ORACLE®

PEOPLESOFT ENTERPRISE

PeopleTools 8.51 PeopleBook: PeopleSoft Cube Manager

August 2010

ORACLE®

PeopleTools 8.51 PeopleBook: PeopleSoft Cube Manager
SKU pt8.51tcbm-b0810

Contents

Chapter 4

Chapter 5

PeopleSoft Cube Manager Preface

This book discusses PeopleSoft Cube Manager.

PeopleSoft Cube Manager

PeopleSoft Cube Manager is a set of PeopleTools pages and processes that you use to create and maintain analytic data stores, also called Online Analytical Processing (OLAP) cubes. This book explains the basic concepts of OLAP and the tools that are supported, as well as how to use PeopleSoft PeopleTools to design analytic components and build and troubleshoot cubes.

Note. The concepts of Outlines, Aggregate Data, Point of View (POV), Block Storage, and Aggregate Storage used in PeopleSoft Cube Manager are the same concepts used in Oracle Hyperion. For more information, see the documentation of Oracle Hyperion product.

PeopleBooks and the PeopleSoft Online Library

A companion PeopleBook called *PeopleBooks and the PeopleSoft Online Library* contains general information, including:

- Understanding the PeopleSoft online library and related documentation.

- How to send PeopleSoft documentation comments and suggestions to Oracle.

- How to access hosted PeopleBooks, downloadable HTML PeopleBooks, and downloadable PDF PeopleBooks as well as documentation updates.

- Understanding PeopleBook structure.

- Typographical conventions and visual cues used in PeopleBooks.

- ISO country codes and currency codes.

- PeopleBooks that are common across multiple applications.

- Common elements used in PeopleBooks.

- Navigating the PeopleBooks interface and searching the PeopleSoft online library.

- Displaying and printing screen shots and graphics in PeopleBooks.

- How to manage the locally installed PeopleSoft online library, including web site folders.

- Understanding documentation integration and how to integrate customized documentation into the library.

- Application abbreviations found in application fields.

You can find this companion PeopleBook in your PeopleSoft online library.

Chapter 1

Getting Started with PeopleSoft Cube Manager

This chapter provides an overview of PeopleSoft Cube Manager and discusses:

- PeopleSoft Cube Manager integrations.

- Other sources of information.

- Implementation

PeopleSoft Cube Manager Overview

PeopleSoft Cube Manager enables you to build Online Analytical Processing (OLAP) databases, or cubes, which are specifically designed for data analysis. OLAP cubes are collections of related data—like a database with multiple dimensions. These dimensions, like database fields, are criteria that let you identify your data. An example dimension might be Time or Product. Dimensions in turn are subdivided into categories, called cube members. For instance, in the Time dimension, one member might be the year 2000.

Note. Cubes are not limited to three dimensions. They can consist of three, four, or more dimensions.

PeopleSoft Cube Manager Business Processes

The following process flow illustrates the Cube Manager business processes.
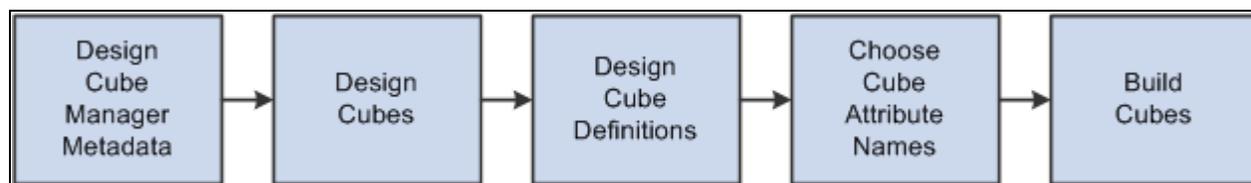
Cube Manager business process flow

PeopleSoft Cube Manager Integrations

You can use PeopleSoft Cube Manager in tandem with the third-party OLAP platforms Cognos PowerPlay and Oracle Essbase. In addition, you can use Oracle's PeopleSoft Tree Manager and Oracle's PeopleSoft Query to describe metadata to PeopleSoft Cube Manager.

Note. You do not need Oracle Essbase or Cognos PowerPlay installed to design cubes and dimensions with PeopleSoft Cube Manager. However, you do need to install one of them before you can build cubes. Basically, you use PeopleSoft Cube Manager to design platform-independent metadata that you can then use to build cubes on any of the supported platforms.

Other Sources of Information

This section provides information to consider before you begin to use Oracle PeopleSoft Cube Manager.

Take advantage of all PeopleSoft sources of information, including the installation guides, release notes, PeopleBooks, red papers, the Updates and Fixes area of My Oracle Support, and curriculum courses of Oracle PeopleSoft.

See "PeopleSoft Cube Manager Preface," page vii.

See *PeopleTools 8.51 PeopleBook: Getting Started with Enterprise PeopleTools.*

PeopleSoft Cube Manager Implementation

Cube Manager implementation can be divided into the following phases:

• Designing cube metadata.

• Designing cubes.

• Designing cube definitions.

• Selecting cube attribute names.

• Building cubes.

Designing Cube Metadata

The initial piece of the process for creating cubes is to define their metadata—data that describes the structure of the cube. You can define metadata with either PeopleSoft trees or queries.

