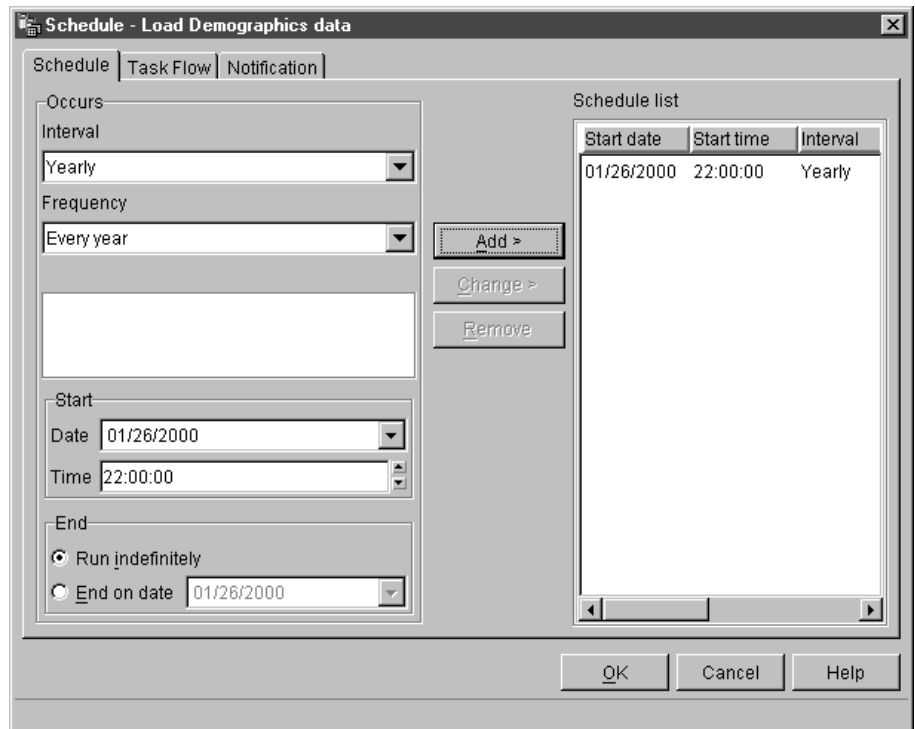


6. Click **OK**.

The specified schedule is created.

## Promoting the steps to production mode

To activate the schedule and the task flow links that you created, you must promote the steps to production mode. Production mode indicates that the steps are in their final format. In production mode, you can change only those settings that will not affect the step's processing, such as changing the description of the step. For more information, see the *Data Warehouse Center Administration Guide*.

To promote the steps to production mode:

1. Right-click the Load Demographics data step.
2. Click **Mode —> Production**, then click **Yes** to save.

   The Data Warehouse Center displays a progress window. Wait until the Data Warehouse Center finishes processing before continuing with this lesson. When the progress window closes, the step icon changes to show two locks indicating that the step is in production mode.

3. Repeat the previous steps for the Select Geographies Data step and Join Market Data steps, in order.

4. Close the Process Model Window.

## What you just did

In this lesson, you scheduled the steps that you created to run once a year at the current date and time in the following order:

1. Load Demographics Data
2. Select Geographies Data
3. Join Market Data

Then you promoted the steps to production mode to implement the schedule.

# Chapter 11. Defining keys on target tables

In this lesson, you will define primary and foreign keys on target tables to use later in a join. To do this lesson, you must have defined the dimension tables and fact table in "Defining the rest of the tables for the star schema (optional)" on page 55.

In each target table, you will select a column that can be used to uniquely identify rows in that table. This will be the table's primary key. The column that you select as a primary key must have the following qualities:

- It must always have a value. The column for a primary key cannot contain null values.
- It must have unique values. Each value in the column must be different for each row in the table.
- Its values must be stable. A value should never change to another value.

For example, the CITY_ID column in the LOOKUP_MARKET table (created in "Chapter 8. Defining data transformation and movement" on page 35) is a good candidate for designation as a primary key. Because each city needs an identifier, no two cities can have the same identifier, and identifiers are unlikely to change.

Defining a primary key for a table is highly recommended because uniquely identifying each row speeds up row access.

You use foreign keys to define relationships between tables. In a star schema, a foreign key defines the relationship between the fact table and its associated dimension tables. The primary key of the dimension table has a corresponding foreign key in the fact table. The foreign key requires that all the values of a given column in the fact table also exist in the dimension table. For example, the CITY_ID column of the FACT_TABLE might have a foreign key defined on the CITY_ID column of the LOOKUP_MARKET dimension table. This means that a row cannot exist in the FACT_TABLE unless the CITY_ID exists in the LOOKUP_MARKET table.

In this lesson, you will define primary keys on the four target tables that you created in "Chapter 8. Defining data transformation and movement" on page 35: LOOKUP_MARKET, LOOKUP_TIME, LOOKUP_PRODUCT, and LOOKUP_SCENARIO. You will define corresponding foreign keys in the FACT_TABLE target table.

**71**

## Defining a primary key

In this exercise, you will define a primary key for the LOOKUP_MARKET, LOOKUP_TIME, LOOKUP_PRODUCT, and LOOKUP_SCENARIO target tables.

To define the primary keys:

1. Go to the Control Center window.
2. Click **View** –> **Refresh**.
3. Find the LOOKUP_MARKET table in the list of tables for the TUTWHS database. Right-click the table and click **Alter**.

   The Alter Table notebook opens.
4. Click the **Primary Key** tab. The Primary Key page opens.
5. In the **Available columns** list, select **CITY_ID** as the primary key.
6. Click **>** to move CITY_ID to the **Primary key columns** list.
7. Leave the **Constraint name** field empty, so that DB2 Universal Database will generate a constraint name for you. A primary key can be considered a constraint, because all values in the selected column must be unique.
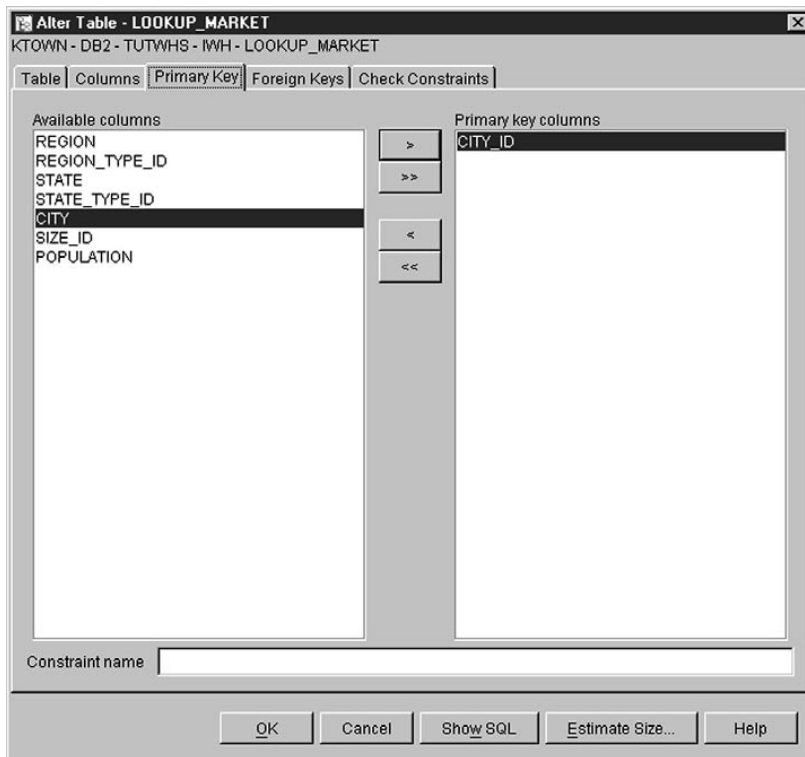


8. Click **OK** to save your definition.

Follow the same steps to define primary keys for the other target tables. Define:

- TIME_ID as the primary key for the LOOKUP_TIME table.
- PRODUCT_KEY as the primary key for the LOOKUP_PRODUCT table.
- SCENARIO_ID as the primary key for the LOOKUP_SCENARIO table.

## Defining a foreign key

You need to define foreign keys for the relationships between the FACT_TABLE and the other target tables.

In this exercise, you will define a foreign key in the FACT_TABLE (dependent table) based on the primary key of the LOOKUP_MARKET table (parent table).

To define the foreign keys:

1. Find the FACT_TABLE table in the list of tables for the TUTWHS database. Right-click the table and click **Alter**.

   The Alter Table notebook opens.
2. Click the **Foreign Keys** tab. The Foreign Key page opens.
3. Click **Add**. The Add Foreign Key window opens.
4. In the **Table schema** field, select **IWH**.
5. In the **Table name** field, specify LOOKUP_MARKET, which is the parent table. The **Primary key** field displays the primary key for LOOKUP_MARKET. The steps that generate the parent table must be in test or production mode in order for the primary key to be available. A primary key must be defined for the parent table before a foreign key can be defined for the FACT_TABLE.
6. In the **Available columns** list, select CITY_ID as the column that you want defined as the foreign key.

7. Click **>** to move CITY_ID to the **Foreign key** list.



8. Click **OK** to save your definitions.

Define foreign keys for the FACT_TABLE to the other target tables by repeating steps 3 on page 73 to 8. Define:

- TIME_ID as a foreign key with the LOOKUP_TIME table as the parent.
- PRODUCT_KEY as a foreign key with the LOOKUP_PRODUCT table as the parent.
- SCENARIO_ID as a foreign key with the LOOKUP_SCENARIO table as the parent.
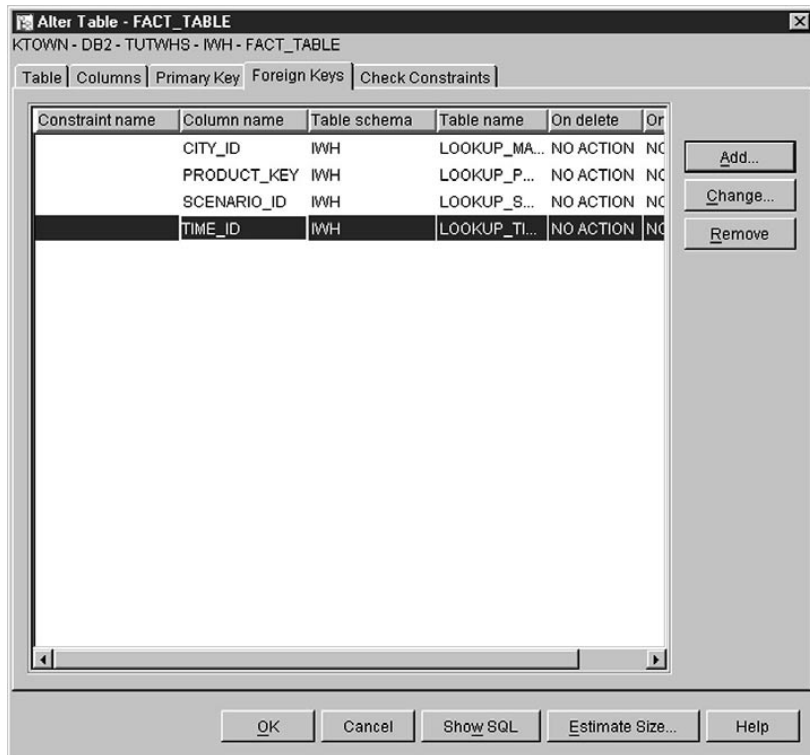
9. Click **OK** to save the information in the Alter Table window.

## Defining foreign keys in the Data Warehouse Center

In this section, you will define foreign keys in the Data Warehouse Center.

To define foreign keys into your data warehouse:

1. In the Data Warehouse Center window, expand the **Warehouse Target** tree.
2. Expand the **Tutorial Targets** tree until you see **Tables** folder.
3. Right-click on the **FACT_TABLE** table.
4. Click **Properties**.

   The table Properties notebook opens.
5. Click the **Warehouse Foreign Keys** tab.
6. Right-click on the **Constraint name** of the first foreign key in the view and click **Remove**. Do this for each foreign key in the view. These foreign key definitions were carried over when the Fact Table Join step was

copied. They refer to tables in the TBC Sample Targets and need to be deleted and replaced with new foreign keys that refer to tables in the Tutorial Targets.

7. Right-click on the white space and click **Define**. The Define Warehouse Foreign Key window opens.
8. Select **IWH** in the **Object schema** list.
9. Select **LOOKUP_MARKET** in the **Object name** list. The primary key for LOOKUP_MARKET appears in the **Warehouse primary key** field.
10. Select **CITY_ID** in the **Available columns** field.
11. Click > to move CITY_ID into the **Warehouse foreign key columns** field.
12. In the **Constraint name** field, type:

    ```
    "Whse Market FK"
    ```

    The constraint names must be different for each foreign key. The constraints for the remaining three LOOKUP tables are shown in the table below.

| Table | Constraint name |
| --- | --- |
| LOOKUP_TIME | "Whse Time FK" |
| LOOKUP_PRODUCT | "Whse Product FK" |
| LOOKUP_SCENARIO | "Whse Scenario FK" |

13. Click **OK** to save your definition and close the Define Warehouse Foreign Key window.
14. Repeat steps 7 to 13 to add foreign keys for the remaining three LOOKUP tables.
15. Click **OK** to save your changes and close the table Properties window.

## What you just did

In this lesson, you defined primary keys for the following target tables:

- LOOKUP_MARKET
- LOOKUP_TIME
- LOOKUP_PRODUCT
- LOOKUP_SCENARIO

You defined foreign keys for the FACT_TABLE that correspond to the primary keys. In "Chapter 16. Creating a star schema from within the Data Warehouse Center" on page 103, you will use these keys to join the tables in the star schema.

# Chapter 12. Maintaining the data warehouse

In this lesson, you will learn to define indexes, use the RUNSTATS and REORG utilities, and monitor the warehouse database.

The amount of maintenance that is required for a database is directly related to the amount of database activity or workload. Because this tutorial does not generate significant activity, this lesson is primarily a guide to the DB2 tools and utilities that you can use when you are using an actual warehouse database.

## Creating an index

You can create an index to optimize queries for end users of the warehouse. An *index* is a set of keys, each pointing to a set of rows in a table. The index is a separate object from the table data. The database manager builds the index structure and maintains it automatically. An index gives more efficient access to rows in a table by creating a direct path to the data through the pointers that it creates.

An index is created when you define a primary key or a foreign key. For example, an index was created on the LOOKUP_MARKET table when you defined CITY_ID as its primary key in "Chapter 11. Defining keys on target tables" on page 71.

To create additional indexes:
1. From the DB2 Control Center, expand the objects within the TUTWHS database until you see the Indexes folder.
2. Right-click the **Indexes** folder and click **Create —> Index Using Wizard**. The Connect window opens.
3. Type the userid and password that you use to log onto the Control Center.
4. Click **OK**. The Index wizard opens.
5. Follow the detailed instructions in the wizard to build a new index.

For more information about creating an index, see the *DB2 Universal Database Administration Guide*.

## Collecting table statistics

*Table statistics* provide information about the physical and logical characteristics of a table and its indexes. You must periodically collect these statistics, so that DB2 Universal Database can determine the best way to access your data. If extensive changes are made to the data in a table, and the last collection of statistics no longer reflects the actual table data, then data access performance can deteriorate. In general, you should update statistics if there are major changes to the data in your table.

To collect statistics on the LOOKUP_MARKET table:

1. From the DB2 Control Center, right-click on the LOOKUP_MARKET table and click **Run Statistics**.

   The Run Statistics window opens.

2. Use the online help to determine the level of statistics that you want to gather for the table and its indexes.

3. Use **Share level** to specify the type of access that users have while statistics are being gathered.

   **Change (Table read/write allowed)**
   Other users can read and write to the table.

   **Reference (Table read only)**
   Other users have read-only access to the table.

**Run Statistics - LOOKUP_MARKET**                    ×

9 - TCP0000 - TUTWHS - IWH - LOOKUP_MARKET

Statistics for the table

○ <u>D</u>o not update

◉ Update <u>w</u>ithout distribution statistics

○ <u>U</u>pdate with distribution statistics

Statistics for the indexes

◉ Do <u>n</u>ot update

○ Upda<u>t</u>e without extended index statistics

○ Update with e<u>x</u>tended index statistics

Share level

◉ <u>C</u>hange (Table read/write allowed)

○ <u>R</u>eference (Table read only)

| <u>O</u>K | Cancel | Sho<u>w</u> Command | Help |

4. Click **OK** to begin collecting the table statistics.

## Reorganizing a table

Reorganizing a table rearranges it in physical storage, eliminating fragmentation and making sure that the table is stored efficiently in the database. You can also use reorganization to control the order in which the rows of a table are stored, usually according to an index.

