Getting Started
# Table of Contents

Introduction  

Chapter 1: IBM Cognos Data Manager  

Chapter 2: Creating Data Source Names  
   - Sample Databases  
   - Set Up Data Source Names  

Chapter 3: Exploring and Creating Catalogs  
   - IBM Cognos Data Manager Catalogs  
   - Explore an Existing Catalog  
   - Explore a User-Defined Folder  
   - Explore a Fact Build  
   - Explore a Dimension Build  
   - Explore a JobStream  
   - Explore a Metadata Export  
   - Explore the Library  
   - Explore the Visualization Pane  
   - Create a Catalog  

Chapter 4: Establishing Data Connections  
   - Connections  
   - Connect to an ODBC Database  
   - Connect to a SQLTXT Database  

Chapter 5: Creating the Dimensional Framework  
   - Dimensional Framework  
   - Identify Business Dimensions  
   - Set Up Reference Dimensions  
   - Set Up Reference Structures  
   - The Retailer Reference Dimension  
   - Set Error Handling  
   - The Time Reference Dimension  
   - The Product and SalesStaff Reference Dimensions  

Chapter 6: Creating a Data Mart  
   - Fact Builds  
   - Transformation Model  
   - Create a Fact Build  
   - Define the Purpose of the Fact Build  
   - Create the DataStream  
   - Assign the Element Types  
   - Define the Dimensions  
   - Define the Fact Delivery  
   - Define the Dimension Delivery  
   - Save Your Work  

Licensed Materials – Property of IBM  
© Copyright IBM Corp. 2005, 2010.
<table>
<thead>
<tr>
<th>Chapter 7: Creating Derivations</th>
<th>45</th>
</tr>
</thead>
<tbody>
<tr>
<td>Derivations</td>
<td>45</td>
</tr>
<tr>
<td>Create a Transformation Model Derivation</td>
<td>46</td>
</tr>
<tr>
<td>Create DataStream Derivations</td>
<td>46</td>
</tr>
<tr>
<td>Create the Total Value of Each Sale</td>
<td>47</td>
</tr>
<tr>
<td>Create the Gross Profit of Each Sale</td>
<td>47</td>
</tr>
<tr>
<td>Execute the DataStream</td>
<td>48</td>
</tr>
<tr>
<td>Map the DataStream to the Transformation Model</td>
<td>48</td>
</tr>
<tr>
<td>Deliver Additional Data to the Data Mart</td>
<td>49</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 8: Maintaining a Data Mart</th>
<th>51</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a Data Source to a Fact Build</td>
<td>51</td>
</tr>
<tr>
<td>Specify the Data Source</td>
<td>51</td>
</tr>
<tr>
<td>Map the Data Source to the DataStream</td>
<td>52</td>
</tr>
<tr>
<td>Deliver Data</td>
<td>53</td>
</tr>
<tr>
<td>Handle Rejected Data</td>
<td>53</td>
</tr>
<tr>
<td>Set Up the Location of Rejected Data</td>
<td>53</td>
</tr>
<tr>
<td>Check for Rejected Data</td>
<td>53</td>
</tr>
<tr>
<td>View Rejected Data</td>
<td>54</td>
</tr>
<tr>
<td>Consolidate the Data</td>
<td>54</td>
</tr>
<tr>
<td>Change the ORDER_DATE Dimension Element</td>
<td>54</td>
</tr>
<tr>
<td>Change the Dimension Table Template</td>
<td>55</td>
</tr>
<tr>
<td>Drop the Target Tables</td>
<td>56</td>
</tr>
<tr>
<td>Deliver Data</td>
<td>56</td>
</tr>
<tr>
<td>Analyze the Results</td>
<td>57</td>
</tr>
<tr>
<td>Multiple Granularity</td>
<td>57</td>
</tr>
<tr>
<td>Add a Data Source</td>
<td>57</td>
</tr>
<tr>
<td>Map the Data Source to the DataStream</td>
<td>58</td>
</tr>
<tr>
<td>Execute the DataStream</td>
<td>59</td>
</tr>
<tr>
<td>Map the DataStream to the Transformation Model</td>
<td>60</td>
</tr>
<tr>
<td>Deliver Data</td>
<td>61</td>
</tr>
<tr>
<td>Analyze the Results</td>
<td>61</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 9: Creating a Dimension Build</th>
<th>63</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimension Builds</td>
<td>63</td>
</tr>
<tr>
<td>Conformed Dimensions</td>
<td>64</td>
</tr>
<tr>
<td>Granularity in Conformed Dimensions</td>
<td>65</td>
</tr>
<tr>
<td>Create a Conformed Dimension</td>
<td>65</td>
</tr>
<tr>
<td>Create a Primary Reference Structure</td>
<td>65</td>
</tr>
<tr>
<td>Deliver the Primary Reference Structure</td>
<td>66</td>
</tr>
<tr>
<td>Create a Secondary Reference Structure</td>
<td>68</td>
</tr>
<tr>
<td>Associate the Secondary Reference Structure With a Fact Build</td>
<td>70</td>
</tr>
</tbody>
</table>
Table of Contents
Introduction

This document is intended for use with IBM® Cognos® Data Manager Designer.

This document is designed so that new users can quickly acquire a basic understanding of Data Manager Designer. The guide does not describe all the advanced features of Data Manager. For more information, see the IBM® Cognos® Data Manager User Guide.

Audience

You should be familiar with Microsoft® Windows® operating system and SQL. You should also have an understanding of multi-dimensional data analysis or Business Intelligence.

Before you begin to use this guide, you must have IBM® Cognos® Data Manager Designer, the Data Manager engine, and the sample data installed on your computer. These are automatically installed during a typical installation.

Finding information

To find IBM® Cognos® product documentation on the web, including all translated documentation, access one of the IBM Cognos Information Centers at http://publib.boulder.ibm.com/infocenter/cogic/v1r0m0/index.jsp. Updates to Release Notes are published directly to Information Centers.

You can also read PDF versions of the product release notes and installation guides directly from IBM Cognos product disks.

Accessibility features

This product does not currently support accessibility features that help users with a physical disability, such as restricted mobility or limited vision, to use this product.

Forward-looking statements

This documentation describes the current functionality of the product. References to items that are not currently available may be included. No implication of any future availability should be inferred. Any such references are not a commitment, promise, or legal obligation to deliver any material, code, or functionality. The development, release, and timing of features or functionality remain at the sole discretion of IBM.

Samples disclaimer

The Great Outdoors Company, GO Sales, any variation of the Great Outdoors name, and Planning Sample depict fictitious business operations with sample data used to develop sample applications for IBM and IBM customers. These fictitious records include sample data for sales transactions, product distribution, finance, and human resources. Any resemblance to actual names, addresses, contact numbers, or transaction values, is coincidental. Other sample files may contain fictional data manually or machine generated, factual data compiled from academic or public sources, or data used with permission of the copyright holder, for use as sample data to develop sample appli-
Introduction
Chapter 1: IBM Cognos Data Manager

The main purpose of IBM® Cognos® Data Manager is to create data warehouses and data repositories for reporting, analysis, and performance management. Data Manager does this by

- extracting operational data from multiple sources
- merging and transforming the data to facilitate enterprise-wide reporting and analysis
- delivering the transformed data to coordinated data marts

Data Manager can be used to transfer data into single database tables and, where more complex transformations are required, to populate data warehouse fact tables and dimension tables.

Data Manager integrates with other IBM® Cognos® products by

- delivering metadata to IBM® Cognos® Framework Manager. This allows target data warehouse and data repositories to be modeled and used in IBM® Cognos® Business Intelligence and performance management projects.
- publishing data movement tasks to IBM® Cognos® Connection where they can be run and scheduled.

Building the Data Foundation for the IBM Cognos Business Intelligence Solution

IBM® Cognos® Data Manager can deliver data to any supported target database into any design of database schema. However, Data Manager is best suited to delivering data into dimensional data warehouses and coordinated data marts. This approach has several advantages over non-dimensional data warehouses and uncoordinated data marts.

Dimensional data warehouses are typically made up of a series of coordinated data marts. The key to coordinated data marts is that common dimensions are shared across subject areas. This offers a common view of your business, while allowing flexibility and rapid incremental deployment. This approach is well suited to Business Intelligence because it allows a fast and iterative development approach to help the successful delivery of projects.

Data Manager supports this development approach by providing many of the tools and functionality required for the delivery of subject orientated data warehouses. The Data Manager dimensional framework allows the shared dimensions, known as conformed dimensions, to be constructed and then shared across all related subject areas. This reuse of the dimensional framework helps ensure that each subject area can be designed incrementally, but integrated consistently with any existing subject areas within the data warehouse.

You can choose to build enough dimensions and fact tables to complete one or more subject areas and then deploy the model to IBM® Cognos® Framework Manager for further modeling. When the model has been published, it can be used for reporting and analysis. Data Manager allows you to build on the model you have created and extend it to additional areas while maintaining maximum reuse of any completed work.
Data Manager provides all of the techniques required to easily build a scalable data foundation for your Business Intelligence project.
The sample data referred to throughout this guide is based on sales and marketing databases for a fictitious camping equipment company, the Great Outdoors Company.

For more information about the sample data that models the Great Outdoors Company, see "Sample Data" (p. 91).

In this chapter, you learn how to create the ODBC data source names that must be set up before you can begin to work with the sample databases supplied with IBM® Cognos® Data Manager.

## Sample Databases

A number of Microsoft® Access sample databases are supplied with IBM® Cognos® Data Manager. These are listed in the table below, together with a brief description.

<table>
<thead>
<tr>
<th>Database</th>
<th>Purpose</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS_Lessons.mdb</td>
<td>Source database for a Data Manager catalog</td>
<td>The main Data Manager catalog used in this guide</td>
</tr>
<tr>
<td>DS_New.mdb</td>
<td>Source database for a Data Manager catalog</td>
<td>An empty database in which you create a Data Manager catalog</td>
</tr>
<tr>
<td>DS_Marts.mdb</td>
<td>Target database for Data Manager catalogs</td>
<td>An empty database to which you deliver data from DS_Lessons, DS_New, and DS_Tutorial</td>
</tr>
<tr>
<td>DS_Tutorial.mdb</td>
<td>Source database for a Data Manager catalog</td>
<td>A Data Manager catalog that contains specimen solutions</td>
</tr>
<tr>
<td>DS_Advanced.mdb</td>
<td>Source database for a Data Manager catalog</td>
<td>A sample Data Manager catalog that contains more advanced examples, such as tracking attribute changes, derived dimensions, and user-defined functions</td>
</tr>
<tr>
<td>DS_Advanced_out.mdb</td>
<td>Target database for a Data Manager catalog</td>
<td>An empty database to store the data marts from DS_Advanced</td>
</tr>
<tr>
<td>MarketResearch.mdb</td>
<td>Source database for Data Manager data</td>
<td>Market research data for the Great Outdoors Company</td>
</tr>
<tr>
<td>GOSales.mdb</td>
<td>Source database for Data Manager data</td>
<td>Sales data for the Great Outdoors Company</td>
</tr>
</tbody>
</table>
### Set Up Data Source Names

You must set up a data source name (DSN) for each sample database. All DSNs are installed as system DSNs using the ODBC Data Source Administrator.

You can access the ODBC Data Source Administrator from within IBM® Cognos® Data Manager, as described here, or from the Microsoft® Windows® operating system Control Panel.

**Steps**

1. From the Start menu, click Programs, IBM Cognos 10, Data Manager.
2. In the Welcome dialog box, click Close.
3. From the Tools menu, click ODBC Administrator.
   
   The ODBC Data Source Administrator dialog box appears.
4. Click the System DSN tab.
5. Click Add.
   
   The Create New Data Source dialog box appears.
6. From the list, click Microsoft Access Driver (*.mdb), and then click Finish.
   
   The ODBC Microsoft Access Setup dialog box appears.
7. In the Data Source Name box, type DS_Lessons, and then click Select.
   
   The Select Database dialog box appears.
8. In the Directories box, browse to the following directory:
   
   `c10_location\webcontent\samples\DataManager\MSAccess`

   **Note:** The default location for `c10_location` is C:\Program Files\ibm\cognos\c10.
9. From the list of databases that appears, click DS_Lessons.mdb, then click OK.
10. Click OK to return to the ODBC Data Source Administrator dialog box.
11. Repeat steps 5 to 10 to create system DSNs for the following databases.