See Chapter 3, "Designing Cube Metadata," Query Types, page 15.

See *PeopleTools 8.51 PeopleBook: PeopleSoft Cube Builder*, "Designing Cube Metadata."

Designing Cubes

When you have established the cube metadata, you can design the cube itself. To design the cube, perform the following steps:

Step	Reference
1. Define the high-level dimensional structures of the cube.	See Chapter 4, "Designing Cubes," Defining the Dimensional Structures of the Cube, page 27.
2. Define dimension options by naming blank members in your hierarchy to view in your cube.	See Chapter 4, "Designing Cubes," Defining Dimension Options, page 32.
3. Define rollup options by identifying prefix and suffix options.	See Chapter 4, "Designing Cubes," Defining Rollup Options, page 34.
4. Specify the source of the metadata for each rollup.	See Chapter 4, "Designing Cubes," Specifying the Source of the Metadata for Each Rollup, page 35.
5. Map fields to a role in the rollup.	See Chapter 4, "Designing Cubes," Mapping Fields to a Role in the Rollup, page 38.

Designing Cube Definitions

In addition, you need to design cube definitions by performing the following steps:

Step	Reference
1. Map data sources to dimensions and select dimensions for the cube.	See Chapter 4, "Designing Cubes," Mapping Data Sources to Dimensions and Selecting Dimensions for the Cube, page 40.
2. Add cube inputs to a cube definition.	See Chapter 4, "Designing Cubes," Adding Cube Inputs to a Cube Definition, page 43.
3. Map source query fields to cube components.	See Chapter 4, "Designing Cubes," Mapping Source Query Fields to Cube Components, page 43.

Selecting Cube Attribute Names

Finally, you need to select cube attribute names by performing the following step:

Step	Reference
Specify user-defined attributes.	See Chapter 4, "Designing Cubes," Specifying User-Defined Attributes for Members, page 45.

Building Cubes

After you have designed the cube, you are ready to build it. To build the cube, perform the following steps:

Step	Reference
1. Build the cube instance.	See Chapter 5, "Building Cubes," Building Cube Instances, page 47.
2. Build the cube itself.	See Chapter 5, "Building Cubes," Building Cubes, page 61.

Chapter 2

Introducing PeopleSoft Cube Manager

This chapter discusses:

- Online Analytical Processing (OLAP).

- Cubes.

- PeopleSoft Cube Manager.

- How to use supported OLAP tools.

- Cube design and components.

Note. The information presented here is not a substitute for your Oracle Essbase, Cognos PowerPlay, or other third-party technology documentation, but should help you integrate your PeopleSoft data with one or more of these tools.

See Also

Understanding OLAP

Most business software users are familiar with Online Transaction Processing (OLTP) applications, which are for used for creating and maintaining information about business operations. The transactions that are stored by these applications are the heart of any business software. At the database level, OLTP applications are designed to allow for speedy creation of data and to reduce redundant information. However, data structures of this design are not well suited to analysis.

The nature of OLTP databases poses a problem: how to analyze data in a database that is not designed for analysis. One solution is to use a product such as Oracle PeopleSoft nVision. Such products perform analysis on selected characteristics of the database. However, using a powerful tool, such as PeopleSoft nVision, on top of an OLTP database, takes time.

In contrast, OLAP applications are designed specifically for data analysis. The source of information for analysis is an OLTP database. To make the OLTP data available to analytical applications, data is extracted and transformed into a format that is easier to analyze. You can store the resulting OLAP database in several different formats, depending on the tools that you used to access the data.

MOLAP

Multidimensional OLAP (MOLAP) is a format that stores all of the data hierarchically. This type of database is mainly for small-to-medium data marts. MOLAP databases, such as Oracle Essbase, summarize and access data quickly. The only drawback of MOLAP systems is that as dimensional information, sizes, or numbers increase, the storage mechanism becomes less efficient.

ROLAP

Relational OLAP (ROLAP) is a format that stores the analytical data in relational tables. The main benefit of the ROLAP format is its ability to store vast amounts of data. However, ROLAP data storage is not as efficient in accessing aggregate information at higher levels of the hierarchy.

The structures of the data schema can be one of two types: snowflake or star schema. Snowflake schemas are designed to keep very little redundant data, whereas star schemas encourage duplicate data. This release of PeopleSoft Cube Manager supports only the star schema.

In a star schema, each dimension is represented in a single table. The *fact data,* data that is to be analyzed, is stored in a separate table. The fact table contains one column to represent each of the dimensions from which the data was created.