To reorganize the LOOKUP_MARKET table:

1. From the DB2 Control Center, right-click on the LOOKUP_MARKET table and click **Reorganize**.

   The Reorganize Table window opens.
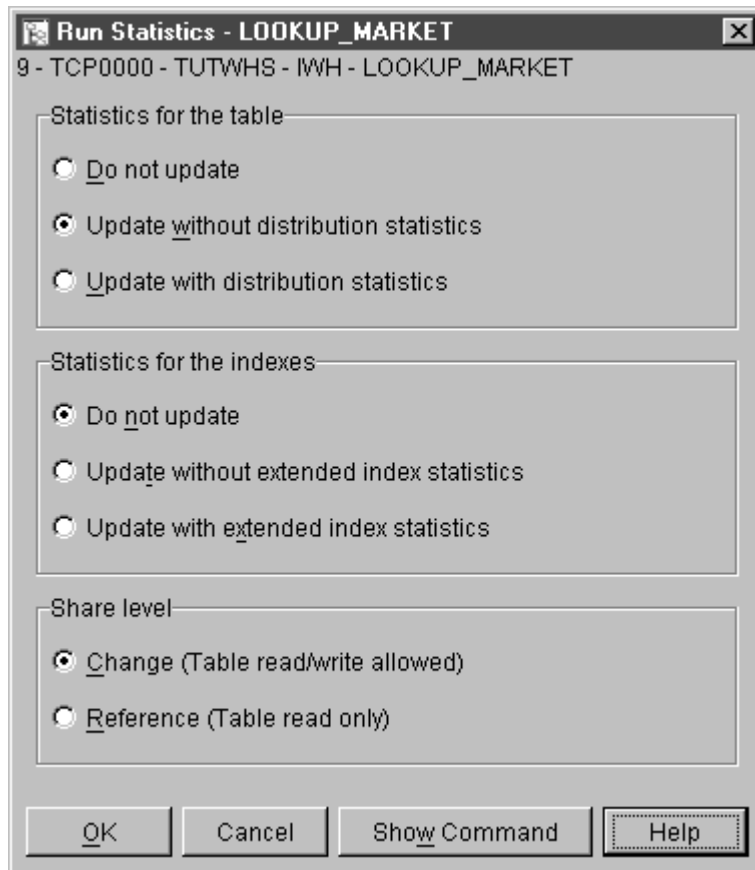
2. In the **Using system temporary table space** field, specify the name of the table space where the table being reorganized can be temporarily stored. If

you do not specify a table space, the temporary copy of the table is stored in the same table space where the table currently resides.

3. In the **Using index** field, specify the index to use to reorganize the table rows. If you do not specify an index, the table rows are reorganized without regard to order.

4. Click **Reorganize Now** to reorganize the table immediately. Click **Schedule** to schedule the reorganization for a specific date and time.
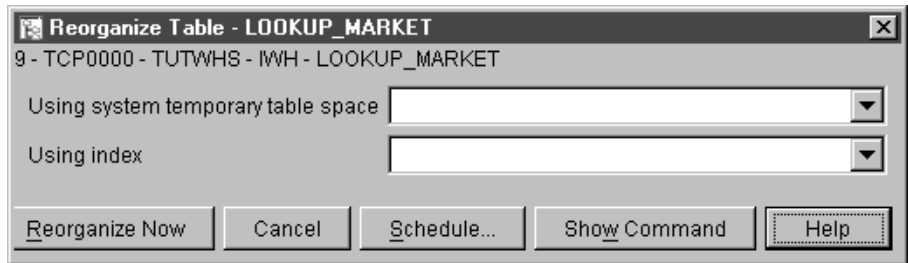


## Monitoring a database

The performance monitor provides information about the state of DB2 Universal Database and the data that it controls, and calls attention to unusual situations. The information is provided in a series of snapshots, each of which represents the state of the system and its databases at a point in time. You can control the frequency of the snapshots and the amount of information collected by each.

The information that is collected by a performance monitor is returned in performance variables. These performance variables are a defined set of elements that can:

- Count the number of times something has occurred (counters), for example, the number of rows read.
- Return the current value (gauge), for example, the number of applications currently connected.
- Represent the time when something occurred (time stamp), for example, the time when the last application completed.
- Record the highest value reached (water mark), for example, the maximum size of the database heap.

To monitor the LOOKUP_MARKET table:

1. From the DB2 Control Center, right-click the LOOKUP_MARKET table, and click **Performance Monitoring —> Start Monitor...**.

   The Start Monitor window opens.

2. The Start Monitor window lists the performance monitors that can take snapshots at the table level. Select the **Default_for_table_level** monitor, and click **OK** to start it.

   The Default_for_table_level is an IBM-supplied monitor that is designed to collect key performance related information at the table level.

3. Right-click on the LOOKUP_MARKET table and select **Performance Monitoring —> Show Monitor Activity...**.

   The Show Monitor notebook opens.

   The Details page lists the performance variables that are being collected by the monitor and the values that are returned by the snapshots. The Default_for_table_level monitor takes snapshots (Regular sample box) every 30 seconds, and does not graph its collected data. You can tailor this monitor to your own preferences (for example, graph performance variables or change snapshot interval). See the online help for details.

4. If there is no activity on the LOOKUP_MARKET table, then the values on the Details page will remain blank. To generate table activity:

   a. Open the Command Center.

   b. On the Script page, do the following:

      1) Issue `connect to TUTWHS user` *userid* `using` *password*, where:
         - *userid* is the user ID that you used to create the database in "Chapter 2. Creating a warehouse database" on page 5.
         - *password* is the password for the user ID.

      2) Issue a number of `SELECT * FROM` *schema*`.LOOKUP_MARKET` statements, where *schema* is the schema that was specified when the database was created."Chapter 2. Creating a warehouse database" on page 5.

      3) Click **Script—>Execute** to run the script.

   c. Return to the Show Monitor notebook. As snapshots are taken, the values for the Rows Read per Second performance variable will reflect your SQL statement activity.

5. Close the Show Monitor notebook. Right-click on the LOOKUP_MARKET table and click **Performance monitoring —> Stop monitor**. The Default_for_table_level monitor stops.

## What you just did

In this lesson, you created an index. You collected statistics on the LOOKUP_MARKET table, reorganized it, and monitored it. In the next lesson, you will authorize the users of the warehouse database to access the table.

# Chapter 13. Authorizing users to the warehouse database

In this lesson, you will define privileges that authorize users to the warehouse database. Security for the warehouse database is managed within DB2 Universal Database. It is separate from the Data Warehouse Center security.

Access within DB2 Universal Database is managed by administrative authorities and user privileges within the database manager. Authorities are usually granted at the database level, and privileges are usually granted for objects within the database (for example, tables).

Privileges are controlled by users with SYSADM or DBADM authority, or by the creator of the object. You can grant privileges to users for the TUTWHS database because you are its creator.

## Granting privileges

To grant privileges to the TUTWHS database:
1. From the DB2 Control Center, expand the objects in the TUTWHS database until you see the **Tables** folder.
2. Click the **Tables** folder. In the right panel, you will see all the tables in the database.
3. Right-click on the LOOKUP_MARKET table, and click **Privileges**.
   The Table Privileges window opens.
4. Click **Add User**.
   The Add User window opens.
5. Select a user or enter a name. Click **OK**. The user is added to the User page.
6. Select one or more users. To grant all privileges to the selected users, click **Grant All**. To grant individual privileges, use the **Privileges** list boxes.
7. Click **Apply** to process your request.
8. Close the Table Privileges window.

## What you just did

In this lesson, you authorized users to use the LOOKUP_MARKET table. In the next lesson, you will catalog the metadata that you have defined in the Data Warehouse Center, so that the users that you authorized in this lesson can find the data that they need more easily.

# Chapter 14. Cataloging data in the warehouse for end users

For this lesson, you must have the Warehouse Manager package to install the Information Catalog Manager, which runs on Windows NT, Windows 95, and Windows 98. You also must have defined the dimension tables and fact table in "Defining the rest of the tables for the star schema (optional)" on page 55.

In this lesson, you will catalog the data in your data warehouse for use by end users. You catalog the data by publishing Data Warehouse Center metadata in an information catalog. An *information catalog* is the set of tables managed by the Information Catalog Manager that contains business metadata that helps users identify and locate data and information available to them in the organization. Users can search the information catalog to find the tables that contain the data that they need to query.

*Publishing metadata* is the process of transferring metadata from the Data Warehouse Center to the Information Catalog Manager. In this lesson, you will publish the metadata for the Market process that you created in "Chapter 8. Defining data transformation and movement" on page 35. When you publish the process, you will publish the metadata for the following objects that are contained in the process:

1. The Load Demographics Data step, its source file DEMOGRAPHICS, and its target table DEMOGRAPHICS_TARGET.
2. The Select Geographies Data step, its source table GEOGRAPHIES, and its target table GEOGRAPHIES_TARGET.
3. The Join Market Data step and its target table LOOKUP_MARKET. (Its source tables will be published with the other two steps.)

## Creating the information catalog

First, you will create the information catalog in which you will publish metadata.

To create the information catalog:

1. Click **Start —> Programs —> IBM DB2 —> Information Catalog Manager —> Initialize Information Catalog**.
   The Initialize Information Catalog window opens.
2. Select the version of DB2 UDB for your operating system
3. Click **OK**.

The Define Catalog on DB2 UDB window opens.

4. From the **Information catalog name** field, type TBC_MD.

   Accept the default value, -, in the **Not-applicable symbol** field.

5. In the **Primary administrator's user ID** field, type your user ID.

   Accept the default for the rest of the selections. For more information about these fields, see "Define Catalog on DB2 UDB" in the online help.

6. Click **Define**.

   The Connect to information catalog window opens.

7. In the **User ID** field, type the user ID that you specified when you installed the sample.

8. In the **Password** field, type the password for the user ID.

9. Click **Connect**.

   The Information Catalog Manager creates the information catalog. The Information Catalog window opens and displays a message stating that the import completed successfully.

10. Close the Information Catalog window.

## Selecting metadata to publish

To select the metadata that you want to publish:

1. From the Data Warehouse Center window, right-click the **Warehouse** folder.

2. Click **Publish metadata**.

   The Publish Metadata window opens.

3. Right-click the white space in the window.

4. Click **Publish Data Warehouse Center metadata**.

The Publish Data Warehouse Center Metadata notebook opens.
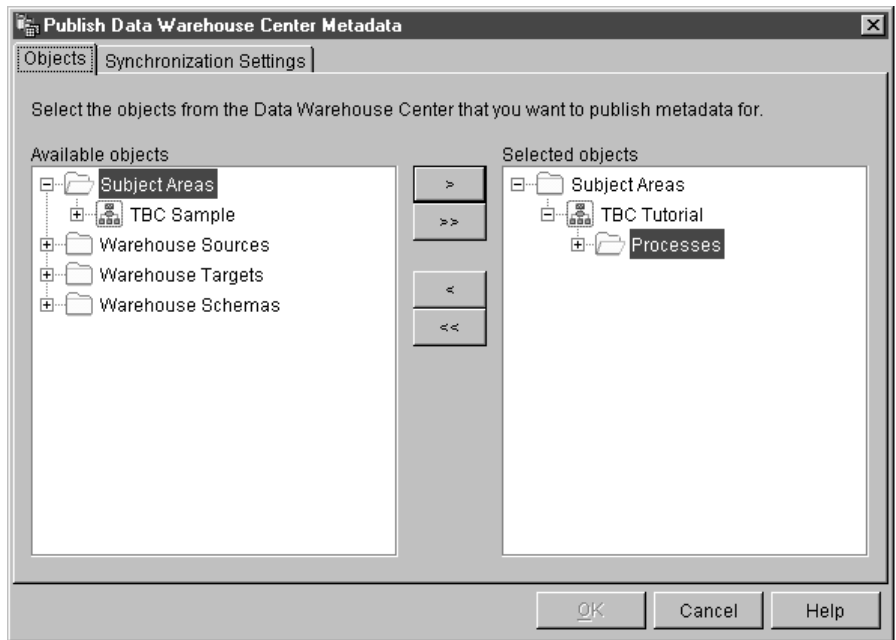


5. In the **Available objects** list, click **TBC Tutorial** located in the **Subject Areas** folder.
6. Click >.

   The TBC Tutorial subject area moves to the **Selected objects** list.

7. Click the **Synchronization Settings** tab.



8. In the **Catalog name** field, type TBC_MD.

   The TBC_MD database contains sample metadata, which includes a sample information catalog. You will add metadata about the Market process to the existing metadata.

9. In the **Administrator user ID** field, type the user ID under which you installed the sample.

10. In the **Administrator password** field, type the password that corresponds to the user ID.

11. In the **Verify password** field, type the password again.

12. From the **Map Source Tables to Target Warehouse** list, select the **At the table level** radio button to map source tables and target tables at the table level. The information catalog will indicate that there is a mapping between the two tables with a Transformation object.

   The other option you can select is **At the column level** to map source tables and target tables at the column level. The information catalog will indicate that there is a mapping between each column in the source table and its corresponding column in the target table. However, if there are many columns, the mappings in the information catalog can be quite complex.

   Accept the rest of the default values on this page. For more information about the fields on this page, see "Publishing Data Warehouse Center metadata" in the online help.

13. Click **OK**.

A status indicator is displayed until the publication of the objects is complete. When the publication is complete, the Publish Information window opens and displays the contents of a log file that lists which metadata objects were published successfully.

14. Click **Close** on the Publish Information window.

The window closes, and the log file is deleted. The Publish Metadata window opens listing your information catalog name in the **List of publications**.

## Updating published metadata

After you publish metadata, you need to update it periodically to capture the changes that have been made in the Data Warehouse Center. To transfer updates of the Data Warehouse Center metadata to the information catalog, you run the step for the publication in the same way that you run any other steps in the Data Warehouse Center.

To update published metadata:

1. In the **List of publications** in the Publish Metadata window, right-click **TBC_MD.DWC.def**.
2. Click **Run**.
3. Go to the main Data Warehouse Center window and click **Warehouse —> Work in Progress**.

The Work in Progress window opens. You should see an entry for the step that is running. While the step is running, the status is Populating. When the step has completed the status is Successful.

For more information about the Work in Progress window, see "Work in Progress—Overview" in the online help.

4. Close the Work In Progress window.
5. Click **Close** in the Publish Metadata window.

## What you just did

In this lesson, you published the Data Warehouse Center metadata that you created in the tutorial to the information catalog. You ran the publication to update the metadata that you published. In the next lesson, you will view the metadata in the information catalog.

# Chapter 15. Working with business metadata

For this lesson, you must have the Warehouse Manager package to install the Information Catalog Manager, which runs on Windows NT, Windows 95, and Windows 98. You also must have defined the dimension tables and fact table in "Defining the rest of the tables for the star schema (optional)" on page 55.

In this lesson, you will view your published metadata in the information catalog and customize the catalog. In the information catalog, the metadata is in the form of *objects*, which are items that represent units or distinct groupings of information, but do not contain the actual information. You will create a collection of objects in the catalog. A *collection* is a container for objects that you define for easy access. You will launch a program from an object that represent a file to view the actual file data.

## Opening the information catalog

To open the information catalog:

1. Click **Start**.
2. Click **Programs —> IBM DB2 —> Information Catalog Manager —> TBC_MD User**.

   The Product Information window opens. **TBC_MD User** represents the information catalog to which you added metadata in "Chapter 14. Cataloging data in the warehouse for end users" on page 85.
3. Click **OK** to close the Product Information window.
4. In the **User ID** field, type the user ID that you used to create the information catalog.
5. In the **Password** field, type the password for the user ID you entered in the **User ID** field.

   The password is case-sensitive.
6. Click **Open**.

   The Information Catalog window opens.