---

### Table: Sample Databases

<table>
<thead>
<tr>
<th>Database</th>
<th>Purpose</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GOVendors.mdb</td>
<td>Source database for Data Manager data</td>
<td>Vendor data for the Great Outdoors Company</td>
</tr>
<tr>
<td>GOSalesDW.mdb</td>
<td>Target database</td>
<td>A simple data warehouse</td>
</tr>
</tbody>
</table>

For details about the further sample data supplied with Data Manager, see "Sample Data" (p. 91).

---

12 IBM Cognos Data Manager
<table>
<thead>
<tr>
<th>Data source name</th>
<th>Database</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS_New</td>
<td>DS_New.mdb</td>
<td>\c10_location\webcontent\samples\DataManager\MSAccess</td>
</tr>
<tr>
<td>DS_Marts</td>
<td>DS_marts.mdb</td>
<td>\c10_location\webcontent\samples\DataManager\MSAccess</td>
</tr>
<tr>
<td>DS_Tutorial</td>
<td>DS_Tutorial.mdb</td>
<td>\c10_location\webcontent\samples\DataManager\MSAccess</td>
</tr>
<tr>
<td>DS_Advanced</td>
<td>DS_Advanced.mdb</td>
<td>\c10_location\webcontent\samples\DataManager\MSAccess</td>
</tr>
<tr>
<td>DS_Advanced_out</td>
<td>DS_Advanced_out.mdb</td>
<td>\c10_location\webcontent\samples\DataManager\MSAccess</td>
</tr>
<tr>
<td>GOMarket</td>
<td>MarketResearch.mdb</td>
<td>\c10_location\webcontent\samples\DataManager\db</td>
</tr>
<tr>
<td>GOSales</td>
<td>GOSales.mdb</td>
<td>\c10_location\webcontent\samples\DataManager\db</td>
</tr>
<tr>
<td>GOVendors</td>
<td>GOVendors.mdb</td>
<td>\c10_location\webcontent\samples\DataManager\db</td>
</tr>
<tr>
<td>GOSalesDW</td>
<td>GOSalesDW.mdb</td>
<td>\c10_location\webcontent\samples\DataManager\db</td>
</tr>
</tbody>
</table>

12. When you have finished, click **OK** to close the ODBC Data Source Administrator.
Chapter 3: Exploring and Creating Catalogs

In this chapter, you familiarize yourself with IBM® Cognos® Data Manager by learning how to

- Explore a catalog.
- Create a catalog.

IBM Cognos Data Manager Catalogs

An IBM® Cognos® Data Manager catalog provides a central repository for the information that defines how Data Manager extracts, transforms, and delivers data. You can store a Data Manager catalog in either a dedicated database or a database that you use for other purposes, such as a target data mart.

A catalog consists of a set of data tables that can reside in any database for which Data Manager has create permissions. You can create only one catalog per database.

The following diagram illustrates

- how Data Manager extracts and transforms data from the data sources and how it delivers data to the target data marts
- the flow of data from the data sources to the target data marts
- how Data Manager creates target conformed models for a data mart that can be exported as metadata
Explore an Existing Catalog

You can explore an existing catalog by opening it and studying its components, such as the fact builds, dimension builds, and the contents of the library.

Steps

1. From the File menu, click Open Catalog.
   
   Tip: You can also open a catalog from the Welcome dialog box that appears when you first open IBM® Cognos® Data Manager.

2. In the left pane of the Open Catalog dialog box, click ODBC.

3. In the Data Source Name box, click DS_Tutorial, and then click OK.

   The left side of the Data Manager window shows the DS_Tutorial catalog in a tree structure. Within the tree, there is a separate folder for each of the following components: builds and JobStreams, metadata exports, dimensions, connections, and functions.

   ![Tree structure of DS_Tutorial catalog]

   At the bottom of the tree there are a number of tabs:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Tab Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Catalog</td>
<td>Shows the full catalog</td>
</tr>
<tr>
<td></td>
<td>Fact Builds</td>
<td>Shows only the fact builds contained in the catalog</td>
</tr>
<tr>
<td></td>
<td>Dimension Builds</td>
<td>Shows only the dimension builds contained in the catalog</td>
</tr>
<tr>
<td></td>
<td>JobStreams</td>
<td>Shows only the JobStreams contained in the catalog</td>
</tr>
<tr>
<td></td>
<td>Library</td>
<td>Shows only the dimensions, connections, and functions contained in the catalog</td>
</tr>
<tr>
<td></td>
<td>Metadata</td>
<td>Shows only the metadata dimensions and collections contained in the catalog</td>
</tr>
</tbody>
</table>
Explore a User-Defined Folder

A user-defined folder is used to group together fact builds, dimension builds, and JobStreams. All user-defined folders are situated within the Builds and JobStreams folder.

Step

- Expand the **ConformedMart** user-defined folder.

  This folder contains a fact build, a number of dimension builds, and a JobStream.

Explore a Fact Build

A fact build specifies how to acquire, transform, and deliver fact and reference data.

Steps

1. Expand the **Example** fact build.

2. Expand **DataStream**.

   A DataStream controls the acquisition of data. It contains the data sources for the fact build and specifies how the columns of each data source map to the DataStream.

3. Expand **Transformation Model**.

   The transformation model lists the elements of the fact build and denotes their type. It specifies how each DataStream item maps to the transformation model.

4. Fully expand **Delivery Modules**.

   This contains fact and dimension deliveries for the fact build.

Explore a Dimension Build

A dimension build specifies the delivery of reference data only. You use dimension builds to manage shared dimension tables for conformed data marts.
Chapter 3: Exploring and Creating Catalogs

Step

- In the ConformedMart user-defined folder, expand the Dates dimension build.

  Note that the structure of a dimension build is similar to that of a dimension delivery in a fact build.

Explore a JobStream

A JobStream is used to group other components into logical processes so that you can automate its execution.

Step

- In the ConformedMart user-defined folder, expand the Example JobStream.

  This JobStream is used to automate the execution of a number of dimension builds, and a single fact build.

Explore a Metadata Export

A metadata export allow you to create descriptions of target conformed models for a data mart or data warehouse.

Steps

1. Expand Metadata.

2. Expand the Dimensions folder.

   This lists all the metadata dimensions that are set up. A metadata dimension contains the description of the conformed dimension that you want to include in the metadata export.

3. Expand the Collections folder.

   This lists all the metadata collections that are set up. A metadata collection groups together all the metadata star models that you want to include in a metadata export. A metadata star model
contains the description of the fact table that you want to include in the metadata export, with reference to the metadata dimensions you have set up.

**Explore the Library**

The library contains dimensions, connections, and user-defined functions. These components are common to all builds and JobStreams contained within the catalog.

**Steps**

1. Expand Library.

2. Expand the **Dimensions** folder.

   You set up the dimensional framework within this folder. At the top level, the folder contains all the reference dimensions used in the catalog. Each reference dimension corresponds to a single business dimension.

3. Expand the **Product** reference dimension.

   A reference dimension contains reference structures (hierarchies, auto-level hierarchies, and lookups) and one or more templates.

4. Expand the **Product** hierarchy.

   Each hierarchy has one or more levels that represent degrees of consolidation.

5. Within the Product dimension, expand the **Templates** folder.

   Each template provides information that is required to properly maintain and use the corresponding reference dimension table.

6. Expand the **Connections** folder.
Each connection specifies a data source or data target and provides the information that IBM® Cognos® Data Manager requires to connect to the data.

7. Expand the **Functions** folder.

```plaintext
<table>
<thead>
<tr>
<th>Function Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Margin</td>
<td>Price, Numeric, Cost, Numeric</td>
</tr>
<tr>
<td>Strip TimePeriod</td>
<td>Date, String</td>
</tr>
</tbody>
</table>
```

User-defined functions can be used to derive new values within build variables, derivations, derived dimensions, fact delivery output filters, and JobStreams.

**Explore the Visualization Pane**

The right side of the IBM® Cognos® Data Manager window contains graphical representations of the builds, reference structures, JobStreams, and metadata exports in the catalog.

For information about the icons used within the Visualization pane, see the IBM® Cognos® Data Manager User Guide.

**Tip:** If you cannot see the Visualization pane, from the View menu, click **Show Visualization**.

**Steps**

1. Click the **Example** fact build.

2. Click the **Example** tab.

   An overview of the fact build appears.

   **Tip:** You can view and set the properties of many components by double-clicking their icons.

3. Click the **Mapping** tab.

   The mapping between the data source columns, DataStream items, and transformation model elements of the fact build appears.
4. Click the Transformation Model tab.

You can see the elements of the fact build, and the mapping between the dimension elements and hierarchies from the dimensional framework.

For each hierarchy, Data Manager shows the levels at which the fact build acquires and delivers data. A check mark to the left of a level indicates that data is to be acquired, a check mark to the right indicates that data is to be delivered.

5. Click the Fact Delivery tab.

You can see the mapping between the transformation model elements and the columns of the delivered fact data tables.

Create a Catalog

You can create a catalog in the DS_New sample database. You will use this catalog again in the next chapter.

Steps

1. From the File menu, click New Catalog. If you are prompted to disconnect from the current catalog, click OK.
Chapter 3: Exploring and Creating Catalogs

The New Catalog dialog box appears.

2. In the Name box, type MyCatalog and then click Next.

3. In the left pane, click ODBC.

4. In the Data Source Name box, click DS_New.

5. Click Finish.

IBM® Cognos® Data Manager creates the catalog tables and opens the new catalog.

6. From the File menu, click Save Catalog.
Chapter 4: Establishing Data Connections

In this chapter, you establish connections within the catalog you created in Chapter 3. You learn how to

- Set up an ODBC connection.
- Set up a SQLTXT connection.

Connections

You use connections to specify data sources and data targets.

A connection provides all the information required to connect to a database. It identifies the particular database and specifies the connection method. It also provides information that the database management system (DBMS) requires, such as the username and password.

IBM® Cognos® Data Manager supports database management systems from many manufacturers, including ORACLE, Microsoft®, Informix®, Sybase, and Teradata using native drivers and ODBC. Data Manager also supports data sourcing from SAP R/3 systems using the add-on IBM® Cognos® Data Manager Connector for SAP R/3 product.

In addition, Data Manager can use the SQLTXT driver to connect to delimited text (ASCII or EBCDIC) data sources. This allows Data Manager to use data from any database or application that supports delimited text exports.

For the current list of supported DBMS versions and manufacturers, see http://www.ibm.com/software/data/support/cognos_crc.html.

You can also use a published IBM® Cognos® Framework Manager package connection to access query subjects from a published IBM® Cognos® Business Intelligence package.

Connect to an ODBC Database

You must define connections to the ODBC databases set up for the Great Outdoors Company. IBM® Cognos® Data Manager adds the connections to the Connections folder.

Steps

1. Open the MyCatalog catalog that you created in Chapter 3.
2. In the Library, right-click Connections, then click Insert Connection.
3. In the Alias box, type GO_Sales
4. Click the Connection Details tab.
5. In the left pane, click ODBC.
6. In the Data Source Name box, click GOSales.
7. Click **Test Connection**, then click **OK** to confirm the message that appears.

8. Click **OK** to close the **Connection Properties** dialog box.

9. Repeat steps 2 to 8 to create the following connections:
   - GO_Vendors that refers to the GOVendors data source
   - DS_Marts that refers to the DS_Marts data source

10. From the **File** menu, click **Save Catalog**.

### Connect to a SQLTXT Database

You must also define a connection to the DS_Sources SQLTXT database for the Great Outdoors Company. This database contains a dates master table and additional source data that you will use later. IBM® Cognos® Data Manager adds the connection to the Connections folder.

Using the SQLTXT connection type, you can use SQL queries to access data in text files of various formats.

**Steps**

1. In the **Library**, click the **Connections** folder.

2. From the **Insert** menu, click **Library**, and then click **Connection**.

3. In the **Alias** box, type **DS_Sources**

4. Click the **Connection Details** tab.

5. In the left pane, click **SQLTXT**.

6. In the **Definition File** box, type:
   
   `SCOG_ROOT\webcontent\samples\DataManager\sqltxt\DS_Sources.def`

7. Click **Test Connection**, then click **OK** to confirm the message that appears.

8. Click **OK** to close the **Connection Properties** dialog box.

9. From the **File** menu, click **Save Catalog**.

   You have now finished creating ODBC and SQLTXT connections.
In this chapter, you create the dimensional framework in the catalog you created in Chapter 3. IBM® Cognos® Data Manager uses this dimensional framework to provide information when delivering dimension data to data marts and when consolidating the data.