The following diagram illustrates a typical star schema:
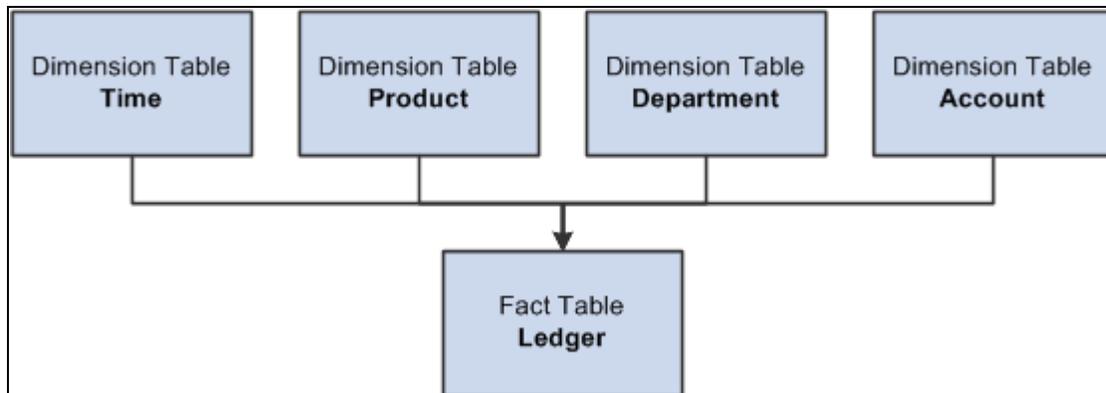
Star schema structure

See *PeopleTools 8.51 PeopleBook: PeopleSoft Cube Builder*, "Designing Cube Metadata," Understanding Cubes.

HOLAP

Hybrid OLAP (HOLAP) is the latest type of analytical structure. As the name implies, the data is stored in a hybrid format. The base fact data is stored in a set of relational tables, whereas the summary data is stored in a multidimensional format. This strategy is powerful, but can pose problems when you are combining data from the relational tables with the multidimensional summary data.

Generally speaking, multidimensional analysis is not a technology. Rather, it is a type of analysis that enables you to easily view data from many perspectives. These tools enable you to explore, interact with, and "slice and dice" complex data, guiding you to the multidimensional information that you can't easily discover with conventional reporting tools.

Understanding PeopleSoft Cube Manager

This section provides an overview of Cube Manager and discusses:

- Cube Manager architecture.

- PeopleSoft metadata.

- PeopleSoft Process Scheduler integration.

- Designing and building OLAP cubes.

Cube Manager Overview

You can build an OLAP database out of PeopleSoft data without using PeopleSoft Cube Manager. In fact, customers have implemented custom OLAP solutions with previous PeopleSoft releases. However, PeopleSoft Cube Manager provides important benefits when connecting between your PeopleSoft application data and your OLAP platform.

PeopleSoft Cube Manager is a PeopleTool used to build OLAP databases or cubes. It provides a framework for modeling your OLAP cubes through its design tools: PeopleSoft Tree Managers and PeopleSoft Query. This framework supports the definition of all dimensions, attributes, measures, and cubes that you might want to build from PeopleSoft sources. You can share all of these definitions across cubes to ensure that OLAP results are consistent across your enterprise.

You can also use components designed in PeopleSoft Cube Manager across all target OLAP platforms supported by Oracle. PeopleSoft Cube Manager provides all these benefits while achieving a nearly platform-independent solution. You can apply the majority of a cube's design attributes to Oracle Essbase, Cognos PowerPlay, or Generic Star Schema. PeopleSoft Cube Manager also enables you to leverage your existing PeopleSoft metadata to define the cube structure.

Cube Manager Architecture

The following diagram illustrates the cube building process:
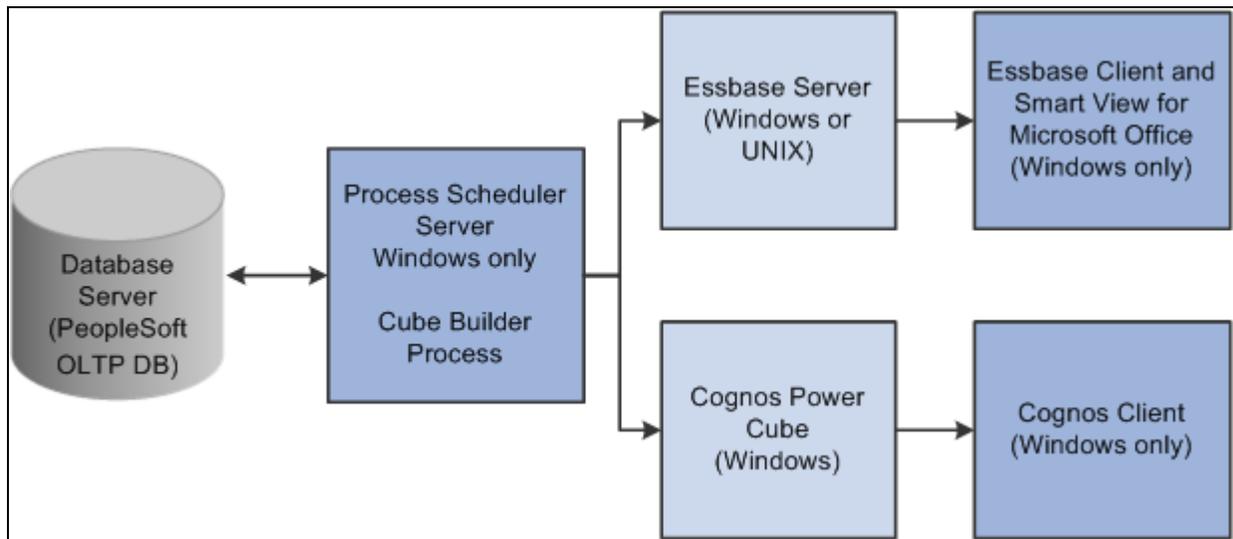
PeopleSoft Cube Manager: The big picture