## Browsing subjects

To browse subjects in an information catalog:

1. Double-click the **Subjects** icon in the Information Catalog window.

   The Subjects window opens, displaying a list of objects in your information catalog. These objects contain other objects, but are not contained by any other object. The Subjects window opens in an icon view

by default, but it can also open as a list view.
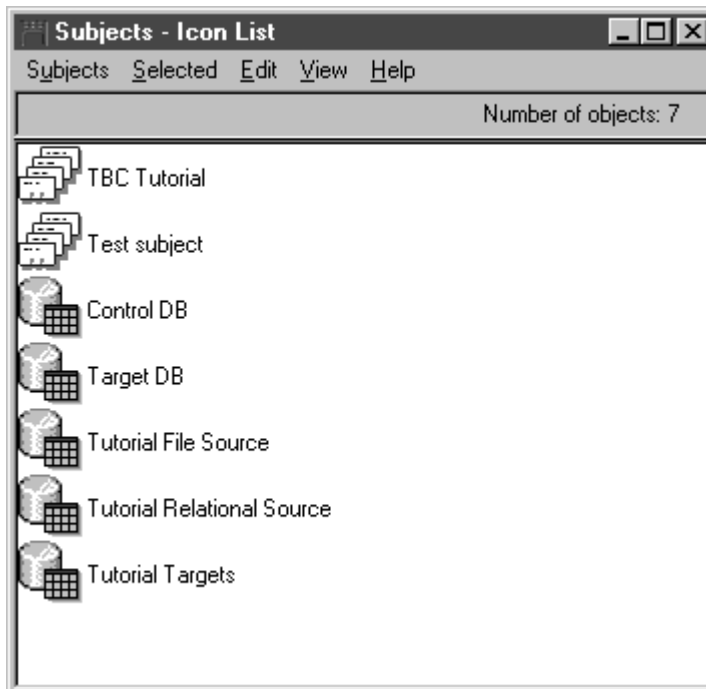


2. To view information that is contained in an object, right-click on the object, and click **Open as —> Tree**.
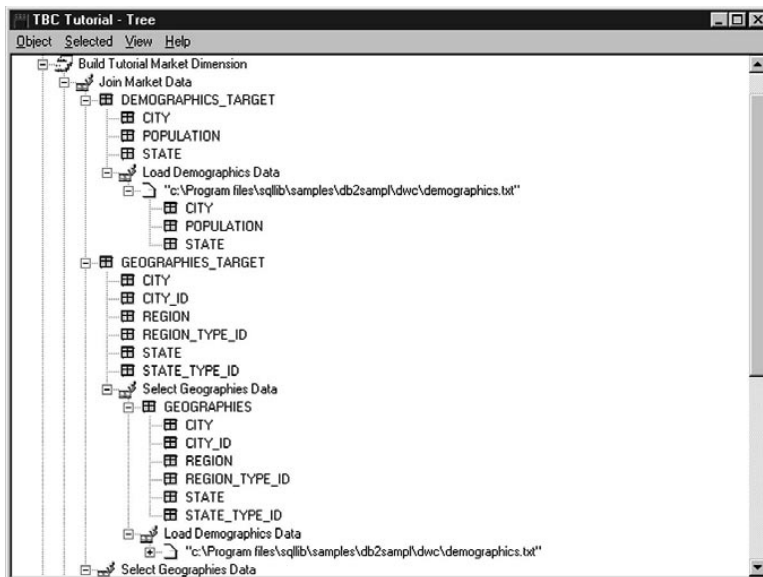
The tree view shows you the relationship of the objects that belong to a particular grouping. The objects in the tree view have plus signs (+) next to them to show that all objects in this view are grouping objects that contain other objects.

3. Close the Tree window.
4. Close the Subjects-Icon List window.

## Searching the information catalog

In this exercise, you will search for the objects that correspond to the dimension tables that you specified in "Chapter 8. Defining data transformation and movement" on page 35.

To search for objects in an information catalog:

1. Double-click the **New search** icon in the Information Catalog window. The Define Search window opens.
2. Scroll through the **Available object types** list. This list displays all of the object types that are defined in your information catalog.
3. Select the **Relational tables and views** object type.
4. Click > to add the selected object type to the **Search for object types** list.
5. From the **Properties/Values** list, click **Name**.
6. In the **Enter value for selected property** field, type the following value:

   LOOKUP
7. Click **Enter**. The value is displayed in the **Value** column of the **Properties/Values** list.
8. Select the **Ignore case** check box to search all occurrences of the value, regardless of case.
9. From the **Search options** list, click **beginning with**.

The Information Catalog Manager will search for objects that begin with the value that you specified in the **Value** column.
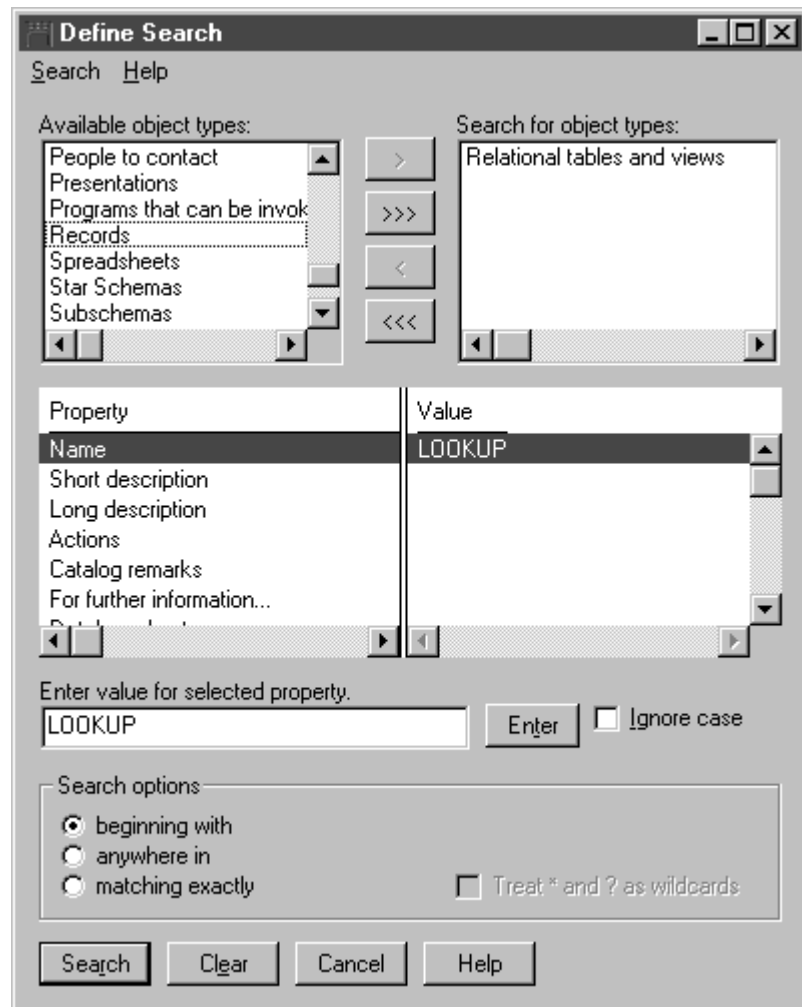
```
┌──────────────────────────────────────────────────────────────┐
│ ▤ Define Search                                      _ □ ✕    │
│ Search  Help                                                  │
│                                                               │
│ Available object types:            Search for object types:   │
│ ┌──────────────────────────┐ ┌───┐ ┌─────────────────────────┐│
│ │People to contact      ▲  │ │ > │ │Relational tables and views││
│ │Presentations             │ └───┘ │                         ││
│ │Programs that can be invok │ ┌───┐ │                         ││
│ │Records                   │ │>>>│ │                         ││
│ │Spreadsheets              │ └───┘ │                         ││
│ │Star Schemas              │ ┌───┐ │                         ││
│ │Subschemas             ▼  │ │ < │ │                         ││
│ │◄│                  ►      │ └───┘ │                         ││
│ │                          │ ┌───┐ │◄│                  ►    ││
│ └──────────────────────────┘ │<<<│ └─────────────────────────┘│
│                              └───┘                            │
│                                                               │
│ Property                         Value                        │
│ ┌──────────────────────────────┬──────────────────────────┐  │
│ │Name                          │LOOKUP                 ▲   │  │
│ │Short description             │                           │  │
│ │Long description              │                           │  │
│ │Actions                       │                           │  │
│ │Catalog remarks               │                           │  │
│ │For further information...    │                       ▼   │  │
│ │◄│              ►    │◄│       │◄│                  ►      │  │
│ └──────────────────────────────┴──────────────────────────┘  │
│                                                               │
│ Enter value for selected property.                            │
│ ┌───────────────────────────────┐ ┌───────┐  ☐ Ignore case   │
│ │LOOKUP                         │ │ Enter │                   │
│ └───────────────────────────────┘ └───────┘                   │
│ ┌─ Search options ──────────────────────────────────────────┐│
│ │  ⦿ beginning with                                          ││
│ │  ○ anywhere in                                             ││
│ │  ○ matching exactly         ☐ Treat * and ? as wildcards   ││
│ └────────────────────────────────────────────────────────────┘│
│                                                               │
│ ┌────────┐ ┌────────┐ ┌────────┐ ┌────────┐                  │
│ │ Search │ │ Clear  │ │ Cancel │ │  Help  │                  │
│ └────────┘ └────────┘ └────────┘ └────────┘                  │
└──────────────────────────────────────────────────────────────┘
```

10. Click **Search**. The Information Catalog Manager searches for objects of the type that you specified and displays the results in the Search Results
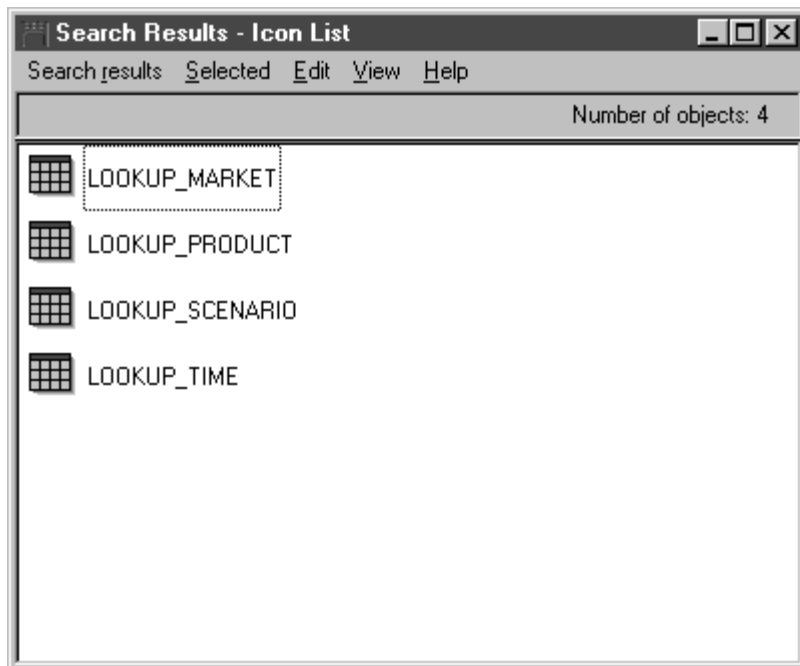
window.



## Creating a collection of objects

You can create a collection that groups different objects together so that you can access them easily. A collection is represented by an icon that is displayed in the Information Catalog window. In this exercise, you will create a collection for the objects that correspond to the dimension tables that you specified in "Chapter 8. Defining data transformation and movement" on page 35.
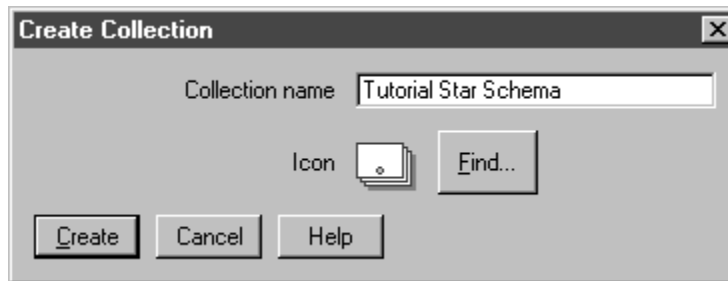
To create a collection:

1. Click **Catalog —> Create collection** from the Information Catalog window. The Create Collection window opens.
2. In the **Collection name** field, type a name for your new collection:

   Tutorial Star Schema

Accept the default icon to represent your collection.



3. Click **Create**. The new collection icon is displayed. You can now add and delete objects to and from your collection.
4. From the Search Results window, right-click on the **LOOKUP_MARKET** object.
5. Click **Copy to collection**.

   The Copy to Collection window opens.
6. From the **Select a collection list**, select the Tutorial Star Schema collection.
7. Click **Copy**. The object is copied to the collection of objects that you selected.
8. Click **OK**.
9. Repeat steps 4 through 7 for the LOOKUP_PRODUCT, LOOKUP_SCENARIO, and LOOKUP_TIME objects.

   After you complete these steps, if you double-click the Tutorial Star Schema collection in the Information Catalog window, you see the same list of tables that were displayed in the Search Results window.
10. Close the Information Catalog.

    In the next section, we will re-open the Information Catalog as an administrator.

## Starting a program

The Information Catalog Manager makes it easy to start a program that can retrieve the actual data that an object describes. For example, if you have objects describing graphic charts, you can set up a graphic program, such as CorelDRAW!, so that you can retrieve the actual charts for editing, copying, or printing.

The Information Catalog Manager can start any program that runs on the Windows platform you are using, or that can be started from an MS-DOS command prompt. The program must be installed on the client workstation.

A single object type can start more than one program (for example, the object type Spreadsheet can have both Lotus 1-2-3® and Microsoft Excel associated with it).

To enable an object to start a program, you create an association between a Programs object and any object type not categorized as Program. In this exercise, you will define a Programs object for Microsoft Notepad. You will create an association between the Notepad Programs object and the Files object type.

## Creating a Programs object

To create a Programs object, you will define the properties of the object and specify the properties to be used as program parameters.

In this exercise, you will create a Programs object.

1. Click **Start —> Programs —> IBM DB2 —> Information Catalog Manager —> TBC_MD Administrator**.
2. In the Open Information Catalog window, type the user ID that you used to create the information catalog in the **User ID** field.
3. In the **Password** field, type the password for the user ID. The password is case-sensitive.
4. Click **Open**.
5. From the Information Catalog window, right-click the **Object types** icon.
6. Click **Open as —> Icon list**.
7. Right-click the icon of the Files object type.
8. Click **Associate programs**.

   The Programs window displays a list of programs currently associated with the selected object type.
9. Click **Add**.

   The Files-Add Program window opens. The **Name** property is selected.
10. In the **Enter value for selected property** field, type a description of the program:

    `View files using Microsoft Notepad`
11. Click on the **Enter** push button to move the value to the **Value** column in the **Properties/Values** list box.

    If you want to erase what you entered in the **Enter value for selected property** field, click on the **Clear** push button.
12. From the **Properties/Values** list box, click the **Class** property.
13. In the **Enter value for selected property** field, type a classification for the program:

    `Word processor`

14. Click on the **Enter** push button to move the value to the **Value** column in the **Properties/Values** list box.
15. From the **Properties/Values** list box, click the **Qualifier 1** property.
16. In the **Enter value for selected property** field, type a qualifier for the program:
    Notepad
17. Click on the **Enter** push button to move the value to the **Value** column in the **Properties/Values** list box.
18. From the **Properties/Values** list box, click the **Qualifier 2** property.
19. In the **Enter value for selected property** field, type a qualifier for the program:
    Windows
20. Click on the **Enter** push button to move the value to the **Value** column in the **Properties/Values** list box.
21. From the **Properties/Values** list box, click the **Qualifier 3** property.
22. In the **Enter value for selected property** field, type a qualifier for the program:
    -

    A hyphen is the default not-applicable symbol.
23. Click on the **Enter** push button to move the value to the **Value** column in the **Properties/Values** list box.
24. From the **Properties/Values** list box, click the **Identifier** property.
25. In the **Enter value for selected property** field, type an identifier for the program:
    notepad.exe
26. Click on the **Enter** push button to move the value to the **Value** column in the **Properties/Values** list box.

    The Files object is already defined for the **Object type that this program handles** property.
27. From the **Properties/Values** list box, click the **Start by invoking** property.
28. In the **Enter value for selected property** field, type the command to start the program:
    START *X:\path*\notepad.exe

    where *X:\path* is the path in which Microsoft Notepad is installed, such as c:\Winnt\System32.

**Tip:** The combination of the Class, Qualifiers 1, 2, and 3, and the Identifier properties must be unique across all objects in the information catalog. Each instance of an object type must be different.
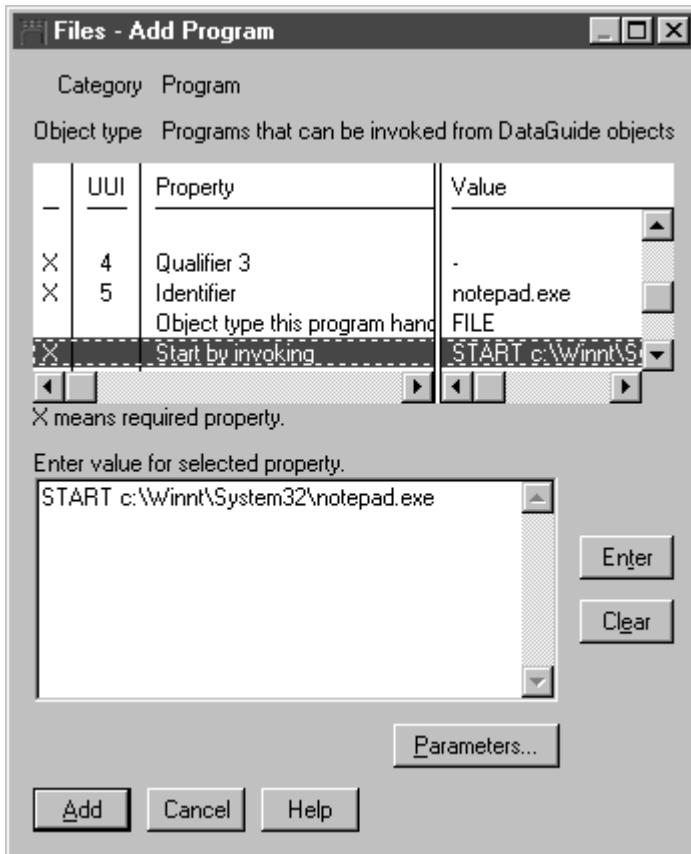
In this exercise, you will specify the properties whose values you want to be used as program parameters.