You learn how to

- Identify business dimensions.
- Set up reference dimensions.
- Set up reference structures.

**Dimensional Framework**

The dimensional framework defines the hierarchy structure of core business elements. It represents the way in which an organization thinks about its data independently of the physical data source. Each dimension is a grouping of related information about a specific aspect of the business. A dimension provides the context for analysis, such as products, customers, and sales staff. IBM® Cognos® Data Manager supports an unlimited number of reference dimensions, and therefore unlimited dimensionality.

A reference dimension contains reference structures (hierarchies, auto-level hierarchies, and lookups) and one or more templates.

**Hierarchies**

A hierarchy presents a particular view of a business dimension. It contains the definition of related reference data that is organized into a fixed number of levels. Each level contains a set of members, which are the unique occurrences of the data for the level. Members at a level can relate to their parents at the level immediately above, and their children at the level immediately below.

**Auto-level Hierarchies**

An auto-level hierarchy does not have a fixed number of levels. It is structured solely in terms of parent-child relationships. For example, employees report to managers, who may report to other managers.

**Lookups**

A lookup is a single level structure used primarily for data validation.

**Templates**

A template defines the attributes, and their behaviors, for a dimension table. A template is required to properly maintain and use the corresponding reference dimension table.
Identify Business Dimensions

Business dimensions are the core components of a business, anything that you want to analyze in reports.

The Great Outdoors Company obtains its profits by selling products. A Product dimension lets you analyze company performance by product, product type, and product line.

Product sales may have seasonal trends. Knowing these trends, management can plan to stock more of each product during the time it is most successful. To identify these trends, the data mart must contain a Time dimension.

A SalesStaff dimension lets you analyze how the different sales territories, countries, sales branches, and individual salespersons perform with respect to product sales and revenue generation.

Great Outdoors retailers are the customers of the Great Outdoors Company. A Retailer dimension can answer queries such as, “What products sell to which retailer?”.

Using the information above, you know that you need to model these components as IBM® Cognos® Data Manager reference dimensions:

- Product dimension
- Time dimension
- SalesStaff dimension
- Retailer dimension

Set Up Reference Dimensions

You must define the reference dimensions for each of the business dimensions you identified for the Great Outdoors data. IBM® Cognos® Data Manager adds all the reference dimensions that you define to the Dimensions folder.

Steps

1. Open the catalog named MyCatalog that you created in Chapter 3.
2. In the Library, right-click Dimensions, then click Insert Reference Dimension.
3. In the Name box, type Product
4. Click OK.
5. Repeat steps 2 to 4 to create the following reference dimensions:
   - Time
   - SalesStaff
   - Retailer
Set Up Reference Structures

For each reference dimension that you have set up, you must set up a hierarchy and the required templates. Each hierarchy models a view of the associated business dimension.

IBM® Cognos® Data Manager supports hierarchies based on

- the columns of one table (star schema)
- multiple tables (snowflake schema)
- the rows of one table (parent-child schema)
- parent-child structures without named levels (auto-level hierarchies)

For more information about hierarchy types, see the IBM® Cognos® Data Manager User Guide.

Here, you use the Hierarchy wizard to create the hierarchies.

Tip: You can also manually create or refine hierarchies. For information about maintaining hierarchies manually, see the IBM® Cognos® Data Manager User Guide.

The Retailer Reference Dimension

For the Retailer reference dimension, you set up a hierarchy that represents the retailers of the Great Outdoors Company’s products.

Create the Retailer Hierarchy

You create the Retailer hierarchy from related tables in the GO_Vendors database. The hierarchy follows a one-to-many relationship between tables. A retailer type can have many retailers, but a retailer can only belong to one retailer type. Each retailer can have many sites, but a site can have only one retailer. These tables represent a snowflake schema.

Steps to Select the Hierarchy Structure

First, you specify the type of hierarchy you want to create.

1. From the Tools menu, click Hierarchy Wizard.
2. Click Create the hierarchy from multiple tables (Snowflake Schema), and then click Next.

Steps to Define the Hierarchy

On this page, you enter a name for the hierarchy and specify the reference dimension to which the hierarchy belongs.

1. In the Enter the name of the hierarchy box, type Retailer
2. In the **Select the reference dimension to use for this hierarchy** box, click Retailer, and then click **Next**.

**Step to Define the All Level**

IBM® Cognos® Data Manager provides the option of creating a static All level at the top of the hierarchy. This is useful for obtaining a sum total of all data at the lower levels of the hierarchy.

- Click **Next** to accept the defaults and create the All level.

**Steps to Define the RetailerType Level**

On this page, you insert the second level of the hierarchy, the RetailerType level.

When defining a level you must specify the source of the data (the database and table) and the attributes (the columns to include from the data source).

1. Click the **ALLRetailer** level of the Retailer hierarchy, and then click **Add**.

   The Level Details window appears.

2. In the **Name** box, type **RetailerType**

3. In the **Source database** box, click **GO_Vendors**.

4. In the **Source table** box, click **Browse for table**  
    and then double-click the **RETAILER_TYPE** table.

5. Move each table column from the **Available attributes** pane to the **Chosen attributes** pane.

6. For the **RETAILER_TYPE_CODE** attribute, select the **Id** check box.

   RETAILER_TYPE_CODE is now the unique identifier for the level.

7. For the **TYPE_NAME_EN** attribute, select the **Caption** check box.

8. Click **OK**.

**Steps to Define the Retailer Level**

Here, you insert the third level of the hierarchy, the Retailer level.

1. Click the **RetailerType** level of the Retailer hierarchy, and then click **Add**.

   The Level Details window appears.

2. In the **Name** box, type **Retailer**

3. In the **Source database** box, click **GO_Vendors**.

4. In the **Source table** box, click **Browse for table**  
    and then double-click the **RETAILER** table.

5. Move the **RETAILER_CODE**, **COMPANY_NAME**, and **RETAILER_TYPE_CODE** table columns from the **Available attributes** pane to the **Chosen attributes** pane.

6. For the **RETAILER_CODE** attribute, select the **Id** check box.

   RETAILER_CODE is now the unique identifier for the level.
7. For the COMPANY_NAME attribute, select the Caption check box.

8. For the RETAILER_TYPE_CODE attribute, select the Parent check box.
   When you execute a build, IBM® Cognos® Data Manager creates links between a level and its
   parent by matching the Parent attribute of the current level with the ID attribute of the level
   above.

9. Click OK.

Steps to Define the Site Level
Here, you insert the lowest level of the hierarchy, the Site level.

1. Click the Retailer level of the Retailer hierarchy, and then click Add.
   The Level Details window appears.

2. In the Name box, type Site

3. In the Source database box, click GO_Vendors.

4. In the Source table box, click Browse for table [ ], and then double-click the RETAILER_SITE table.

5. Move the RETAILER_SITE_CODE, RETAILER_CODE, and CITY table columns from the
   Available attributes pane to the Chosen attributes pane.

6. For the RETAILER_SITE_CODE attribute, select the Id check box.
   RETAILER_SITE_CODE is now the unique identifier for the level.

7. For the CITY attribute, select the Caption check box.

8. For the RETAILER_CODE attribute, select the Parent check box.

9. Click OK.
   The Retailer hierarchy should now have these levels:
   • ALLRetailer
   • RetailerType
   • Retailer
   • Site

10. Click Next.

Steps to Save Your Work
1. Click Finish to close the Hierarchy wizard.

2. From the File menu, click Save Catalog [ ].
   IBM® Cognos® Data Manager adds the Retailer hierarchy to the Retailer reference dimension.
   It also automatically creates a template in the Templates folder for each hierarchy level.
View the Retailer Hierarchy Visualization

Here, you view the Retailer hierarchy you have created.

**Step**
- In the Tree pane, click the Retailer hierarchy. The Visualization pane should look like this:

- Explore the Structure of the Retailer Hierarchy

You can use the Reference Explorer to view the structure of a hierarchy and the members it contains at each level. You can view an entire hierarchy or restrict your view to specified levels. This means that you can test the hierarchy as you construct each level.

**Steps**
1. Expand the Dimensions folder, and then the Retailer reference dimension.
2. Right-click the Retailer hierarchy, and then click Explore.
3. In the Reference Explorer dialog box, click OK to confirm that you want view the entire Retailer hierarchy.

   A warning appears about non-unique IDs in the hierarchy. For further information on non-unique IDs, see "Set Error Handling" (p. 31).
4. Click OK to accept the warning and open the hierarchy in the Reference Explorer.
5. In the Elements pane, expand the hierarchy to examine its structure.
The Attributes pane shows information about the currently selected member of the hierarchy.

**Set Error Handling**

You can specify the action that IBM® Cognos® Data Manager should take when it encounters particular situations. The non-unique IDs that you encountered whilst viewing the Retailer hierarchy are an example of one such situation.

For a full list of these situations, and the actions that you can specify, see the IBM® Cognos® Data Manager User Guide.

Non-unique IDs mean that more than one member of the hierarchy has the same ID. You can see this in the previous illustration where the retailer Donovan’s Sports and retailer site New York both have an ID of 106. Non-unique IDs should cause concern only where a build acquires transaction data at more than one level of the hierarchy.

By default, Data Manager warns you when it detects non-unique IDs. You can change the default action from warn to accept.

**Steps to Accept Non-Unique IDs**

1. Right-click the Retailer hierarchy, and then click Properties.
2. Click the Features tab.
3. In the box that lists actions for non-unique IDs, click Accept.
4. Click OK.

**The Time Reference Dimension**

For the Time reference dimension, you set up a hierarchy that represents the fiscal periods.
Create the Dates Hierarchy

You create the Dates hierarchy from the Fiscal table in the DS_Sources database. In this table, every data row identifies the day, week, month, quarter, and year to which that row relates. This table represents a star schema.

<table>
<thead>
<tr>
<th>Yearid</th>
<th>Quarterid</th>
<th>Monthid</th>
<th>Weekid</th>
<th>Dayid</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005</td>
<td>2005Q1</td>
<td>200501</td>
<td>2005W1</td>
<td>2005-01-02</td>
</tr>
</tbody>
</table>

Steps to Select the Hierarchy Structure
First, you specify the type of hierarchy you want to create.

1. From the Tools menu, click Hierarchy Wizard.
2. Click Create the hierarchy from the columns of one table (Star Schema), and then click Next.

Steps to Define the Hierarchy
On this page, you enter a name for the hierarchy and specify the reference dimension to which the hierarchy belongs.

1. In the Enter the name of the hierarchy box, type Dates
2. In the Select the reference dimension to use for this hierarchy box, click Time, and then click Next.

Steps to Specify the Hierarchy Source
On this page, you specify the source of the data to include in the hierarchy.

1. In the left pane, expand the DS_Sources database.
2. Click the Fiscal table, and then click Next.

Step to Configure the Top Level
The Great Outdoors Company does not aggregate data over more than one year, so a static member at the top level is not required.

- Clear the Include a top level with an All member check box, and then click Next.
Steps to Define the Year Level
On this page, you insert the top level of the hierarchy, the Year level. When defining a level you must specify the attributes (the columns to include from the data source).

1. Click Add.
   The Level Details window appears.

2. In the Name box, type Year

3. In the Source column for ID box, click Browse for column, click the YearId attribute, and then click OK.

4. In the Source column for caption box, click Browse for column, click the YearCaption attribute, and then click OK.

5. Click OK.

Steps to Define the Lower Levels
Here, you insert the lower levels of the hierarchy: Quarter, Month, Week, and Day.

1. Follow the steps from "Steps to Define the Year Level" (p. 33) to create each of the following hierarchy levels.

<table>
<thead>
<tr>
<th>Level name</th>
<th>Source column for ID</th>
<th>Source column for caption</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quarter</td>
<td>QuarterId</td>
<td>QuarterCaption</td>
</tr>
<tr>
<td>Month</td>
<td>MonthId</td>
<td>MonthCaption</td>
</tr>
<tr>
<td>Week</td>
<td>WeekId</td>
<td>WeekCaption</td>
</tr>
<tr>
<td>Day</td>
<td>DayId</td>
<td>DayCaption</td>
</tr>
</tbody>
</table>

When you have finished, the Dates hierarchy should contain the following levels:

- Year
- Quarter
- Month
- Week
- Day

2. Click Next.

Steps to Save Your Work
1. Click Finish to close the Hierarchy wizard.
2. From the File menu, click **Save Catalog**.