The cube building process includes the following steps:

1. PeopleSoft Cube Manager extracts data from the PeopleSoft application (OLTP) database.

2. PeopleSoft Cube Manager writes to the OLAP data store.

3. A third-party reporting tool directly analyzes the OLAP data store.

PeopleSoft Metadata

Metadata is data that defines data. Metadata conveys information about how data is formatted, structured, and stored. In an OLAP cube, metadata defines dimensions, levels, members, member attributes, and interrelationships of the cube. PeopleSoft Cube Manager uses two types of PeopleSoft structures—trees and queries—to define cubes.

See Also

Chapter 3, "Designing Cube Metadata," page 15

PeopleSoft Process Scheduler Integration

Oracle PeopleSoft Process Scheduler includes a process type definition specifically for use with PeopleSoft Cube Manager. This process type is the Cube Manager process type, and you invoke it whenever you launch the process to create a cube from the standard run control page. During this process, depending on the OLAP tool that you specify, the data and metadata are translated into a format that is understood by Oracle Essbase, Cognos PowerPlay, or another ROLAP tool that can read a star schema.

See Also

Chapter 5, "Building Cubes," Defining a Cube Build Process Using Process Scheduler Manager, page 67

Designing and Building OLAP Cubes

Identifying requirements is the most important part of any analytical application. An analytical application always produces results, regardless of whether those results are meaningful. Identifying requirements is essential to generating meaningful results.

Decide what aspects or processes of the business you want to capture. Examples can be sales activity, claims processing, or marketing return on investment. You might want to analyze several unrelated aspects of your business. Do not try to make one cube for all of the aspects. Instead, treat each set of related information as a single cube.

Next, identify the measures that you use to quantify those results, such as sales amounts in dollars or units. This information is almost always numeric. Then, identify the criteria with which you want to view the data and the granularity of the data. These criteria form the dimensions. The most common dimensions are time, accounts, geography, products, and department.

See the Oracle Essbase, Cognos PowerPlay, or other third-party technology documentation.

Using Supported OLAP Tools

Each PeopleSoft customer has unique reporting and analysis needs. To address these needs, Oracle provides support for various OLAP databases and tools, such as Oracle Essbase and Cognos PowerPlay.

If you have not selected an OLAP platform, the following descriptions should help you decide which platform best suits your needs.

This section discusses:

•   Oracle Essbase

•   Cognos PowerPlay

Oracle Essbase

Oracle Essbase is a robust, client/server-based product, and part of the Oracle Enterprise Performance Manager, the scope of which goes beyond individual users to the level of a data mart. However, because it supports metadata updates, Essbase enables you to keep a persistent data store, whereas you must re-create PowerCubes whenever any metadata changes.

Essbase includes the Smart View for Microsoft Office add-in that enables you to view OLAP cubes using Microsoft Excel and other Microsoft Office products—much like PS/nVision. Essbase is shipped separately from PeopleTools (separate licensing components are available for the integration and the end-user product).

Cognos PowerPlay

Cognos PowerPlay includes a product called Transformer, which builds the portable data cube. This cube is easy to attach as a file to email.

Cognos PowerPlay includes two components: a database engine component (the PowerCube) and an end-user component. This front-end component can be used not only for Cognos PowerPlay databases, but also for other OLAP databases, including Oracle Essbase.

Several new components exist in Cognos PowerPlay 7.x (Enterprise Server version) that work together to give you more functionality and to make Cognos PowerPlay cubes available to remote users. PeopleSoft Cube Manager has been modified to work with some of these new pieces, as described in this section.

See Supported Platforms on My Oracle Support for exact certified versions of third-party products such as Cognos PowerPlay.

This section discusses:

- Enterprise Server.

- PP Enterprise Server Administrator.

- PPAdmtool.

- PPApplications.

- How PeopleSoft Cube Manager uses the EP Server.

Enterprise Server

Enterprise Server (EP Server) is a Microsoft Windows service that maintains cubes at a given location. Users from remote locations connect to this service and open the cubes in their choice of Cognos PowerPlay for Windows, Cognos PowerPlay for Excel, or Cognos PowerPlay for Web.

You can configure PeopleSoft Cube Manager to register cubes automatically with the EP Server. The EP Server needs a port number, server or machine name, user name, and password for registering a cube. However, PeopleSoft Cube Manager can add cubes only to the EP Server that is running locally, and only to the admin account. This means that users are required to specify only the port number and password for the admin account.

See *Cognos PowerPlay Enterprise Server* documentation.

Note. PeopleSoft Cube Manager and the EP Server must run on the same machine. When updating a cube, you should make sure that it is not being accessed by anyone prior to running PeopleSoft Cube Manager.

PP Enterprise Server Administrator

PP Enterprise Server Administrator is a Microsoft Windows application that can communicate with the EP Server, either locally or remotely, and can display the cubes that the EP Server is maintaining. Depending on the security privileges, you can add or remove cubes from the EP Server by using this application. You can also change passwords from this application.