1. From the Files-Add Program window, click on the **Parameters** push button. The Files-Invocation Parameters window opens, listing the properties of the File object type.
2. From the **Property** list box, click **Name**.
3. Click **Append**.

%NAME% appears in the **Enter program invocation parameters** list box.
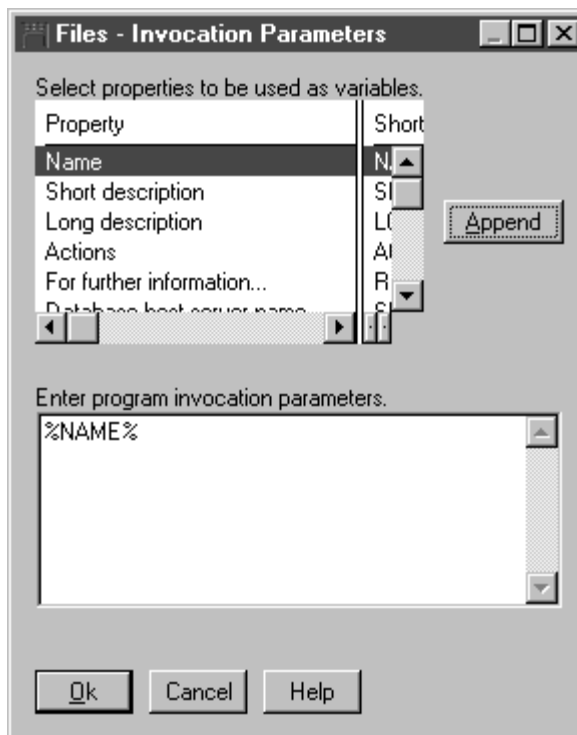


4. Click **OK**.
5. From the Files-Add Program window, click the **Add** push button.
6. Close the Files-Programs window.
7. A message box opens in the Information Catalog Manager. Click **OK**.
8. Close the Object Types window.

## Starting the program from a Files object

In this exercise, you will start Microsoft Notepad from the Files object for the demographics file. You will search for the object and then start the program.

To start the program from the Files object:

1. Double-click the **New search** icon in the Information Catalog window.
   The Define Search window opens.
2. Scroll through the **Available object types** list. This list displays all of the object types that are defined in your information catalog.
3. Select the Files object type.
4. Click **>** to add the selected object type to the **Search for object types** list.
5. From the **Properties/Values** list, click **Name**.

6. In the **Enter value for selected property** field, type the following value:
   ```
   demographics.txt
   ```
7. Click **Enter**. The value is displayed in the **Value** column of the **Properties/Values** list.
8. Select the **Ignore case** check box to search all occurrences of the value, regardless of case.
9. Verify that **anywhere in** is selected in the **Search options** list.

   The Information Catalog Manager will search for objects that contain the value that you specified in the **Value** column.
10. Click **Search**. The Information Catalog Manager searches for objects of the type that you specified and displays the results in the Search Results window.
11. From the Search Results window, right-click the `X:\sqllib\samples\db2sampl\dwc\demographics.txt` object, where *X* is the drive on which you installed the sample.
12. Select **Start program**.

    The Microsoft Notepad program is launched. You see the same data that you viewed with Microsoft Notepad in "Chapter 3. Browsing the source data" on page 9.
13. Close all of the Information Catalog windows that are still open.

## What you just did

In this lesson, you opened the information catalog and browsed the metadata that you transferred from the Data Warehouse Center. You searched for the objects corresponding to the LOOKUP* tables and grouped them together in a collection. In "Chapter 16. Creating a star schema from within the Data Warehouse Center" on page 103, you will create a star schema from the LOOKUP* tables. You defined a Programs object for Microsoft Notepad, and started Microsoft Notepad from the object for the demographics.txt file.

# Chapter 16. Creating a star schema from within the Data Warehouse Center

In this lesson, you will create a star schema from the warehouse tables that you created in this tutorial. You can use this star schema for queries in the warehouse database. You can also export the star schema to OLAP Integration Server to create an OLAP database. In this lesson, you will export the star schema to OLAP Integration Server.

To do this lesson, you must have installed the OLAP Starter Kit. You also must have defined the dimension tables and fact table in "Defining the rest of the tables for the star schema (optional)" on page 55.

## Defining a star schema

In this exercise, you will define the star schema that is to contain the dimension and fact tables that you have defined in this tutorial.

To define a star schema:

1. From the Data Warehouse Center, right-click the **Warehouse Schemas** folder.
2. Click **Define**.

   The Define Warehouse Schema notebook opens.
3. In the **Name** field, type the schema name:

   ```
   Tutorial Schema
   ```
4. In the **Administrator** field, type your name as the contact for the schema.
5. In the **Description** field, type the description of the schema:

   ```
   This is the TBC star schema
   ```
6. Select the **Use only one database** check box.
7. From the **Warehouse target database** list, select **TUTWHS**.

   Accept the rest of the values. For more information about the fields on this page, see the "Defining a warehouse schema" in the online help.
8. Click **OK** to define the warehouse schema.

   The star schema is added to the tree underneath the **Warehouse Schemas** folder.

## Opening the schema

In this exercise, you will open the star schema so that you can add the dimension tables and fact table to the schema in the next exercise.

To open the Tutorial Schema:
1. Expand the **Warehouse Schemas** tree.
2. Right-click on **Tutorial Schema**.
3. Click **Open**.

## Adding tables to the schema

In this exercise, you will add the dimension tables and fact tables that you defined in this tutorial—the LOOKUP_PRODUCT, LOOKUP_SCENARIO, LOOKUP_TIME, and FACT_TABLE tables—to the star schema.

To add the dimension tables and fact table to the star schema:
1. Click the **Add Data** icon:

   

2. Click the canvas at the spot where you want to place the tables.

   The Add Data window opens.
3. Expand the Warehouse Targets tree until you see a list of tables underneath the **Tables** folder.
4. Select the LOOKUP_MARKET table.
5. Click **>** to add the LOOKUP_MARKET table to the **Selected source and target tables** list.
6. Repeat step 4 and step 5 to add the LOOKUP_PRODUCT, LOOKUP_SCENARIO, LOOKUP_TIME, and FACT_TABLE tables.
7. Click **OK**. The tables that you selected are displayed on the window.
8. Arrange the tables in the window so that the FACT_TABLE is in the center and there is one LOOKUP table in each corner.

## Autojoining the tables

In this exercise, you will use the primary and foreign keys that you defined in "Chapter 11. Defining keys on target tables" on page 71 to automatically join the dimension tables and the fact tables.

To autojoin the tables:

1. Click the **Save** icon in the toolbar to save your work:

2. Select the LOOKUP_PRODUCT, LOOKUP_SCENARIO, LOOKUP_TIME, and FACT_TABLE tables by holding down the Ctrl key and clicking each table.

3. Click the **Autojoin** icon in the toolbar:

   The Data Warehouse Center draws green lines between the primary keys in the dimension tables and the foreign keys in the fact table. You can change these lines until you save. (To remove these lines, multiple-select them, right-click, and click **Remove**.

4. Click the **Save** icon on the toolbar to save your work:

   The green autojoin lines become black.

5. Close the Warehouse Schema Model window.

## Exporting the star schema

In this exercise, you will export the star schema for use in OLAP Integration Server.

**Before you begin:** Before you begin this section, you must have installed the OLAP Starter Kit, or the full DB2 OLAP Server® product.

To export the star schema:

1. From the Data Warehouse Center window, right-click the **Warehouse** icon.

2. Click **Export Metadata —> OLAP Integration Server**.

   The Export Warehouse Schema notebook opens.

3. From the **Available warehouse schemas** list, select **Tutorial Schema**.

4. Click >

   The Tutorial Schema moves to the **Selected warehouse schemas** list.

5. Click the **Integration Server** tab.

6. In the **Catalog name** field, type the name of the catalog for the Multidimensional analysis part of the tutorial:

   OLAPCATD

7. In the **Catalog table schema name** field, type the schema name associated with the OM_INFO table.

   To find the name of the schema associated with the OM_INFO table:

   a. From the DB2 Control Center, expand the tree until you see the **OLAPCATD** database under **Databases**.

   b. Click on **Tables** and find the OM_INFO table in the **Name** column on the right. The schema associated with the OM_INFO table is visible in the **Schema** column.

   c. Return to the Export Warehouse Schema notebook and enter the schema name.

8. In the **Time dimension table** field, select the LOOKUP_TIME table.

   You cannot change the dimension type after the schema is stored in the OLAP Integration Server catalog.

9. In the **User ID** field, type the user ID under which you created the TBC_MD sample database. This is the user ID for the sample OLAP Integration Server catalog.

10. In the **Password** field, type the password for the user ID. This is the password for the sample OLAP Integration Server catalog.

11. In the **Verify password** field, type the password again.



12. Select **Overwrite the existing model**.

13. Click **OK** to export the selected warehouse schemas to the OLAP Integration Server catalog.

   The notebook closes, and a progress indicator is displayed until the export is complete. When all the specified warehouse schemas are exported, the Export Information window opens, displaying success information or failure information about the export. Click **Close** to close this window.

## What you just did

In this lesson, you created a star schema that consists of the LOOKUP_MARKET, LOOKUP_PRODUCT, LOOKUP_SCENARIO, LOOKUP_TIME, and FACT_TABLE tables. You exported the star schema to OLAP Integration Server.

# Chapter 17. Summary

Congratulations! You now have completed the Data warehousing part of the tutorial. In this tutorial, you defined a warehouse that contained dimension tables and a fact table for a star schema, as follows:

- You defined a warehouse database to contain the star schema.
- You viewed the source data that is supplied with the sample.
- You defined Data Warehouse Center security by defining a warehouse user and a warehouse group.
- You defined one of the dimension tables in the star schema, LOOKUP_MARKET, by defining warehouse sources for a source table and a source file, and then defining steps that moved the source data into the warehouse and that joined the tables.
- You copied the rest of the dimension tables and the fact table from the sample.
- You promoted the steps, tested them, and scheduled them.
- You defined primary and foreign keys on the tables in the star schema.
- You performed maintenance on the warehouse database to improve performance.
- You authorized users to the warehouse database.
- You published metadata to the information catalog, and enhanced the information catalog by associating a program with one of the objects.
- You defined the star schema to the Data Warehouse Center, and exported the star schema to OLAP Integration Server.

If you have the OLAP Starter Kit installed, your next step is to perform the "Part 2. Multidimensional data analysis" on page 111 part of this tutorial.

# Part 2. Multidimensional data analysis

# Chapter 18. About multidimensional analysis

In this lesson, you will explore the concepts of creating applications you can use to analyze relational data using Online Analytical Processing (OLAP) techniques. You will use the DB2 OLAP Starter Kit, a scaled-down version of the full-function DB2 OLAP Server product. Both the DB2 OLAP Starter Kit and DB2 OLAP Server are based on OLAP technology from Hyperion Solutions.

Within the DB2 OLAP Starter Kit, your primary tool for creating OLAP applications is the DB2 OLAP Integration Server, which runs on top of the Essbase multidimensional server. With these applications, users can analyze DB2 data using Lotus 1-2-3 or Microsoft Excel.

## What is multidimensional analysis?

Tools that employ OLAP technology, such as the DB2 OLAP Starter Kit, empower users to ask intuitive and complex ad hoc questions about their business, such as, "What is my profitability for the third quarter across the southeast region for my focus products?" Such a question requires multiple perspectives on the data, such as time, regions, and products. Each of these perspectives are called *dimensions*. The DB2 OLAP Starter Kit lets you organize your data into multiple dimensions for analysis.

Relational data can be considered two-dimensional because each piece of data, which you can also call a fact, correlates to one row and one column, each of which can be considered a dimension. The dimensions in a multidimensional database are higher-level perspectives of the data that represents the core components of your business plan, such as Accounts, Time, Products, and Markets. In an OLAP application, these dimensions tend not to change over time.

Each dimension has individual components called *members*. For example, the quarters of the year can be members of the Time dimension, and individual products can be members of the Products dimension. You can have hierarchies of members in dimensions, such as months within the quarters of the Time dimension. Members tend to change over time, for example, as your business grows and new products and customers are added.

## Lesson overview

In this tutorial, you will:

- Create an OLAP *model*, which is a logical structure that describes your overall business plan. The model takes the form of a *star schema* which represents the relationships between its components with a star-like structure. At the center of the star schema is a *fact table*, which contains the actual data that you want to analyze, such as product sales figures. Radiating from the fact table are the dimension tables which contain data that define the OLAP dimensions, such as account numbers, months, product names, and so on. One or just a few OLAP models can represent most or all aspects of your business. In the scenario of this tutorial, you will create an OLAP model that covers most of your business, The Beverage Company (TBC). Because you are creating a new model, you are not using the star schema you created in "Chapter 16. Creating a star schema from within the Data Warehouse Center" on page 103.
- Create an OLAP *metaoutline*, which is generally a subset of the model that you use to create an OLAP application. The idea is to create one or a few OLAP models, from which you can create many metaoutlines, each of which can look at a specific aspect of your business. Metaoutlines describe how the multidimensional database outline will look to the OLAP user. You can tailor the scope of a metaoutline by selecting which dimensions will be visible to OLAP users, and by setting filters that determine what data is retrieved. In the tutorial, you will create a metaoutline that is specific to the sales data in the Central states region of the TBC company..
- Load and calculate your data to create an *OLAP application*. An OLAP application contains data structured by an *Essbase outline*, or template, that is based on the metaoutline.
- Take a brief look at the other components of the DB2 OLAP Integration Server.

After you have finished the tutorial and created the OLAP application, you can analyze the TBC sales data from the Central states region using either the Microsoft Excel or Lotus 1-2-3 spreadsheet programs. See the *OLAP Spreadsheet Add-in User's Guide for 1-2-3* or *OLAP Spreadsheet Add-in User's Guide for Excel* for more information.

The DB2 OLAP Integration Server desktop contains the following components:

- The OLAP Model interface is a full-function tool for creating OLAP models. The DB2 OLAP Integration Server also includes an OLAP Model Assistant, which guides you to create a simple OLAP model. The OLAP Model Assistant does not have all the function of the full OLAP Model interface. The lessons in this tutorial show how to use the Assistant.
- The OLAP Metaoutline interface is a full-function tool for creating OLAP metaoutlines. The DB2 OLAP Integration Server also includes an OLAP Metaoutline Assistant, which guides you to create a simple OLAP

metaoutline. The OLAP Metaoutline Assistant does not have all the function of the full OLAP Metaoutline interface. The lessons in this tutorial show how to use the Assistant.

- With the Administration Manager tool you can perform simple OLAP database administration tasks such as exporting data to the Data Warehouse, managing storage, and creating new users and granting them access to applications. See the *OLAP Integration Server Administrator's Guide* for more information.
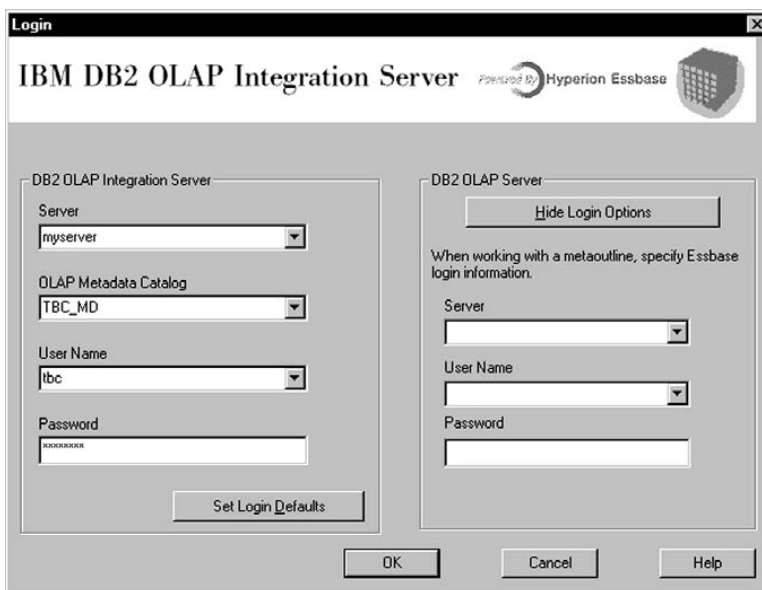
# Chapter 19. Starting the OLAP model

In this lesson, you will start the DB2 OLAP Integration Server desktop and log on to the DB2 database that will become the *OLAP catalog*, which contains the metadata that defines the model and metaoutline. You will also start the OLAP Model assistant.

## Starting the OLAP Integration Server desktop

Start the DB2 OLAP Integration Server desktop interface. Click *Start —> Programs —> IBM DB2 —> DB2 OLAP —> Desktop*. The Login window is displayed.