IBM® Cognos® Data Manager adds the Dates hierarchy to the Time reference dimension. It also automatically creates a single template in the Templates folder for the hierarchy.

**View the Dates Hierarchy Visualization**

**Step**
- In the Tree pane, click the **Dates** hierarchy. The Visualization pane should look like this:

![Dates Hierarchy Visualization](image)

**The Product and SalesStaff Reference Dimensions**

The Product and Staff hierarchies, that you need to set up for the Product and SalesStaff reference dimensions, are of the same type as the Retailer hierarchy. They are both hierarchies from multiple tables that represent the snowflake schema.

The DS_Lessons catalog already contains the Product and Staff hierarchies and corresponding templates, so you do not need to set them up yourself. It also contains the Retailer and Time hierarchies that you created here. You will use the DS_Lessons catalog in the following chapters.
Chapter 6: Creating a Data Mart

In the previous chapters, you set up all the components required in a catalog to create builds. In this chapter, you learn how to create a fact build that delivers a complete data mart. You base the fact build on the IBM® Cognos® BI Mart (Star) type, one of the types that the Fact Build wizard offers.

For more information about the build types that the Fact Build wizard offers, see the IBM® Cognos® Data Manager User Guide.

You learn how to

- Create a fact build.
- Deliver data to a data mart.
- View data in the data mart.
- Use the execution log.

Fact Builds

An IBM® Cognos® Data Manager fact build specifies how to extract, transform, and deliver data. Data may come from multiple data sources, each of which has its own naming and storage conventions. For example, the data sources may store dates as Julian dates, as strings in American or European format, or in a date format of the source DBMS. Data Manager can transform such data so that it exists in standard format in the target data marts.

A fact build can

- acquire data from one or more data sources
- merge and clean the source data
- aggregate the data with reference to the dimensional framework
- deliver the fact and dimension data to one or more data marts

Transformation Model

The transformation model is central to each IBM® Cognos® Data Manager fact build and can contain these types of element:

- **Attribute**

  An attribute element holds additional information that is not a dimension or measure, but that may be of interest. Attributes differ from measures in that they cannot be aggregated.

- **Derivation**
A derivation element is a value that Data Manager calculates rather than obtains directly from the source data, using an expression that you define.

- **Dimension**
  A dimension element contains data that is used to give context to a measure element. For example, a product number dimension element gives context to a quantity measure element.
  
  **Note:** Dimension elements provide the link to the dimensional framework.

- **Derived Dimension**
  A derived dimension element allows dimensional lookups to be performed against reference structures. The lookup value can be obtained from a previous lookup or calculated using a derivation.

- **Measure**
  A measure element is a value that holds a piece of information for analysis, such as units, revenue, or cost.

When Data Manager performs aggregation, it consolidates measures and derivations along dimension elements.

## Create a Fact Build

A fact build can deliver fact data and dimension data together. That is, it can deliver a complete data mart.

Here, you use the Fact Build wizard to create fact builds.

**Tip:** You can also manually create or refine fact builds. For information about maintaining fact builds manually, see the IBM® Cognos® Data Manager *User Guide*.

## Define the Purpose of the Fact Build

First, you enter the basic fact build details, specify the type of fact build to create, and define the database to which to connect.

**Steps**
1. Open the DS_Lessons catalog.
2. From the Tools menu, click Fact Build Wizard.
3. In the Enter the name of the build box, type BIMart.
4. In the Select the type of fact build to create box, click IBM Cognos BI Mart (Star).
5. In the Select the connection into which the build is to deliver data box, click DS_Marts.
6. Select the Perform a full refresh on the target data check box, and then click Next.
Create the DataStream

On this page, you specify a data source for the fact build, the data columns from which to read, and create the transformation model elements.

Steps
1. Click Data Source, Add.

   The Data Source wizard appears.

2. In the Select the connection from which the data source is to read box, click GO_Sales, and then click Next.

3. In the left pane, select the check boxes for the tables ORDERDETAILS and ORDER_HEADER.

4. Clear the check boxes for all data columns except those shown selected here:

   ```
   SELECT a.`PRODUCT_NUMBER`,
   a.`QUANTITY`,
   a.`UNIT_COST`,
   a.`UNIT_PRICE`,
   b.`RETAILER_SITE_CODE`,
   b.`SALES_STAFF_CODE`,
   FROM `ORDER_DETAILS` a, `ORDER_HEADER` b
   WHERE a.`ORDER_NUMBER` = b.`ORDER_NUMBER`
   ```

5. Click Finish to close the Data Source wizard, and then click Next.

   For each column that the SQL statement returns, the wizard creates a corresponding transformation model element.

Assign the Element Types

On this page, you can change the element types assigned by IBM® Cognos® Data Manager, and the order of elements in the transformation model, if necessary.

Steps
1. Change the order of the elements, using Move Up and Move Down, so that they are as follows.
Define the Dimensions

On this page, you associate each dimension element with a reference item. When you execute the fact build, each row of fact data is related to the correct member at each level of the reference structure. For example, for the PRODUCT_NUMBER element, it relates each fact data row to the correct product name, product type, and product line.

Associate the PRODUCT_NUMBER dimension element with the Product hierarchy as follows:

Steps
1. For the PRODUCT_NUMBER element, click the Use reference column, and then click the browse button that appears.
2. Expand the Product reference dimension, and click the Product hierarchy.
3. Repeat steps 1 to 2 to associate each of the following dimension elements with a hierarchy.

<table>
<thead>
<tr>
<th>Dimension element</th>
<th>Reference dimension</th>
<th>Hierarchy</th>
</tr>
</thead>
<tbody>
<tr>
<td>RETAILER_SITE_CODE</td>
<td>Retailer</td>
<td>Retailer</td>
</tr>
<tr>
<td>SALES_STAFF_CODE</td>
<td>SalesStaff</td>
<td>Staff</td>
</tr>
</tbody>
</table>

4. Click Next.

Define the Fact Delivery

On these pages, you specify where to deliver the fact data, and you define the naming conventions.

Steps
1. Click Next to accept the default fact delivery.
2. Click Next to accept the default naming conventions.
   - The Fact Build wizard summarizes of the fact delivery that it will create.
3. Click Next.
Define the Dimension Delivery

On these pages, you specify the schema that you want to use for the dimension deliveries, and you define the naming conventions for tables and columns.

Steps
1. Click Next to accept the default schema type.
2. Click Next to accept the default naming conventions.
   The Fact Build wizard summarizes of the dimension deliveries that it will create.
3. Click Next.

Save Your Work

The final page of the Fact Build wizard shows an overview of the fact build.

Steps
1. Click Finish to close the Fact Build wizard.
2. From the File menu, click Save Catalog.

Deliver Data to the Data Mart

IBM® Cognos® Data Manager can deliver these types of data:

- Fact data
  This contains values that represent facts, such as the number of units sold of a product at a particular store on a particular day.

- Dimension data
  This defines the core business components, such as the range of products that the company produces.
Data Manager can deliver all the data to one or more data marts in the same or different physical locations, or partition the data horizontally and vertically by criteria that you choose.

**Steps**
1. Right-click the BIMart fact build, and then click **Execute**.
   The **Execute Build** dialog box appears.
2. Click **Execute locally**.
3. In the **Execution mode** box, click **Normal**, and then click **OK**.
   Data Manager executes the fact build in a command window, writing progress messages to the screen and the execution log.
   **Note:** Depending on the configuration of your computer, execution may take several minutes.
4. When the fact build execution is complete, press Enter to close the command window.

**View Data in the Data Mart**

The BIMart fact build that you executed delivers the following data tables.

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_BIMart_Product</td>
<td>Product dimension</td>
</tr>
<tr>
<td>D_BIMart_Staff</td>
<td>Staff dimension</td>
</tr>
<tr>
<td>D_BIMart_Retailer</td>
<td>Retailer dimension</td>
</tr>
<tr>
<td>F_BIMart</td>
<td>Facts</td>
</tr>
</tbody>
</table>

You can view the delivered fact and dimension tables using SQLTerm, which is described here, or other open relational database access tools.

You use SQLTerm to analyze the data that you delivered to the data mart. SQLTerm provides a simple, integrated way to edit and run SQL queries.

**Steps**
1. From the **Tools** menu, click **SQLTerm**.
2. In the **Database** box, click **DS_Marts**.
3. In the **Database objects** pane, expand the **DS_Marts** database.
4. Right-click the **D_BIMart_Staff** table, and then click **Add table select statement**.
   SQLTerm creates an SQL SELECT statement for the table.
5. On the SQLTERM toolbar, click **Return all rows**.
Use the Execution Log

The execution log provides valuable information when debugging a fact build or a dimension build. It records the messages that IBM® Cognos® Data Manager writes to the screen as it executes the build. You can specify the level of detail that you want to include in the log file and the frequency of input and output messages:

- **Progress** records the overall progress of the build execution.
- **Detail** includes more detailed progress messages.
- **Internal** records all internal Data Manager activity messages, including resource usage.
- **SQL** records all SQL statements that Data Manager uses to acquire and deliver data.
- **ExecutedSQL** includes executed SQL for SELECT statements.
- **Variable** records all variables used in the build.
- **User** records messages that you define, for example, with the LogMsg function.

By default, the Fact Build wizard configures fact builds to log only progress messages.

Examine the Execution Log

Here, you examine the execution log for the BIMart fact build.

Steps

1. From the **Tools** menu, click **Browse Log Files**.
   
   The log file directory appears.
Chapter 6: Creating a Data Mart

The name of each build execution log file is Build_<build_name>_<number>.log, where <build_name> is the name of the build and <number> is a four digit number. The latest execution log file for a build has the highest number.

2. Double-click the latest log file for the BIMart fact build.

The log file appears.

Tip: If your computer cannot automatically open the log file, the Open With dialog box appears. Choose a standard text editor from the list of programs.

Add More Information to the Execution Log

Here, you override the default log settings for the BIMart fact build. For further details on different log settings, see "Use the Execution Log" (p. 41).

Steps
1. Right-click the BIMart fact build, and then click Execute.

The Execute Build dialog box appears.

2. Click Execute locally.

3. Select the Override build settings check box.

4. Select the check boxes for each message type.

The Execute Build dialog box should now look like this.

5. Click OK.
IBM® Cognos® Data Manager executes the fact build in a command window. Note the increased detail that Data Manager writes to the screen.

6. When fact build execution is complete, press Enter to close the command window.

7. View the new execution log.

You have now finished creating a data mart. In the following chapter, you will add more advanced features to a fact build.
Chapter 7: Creating Derivations

In this chapter, you learn how to add derivations to a fact build.

You learn how to

- Create a transformation model derivation.
- Create DataStream derivations.
- Deliver additional data to a data mart.

Derivations

A derivation is a value that IBM® Cognos® Data Manager calculates rather than obtains directly from the source data, using an expression that you define. You can create derivations in

- DataStreams
- data sources
- the transformation model

You use the Expression box, shown below, to enter a derivation expression. You can use the tree structure in the left pane to select operators, functions, and so on, from the appropriate folders.

![Expression box](image)

You can select an individual item from a folder by

- dragging it into the right pane
- double-clicking it
Create a Transformation Model Derivation

Here, you create a derivation element, named GROSS_Margin, in the transformation model. This derivation element represents the gross profit margin expressed as a percentage of each sale. The derivation uses a user-defined function named Margin.

Steps
1. Open the DS_Lessons catalog, and then expand the Additional fact build.
2. Right-click Transformation Model, and then click Insert Derivation.
   The Derivation Properties window appears.
3. In the Name box, type GROSS_Margin
4. Click the Calculation tab.
5. Add the Margin function to the right pane by double-clicking the Functions folder, then the User Defined folder, and then double-clicking the Margin function.
   The expression in the right pane should now read Margin( Price, Cost )
6. Replace the numeric arguments in the Margin function with your required parameters.
   First, select Price and replace it by typing UNIT_SALE_PRICE. Then select Cost and replace it by typing UNIT_COST
   Tip: You can select the UNIT_SALE_PRICE and UNIT_COST parameters from the Elements folder.
   The expression should now read Margin( UNIT_SALE_PRICE, UNIT_COST )
7. Click OK.
   IBM® Cognos® Data Manager provides the option to add the new element to existing delivery modules.
8. Click Yes.
9. From the File menu, click Save Catalog.