PPAdmtool

PPAdmtool is a command-line program that can communicate with the EP Server. You can add or remove cubes from the EP Server by using this program. To carry out any commands using this program, you must specify the server or machine name, port number, and password.

The difference between the PP Enterprise Server Administrator and PPAdmtool is that PPAdmtool runs at the command prompt. The PP Enterprise Server Administrator is a Microsoft Windows application that you must start before viewing or manipulating the cubes. You can use PPAdmtool from the prompt to connect to, view, add, or remove cubes without performing the extra step of starting a program.

PPApplications

PPApplications are the actual Microsoft Windows applications (Cognos PowerPlay for Windows, Cognos PowerPlay for Excel, or Cognos PowerPlay for Web) that can open the cube .mdc files. These applications need either direct access to the .mdc file (locally or on the network) or access to the EP Server that is maintaining a particular cube.

How PeopleSoft Cube Manager Uses the EP Server

This example shows how PeopleSoft Cube Manager uses the new functionality of Cognos PowerPlay 7.x to manage a cube that exists on the EP Server.

Step 1:
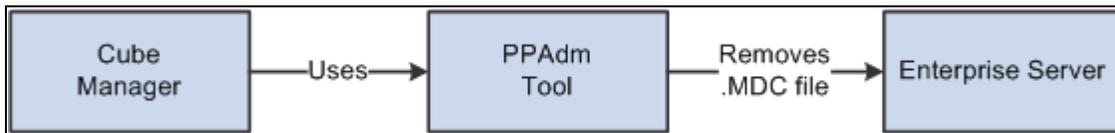
PPAdmtool removes the .mdc file from the EP Server

Step 2:
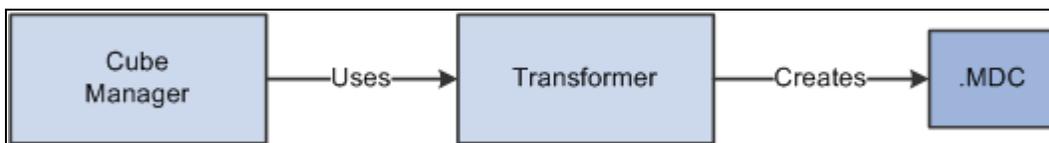
PeopleSoft Cube Manager uses the transformer to create the .mdc file

Step 3:
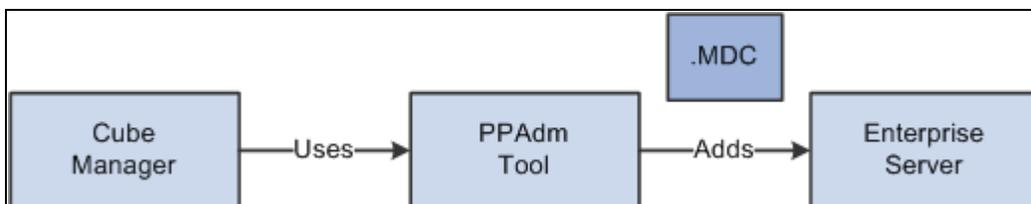
PeopleSoft Cube Manager uses PPAdmtool to add the .mdc file to the EP Server

The Platform Options page (PeopleTools, Cube Builder, Cube Manager, Cube Instances, Platform Options) for Cognos PowerPlay has been modified to take the extra information needed to communicate with the EP Server.

See Also

Understanding Cube Design and Components

Before using PeopleSoft Cube Manager, you should define the specific goals and results that you expect from online data analysis. After defining your goals, you should design the PeopleSoft trees and queries that are appropriate for creating both the structure and the data of the cube that you plan to build. These trees and queries supply data from your PeopleSoft application to any cubes that you create. After you establish your goals and create the necessary trees and queries upon which the resulting cube will be built, use PeopleSoft Cube Manager to begin designing a cube.

Many different kinds of components make up the cubes that you build with PeopleSoft Cube Manager:

|	
Dimensions	The most basic component of a cube is a dimension. When you create a dimension, you specify the PeopleSoft metadata to be used to create the rollup structure of the dimension. PeopleSoft Cube Manager dimensions are platform-independent; you can reuse them again and again to build different cubes.
Member Properties	Dimension members may have various member properties associated with them. For example, if you have a product dimension, you might want to add color as a member property of the actual product.
Outlines	An outline defines the basic structure of a cube. You select the dimensions and measures that make up the cube and the data source queries that populate the members and cube cells with data. Like dimensions, outlines are platform-independent and reusable.
Cube Instance Definitions	Cube instance definitions are platform-specific. In a cube instance definition, you specify the platform for which the cube will be built and select the cube definition on which the cube will be based. You then set various other options that are specific to the chosen platform.
Platform Options	Each of the Online Analytical Processing (OLAP) target databases has an associated set of options by which PeopleSoft Cube Manager controls how the cube is built. These options can provide PeopleSoft Cube Manager with any required security information or information about how to allocate database space for dimensions in the cube.