## Connecting to the OLAP catalog

On the Login panel, log in to the DB2 database that will contain the OLAP catalog.

1. Type the following values:

   **Server** The name of the server machine on which the DB2 OLAP Starter Kit server has been installed. Contact your database administrator for this information.

**Catalog**

> The name of the DB2 database in which you will store the metadata for your OLAP model. This tutorial uses the sample catalog shipped with DB2 Universal Database, which is called TBC_MD.

**User Name**

> The user ID you will use to access DB2 UDB. In the examples in this tutorial, the user ID is tbc.

**Password**

> The password for the user ID entered in *Username*.

2. You could click *Set Login Defaults* to store the server, catalog, and user ID you entered here as defaults for future use. For security reasons, the password is not stored. Do not save the defaults for this lesson.

3. Click *OK*. The Welcome window of the DB2 OLAP Integration Server Desktop is displayed:



.
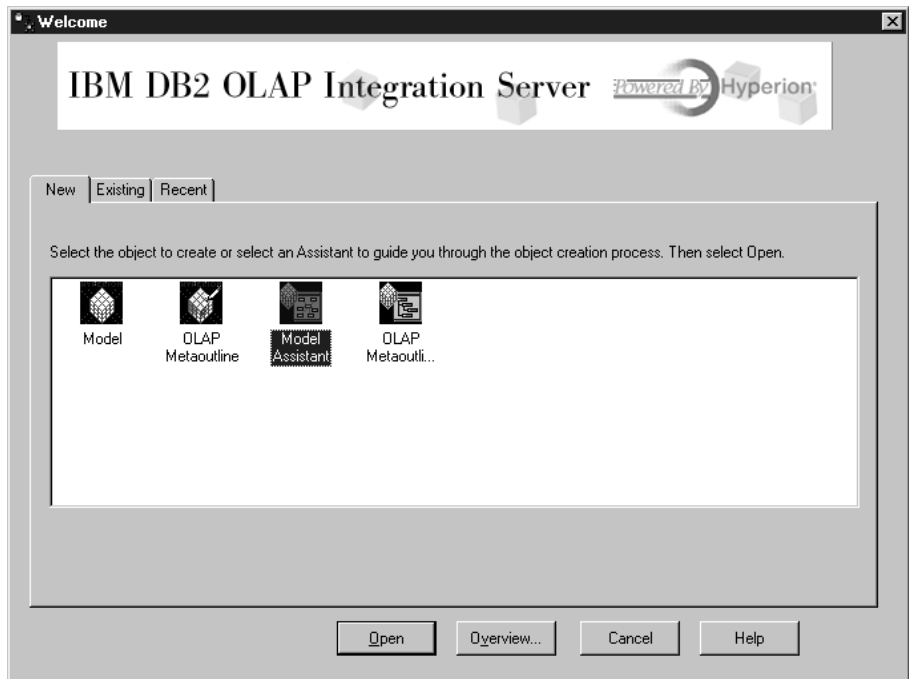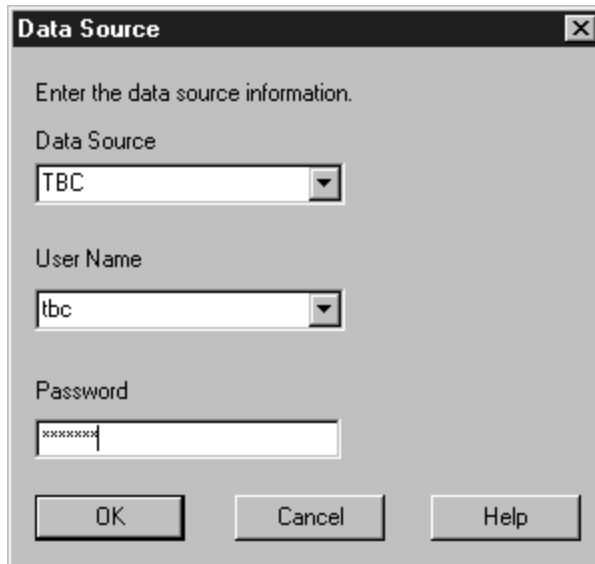
## Starting the Model Assistant

The first step in creating an OLAP model is to decide whether to use the OLAP Model interface, which offers full function, or the Model Assistant, which offers a simpler, guided approach. In this lesson, you will use the OLAP Model Assistant.

1. Double-click the **OLAP Model Assistant** icon.
2. You will be prompted to log on to the TBC database, which contains the source data. The Data Source window is displayed.



Type the following values:

**Data Source**
> The name of the DB2 database in which your business data is stored. This tutorial uses one of the sample databases shipped with DB2 Universal Database, which is called TBC.

**User Name**
> The user ID you will use to access DB2 UDB. In the examples in this tutorial, the user ID is tbc.

**Password**
> The password for the user ID entered in *Username*.

Click *OK*, and the Select Fact Table page of the OLAP Model Assistant is displayed.

## What you just did

In this lesson, you started the Integration Server desktop interface and connected to the metadata in the OLAP catalog. You also started the OLAP Model Assistant and connected to the DB2 database that holds your data source.

# Chapter 20. Selecting the fact table and creating dimensions

In this lesson, you will select a fact table and create the Accounts and Time dimensions. Then you will create the Product, Market, and Scenario dimensions.

## Selecting the fact table

Every model needs a fact table, which is the center of the star schema. The fact table is composed of one or more relational tables that contain facts, such as units sold or cost of goods, and foreign keys that link the fact table to each dimension table. In this lesson, the fact table will also be used as the Accounts dimension. The Accounts dimension contains numeric data that you want to analyze and track over time, such as sales or inventory; this data is also called variable *measures* data and enables accounting intelligence during online analytical processing.

To select the fact table:

1. On the Select Fact Table page, notice the default value of Accounts in the *Name* field. Do not change this value. In this lesson, you will automatically create an Accounts dimension using the fact table.
2. In the *Available Relational Tables* field, click **TBC.FACT_TABLE**.

3. Click the right arrow button, and **TBC.FACT_TABLE** is displayed in the *Selected Relational Tables* field. The window looks like this:
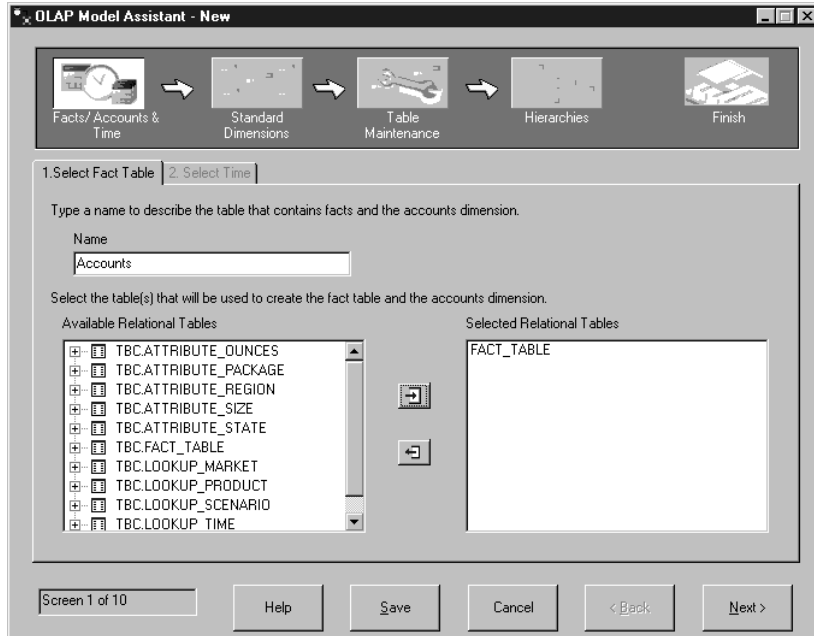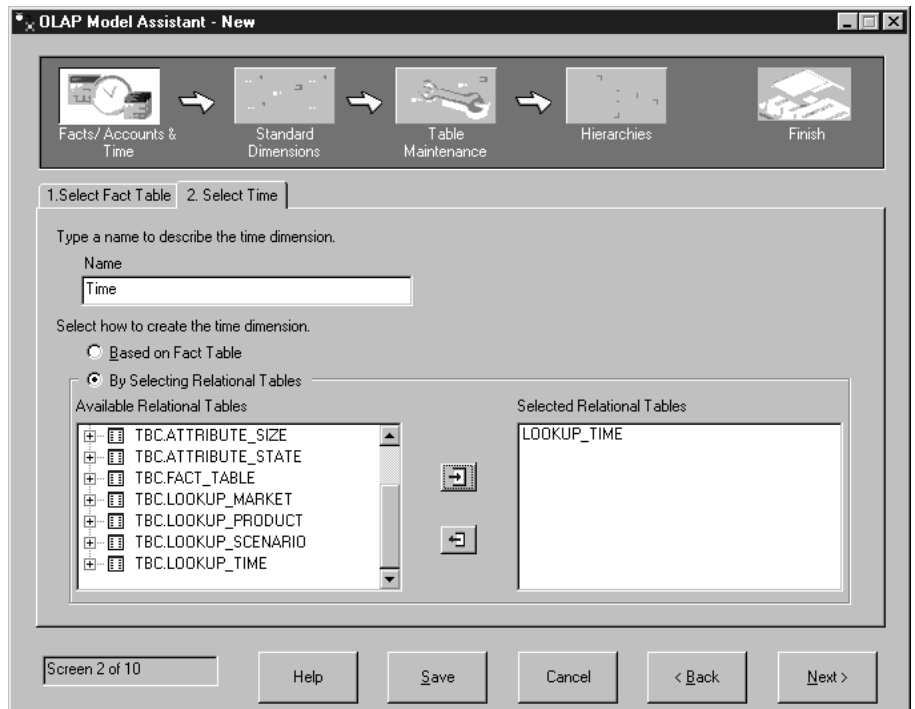


4. Click *Next* and the Select Time page is displayed.

## Creating the time dimension

In this exercise, you will create a Time dimension, which you use to describe how often you collect and update data. For example, you can organize time into years, quarters, and months.

1. On the Select Time page, you can accept the default value of Time in the *Name* field.
2. Click *By Selecting Relational Tables*, which activates the *Available Relational Tables* field.
3. Select the **TBC.LOOKUP_TIME** table and click the right arrow button. The table is displayed in the *Selected Relational Tables* field. The window

looks like this:



4. Click *Next* and the Name Dimensions page is displayed.

## Creating the standard dimensions

In this exercise, you will create standard dimensions for the model, and associate each standard dimension with a relational table that contain the data for that dimension. The dimensions are:

**Scenario dimension**
> Describes likely scenarios for analyzing data.

**Product dimension**
> Describes the products in your business. In this tutorial, your products are soft drinks.

**Market dimension**
> Describes the markets in which you operate. For example, you can organize your markets into regions and cities.

1. On the Name Dimensions page, type Scenario in the *Name* field and click *Add to list*. The Scenario dimension is added to the *Dimension List* field.

2. Follow the same process for the Product and Market dimensions. The window now looks like this:
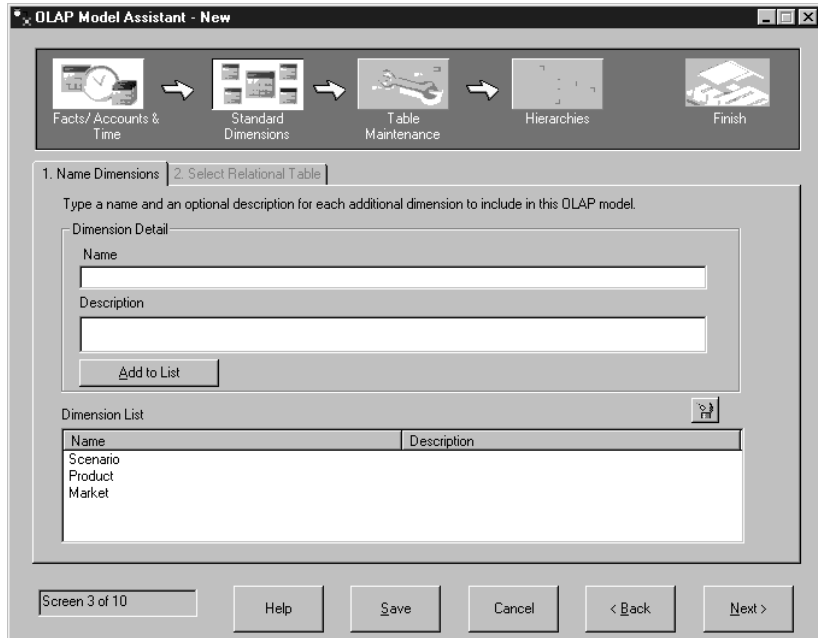


3. Click *Next* and the Select Relational Tables page is displayed.

On the Select Relational Tables page, you can associate one or more tables with the dimensions you created. Each dimension must have at least one table. The Accounts and Time dimensions are not listed because you have already created them.

1. In the *Dimension List* field, click the *Scenario* dimension.

2. Scroll down the *Available Relational Tables* list to the **TBC.LOOKUP_SCENARIO** table. Select the table and click the right arrow button next to the *Primary Dimension Table* field, and the table is added to the field. The table is also added under the *Primary Table* heading in the *Dimension List* field.

   If you wanted to associate additional tables for this dimension, you could select the table and click the right arrow next to the *Additional Dimension Tables* field. But for this lesson, do not add additional tables.

3. Follow the same process for the Product and Market dimensions. For the Product dimension, use the **TBC.LOOKUP_PRODUCT** table. For the Market dimension, use the **TBC.LOOKUP_MARKET** table. The window

looks like this:
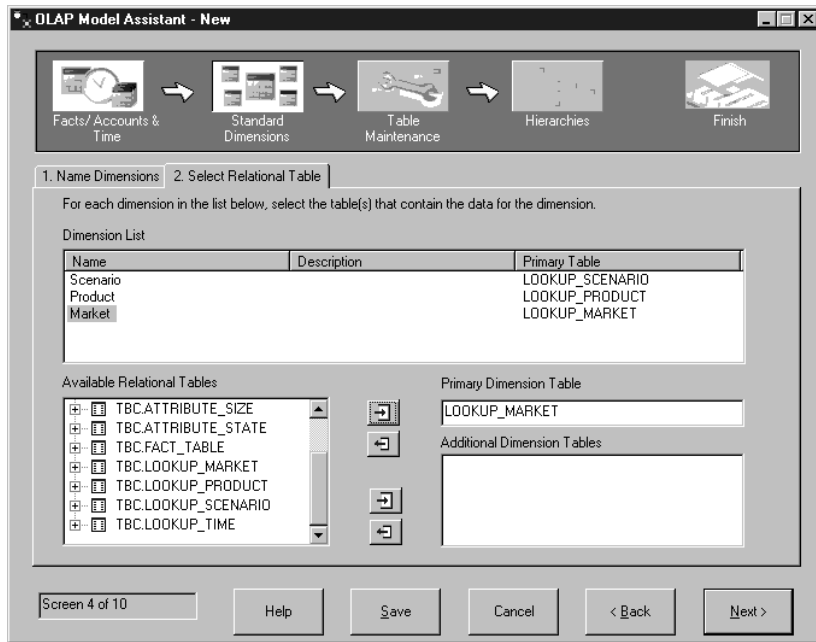


4. Click *Next* and the Fact Table Joins page is displayed.

## What you just did

In this lesson, you selected a fact table and created the Accounts and Time dimensions for your model. Then you created the Product, Market, and Scenario dimensions.

# Chapter 21. Joining and editing dimension tables

The star schema represents the relationships between the fact table and the other dimensions in the model. In this lesson, you will see how the structure of the star schema is defined by joins between the dimension tables and the fact table. You will learn how to hide columns in the dimension tables so that the columns do not appear as members of the dimensions in the model.

The left side of the Fact Table Joins page lists all the dimensions in the model. The right side shows which columns are joined between the dimension tables and the fact table, if a join exists. In the *Dimension List* field, an X next to a dimension means the dimension is joined with the fact table. Notice that all the dimensions are joined to the fact table.