Create DataStream Derivations

The following sections describe how you create the following DataStream derivations in the Additional fact build:
- total value of each sale
- gross profit of each sale
Create the Total Value of Each Sale

Here, you set up a DataStream derivation named SALES_TOTAL.

**Steps**

1. In the Additional fact build, right-click DataStream, and then click Properties. The DataStream Properties window appears.
2. Click the Derivations tab, and then click Add. The Derivation Properties window appears.
3. In the Name box, type SALES_TOTAL
4. Click the Calculation tab.
5. Type the expression QUANTITY * UNIT_SALE_PRICE
6. Click Test and test the expression.
   For information, see the IBM® Cognos® Data Manager User Guide.
7. Click OK in the Derivation Properties window.
   Leave the DataStream Properties window open so that you can create another derivation.

Create the Gross Profit of Each Sale

Here, you set up a DataStream derivation named GROSS_PROFIT.

**Steps**

1. In the DataStream Properties window, click the Derivations tab, and then click Add. The Derivation Properties window appears.
2. In the Name box, type GROSS_PROFIT
3. Click the Calculation tab.
4. Type the expression QUANTITY * (UNIT_SALE_PRICE - UNIT_COST)
5. Click Test and test the expression.
   For information, see the IBM® Cognos® Data Manager User Guide.
6. Click OK in the Derivation Properties window, and then in the DataStream Properties window.
7. Click the Mapping tab in the Visualization pane. The mapping should look like this:
Notice that the DataStream derivations you created are shown in an alternative color to indicate that they are not yet mapped to the transformation model.

**Execute the DataStream**

Here, you execute the DataStream for the Additional fact build to check the source data, including the two derivations you have created.

**Steps**

1. In the Additional fact build, click the DataStream.

2. From the Actions menu, click **Execute**.

   The **Execute DataStream** window appears as follows:

   ![Execute DataStream window](image)

3. Click **Close**.

**Map the DataStream to the Transformation Model**

The SALES TOTAL and GROSS PROFIT DataStream derivations must be mapped to the transformation model for the Additional fact build.

You must create two transformation model elements to map these derivations. IBM® Cognos® Data Manager prompts you to create these elements during the mapping process.

**Steps**

1. In the Additional fact build, right-click the Transformation Model, then click **Mapping**.

   The Transformation Model Mapping window appears.

2. Click **Auto Map**.

   Data Manager prompts you to choose the type of element to create for the required transformation model elements.

3. Click **Create New Elements as Measure**.
Data Manager maps the selected DataStream items to the new transformation model elements, as shown here.

4. Click OK.

Data Manager prompts you to add the new elements to fact deliveries.

5. Ensure the check boxes are selected for each listed element, and then click OK.

6. Click the Mapping tab in the Visualization pane. The mapping should look like this:

You can clearly see that the DataStream derivations are now mapped to the transformation model.

7. From the File menu, click Save Catalog.

**Deliver Additional Data to the Data Mart**

Because the Additional fact build contains additional columns for the derivation that you created, the fact deliveries contain additional information to describe those columns. As a result, the deliv-
eries cannot map successfully to the tables in the data mart. IBM® Cognos® Data Manager detects such inconsistencies and offers to drop the inconsistent tables.

**Steps**
1. Right-click the Additional fact build, and then click Execute.
   Data Manager reports an inconsistency between the fact delivery and the target fact table.
2. Click Drop Tables, and then click Yes.
   The Execute Build dialog box appears.
3. Click Execute locally.
4. In the Execution mode box, click Normal, and then click OK.
   Data Manager executes the fact build in a command window.
   **Note:** Depending on the configuration of your computer, execution may take several minutes.
5. When fact build execution is complete, press Enter to close the command window.
   You have now finished adding derivations to the Additional fact build and delivering the data mart.
Chapter 8: Maintaining a Data Mart

In this chapter, you learn how to
❑ Add a data source to an existing fact build.
❑ Handle rejected data.
❑ Configure a fact build to consolidate data.
❑ Handle source data at different levels of granularity.

Add a Data Source to a Fact Build

Here, you add a data source to the Additional fact build. The additional data contains some data that IBM® Cognos® Data Manager rejects. You learn how to handle the rejected data in "Handle Rejected Data" (p. 53).

Specify the Data Source

Here you open the DS_Sources database, select the AdditionalSales1 table and list the column names from within it.

Steps
1. Open the DS_Lessons catalog.
2. Expand the Additional fact build.
3. Right-click the DataStream, and then click Insert Data Source.
4. Click the Query tab.
5. In the Database box, click DS_Sources.
6. In the Database Objects pane, expand the DS_Sources database.
7. Within the DS_Sources database, right-click the AdditionalSales1 table, and then click Add table select statement.
   IBM® Cognos® Data Manager adds an SQL statement for the table to the Query pane.
8. Click Return All Rows to test the SQL statement.
9. Click the Result Columns tab.
10. Click Prepare, and then click Refresh.
    A list of column names from the data source appears.
11. Click OK to close the Data Source Properties window.
Map the Data Source to the DataStream

Here, you map data source columns in DataSource2 to the DataStream for the Additional fact build.

Note: The DataStream items required for mapping have already been set up for DataSource1.

Steps

1. In the Additional fact build right-click the DataStream, and then click Properties.
2. Click the DataStream Items tab.
   Within DataSource2, double-click each of the following data source columns in the left pane to map them to the corresponding DataStream items:
   - ORDER_DATE
   - PRODUCT_NUMBER
   - QUANTITY
   - UNIT_COST
   - UNIT_PRICE
   - UNIT_SALE_PRICE
3. When the mapping table looks like this, click OK.
4. From the File menu, click Save Catalog.
Deliver Data
Here you select the Additional fact build and execute the build to deliver the data.

Steps
1. Right-click the Additional fact build, and then click Execute.
   The Execute Build dialog box appears.
2. In the Execution mode box, click Normal, then click OK.
   IBM® Cognos® Data Manager executes the fact build in a command window.
   Note: Depending on the configuration of your computer, execution may take several minutes.
3. When fact build execution is complete, press Enter to close the command window.

Handle Rejected Data
When you executed the Additional fact build, IBM® Cognos® Data Manager rejected data that violated the referential integrity of one or more of the dimension elements. The rejected data is saved in a separate file.

Set Up the Location of Rejected Data
Here you set up the location of the text files in which reject data is stored.

Steps
1. Right-click the Additional fact build, and then click Properties.
2. Click the Input tab.
3. In the Reject record handling box, ensure Write reject records to file is selected.
   In the File name box, browse to the following directory:
   c10_location\datamanager\data
   This is the default directory in which IBM® Cognos® Data Manager stores reject files.
   Note: The default location for c10_location is C:\Program Files\ibm\cognos\c10.
4. Ensure the value in the File name box is [SDS_BUILD_NAME].rej, and then click Save.
5. Click OK.

Check for Rejected Data
Here, you check for rejected data using the execution log.

Steps
1. Open the latest execution log file for the Additional fact build.
   For more information, see "Use the Execution Log" (p. 41).
2. Search the file for a line similar to

   [PROGRESS - hh:mm:ss ] Acquisition: 43157 accepted, 11 rejected

   This line indicates that IBM® Cognos® Data Manager rejected some of the source data.

---

**View Rejected Data**

Here, you view the data rejected from the Additional fact build.

**Steps**

1. From the **Tools** menu, click **Browse Data Files**.

2. Double-click the file named `additional.rej`

   The reject file appears in your default text editor.

3. View the reject file, noting that

   - the first line contains column headings
   - each subsequent line contains the details of one rejected record
   - the first column contains the reason for rejecting the record

   **Note:** In this example, some records are rejected because there are product numbers and order dates for which no members of the Product and Dates hierarchies correspond.

   **Tip:** Because the reject file is a tab-delimited text file, you can include it in a SQLTXT database for reprocessing. For more information, see the IBM® Cognos® Data Manager *User Guide*.

---

**Consolidate the Data**

You can modify the Additional fact build to consolidate the data along the Time business dimension.

**Change the ORDER_DATE Dimension Element**

Here, you configure the ORDER_DATE dimension element to acquire data at the Day level, to deliver fact data at the Month level, and to deliver dimension data for the Month, Quarter, and Year levels. Because the fact build delivers summary data for the Time business dimension, you must configure the ORDER_DATE element to perform aggregation.

**Steps**

1. Expand the **Additional** fact build, and then its transformation model.

2. Right-click the **ORDER_DATE** dimension element, then click **Properties**.

3. Click the **Reference** tab.

4. Select the **Aggregate** check box.

5. In the **Output** column, select the check box for the **Month** level and clear the others.

6. In the **Dimension** column, clear the check boxes for the **Day** and **Week** levels.
7. When the level selections look like this, click **OK**.

![Dimension Properties window](image)

### Change the Dimension Table Template

The dimension table that the Additional fact build created has columns for the business key and business name for each of the Day, Week, Month, Quarter, and Year levels of the Dates hierarchy. Because the changed fact build no longer delivers dimension data for the Day and Week levels, you must remove these levels from the delivered dimension table.

The template associated with the dimension table controls the configuration of this table. To remove the columns that are not required, you must change the associated template.

**Steps**

1. Expand the Additional fact build, Delivery Modules, Dimension Delivery, ORDER_DATE, and Dates (Relational Table).

2. Right-click `D_Additional_Dates`, and then click **Properties**.

   The Dimension Table Properties window appears.

3. Click the **Columns** tab.

4. In the **Columns** column, clear the checkboxes for the Dayid, Daycaption, Weekid, and Week-caption columns.

5. Click **Edit**.

   The Template Properties window appears.

6. Click the **Attributes** tab.

7. Delete the following attributes:
   - Dayid
   - Daycaption
8. For the Monthid attribute, in the Value box to the right of the Primary Key property, click True. This identifies the new primary key of the dimension table.

9. Click OK in the Template Properties window, and then in the Dimension Table Properties window.

10. From the File menu, click Save Catalog.

**Drop the Target Tables**

In the Dates hierarchy, the identifiers for days are date values. Because of this, the Day column of the existing fact table is of type DATETIME. The identifiers for months are string values and are of the wrong type for delivery to the existing fact table. The simplest way to avoid the data type conflict is to use SQLTerm to drop the target fact table. IBM® Cognos® Data Manager then recreates the fact table when you execute the fact build.

The existing dates dimension table has columns for days and weeks that no longer map to template attributes. Therefore, you should either remove those columns from the target dimension table, or drop the table itself.

**Steps**

1. From the Tools menu, click SQLTerm.

2. In the list of connections, click DS_Marts.

3. In the Database Objects pane, expand the database DS_Marts.

4. Right-click the table F_Additional, and then click Drop table. SQLTerm adds the following statement to the Source SQL tab:
   DROP TABLE 'F_Additional';

5. Repeat step 4 to drop the D_Additional_Dates table.

6. Click Execute Statement(s).

7. Close SQLTerm.

**Deliver Data**

Here you select the Additional fact build and deliver the data.

**Steps**

1. Right-click the Additional fact build, and then click Execute.

   The Execute Build dialog box appears.
2. Ensure that **Execute locally** is selected, and then click **OK**.
   IBM® Cognos® Data Manager executes the fact build in a command window.

3. When fact build execution is complete, press Enter to close the command window.

### Analyze the Results

Use SQLTerm to analyze the delivered data. Note that the Dates dimension and the fact data now have a granularity of Month.

### Multiple Granularity

With IBM® Cognos® Data Manager, you can merge and reconcile data with different levels of granularity. Here, you add a third data source to the Additional fact build. The new data has a granularity of month. The other data sources have transactions at a daily granularity. Therefore, the modified fact build acquires data at both the Day and Month levels of the Time dimension.

The new data comes from the ProductForecast table of the GO_Sales database. The existing data sources of the fact build have a string value for the month identifier of the form YYYYMM, such as, 200102. The new data has numeric columns for year and month. You must create the SQL and a new data source derivation for the new data to return month identifiers that are compatible with the existing data.