Run Control IDs   When you have defined your dimensions, cube definition, and cube instance definition, you are ready to start the build process. Oracle PeopleSoft Process Scheduler runs this process and—as with all PeopleSoft Process Scheduler processes—you create a run control ID that defines the specifics of the process:

- Specify which outline and which connection definition you want to use to build the cube.

- Define whether the build process is to create a new cube or to update an existing one.

- Specify the metadata actions and the restructuring options.

Note. Using run control ID, you can also run a single dimension or a single data source.

Cube Instance   A cube instance is the output of the PeopleSoft Cube Manager. It is a physical cube that is built by the Cube Manager process in Cognos PowerPlay or Essbase.

Chapter 3

Designing Cube Metadata

This chapter provides an overview of query types.

See Also

PeopleTools 8.51 PeopleBook: PeopleSoft Tree Manager, "Using PeopleSoft Tree Manager"

PeopleTools 8.51 PeopleBook: PeopleSoft Query, "Creating and Running Simple Queries"

Query Types

This section provides an overview of query types and discusses:

- Dimension queries.

- Attribute Queries

- Data source queries.

Understanding Query Types

You can create several types of queries to use with PeopleSoft Cube Manager, all of which you must define as *user(ad hoc)* queries, as opposed to *role* queries, or *database agent* queries.

Because you cannot have duplicate member names in a dimension, unless those members are shared (Essbase is the only platform that handles shared members), you cannot have duplicate query column names in your query metadata. PeopleSoft Cube Manager treats uppercase and lowercase characters as distinct, so the names *ABC,Abc,* and *abc* are all considered unique member names. However, Essbase offers an option to change all member names to uppercase. If you enable this option, you create problems with members that are identical except for their letter casing.

Note. PeopleSoft Cube Manager permits duplicate node names if you cannot avoid the duplication.

Dimension Queries

Dimension queries enable you to define the dimension structure using query results instead of, or in addition to, a tree. However, remember that you are using queries to create a tree-like structure. For narrow query definitions, each dimension query maps child members at a particular level to parent members at the next higher level. For wide query definitions, you need only one query to build the dimension. Optionally, these queries might contain extra members or level attributes, as well as the relationship information.

You can convey hierarchical information in one of two ways:

• By parent/child relationship, or a narrow query.

• By level specification, or a wide query.

Narrow Query Definition

When you use multiple queries to define dimensional structure, the first query that you specify defines the first two levels of the dimension. To add lower levels, you must write one additional query for each additional level.

Suppose that you want to build a department dimension that contains the following departments within an organization:
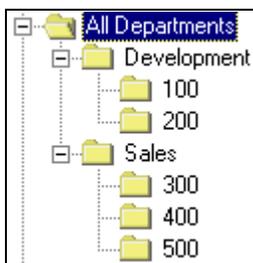
Example of departments within an organization

The levels of the organization might be described in the following way:

Level 1 (top of dimension)	Level 2	Level 3
ALL DEPARTMENTS	DEVELOPMENT	100
		200
	SALES	300
		400
		500

Example of levels of the organization

To create the dimension, you must write two queries to provide the preceding information: one to define the child members at level 2 and one to define the child members at level 3.

Query 1:

Parent	Member (Child)
ALL DEPARTMENTS	DEVELOPMENT
ALL DEPARTMENTS	SALES

Query 1

Note. During the build, PeopleSoft Cube Manager knows that child members with an unspecified parent become level 2 members, directly under the top of the dimension. You can create an empty column in a query by adding a blank *("")* expression. Be sure to enter some meaningful text for the heading text and unique field name (such as *Top of Dimension*) so that you can easily identify the blank column when mapping query columns to dimension levels.

Query 2 must return one column for the second-level members (DEVELOPMENT and SALES) and one column for the third-level members (100, 200, 300, 400, and 500).

The query results look like this:

Parent	Member (Child)
DEVELOPMENT	100
DEVELOPMENT	200
SALES	300
SALES	400
SALES	500

Query 2

To add additional levels, write one query for each additional level. Building on the previous example, assume that you want to create an employee dimension in addition to a department dimension. In this case, include a fourth level showing the employees in each department:
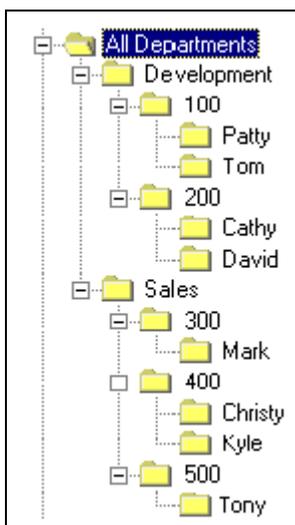
Example of employees within an organization

Now you might describe the levels of the organization in the following way:

Level 1	Level 2	Level 3	Level 4
ALL DEPARTMENTS	DEVELOPMENT	100	Patty
			Tom
		200	Cathy
			David
	SALES	300	Mark
		400	Christy
			Kyle
		500	Tony

Description of the levels of the organization

In addition to the two queries that you create, you must create a third query with the following results to add the fourth level to the dimension:

Parent	Child
100	Patty
100	Tom
200	Cathy
200	David
300	Mark
400	Christy
400	Kyle
500	Tony

Results to add to the fourth level to the dimension

The following diagram illustrates how you supply three queries to create a dimension with four levels:

Query 1	Level 1 Parent	Level 2 Child		
Query 2		Level 2 Parent	Level 3 Child	
Query 3			Level 3 Parent	Level 4 Child

Four-level dimension created using three queries

Your dimension queries should always produce at least two columns in the result set: one for the parent and one for the child (or member). Except for the top-level query, a parent column for a given level must always correspond to the child column of the previous level.