1.  In this exercise you will show which column joins the fact table to the Time dimension. In the *Dimension List* field, select the **Time** dimension.
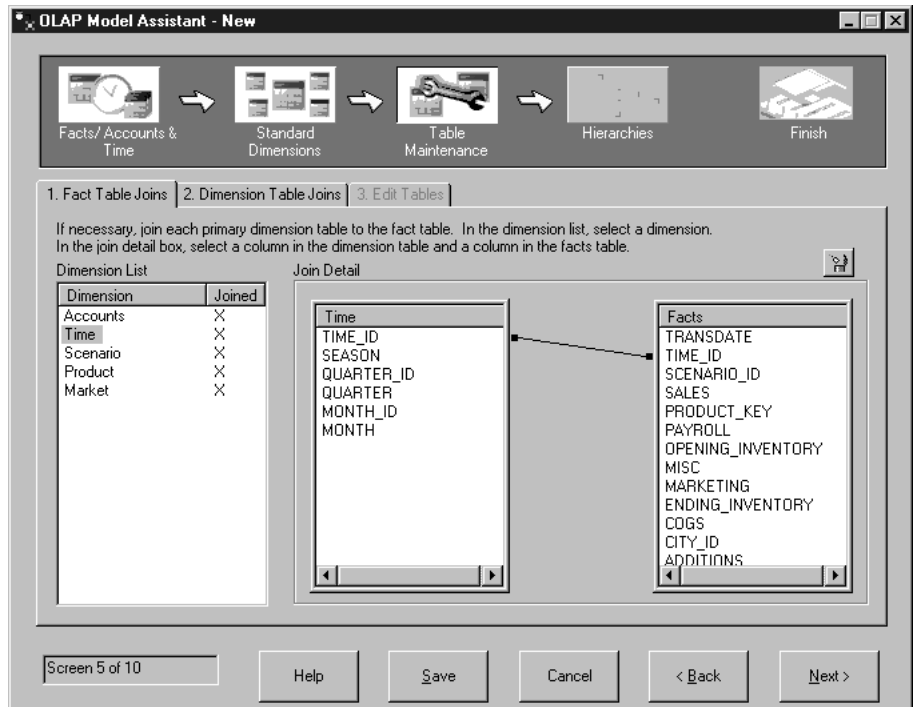


Notice that the `TIME_ID` column joins the fact table with the Time dimension.

2.  Click *Next* and the Dimension Table Joins page is displayed. You can use this page to create joins between the primary tables for the dimensions

and any additional dimension tables you added on the Select Relational
Tables page. In this tutorial, you did not add any additional relational
tables, so none are listed.

3. Click *Next* and the Edit Tables page is displayed.

## Editing dimension tables

In this exercise, you will hide a column in the Product dimension so that it
does not appear in the model:

1. In the *Dimension List* field, select the **Product** dimension.
2. Look for CAFFEINATED in the *Column Detail* field. Under the *Hidden* heading
   next to CAFFEINATED, click the check box. The window looks like this:
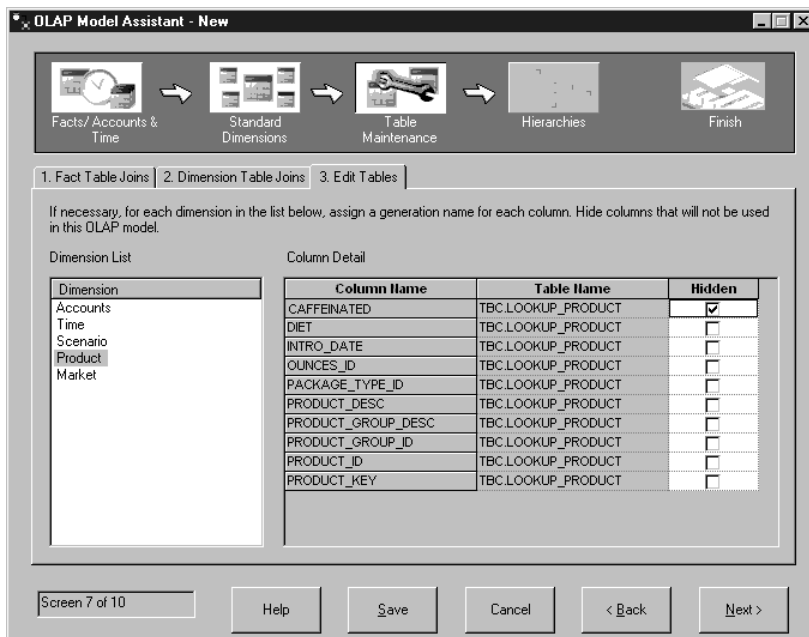


You could also give the columns more descriptive names without having
to change the column names in the source data. These names are called
*Essbase generation names* and they identify the columns in the final OLAP
application. If you do not assign Essbase generation names, they default to
the columns names. Do not assign generation names at this time.

3. Click *Next* and the Define Hierarchies page is displayed.

## What you just did

In this lesson, you saw how the structure of the star schema is defined by joins between the dimension tables and the fact table. You also learned how to hide columns in the dimension tables so that the columns do not appear as members of the dimensions in the model.

# Chapter 22. Defining hierarchies

In this lesson, you will create a hierarchy in one of the dimensions.
Hierarchies organize parent-child relationships among the columns of a
dimension and are displayed as a tree structure. For example, in the Time
dimension, you might define the Year member as the top of the hierarchy. The
Quarter member would be a child of Year, and Month would be a child of
Quarter.

## Creating hierarchies

In this exercise, you will create a hierarchy in the Markets dimension.

1. Select the Market dimension in the field on the left side of the Define
   Hierarchy page and click *Add Hierarchy*. The Add Hierarchy window is
   displayed.
2. In the *Name* field, type `Region-City` exactly as shown here (without
   spaces) and click *Done*. Notice that columns in the Market dimension are
   now displayed in the *Dimension Columns* field on Define Hierarchy page.
3. Select the **Region** column in the *Dimension Columns* field and click the
   right arrow button. The Region column is added to the *Parent/Child
   Relationship* field.
4. Select the **City** column in the *Dimension Columns* field and click the right
   arrow button. The City column is displayed as a child of Region column in

the *Parent/Child Relationship* field. The window looks like this:
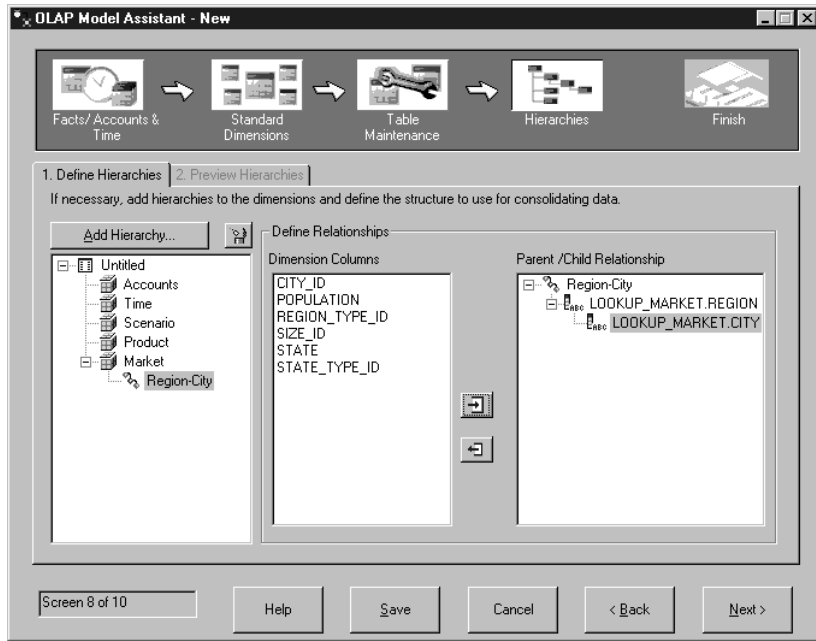


5. Click *Next* and the Preview Hierarchies page is displayed.

## Previewing hierarchies

In this exercise, after you have created all the hierarchies you want, you can see what kind of data they will present in the Preview Hierarchies page.

1. Open the tree structure for the Sample Outline in the *Essbase Outline Hierarchies* field. The window looks like this:
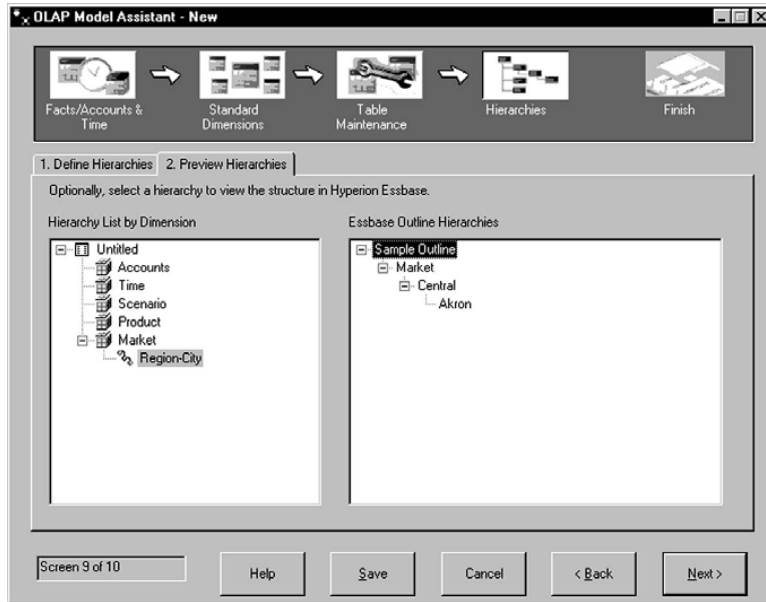


2. Click *Next* and the final window of the OLAP Model Assistant is displayed.

## What you just did

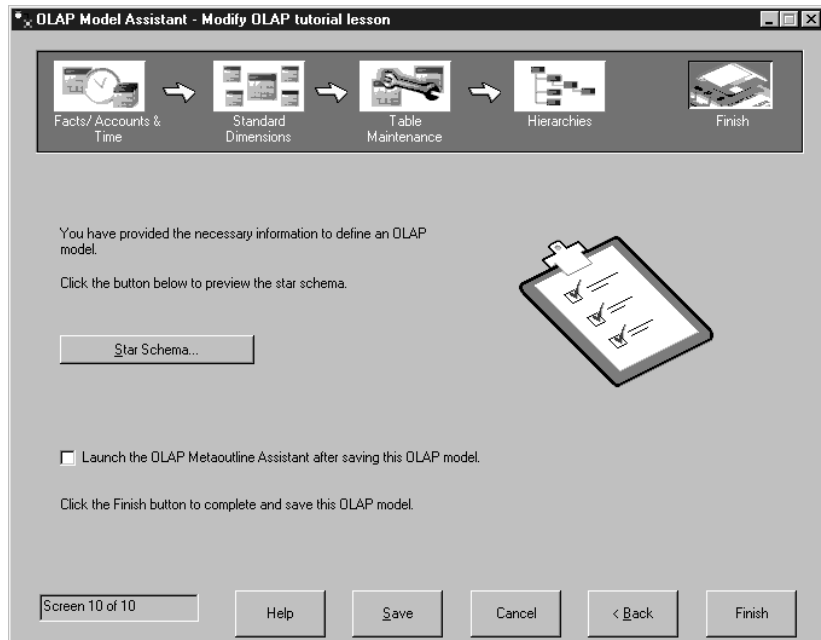In this lesson, you created and reviewed a hierarchy in the Market dimension.

# Chapter 23. Previewing and saving the OLAP Model

In this lesson, you will finish your OLAP model. You will look at a preview of the star schema you created and save the model in the database.

1. The final window of the Model Assistant looks like this:

2. Click the *Star Schema* button to view the star schema. This shows how the fact table is joined with all the dimension tables.
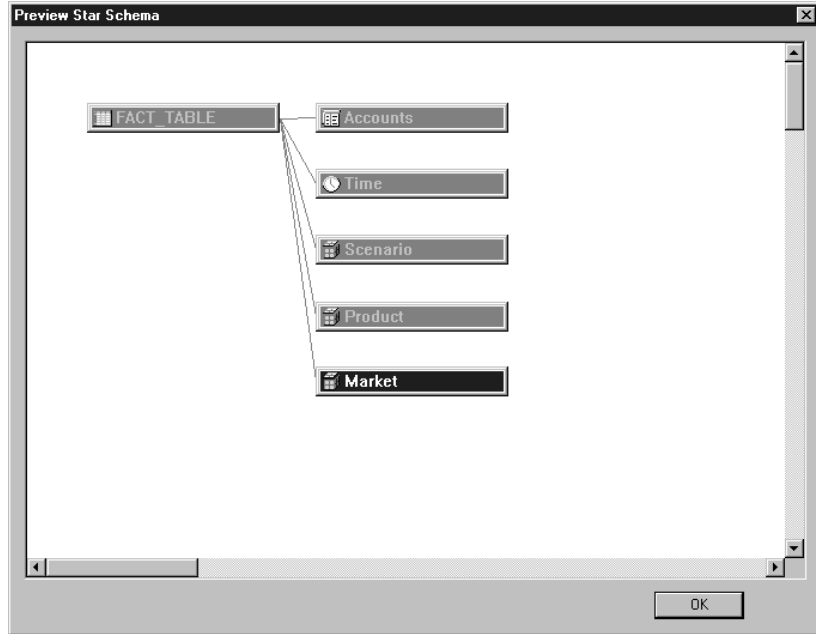


3. Click *OK*.
4. Do not check the *Launch the Metaoutline Assistant after Saving* box. In the rest of this tutorial, you will create a metaoutline based on the sample OLAP model supplied with DB2 Universal Database, not the model you just created, because the sample model provides more detail. In the next lesson, you will launch the Metaoutline Assistant manually.
5. Click *Finish* and then click *Yes* on the Save message. On the Save New OLAP Model page, you are prompted for a name for your model. Type My

tutorial model and click *OK*.
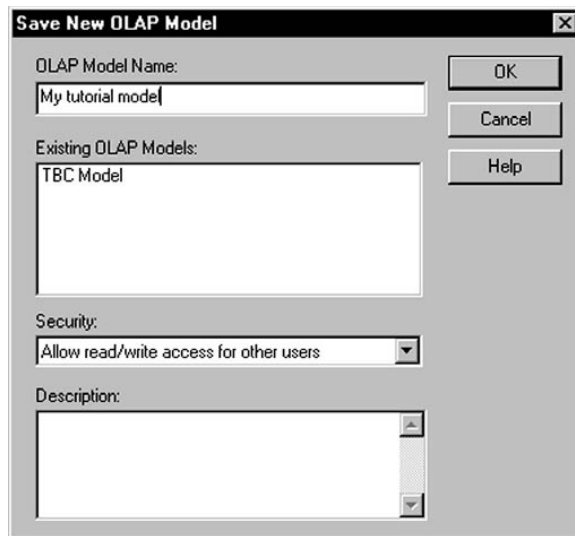


Your OLAP model is saved in the TBC database and the Integration Server desktop is displayed.

## What you just did

In this lesson, you previewed your star schema, finished your OLAP model, and saved it in the database.

# Chapter 24. Starting the OLAP metaoutline

Your ultimate goal is to create a OLAP metaoutline that includes measures in the Central States region, and excludes the other regions. From that metaoutline you will create an OLAP application that you can use to examine how sales data from the Central States changed during 1996.

The first step in creating an OLAP metaoutline is to decide whether to use the OLAP Metaoutline interface, which offers full function, or the Metaoutline Assistant, which offers a simpler, guided approach. In this lesson, you will start the OLAP Metaoutline Assistant, select an OLAP model to base your metaoutline on, and connect to the database.

## Starting the Metaoutline Assistant

After you log on to the DB2 OLAP Integration Server, the Desktop is displayed.