The new data source also contributes forecast information, which you must add to the transformation model.

### Add a Data Source

Here, you add a new data source to the Additional fact build.

**Steps**

1. Expand the **Additional** fact build.

2. Right-click the **DataStream**, and then click **Insert Data Source**.

3. Click the **Query** tab.

4. In the **Database** box, click **GO_Sales**.

5. In the **Database Objects** pane, expand the **GO_Sales** database.

6. Within the **GO_Sales** database, right-click the **PRODUCT_FORECAST** table, and then click **Add Table Select Statement**.
   IBM® Cognos® Data Manager adds an SQL statement for the table to the **Query** pane.

7. Click **Return 1 Row** to test the SQL statement.

8. Click the **Result Columns** tab.

9. Click **Prepare**, and then click **Refresh**.
A list of column names from the data source appears.

10. Click the **Derivations** tab, and then click **Add**.

11. In the **Name** box, type **ORDER_DATE**

12. Click the **Calculation** tab.

13. Enter the following derivation expression:

   \[ \text{Concat(YEAR,LPad(ToChar(MONTH),2,'0'))} \]

14. Click **OK** to close the **Derivation Properties** window.

15. Click **OK** to close the **Data Source Properties** window.

**Map the Data Source to the DataStream**

The new data contains sales forecast information for which a new DataStream item is required. IBM® Cognos® Data Manager automatically creates this DataStream item during the mapping process.

**Steps**

1. In the **Additional** fact build, right-click the **DataStream**, and then click **Properties**.

2. Click the **DataStream Items** tab.

3. Within DataSource3, double-click each of the following data source columns in the left pane:
   - **PRODUCT_NUMBER**
   - **EXPECTED_VOLUME**
   - **ORDER_DATE**

   Data Manager creates the **EXPECTED_VOLUME** DataStream item and maps the selected columns to the appropriate DataStream item.

4. When the mapping table looks like this, click **OK**.
5. From the File menu, click Save Catalog.

**Execute the DataStream**

Here, you execute the DataStream for the Additional fact build to check the source data, including the derivation you have created.

**Steps**

1. In the Additional fact build, click the DataStream.

2. From the Actions menu, click Execute.

   The Execute DataStream window appears as follows:

3. Click Close.
Map the DataStream to the Transformation Model

The new data contains sales forecast information for which a new transformation model element is required. IBM® Cognos® Data Manager prompts you to create this element item during the mapping process.

**Steps**

1. In the Additional fact build, right-click the Transformation Model, and then click Mapping.
2. Click Auto Map.
3. Data Manager prompts you to choose the type of element to create for the required transformation model element.
4. Click Create New Elements as Measure.
   
   Data Manager maps the EXPECTED_VOLUME DataStream item to the new transformation model element.
   
   The new data has a granularity of month for the Time dimension. As a result, you must modify the mapping for the ORDER_DATE DataStream item to also accept data at the month level.
5. In the Transformation Model pane, expand the ORDER_DATE dimension, Dates hierarchy, and Month level. Select the Monthid attribute ID.
6. Hold down the Ctrl key, and drag the Monthid attribute ID from the Transformation Model pane to the ORDER_DATE DataStream item in the Maps To column.

   The mapping should look like this.

   ![Diagram](image.png)

7. Click OK.

   Data Manager prompts you to add the new element to fact deliveries.
8. Select both check boxes, then click OK.
9. From the File menu, click Save Catalog.

**Deliver Data**

Here you select the Additional fact build and deliver the data.

Because the Additional fact build contains an additional column for DataSource3, the fact delivery contains additional information to describe this column. As a result, the delivery cannot map successfully to the tables in the data mart. IBM® Cognos® Data Manager detects such inconsistencies and offers to drop the inconsistent table.

**Steps**

1. Right-click the Additional fact build, and then click Execute.
   
   Data Manager reports an inconsistency between the fact delivery and the target fact table.

2. Click Drop Tables, and then click Yes.
   
   The Execute Build dialog box appears.

3. Click OK.
   
   Data Manager executes the fact build in a command window.

4. When fact build execution is complete, press Enter to close the command window.

**Analyze the Results**

Use SQLTerm to analyze the delivered data. Note that the Dates dimension and the fact data now have a granularity of Month.
Chapter 9: Creating a Dimension Build

In this chapter, you learn how to create a dimension, using a dimension build, that can be used in a coordinated data mart.

**Dimension Builds**

A dimension build can deliver dimension data for many fact builds. Fact builds can also be used to deliver dimension data, but this may mean several fact builds delivering the same data, resulting in a duplication of effort.

Dimension builds deliver only dimension data. The main use for dimension builds is to deliver conformed dimensions that form the dimensional reference for coordinated data marts.

Each dimension build acquires data from a reference structure that you specify and delivers this, as dimension data, to the target data mart. You can choose to deliver the dimension data to a single table (star schema), to one table for each level of the hierarchy (snowflake schema), partitioned by custom criteria, or as a parent-child schema.

Dimension builds are essential for tracking attribute changes and surrogate keys. To detect changes to attributes, IBM® Cognos® Data Manager must have access to the current and historical states of each member, for every dimension that is tracked. It is unlikely that a transaction system contains sufficient historical information. Transaction systems contain no information about the surrogate keys that reside in the data mart. Data Manager can obtain this information from the dimension tables that reside in the data mart.

The requirement to populate the dimension tables before processing the fact data implies the following two stages of execution.

The dimensional framework contains two reference structures for each dimension:
• Primary reference structure

This is constructed from structure data. It does not contain the attributes to support attribute change tracking or surrogate keys. A dimension build delivers the primary reference structure to the data mart. The delivered dimension data defines the semantics of the source fact data and contains information to track attribute changes and surrogate keys.

• Secondary reference structure

This is constructed from the dimension table in the data mart. It uses the template used to deliver the dimension table, allowing Data Manager to access the information to track attribute changes and surrogate keys.

Conformed Dimensions

A conformed dimension is a single dimension that is delivered once, using a dimension build, and can be shared across multiple data marts.

In the following example, the Time, Product, and Staff dimensions are set up as conformed dimensions and used in the Sales and Inventory data marts.

Note: It is recommended that you use star schemas (as illustrated above) when setting up data marts that use conformed dimensions.

Some advantages of using conformed dimensions are

• Independent data marts become part of a fully integrated data warehouse.

• The development time for a data warehouse is reduced because each dimension is analyzed, designed and created only once.

• Conformed dimensions deliver a consistent view of a business, allowing you to drill from one area of the business to another.
Granularity in Conformed Dimensions

Fact tables normally contain data at the lowest level of granularity. For example, product data would normally contain details of individual products.

A conformed dimension should also be defined at the lowest level of granularity. For example, the granularity of a product dimension would be an individual product. As a result, each record in the reference table corresponds to a single record in the fact table.

Designing conformed dimensions in this way provides maximum flexibility for businesses because it allows questions to be answered at any level of granularity.

Create a Conformed Dimension

ConformedMart is a fact build set up in the DS_Lessons catalog. It uses reference data from the four dimensions that you set up in Chapter 5: Product, Dates, Staff, and Retailer.

Because the Great Outdoors Company intends to use the reference data from these dimensions in more than one data mart, they should be set up as conformed dimensions. Three of the conformed dimensions, Dates, Staff, and Product, have been set up for you. Here, you create the Retailer conformed dimension.

The main tasks required to define and use a conformed dimension are

- Create a primary reference structure.
- Deliver the primary reference structure using a dimension build.
- Create a secondary reference structure using the dimension data delivered for the primary reference structure.
- Associate the secondary reference structure with a fact build.

Create a Primary Reference Structure

The Retailer hierarchy is the primary reference structure on which the Retailer conformed dimension will be based.

You created the Retailer hierarchy in Chapter 5.
Deliver the Primary Reference Structure

The next task is to create the Retailer dimension build and use this to deliver the Retailer hierarchy.

Create a Dimension Build

Here, you use the Dimension Build wizard to create the Retailer dimension build.

Tip: You can also manually create or refine dimension builds. For information about maintaining dimension builds manually, see the IBM® Cognos® Data Manager User Guide.

Steps

1. Open the DS_Lessons catalog.

2. From the Tools menu, click Dimension Build Wizard.

   The Dimension Build wizard appears.

3. In the Name box, type Retailer and then click Next.

   On this page, you specify the schema that you want to use for the dimension delivery, the dimension to deliver, and the database to which you want to deliver the data.

4. In the Schema Type box, click Star Schema.

5. In the Dimension to be delivered box, click Retailer.

6. In the Reference item to be delivered box, click Retailer (H).

7. In the Deliver into connection box, click DS_Marts, and then click Next.

8. Click Next to accept the default naming conventions.

9. Click Next to accept the default build features.

10. Click Next to accept the default build properties.

   When using conformed dimensions in a data mart, it is a good idea to use surrogate keys rather than business keys to reference the dimensions. On this page, you add support for surrogate keys.

11. Select the Add surrogate keys to the dimension tables check box, and then click Next.

    For information about surrogate keys, see the IBM® Cognos® Data Manager User Guide.

    The final page of the Dimension Build wizard shows an overview of the dimension build.

12. Click Finish to close the Dimension Build wizard.

13. From the File menu, click Save Catalog.
View the Dimension Build

Here you view the Retailer dimension build in the Visualization pane.

Step

- In the Tree pane, click the Retailer dimension build. The Visualization pane should look like this:

```
  Retailer  Retailer  D_Retailer  D_Retailer  DS_Marts
```

Note that IBM® Cognos® Data Manager has created a template, D_Retailer, which is used to deliver the primary reference structure. You will use this template when you create the secondary reference structure.

Deliver the Dimension Build

Here, you execute the Retailer dimension build to deliver the primary reference structure.

Steps

1. Right-click the Retailer dimension build, and then click Execute.
   The Execute dialog box appears.
2. Click Execute locally.
3. In the Execution mode box, click Normal.
4. Click OK.
   IBM® Cognos® Data Manager executes the dimension build in a command window.
5. When the dimension build execution is complete, press Enter to close the command window.

View the Retailer Dimension Data

The dimension build that you executed delivers the D_Retailer dimension table. Here, you use SQLTerm to analyze the delivered dimension data.

Steps

1. From the Tools menu, click SQLTerm.
2. In the Database box, click DS_Marts.
3. In the Database Objects pane, expand the DS_Marts database.
4. Right-click the D_Retailer table, and then click Add table select statement.
   SQLTerm creates an SQL SELECT statement for the table.
5. On the toolbar, click Return all rows.
   SQLTerm shows the results in the Test pane.
Notice that the dimension data includes the surrogate keys (skey column) as well as the business keys (RETAILER_SITE_CODE column). It is a good idea to use surrogate keys in data marts when referencing data from the conformed dimension.

Create a Secondary Reference Structure

The Retailer dimension build specified all levels of the hierarchy, and delivered all the relevant attributes to the dimension table. These attributes and levels may be required for reporting and analysis from the data mart. However, when creating a fact build, you do not always need all the attributes in the dimension table.

The ConformedMart fact build uses dimensional data to ensure that the dimensional elements in the fact data match existing dimension table members. For example, to check that a retailer site code exists in the Retailer dimension. To perform this data integrity checking, you only need the RETAILER_SITE_CODE business ID, and surrogate ID from the dimension table.

The next task is to create a secondary reference structure, RetailerLookup, within the Retailer reference dimension. This lookup only requires the business ID and surrogate ID attributes, and so will read data faster from the database because all other attributes that are not required are removed.

For more information about lookups, see the IBM® Cognos® Data Manager User Guide.

Create a Lookup

Here, you create a lookup, named RetailerLookup, within the Retailer reference dimension.

Steps

1. Right-click the Retailer reference dimension, and then click Insert Lookup.

   The Lookup Properties window appears.

2. In the Name box, type RetailerLookup

3. Click the Attributes tab.
4. In the Template box, click D_Retailer.
   Note: D_Retailer is the template that was created by the Retailer dimension build.

5. Move the skey and RETAILER_SITE_CODE attributes from the Available Attributes pane to the Chosen Attributes pane.
   This specifies the reference attributes required by the lookup for data integrity checking. Typically, the attributes required are the business ID and surrogate key of the dimension level you are checking.