Optionally, you can include attributes in the query. All attributes that you specify in a narrow query are associated with the field that is specified as the member. You must add attributes for the topmost member by using an attribute query.

Wide Query Definition

When defining your dimension with a wide query, follow the same logic that is used with narrow queries.

Using the previous example for narrow queries, the following table illustrates the dimension that you want to build with a wide query:

Level 1	Level 2	Level 3	Level 4
ALL DEPARTMENTS	DEVELOPMENT	100	Patty
			Tom
		200	Cathy
			David
	SALES	300	Mark
		400	Christy
			Kyle
		500	Tony

Dimension that you want to build with a wide query

Instead of writing multiple queries to build this dimension, write one that contains only this information. The query result set might look like the data in this table:

Level 1	Level 2	Level 3	Level 4
ALL DEPARTMENTS	DEVELOPMENT	100	Patty
ALL DEPARTMENTS	DEVELOPMENT	100	Tom
ALL DEPARTMENTS	DEVELOPMENT	200	Cathy
ALL DEPARTMENTS	DEVELOPMENT	200	David
ALL DEPARTMENTS	SALES	300	Mark
ALL DEPARTMENTS	SALES	400	Christy
ALL DEPARTMENTS	SALES	400	Kyle
ALL DEPARTMENTS	SALES	500	Tony

Query results of writing one query to build dimension

As with narrow queries, the query can return attributes that you can associate with members. However, in the case of wide queries, you can assign the attributes to any member.

Attribute Queries

You can use attribute queries to set optional attributes on the members within a dimension. Set attributes on either node members or detail members of a dimension.

An attribute query should return at least two result columns: one that identifies the members whose attributes you want to define and one for each type of attribute to be set.

Member	Attribute 1	Attribute 2 (Optional)	Attribute 3 (Optional)
1000	XXX	XXX	
1100	XXX	XXX	XXX
1200	XXX		XXX
1300	XXX	XXX	

Example of an attribute query with at least two result columns

Most attribute types are platform-specific: they are used by one of the target platforms, but not all. Each attribute type uses its own set of conditions to determine whether to apply the attribute. In some cases, for the attribute to be applied, the attribute column must contain a specific value. In other cases, the attribute column simply must not be blank. An attribute query can return a subset of the members, in which case the specified attributes are set only on those members returned.

If an attribute query is on a table that uses a setID, you must select only the setID in which you're interested. Thus, only one member ID field exists to tie the query results to the dimension.

Note. In the preceding example, one query sets three attributes. You can also create three separate queries in which you return the member and one attribute.

General Attributes

You can define numerous types of attributes by using queries and PeopleSoft Cube Manager. We discuss the following valid attribute types:

• Flip sign.

Flip sign attribute refers to the reversal of + (plus) and − (minus) signs for the member, which is sometimes necessary for accounting purposes. Valid values are blank (do not flip the sign) or non-blank (flip the sign). In a star schema, an attribute column is populated with a value of − 1 when a member has a sign to flip. In Cognos PowerPlay and Oracle Essbase, the data that is populated in a flip-signed cell has the opposite sign of the source query.

• Label.

Label attribute provides a description for a member. In an Essbase database, this label equates to the default alias. In a Cognos PowerPlay PowerCube, it equates to the label. For star schemas, the label is a description of the member.

- User-defined.

  User-defined attribute applies to Oracle Essbase only. Oracle Essbase enables you to specify user-defined attributes for members. You can then use calculation (calc) scripts to search for and manipulate members that have particular user-defined attribute values.

  PeopleSoft Cube Manager supports user-defined attributes. To design your own user-defined attributes, select PeopleTools, Cube Builder, Cube Manager, Attribute Definitions.

Oracle Essbase Properties

Oracle Essbase has the following valid property types:

- Currency category.

  Currency category applies to account-type dimensions. This attribute specifies a member that requires currency conversion to a specific category type. In the attribute query field, supply the type of conversion that is required (a value, normally in dollars).

- Currency conversion type.

  Currency conversion type applies to account dimensions. This attribute equates to the Currency Conversion buttons on the Account tab of the Attributes dialog box for an account dimension in the Oracle Essbase Application Manager.

  If the Currency Category attribute is set to a non-blank value on a member, that non-blank value is automatically applied to the Currency Conversion Type attribute. If the Currency Category attribute is set to blank, the Currency Conversion Type attribute is automatically set to inherit-use ancestor. If the Currency Category attribute is not applied at all, the Currency Conversion Type attribute is automatically applied with a value of no conversion ("").

  If you set this attribute manually, valid values are blank (no conversion) or non-blank. If the query returns a non-blank value, that value is used as the currency category.

- Currency name.

  Currency name applies to country dimensions. The value of this attribute defines what type of currency the country or market region uses. This value identifies the type of currency in a currency cube.