1. In the DB2 OLAP Server Desktop, click *File —> New*. The Welcome window is displayed.
2. Click the **OLAP Metaoutline Assistant** icon.
3. Notice that a new field, *Select the model the metaoutline should be based on*, is displayed at the bottom of the window. From the listbox, select **TBC Model**. This is not the model you created earlier in this tutorial; this

model is more detailed.
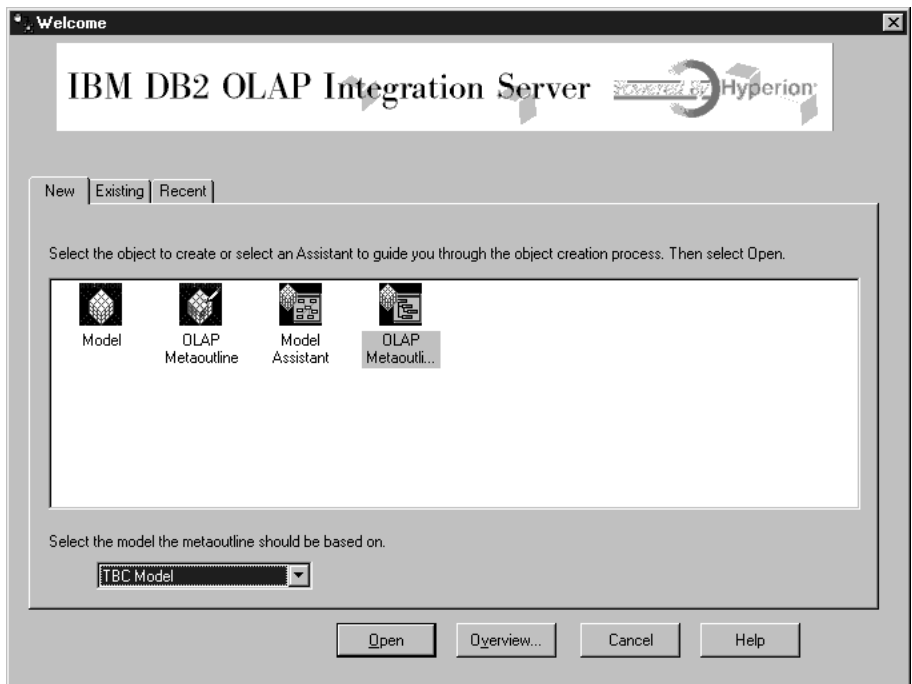


4. Click *Open* and you are prompted to log on to the source database.

## Connecting to the source database

1. In the Data Source window, type the following values:

**Data Source**
> The name of the DB2 database in which your business data is stored. . This tutorial uses one of the sample databases shipped with DB2 Universal Database, which is called TBC.
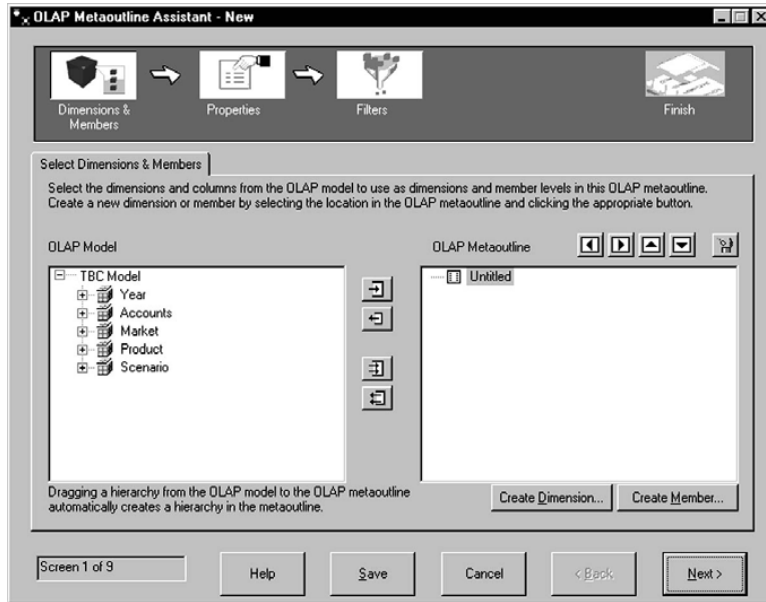
**User Name**
> The user ID you will use to access DB2 UDB. In the examples in this tutorial, the user ID is tbc.

**Password**
> The password for the user ID entered in *Username*.

Click *OK*. The Select Dimensions and Members page of the Metaoutline Assistant is displayed.



## What you just did

In this lesson, you started the OLAP Metaoutline Assistant, selected an OLAP model to base your metaoutline on, and connected to the database.

# Chapter 25. Selecting dimensions and members

In this lesson, you will select from the model the dimensions and members that you want to use in the metaoutline.

1. On the Select Dimensions and Members page, open the tree view of the Accounts dimension in the *OLAP Model* field until the tables are visible.
2. Select all the columns in the Accounts dimension and click the right arrow (Add) button. The accounts dimension and its columns are copied to the *OLAP Metaoutline* field.
3. In the *OLAP Model* field, open the tree view of the Year dimension until the hierarchies are visible. Click the **Year, Quarter, Month** hierarchy, and click the right arrow (Add) button. The hierarchy is copied to the *OLAP Metaoutline* field.
4. Open the tree structure of the Market dimension until the hierarchies are visible. Click the **Central Region, State** hierarchy, and click the right arrow (Add) button. The hierarchy is copied to the *OLAP Metaoutline* field. The window looks something like this:
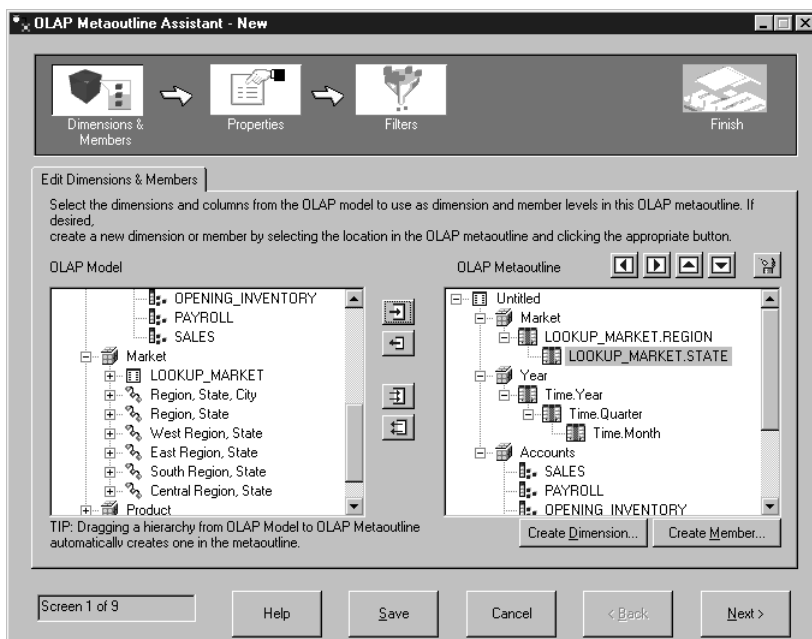


Notice that the metaoutline you are creating is a subset of the TBC model, not an exact duplicate. You selected the entire Accounts dimension but only one of the Time hierarchies, and only one Market region.

5.  Click *Next* and the Set Dimension Properties page is displayed.

## What you just did

In this lesson, you added the Accounts, Time, and Market dimensions to your metaoutline.

# Chapter 26. Setting properties

In this lesson, you will explore the properties of dimensions and members, and change one of the properties of a member of the Accounts dimension. These properties control how the Essbase outline is built in the OLAP application. You will also examine special properties of the Accounts dimension.

## Setting dimension properties

1. On the Set Dimensions Properties page, notice how the properties of the dimension are displayed to the right of the dimension name.
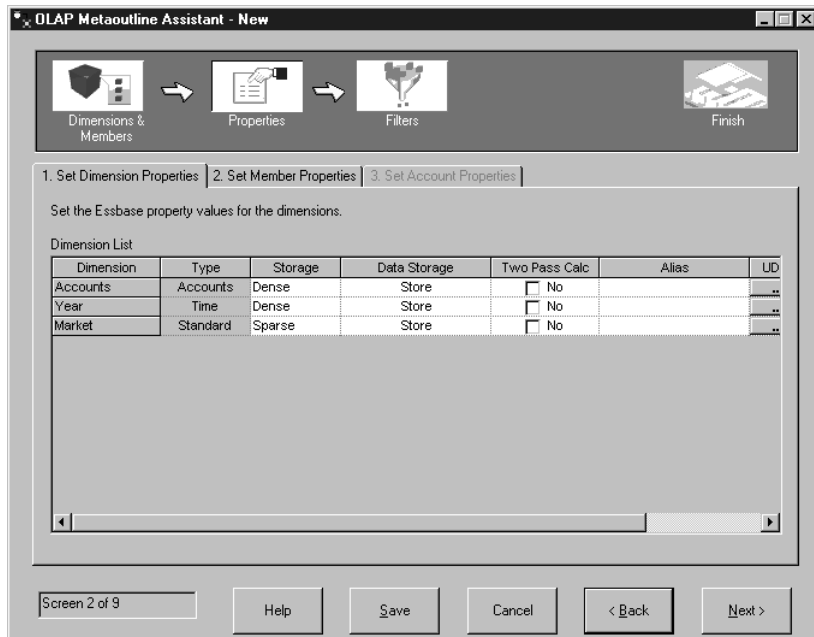


The fields in white are properties of the dimension that you can change. These properties affect all the members of a dimension.

**Storage**

Dimensions can be either Dense or Sparse. A Dense dimension is likely to contain data for every combination of dimension members, for example, the Time dimension. A Sparse dimension has a low probability that data exists for every combination of dimension members, for example, the Product or Market dimensions.

**Data Storage**

This property determines how and when data values are stored for a member. For example, you can store the value (the default), dynamically calculate and store the value, indicate that a member is shared between dimensions, and so on.

**Two Pass Calc**

Calculations are performed from the bottom up, from child values to parent values. The values of some child members might depend on parent values, requiring two calculations.

**Alias**   You can assign an alias name to the dimension.

**UDAs**   You can create a user-defined attribute (UDA) for the dimension.

2. Click *Next* and the Set Member Properties page is displayed.

---

## Setting member properties

In this exercise you will change a property for a dimension member.

1. Notice how the properties of the member are displayed to the right of the member name. The fields in white are properties of the dimension that you can change:

**Data Storage**

This property determines how and when data values are stored for a member. For example, you can store the value (the default), dynamically calculate and store the value, indicate that a member is shared between dimensions, and so on.

**Two Pass Calc**

Calculations are performed from the bottom up, from child values to parent values. The values of some child members might depend on parent values, requiring two calculations.

**Consolidation**

This property determines how child values are rolled up to parents. The default is a plus sign (+), which means the child values are added to the parent value.

**Alias**   You can assign an alias name to the dimension.

**UDA**   You can create a user-defined attribute (UDA) for the dimension.

2. On the Set Member Properties page, you can change some of the properties for individual members that you previously set for the entire dimension, plus one that is unique to members. Under the *Dimension* heading, click the Accounts dimension. Notice that the members are displayed in the *Member List* field.

3. Under the *Data Storage* heading, click the value in the *Consolidation* column for the *MISC* member. A list box button opens in the cell. Change the

property to a minus sign (–).
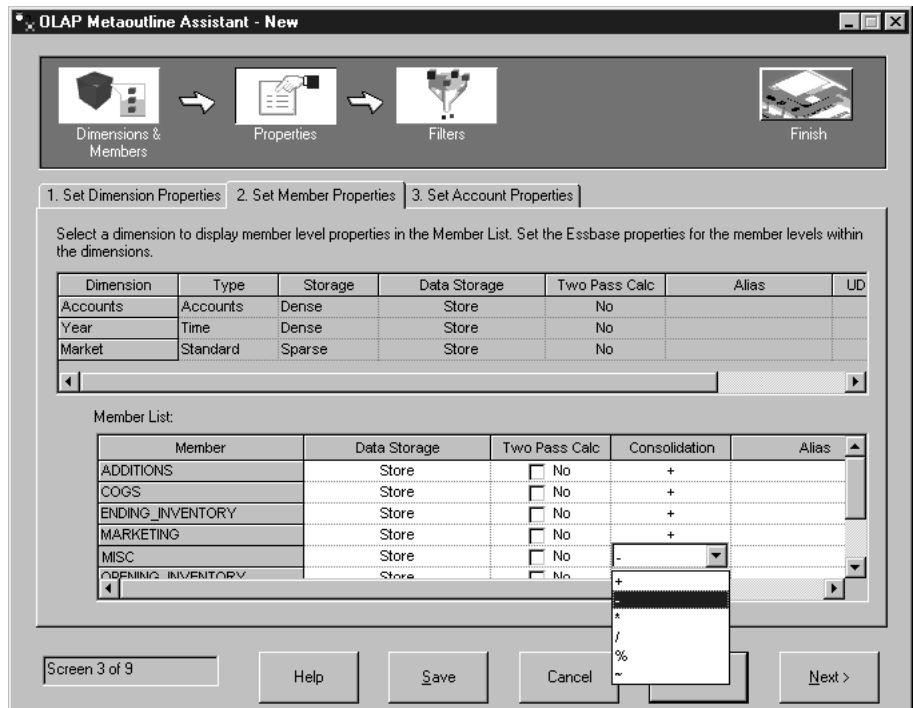


Now, when the values in the Misc member are rolled up into the Accounts dimension, the Misc values will be subtracted, not added.

4. Click *Next* and the Set Accounts Properties page is displayed.

## Examining account properties

In this exercise you will examine special properties for members of the accounts dimension:

1. On the Set Account Properties page, you can change the following properties for members of the Accounts dimension:

**Time Balance**
> This property defines how a member's parent is calculated in the Time dimension. The parent can represent the value of the first member. The default value of None allows an existing calculation associated with the member to determine the parent. Other values include First (the first value in a time period), Last (the last value in a time period), or Average (an average of all the values in the time period).

**Skip**   This property determines whether to skip a member when calculating a parent.

**Variance Reporting**

In Variance Reporting, the difference between budget and actual data is calculated.
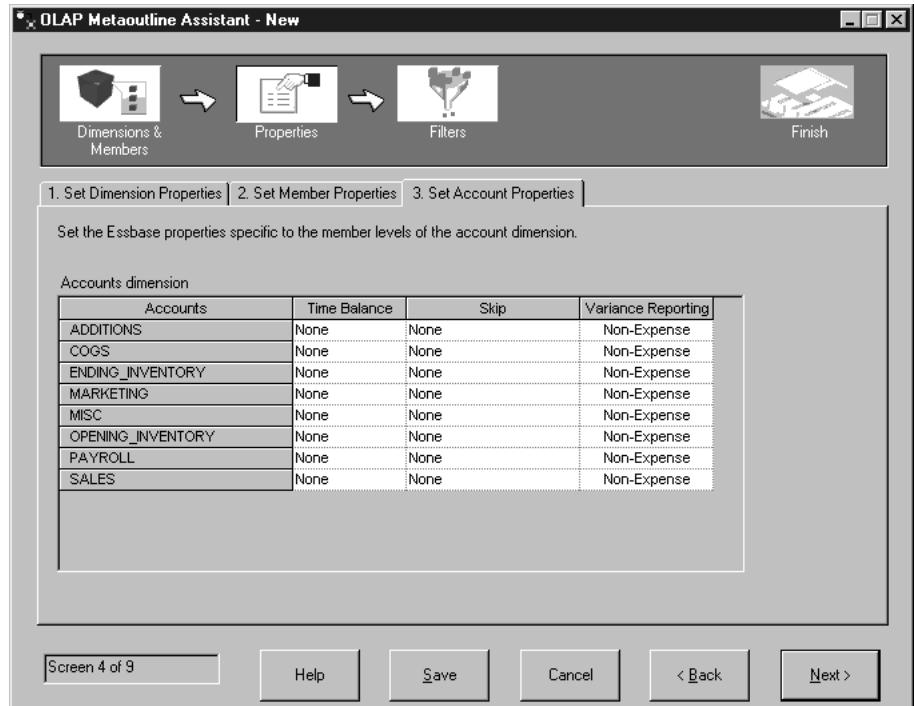


2. For the Accounts dimension, you can set these properties for each member:
3. Click *Next* and the Name Filters page is displayed.

## What you just did

In this lesson, you explored the properties of dimensions and members, and changed one of the properties of a member of the Accounts dimension. You also examined special properties of the Accounts dimension.

# Chapter 27. Setting filters

In this lesson, you will set filters that limit the members or data that are loaded into an OLAP application. For example, if the OLAP model contains data for the entire year, you can set a filter to load only the data for the second quarter.
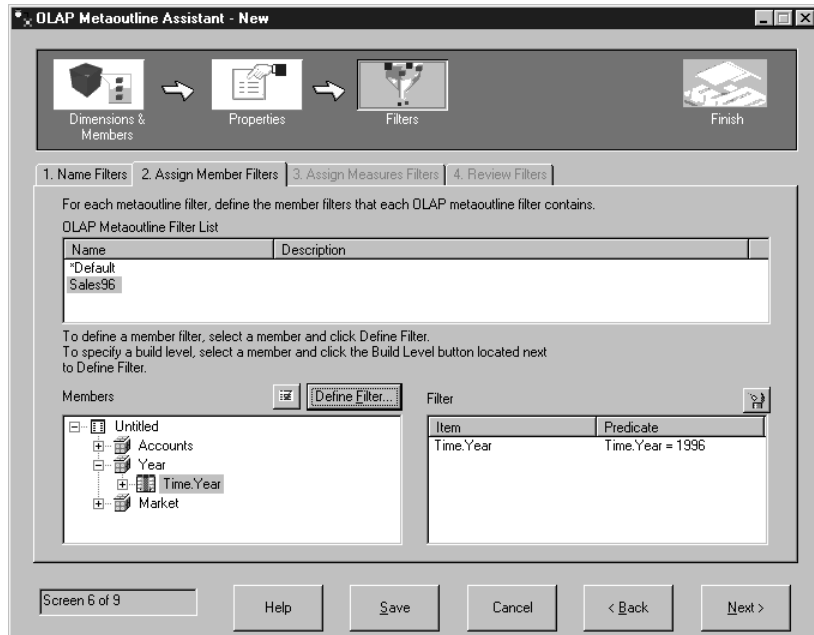
In this exercise, you will create a filter that limits the data loaded in your OLAP application to data from 1996.