6. In the Chosen Attributes pane, select the Id check box for the RETAILER_SITE_CODE attribute.

7. Click the Data Access tab.

8. Click Use Template for data access.

9. In the Connection box, click DS_Marts.
   DS_Marts is the target database into which the D_Retailer dimension table was delivered.

10. In the Table name box, click Browse.
    The Select Table dialog box appears.

11. Click the D_Retailer table, and then click OK.

12. Click OK to close the Lookup Properties window.

13. From the File menu, click Save Catalog.
    You have now finished creating the conformed dimension.

**Explore the RetailerLookup Data**

Here, you use the Reference Explorer to view the data in the lookup that you have created.

**Steps**

1. Expand the Dimensions folder, and then the Retailer reference dimension.

2. Right-click RetailerLookup, and then click Explore.

3. In the Reference Explorer dialog box, click OK to confirm that you want view the lookup named RetailerLookup.
   You can see from the data in the Elements pane that the reference data consists of retailer site codes only.
When you select a retailer site code number, the Attributes pane shows the surrogate key assigned to the retailer site code.

**Associate the Secondary Reference Structure With a Fact Build**

The final task is to amend the ConformedMart fact build to use the lookup that you have created.

**View the ConformedMart Visualization**

Here you view the ConformedMart in the Visualization pane.

**Step**

- In the Tree pane, click the **ConformedMart** build. The Visualization pane should look like this:

![Visualization pane](image)

You can see that the ConformedMart fact build has already been associated with the Dates-Lookup, ProductLookup, and StaffLookup lookups. These are the secondary reference structures set up for the Dates, Product, and Staff conformed dimensions.

**Associate a Dimension Element With a Reference Item**

Within the ConformedMart fact build, the RETAILER_SITE_CODE dimension element in the transformation model is currently not associated with any reference data. Here, you associate this dimension element with the lookup named RetailerLookup.

**Steps**

1. Expand the **ConformedMart** fact build.

2. Expand **Transformation Model** and double-click **RETAILER_SITE_CODE**. The **Dimension Properties** window appears.

3. Click the **Reference** tab.
4. In the Dimension box, click Retailer.

5. In the Structure box, click RetailerLookup.
   This specifies that the dimension element will use reference data from the lookup named RetailerLookup in the Retailer dimension.

6. Select the Use surrogates when available check box.
   This ensures that the fact build delivers surrogate keys from the transformation model to the fact table.

7. Click OK.

8. In the Tree pane, click the ConformedMart fact build. The Visualization pane should look like this:

9. From the File menu, click Save Catalog.
   You have now finished associating the conformed dimension with a fact build. In the next chapter, you will execute all four conformed dimensions and the ConformedMart fact build using a JobStream.
Chapter 10: Creating a JobStream

In the previous chapter you learnt about conformed dimensions and examined the way in which the ConformedMart fact build uses conformed dimensions.

In this chapter, you create a JobStream that executes the conformed dimensions from which ConformedMart reads reference data, and then executes ConformedMart.

JobStreams

Use JobStreams to group IBM® Cognos® Data Manager components into logical processes. A JobStream contains nodes, which are the individual steps within a process.

You can set up JobStreams for many different applications. Examples include conformed data marts, build status notification, preceding dimension builds with staging builds, data staging, data cleansing, both data staging and cleansing prior to a mart build, index maintenance, coping with different arrival rates of source data, partitioning of tasks to make use of multiple CPUs, and custom application logging.

To support these various applications, a JobStream can contain nodes for

- fact builds
- dimension builds
- SQL execution
- procedures, which can use all the functions, variables and control logic of the Data Manager scripting language
- conditional branching
- nested JobStreams
- email notification
- alerts

For more information about JobStreams, see the IBM® Cognos® Data Manager User Guide.

Create the JobStream

Here, you create a JobStream in which you execute the Dates, Product, Staff, and Retailer dimension builds in parallel. After execution, IBM® Cognos® Data Manager checks the status of each dimension build to determine whether it was successful or not. If all the dimension builds are successful, the ConformedMart fact build is executed. If, however, any of the dimension builds fail, Data Manager stops processing without executing the ConformedMart fact build.

The JobStream you will create will look like this.
So that Data Manager can check the status of each dimension build, you must create four status variables in the JobStream, one to check each dimension build.

Create a JobStream and Variables

Steps
1. Open the DS_Lessons catalog.
2. Right-click the Builds and JobStreams folder, and then click Insert JobStream.
3. In the Name box, type Example
4. Click the Variables tab, and then click Add.
   The Variable window appears.
5. In the Name box, type StatusDates
6. In the Type box, click BOOLEAN.
7. Click OK.
8. Repeat steps 4 to 7 to create the following Boolean variables:
   - StatusProduct
   - StatusStaff
   - StatusRetailer
9. Click OK.
Add Dimension Build Nodes

Here, you specify the dimension builds for which you want to deliver dimension data to the data mart.

**Steps**

1. Click the Example JobStream.
2. On the toolbar, click Insert Dimension Build Node. The pointer changes to a cross-hair.
3. Click anywhere in the Visualization pane to create a dimension build node. The Dimension Build Node Properties window appears.
4. In the Associated Build box, click Browse and select Dates.
5. Click the Details tab.
6. In the Result variable box, type StatusDates
7. Click OK.
8. Repeat steps 3 to 7 to create further nodes for these dimension builds.

<table>
<thead>
<tr>
<th>Associated build</th>
<th>Result variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product</td>
<td>StatusProduct</td>
</tr>
<tr>
<td>Staff</td>
<td>StatusStaff</td>
</tr>
<tr>
<td>Retailer</td>
<td>StatusRetailer</td>
</tr>
</tbody>
</table>

Add a Condition Node

Here, you specify a condition so that you can check the status of each dimension build after its execution to determine whether it was successful.

**Steps**

1. On the toolbar, click Insert Condition Node. The pointer changes to a cross-hair.
2. Click anywhere in the Visualization pane to create a condition node. The Condition Node Properties window appears.
3. In the Business Name box, type CheckStatus
   Click the Action tab, and in the Action box, type
   IF ($StatusDates AND
$StatusProduct AND
$StatusStaff AND
$StatusRetailer)
THEN
Return TRUE;
ELSE
Return FALSE;

4. Click OK.

**Add a Fact Build Node**

Here, you specify the fact build for which you want to deliver fact data to the data mart.

**Steps**

1. On the toolbar, click **Insert Fact Build Node**.
The pointer changes to a cross-hair.

2. Click anywhere in the **Visualization** pane to create a fact build node.
The **Fact Build Node Properties** window appears.

3. In the **Associated Build** box, click **Browse**, and select **ConformedMart**.

4. Click **OK**.

**Add a Procedure Node**

Here, you specify a procedure that terminates the fact build if any of the dimension builds fail.

**Steps**

1. On the toolbar, click **Insert Procedure Node**.
The pointer changes to a cross-hair.

2. Click anywhere in the **Visualization** pane to create a procedure node.

3. In the **Business Name** box, type **Abort**

4. Click the **Action** tab, and in the **Action** box, type
   
   `LogMsg('Node CheckStatus failed. ConformedMart build aborted.');`

5. Click **OK**.
**Link the Nodes**

Here, you link the nodes so that IBM® Cognos® Data Manager knows the correct order in which to process them.

**Steps**

1. On the toolbar, click **Insert Link**.
2. Click the **Start** node and drag the link to the **Dates** node.
3. Repeat step 2 to connect the remaining nodes. Refer to the following table for full information on which nodes to connect.

<table>
<thead>
<tr>
<th>Source node</th>
<th>Target node</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>Dates</td>
</tr>
<tr>
<td>Start</td>
<td>Product</td>
</tr>
<tr>
<td>Start</td>
<td>Staff</td>
</tr>
<tr>
<td>Start</td>
<td>Retailer</td>
</tr>
<tr>
<td>Dates</td>
<td>CheckStatus</td>
</tr>
<tr>
<td>Product</td>
<td>CheckStatus</td>
</tr>
<tr>
<td>Staff</td>
<td>CheckStatus</td>
</tr>
<tr>
<td>Retailer</td>
<td>CheckStatus</td>
</tr>
<tr>
<td>CheckStatus</td>
<td>ConformedMart</td>
</tr>
<tr>
<td>CheckStatus</td>
<td>Abort</td>
</tr>
</tbody>
</table>

The order in which you connect the Check Status node to the ConformedMart and Abort nodes is significant. By default, when you link a condition node, Data Manager allocates a status of True to the first link that you create, and False to the other.

**Tip:** If required, you can reverse the status of each link. On the toolbar, click **Select Mode**, right-click either the True or False link, and then click **Reverse Logic**.

The JobStream should now look like this.
4. From the File menu, click Save Catalog.

**Execute the JobStream**

Here, you execute the JobStream.

**Steps**

1. Right-click the Example JobStream, and then click Execute.
   
   The Execute JobStream dialog box appears.

2. Click Execute locally, and then click OK.
   
   IBM® Cognos® Data Manager executes the JobStream in a command window.

3. When the JobStream execution is complete, press Enter to close the command window.
   
   You have now finished creating and executing a JobStream.
Chapter 11: Creating User-Defined Folders

You can group together facts builds, dimension builds and JobStreams into user-defined folders. The advantage of using folders is the improved usability of catalogs that contain a large number of builds or JobStreams.

In the previous chapter, you set up a JobStream that executes conformed dimensions and the ConformedMart fact build. In this chapter you learn how to group the individual dimension builds, fact build, and JobStream into a user-defined folder.

Set Up a User-Defined Folder

Here you create the folder that is used to store the builds and JobStream.

Steps
1. Open the DS_Lessons catalog.
2. Right-click the Builds and JobStreams folder, and then click Insert Folder.
3. In the Name box, type ConformedMart
4. Click OK.

IBM® Cognos® Data Manager adds the user-defined folder to the bottom of the Builds and JobStreams folder.

Move the Builds and JobStream

Here you move the ConformedMart fact build, the Dates, Product, Retailer, and Staff dimension builds, and the Example JobStream into the ConformedMart folder you created.

Steps
1. Right-click the ConformedMart fact build, and then click Move to Folder.

The Move to Folder window appears.

2. Select the ConformedMart folder, and then click OK.
3. Repeat steps 1 and 2 to move the following items to the user-defined folder.

<table>
<thead>
<tr>
<th>Name</th>
<th>Component type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dates</td>
<td>Dimension build</td>
</tr>
<tr>
<td>Product</td>
<td>Dimension build</td>
</tr>
<tr>
<td>Retailer</td>
<td>Dimension build</td>
</tr>
<tr>
<td>Staff</td>
<td>Dimension build</td>
</tr>
<tr>
<td>Example</td>
<td>JobStream</td>
</tr>
</tbody>
</table>

4. From the File menu, click Save Catalog.

You have now finished creating a user-defined folder.
Chapter 12: Exporting Metadata

In this chapter, you set up a metadata export that includes the conformed dimensions and the ConformedMart fact build that you created in previous chapters.

Metadata Exports

A metadata export allows you to create descriptions of target conformed models for a data mart or data warehouse.

A conformed model contains dimensions that are shared by more than one fact table. Fact tables are grouped together using these shared dimensions to form stars. Similarly, stars are grouped into subject areas called collections.

When the conformed model descriptions have been created, you can export the metadata to an XML model. This model can then be used by IBM® Cognos® Framework Manager for deployment to IBM® Cognos® Business Intelligence.

Set Up Metadata

The main tasks required to set up the metadata are

- Create the metadata dimensions.
- Create the metadata collection.
- Add the star model to the metadata collection.

Create the Metadata Dimensions

Here, you create metadata dimensions for each of the conformed dimensions you want to export: the Dates, Product, Retailer, and Staff dimensions.

Steps

1. Open the DS_Lessons catalog.
2. In the ConformedMart folder, right-click the Dates dimension build, and then click Add to Metadata.
   The Add Metadata Dimension window appears.
The dimension table D_Dates is selected by default.

3. Click **Next**.

4. Click **Finish** to accept the default name assigned to the metadata dimension. IBM® Cognos® Data Manager creates a metadata dimension, named Dates, and adds it to the Metadata Dimensions folder.

Follow steps 2 to 4 to create the following metadata dimensions.