1. On the Name Filters page, type `Sales96` in the *Name* field and click *Add to List*. The name is added to the *Metaoutline Filter List* field.
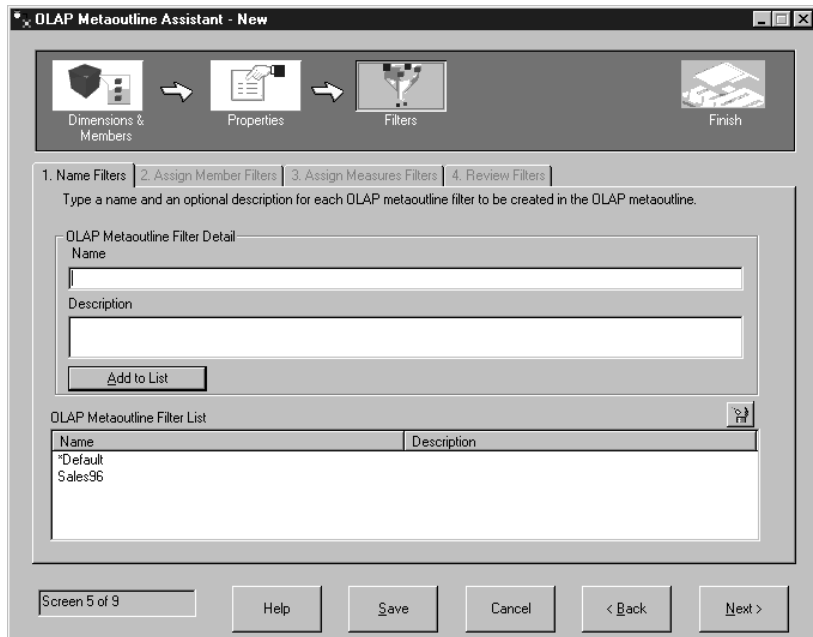


2. Click *Next* and the Assign Member Filters page is displayed.
3. Select **Sales96** in the *Metaoutline Filter List* field.
4. In the *Members* field, open the tree view of the Year dimension and select the **Time, Year** hierarchy.
5. Click *Define Filter* and the Filters window is displayed. Use this window to specify the data to be filtered from the OLAP application. You will create a filter on the time dimension that has one condition.
6. For the first condition, keep the value of `Year` in the *Column* field. In the *Operator* listbox, select the equals sign (=).

7. Click the button to the right of the *Condition* field and the *Select Values from Time, Year* window is displayed.

8. Select **1996** from the list and click *OK* to return to the Filters window.

9. On the Filters window, click *Add*, and the filter is displayed in the *Filters* field. If you want, you can edit the filter directly in the *Filters* field, but do not do that now.

10. Click *Verify* to make sure the syntax of the filter is correct, and click *OK* to close the Filters window. The Assign Member Filters page looks like this:



## Reviewing filters

In this exercise, you will examine how to set filters on dimension members, and review the filters you have created.

- Click *Next* and the Assign Measure Filters page is displayed. On this page you can define filters for dimensions that contain measures, such as the Accounts dimension. For example, you could open the tree view of the Accounts dimension, select the Sales table, and define a filter that limits sales to those greater than 100.

- Click *Next* and the Review Filters page is displayed. On this page you can look at all your filters. You can also go back to previous pages to edit existing filters or add more filters.

- Click *Next* and the Finish window is displayed.

## What you just did

In this lesson, you set a filter that limits the data loaded into your OLAP application to data from 1996.
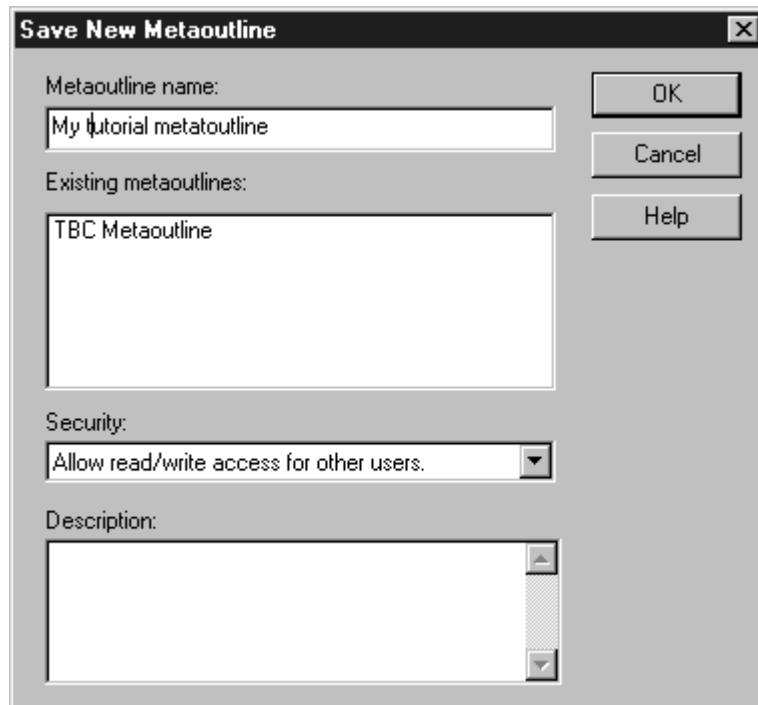
# Chapter 28. Creating the OLAP application

In this lesson, you will look at a preview of the metaoutline you created, save the metaoutline, and load and calculate the data, which creates the OLAP application.
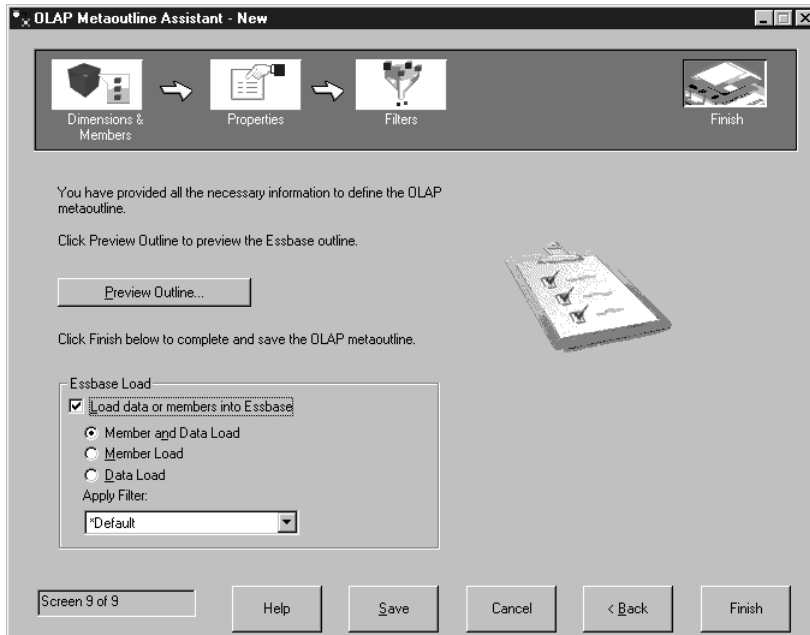


1. Click the preview button to view the metaoutline. The Sample Outline window is displayed. Click *Close*.
2. Keep the default for the *Load data and members into Essbase* check box.
3. Make sure the *Member and Data Load* button is selected.
4. In the *Apply Filter* field, select *\*default*. This is not the filter you created in "Chapter 27. Setting filters" on page 149.
5. Click *Finish* and you might be prompted to log into the TBC database. Then you are prompted for a name and other information about your model. Enter `MyMetaoutline`. The metaoutline is saved in the TBC database
6. You are prompted for the following information:
   - The name of the OLAP application that will contain the database into which you want to load data. In the *Application Name* field, type `MyAppl`.
   - The name of the OLAP database into which you want to load data. In the *Database Name* field, type `MyOLAPdb`.

- Calc Scripts. Select *Use Default Calc Script*.
- When to load the data. Select *Now* and click *Finish.*.

The Load and Calculate process begins. When it is complete, the OLAP application is created and you can analyze it using either the Microsoft Excel or Lotus 1–2–3 spreadsheet programs.

```
Save New Metaoutline                                    ×

  Metaoutline name:                                   OK
  My tutorial metatoutline
                                                    Cancel
  Existing metaoutlines:
                                                     Help
  TBC Metaoutline




  Security:
  Allow read/write access for other users.      ▼

  Description:

                                              ▲


                                              ▼
```

7. Click *Cancel* to close the Metaoutline Assistant.

## What you just did

In this lesson, you previewed the metaoutline you created, saved the metaoutline, and loaded and calculated the data, which created the OLAP application.

# Chapter 29. Exploring the rest of the Starter Kit

In this lesson, you will explore the OLAP Model, OLAP Metaoutline, and
Administration Manager interfaces of the DB2 OLAP Integration Server.

## Exploring the OLAP Model interface

In this exercise, you will open the TBC model in the OLAP model interface of
the DB2 OLAP Integration Server.

1. Click *File —> Open* to display the Existing page of the Welcome window.
2. Select *TBC Model* in the field on the left side of the window, and click
   *Open*.
3. Answer the prompts for the Data Source, User Name, and Password. The
   Data Source is TBC.
4. The OLAP Model interface is displayed with the star schema shown on
   the right pane. Click *View—> View all columns* to expand the fact table and
   dimensions. You can move the rearrange the dimensions in the right pane
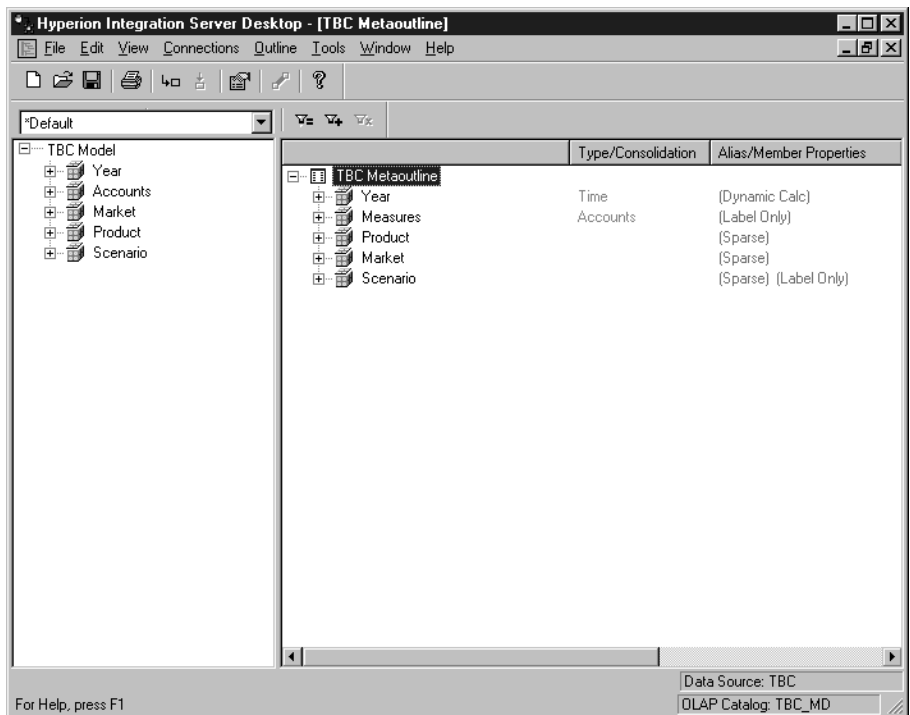   with the mouse. The window looks something like this:
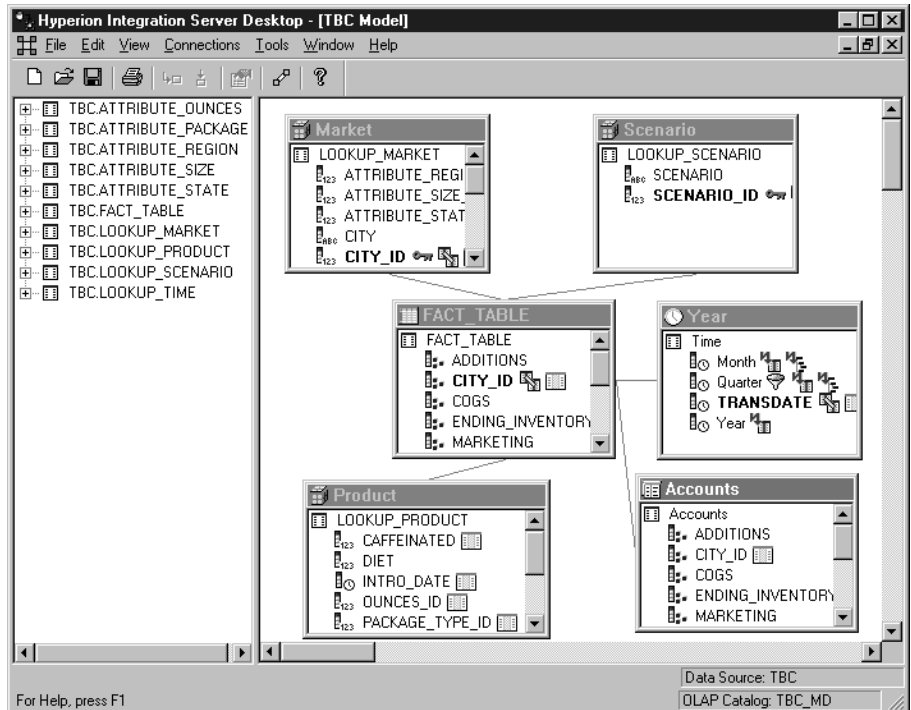
5. When you are finished, click *File —> Close*. Do not save changes.

## Exploring the OLAP Metaoutline interface

In this exercise, you will open the TBC model in the OLAP Metaoutline interface of the DB2 OLAP Integration Server.

1. Click *File —> Open* to display the Existing page of the Welcome window.

2. Expand the tree view of TBC Model and select *TBC Metaoutline*. Click *Open*.

3. Answer the prompts for the Data Source, User Name, and Password. The Data Source is TBC.

4. The OLAP Metaoutline interface is displayed with the metaoutline shown on the right side of the window. The window looks something like this:



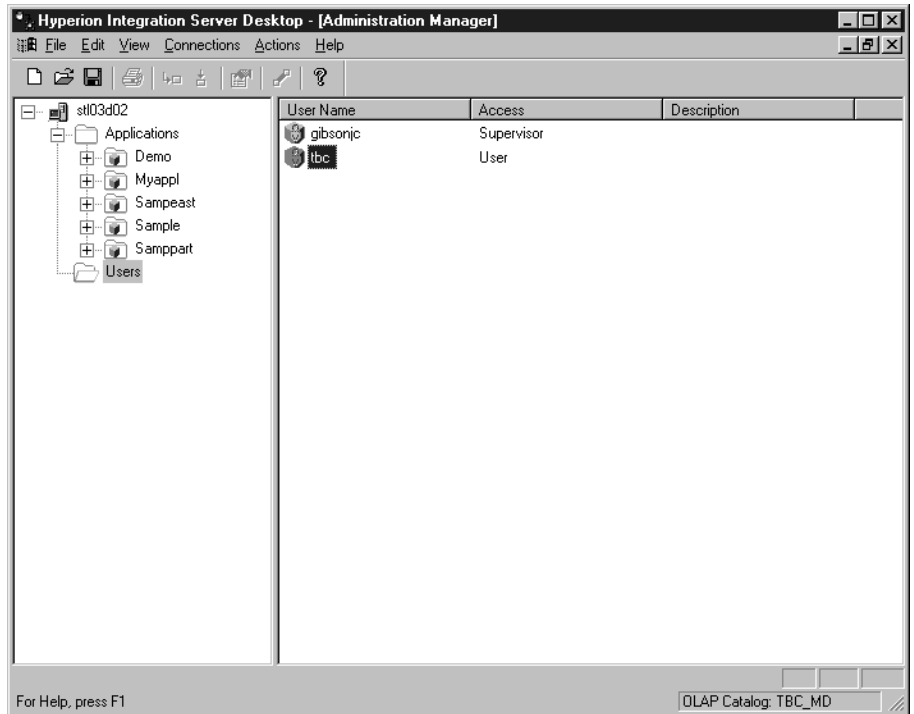5. When you are finished, click *File —> Close*. Do not save changes.

## Exploring the Administration Manager

In this exercise, you will examine the Administration Manager tool of the DB2 OLAP Integration Server.

1. Click *Tools —> Administration Manager* to display the Administration Manager.

2. Open the tree view in the left pane and click on *Users*. In this part of the
Administration Manager, you can create new users and grant them access.
The window looks something like this:



3. When you are finished, click *File —> Close*.
4. Click *File —> Exit* to exit the DB2 OLAP Integration Server.

## What you just did

In this lesson, you explored the OLAP Model, OLAP Metaoutline, and
Administration Manager interfaces of the DB2 OLAP Integration Server.

# Part 3. Appendixes

# Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will

be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Canada Limited
Office of the Lab Directory
1150 Eglinton Ave. East
North York, Ontario
M3C 1H7
CANADA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

## Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

| | |
|---|---|
| AIX | MVS |
| DB2 | OS/2 |
| DB2 OLAP Server | OS/390 |
| DB2 Universal Database | QMF |
| IBM | UNIX |
| IMS | |

Lotus and 1-2-3 are trademarks of Lotus Development Corporation in the United States, other countries, or both.

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.

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

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

**IBM** ®

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

Spine information:

IBM® DB2® Universal
Database

Business Intelligence Tutorial

Version 7

IBM