<table>
<thead>
<tr>
<th>Dimension build</th>
<th>Dimension table</th>
<th>Metadata dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product</td>
<td>D_Product</td>
<td>Product</td>
</tr>
<tr>
<td>Retailer</td>
<td>D_Retailer</td>
<td>Retailer</td>
</tr>
<tr>
<td>Staff</td>
<td>D_Staff</td>
<td>Staff</td>
</tr>
</tbody>
</table>

5. From the **File** menu, click **Save Catalog**.

**Create the Metadata Collection**

Here, you create the metadata collection that will be used to group together the metadata star models that you want to include in the metadata export.

**Steps**

1. In the **Metadata** folder, right-click **Collections**, and then click **Insert Metadata Collection**. The **Metadata Collection Properties** window appears.

2. In the **Name** box, type **Example**

3. Click **OK**.
IBM® Cognos® Data Manager creates a metadata collection, named Example, and adds it to the Metadata Collections folder.

**Add a Star Model to the Metadata Collection**

Here, you add a metadata star model for the F_ConformedMart fact table. This ensures that it is included in the metadata export, with reference to the metadata dimensions you have already set up.

**Steps**

1. Right-click the ConformedMart fact build, and then click Add to Metadata.

   The Add Metadata Star window appears.

   ![](image)

   The fact table F_ConformedMart is selected by default.

2. Click Next.

   On this page, the Elements column lists all the transformation model elements from the fact table that will be included in the star model.
By default, for every dimension element in the fact table, a check mark appears in the Is Dimensional column here. This indicates that the element is marked as a dimensional column in a metadata star description.

For each dimensional column shown, IBM® Cognos® Data Manager has automatically linked the associated column from the dimension table. These links are shown in the Details column.

3. Click Next.

4. Click Finish to accept the default name assigned to the star model.

Data Manager creates a star model, named F_ConformedMart, and adds it to the metadata collection that you created previously.

5. From the File menu, click Save Catalog.

View the Target Conformed Model

Here you use the Visualization pane to obtain different views of the target conformed mart.

Steps
1. In the Tree pane, click the Metadata folder. The Visualization pane should look like this:
2. Expand the **Dimensions** folder and click a metadata dimension to obtain a visual representation of the dimension, together with the fact tables referenced to it. The Dates metadata dimension is shown below.

![Diagram of Dates metadata dimension](image1)

3. Expand the **Collections** folder and click the **Example** collection to obtain a visual representation of the star model you created.

![Diagram of Example collection](image2)

4. Expand the **Example** collection and click the **F_ConformedMart** star model to obtain a visual representation of the star, together with the metadata dimensions it references.

![Diagram of F_ConformedMart star model](image3)

**Export the Metadata**

Here you export the conformed model you have set up to an XML model.

**Steps**

1. Right-click the **Metadata** folder, and then click **Export Metadata**.

   The **Export Metadata** window appears with the Example metadata collection selected by default.

2. Click **OK**.

3. In the **File Name** box, type **Metadata_Export**, then click **Save**.
In IBM® Cognos® Data Manager, the following catalog objects are referred to as components: connections, builds, reference dimensions, reference structures, JobStreams, templates, functions, metadata dimensions, and metadata collections.

In this chapter, you learn how to

- Copy components using packages.
- Search for individual components and component dependencies using the navigator.

**Copy Components**

You can use packages to copy individual components from one catalog to another. This is particularly useful if you want duplicate components in different catalogs without having to recreate them.

**Create a Package**

First, you must create a package that contains the components you want to copy.

**Steps**

1. Open the DS_Lessons catalog.
2. From the File menu, click Create Package.
   
   The Create Package window appears. The Available components pane shows all the components in the catalog.
3. In the Available components pane, expand the Dimensions folder and the Product reference dimension.
4. Select the check box for the Product hierarchy to include it in the package.
   
   IBM® Cognos® Data Manager copies the Product hierarchy, and all its dependents, to the Components in package pane.
   
   In the Available components pane, Data Manager automatically selects the check boxes for copied components.
Tip: To remove a component from the Components in package pane, clear its check box. Data Manager also removes any dependents of that component.

5. Click OK.

The Package File dialog box appears.

6. In the File name box, type Product, and then click Save.

**Import a Package**

You can now import the components of the Product package into another catalog.

**Steps**

1. Open the catalog named MyCatalog, which you created in Chapter 3.

2. From the File menu, click Import Package.

   IBM® Cognos® Data Manager prompts you to back up your catalog before importing a package.

3. Click No to continue.

   The Package File dialog box appears.

4. Click the file Product.pkg, and then click Open.

   The Import Package Wizard appears.
5. Click the check box at the top of the left column to select all components for import, and then click Next.

6. Click Finish to import the components and overwrite existing components without renaming them.

**View Component Dependencies**

You can use the navigator to view the dependencies of individual components.

Here, you search for all components that use the D_Product template, created by IBM® Cognos® Data Manager when you created the Product conformed dimension in Chapter 9.

**Steps**

1. Open the DS_Lessons catalog.
2. Expand the Dimensions folder, the Product dimension, and the Templates folder.
3. Click the D_Product template.
4. From the View menu, click Navigate.

A list of the components that are dependent upon the D_Product template appears in the Navigator dialog box.
5. Click the Product dimension build to view its dependencies.

You can see that the dimension build is used in the Product dimension build node in the Job-Stream that you created.

6. Click the back navigation button to return to the list of D_Product template dependencies.

7. Click the ProductLookup reference structure to view its dependencies.

You can see the reference structure is used in the ProductNumber dimension element in the ConformedMart fact build.

8. Continue exploring the dependencies of individual components as you require.

**Tip:** Use the navigation buttons to move back and forth between the components that you previously selected.

Leave the navigator open so that you can complete the next section.

---

**Search for Components**

Here, you learn the different methods that you can use to search for components in IBM® Cognos® Data Manager.

**Steps**

1. In the **Navigator** dialog box, click the **Find Components** tab.

2. In the **Search for the text** box, type **product**, and then click **Find Now**.

   Data Manager lists each component named product, together with the component type and location of each one.

3. Try different searches to see how the results differ:
   - Type **product** to search for all components that begin with the word product.
   - Type **product** to search for all components that contain the word product.

   For more information about using wildcard characters, see the IBM® Cognos® Data Manager User Guide.

4. When you have finished using the navigator, close the dialog box.
Appendix A: Sample Data

The data available for the Great Outdoors Company begins in January 2004 and continues until the end of December 2006.

This company sells a wide variety of outdoor equipment and accessories. Their product lines are

- Camping Equipment
- Personal Accessories
- Golf Equipment
- Outdoor Protection
- Mountaineering Equipment (introduced in January 2005)

Products from these lines include items such as tents, watches, putters, insect repellent, and climbing tools.

The Great Outdoors Company has a large sales staff who sell their products to retailers around the world. These retailers, in turn, sell these products to the public. Because the Great Outdoors Company does not sell directly to the public, retailers are their customers.

Retailers are located in 21 countries around the world. Each retailer sells a variety of products from each product line, but not necessarily all products. Some products are very successful in certain regions. For example, the Golf Equipment line is the most successful product line in Asian countries.

Some of the samples are common to all IBM® Cognos® products, and others are specific to IBM® Cognos® Data Manager. The sample data includes a Data Manager catalog, DS_Tutorial, that contains specimen solutions, which you can use to check your work.

This guide uses sample databases and other sample data sources that contain sales and marketing data for the Great Outdoors Company.

The information that follows is not essential to work through this guide. However, for a more thorough understanding of the business that you model here, you should have some familiarity with these samples.

SQLTXT Sample Data

The samples include a SQLTXT database that provides additional data for some of your work. By default SQLTXT sample data is located in c10_location\webcontent\samples\DataManager\SQLTXT

The SQLTXT sample data consists of a SQLTXT catalog and text data files.

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ds_sources.def</td>
<td>The SQLTXT catalog.</td>
</tr>
</tbody>
</table>
### DS_Advanced.mdb

This catalog contains examples that illustrate some of the more advanced features of IBM® Cognos® Data Manager.

<table>
<thead>
<tr>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DerivedDimensionsExample</td>
<td>A fact build that uses a derived dimension to perform a 'lookup on a lookup'.  &lt;br&gt;  The fact build is referenced by three lookups, StaffLookup, DatesLookup, and ProductDDLookup.</td>
</tr>
<tr>
<td>JobSequence</td>
<td>A JobStream that consists of  &lt;br&gt;  • four dimension builds (ProductDim, SiteDim, StaffDim, and TimeDim) that run in parallel  &lt;br&gt;  • a condition node that tests for the successful completion of the dimension builds  &lt;br&gt;  • another JobStream (FactBuild) that depends on the successful completion of all the dimension builds  &lt;br&gt;  • a procedure node that notifies whether the job sequence succeeded or failed</td>
</tr>
<tr>
<td>Example</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>StaffSCD, UpdateStaffSCD</td>
<td>Dimension builds that illustrate attribute change tracking. They include changes in the Staff dimension that result from sales staff moving between sales branches. Execute the build StaffSCD to establish the dimension table in the target data mart. Then execute UpdateStaffSCD to see how Data Manager tracks changes in the data.</td>
</tr>
<tr>
<td>various user-defined functions</td>
<td>The following internal and external user-defined functions are included in DS_Advanced:</td>
</tr>
<tr>
<td></td>
<td>• IsEvenNum( num[integer] ) An internal function that determines whether an integer is even.</td>
</tr>
<tr>
<td></td>
<td>• NextWeek( ) An internal function that returns a date seven days from the current system date.</td>
</tr>
<tr>
<td></td>
<td>• StripTimePortion( Date[date] ) An internal function that converts the time portion of a date to 00:00:00.</td>
</tr>
<tr>
<td></td>
<td>• TempConv( Argument1 [float] ) An external function that converts from Celsius to Fahrenheit.</td>
</tr>
<tr>
<td></td>
<td>• Time( Argument1[char] ) An external function that returns Greenwich Mean Time if the first character is G and the local (system) time otherwise.</td>
</tr>
</tbody>
</table>

**Note:** External functions require the file, sampleUDF.dll. This file is located in the following directory:

c10_location\webcontent\samples\DataManager\udfs

Before using external functions, copy this file to one of these locations:
• the current working directory of Data Manager (\bin)
• one of the directories specified in the PATH environment variable
• the Microsoft® Windows® or Windows® system directory
Index

B
business dimensions
   overview, 26

catalogs
   creating, 21
   dimension builds, 17
   fact builds, 17
   library, 19
   opening, 16
   overview, 15
   Visualization pane, 20

components
   copying, 87
   overview, 87
   searching for, 90
   viewing dependencies of, 89

conformed dimensions
   creating, 65
   overview, 64

connections
   ODBC, 23
   SQLTXT, 24

data source names
   setting up, 12

data warehouses
   overview, 9
delivering
   dimension data, 39
   fact data, 39
derivations
   DataStream, 46
   overview, 45
   transformation model, 46
dimensional framework
   overview, 25
dimension builds
   overview, 17, 63
dimension data
   delivering, 39
delivering
   fact builds, 39
   JobStreams, 78
   execution logs, 41

fact builds
   additional data sources, 51
   executing, 39
   overview, 17
   setting up, 35

fact data
   delivering, 39

JobStreams
   creating, 73
   executing, 78
   overview, 73

library
   overview, 19

metadata
   collections, 82
   creating, 81
   dimensions, 81
   exporting, 85
   overview, 81
   star model, 83
   multiple granularity, 57
Index

O
ODBC connections
setting up, 23
opening
catalogs, 16
overview
business dimensions, 26
catalogs, 15
components, 87
conformed dimensions, 64
data warehouses, 9
dimensional framework, 25
dimension builds, 17, 63
fact builds, 17
JobStreams, 73
library, 19
metadata, 81
transformation model, 35
user-defined folders, 79
Visualization pane, 20
reference dimensions, 26
reference structures, 27
SQLTXT connections, 24
user-defined folders, 79
SQLTXT connections
setting up, 24

T
transformation model
overview, 35

U
user-defined folders
creating, 79
overview, 79

V
Visualization pane
overview, 20

P
packages
creating, 87
importing, 88

R
reference dimensions
setting up, 26
reference structures
setting up, 27
rejected data, 53

S
sample data, 11, 91
sample databases, 11
setting up
catalogs, 21
conformed dimensions, 65
data source names, 12
derivations, 45
fact builds, 35
JobStreams, 73
metadata, 81
ODBC connections, 23
packages, 87