- Data storage.

  Data storage enables Essbase to recognize what type of storage to allocate for the member. Valid values are *0* or blank (store data), *1* (never share), *2* (label only), *3* (shared member), *4* (dynamic calculation and store), and *5* (dynamic calculation, no store).

  PeopleSoft Cube Manager sets the default value as store data for all members in the first rollup and the non-detail nodes of all other rollups. Detail nodes in secondary rollups are set to shared members.

- Expense item.

  Expense item applies to account dimensions only. Essbase has certain built-in formulas that can take advantage of the knowledge that an item is an expense. To pass this knowledge on to Essbase, you should use this property. Valid values are *Blank* (set) and *non-Blank* (do not set).

- Time balance.

  Time balance affects how the parent time value is calculated. Valid values are *0, 1,2,* and *3,* which correspond to *none, first, last,* and *average,* respectively.

- Consolidation.

  This property enables you to define the mathematical operator used for rolling up members. Most often, you expect that data is added (using the + operator) when rolled up. However, you might occasionally need to specify other operators, such as those listed in the following table:

Valid Value	*Action*
+ (plus sign)	Add (default)
– (minus sign)	Subtract
* (asterisk)	Multiply
/ (forward slash)	Divide
[Blank]	Do not consolidate.
~ (tilde)	Do not consolidate.
% (percent sign)	Divide the total of previous member calculations by this member and multiply by 100.

Cognos PowerPlay Attributes

Cognos PowerPlay has the following valid attribute types:

- Long description.

  Long description equates to any valid description in Cognos PowerPlay, meaning that it can contain unlimited text.

- Short label.

  Short label equates to the short name. Valid values are any valid Cognos PowerPlay short name.

See Also

Oracle Essbase and Cognos PowerPlay Services documentation.

Data Source Queries

Data source queries define the data that you bring into the cube. Writing a data source query is straightforward; the query must return one column for each dimension and one column for each measure. Assume that you want to build a data source query for a cube containing amounts that are dimensioned by account, department, and period.

See *PeopleTools 8.51 PeopleBook: PeopleSoft Cube Builder*, "Designing Cube Metadata," Using Data Source Queries.

Chapter 4

Designing Cubes

This chapter discusses how to:

- Design a cube.

- Design cube definitions.

- Design attribute names.

Designing a Cube

This section discusses how to:

- Define the dimensional structures of the cube.

- Define dimension options.

- Define rollup options.

- Specify the source of the metadata for each rollup.

- Map fields to a role in the rollup.

Pages Used to Design a Cube

Page Name	Definition Name	Navigation	Usage
Dimension	DIMENSION	PeopleTools, Cube Builder, Cube Manager, Dimensions	Define the structure of each dimension, and define the PowerPlay time dimension.

Page Name	*Definition Name*	*Navigation*	*Usage*
Dimension Options	DIM_OPTIONS	• PeopleTools, Cube Builder, Cube Manager, Dimensions Add a new dimension or search for an existing one, and click the Dimension Options link on the Dimension page. • PeopleTools, Cube Builder, Cube Manager, Dimensions Select the Rollup Inputs tab, and then click the Dimension Options link on the Rollup Inputs page.	Name blank members of a hierarchy so that you can see them in your cube.
Rollup Options	DIM_ROLLUP_OPTIONS	• PeopleTools, Cube Builder, Cube Manager, Dimensions Add a new dimension or search for an existing one, and click the Rollup Options link on the Dimension page. • PeopleTools, Cube Builder, Cube Manager, Dimensions Select the Rollup Inputs tab, and then click the Rollup Options link on the Rollup Inputs page.	Name blank members of a hierarchy so that you can see them in your cube.
Rollup Inputs	DIM_ROLLUP_INPUTS	PeopleTools, Cube Builder, Cube Manager, Dimensions Add a new dimension or search for an existing one. Select the Rollup Inputs tab.	Define the source of the metadata for each rollup.
Dim Input Field (dimension input field)	DIM_INPUT_FLD	PeopleTools, Cube Builder, Cube Manager, Dimensions Add a new dimension or search for an existing one. Select the Rollup Inputs tab, and then click the Field Map link on the Rollup Inputs page.	Map the fields to a role in the rollup.

Defining the Dimensional Structures of the Cube

To define the dimensional structures of the cube:

1. Select PeopleTools, Cube Builder, Cube Manager, Dimensions.

2. Create a new dimension, or search for an existing one.

   The Dimension page appears.

Dimension page

Dimension ID	Identifies the dimension of the target Online Transaction Processing (OLAP) database.
	Note. You cannot rename or delete a dimension ID after you save it.
Description	Appears by default for the dimension, but you can provide another description.
	This is used as the description of the dimension in the resulting cube.
Dimension Options	Click to access the Dimension Options page, where you can name the blank members of a hierarchy so that you can see those members in your cube.
	See Chapter 4, "Designing Cubes," Defining Dimension Options, page 32.

Dim Type (dimension type)	Select one of the following dimension types:
	Standard: Select this option for every kind of dimension that does not contain account, time, currency, or country data. You can have many standard dimensions in each cube.
	Account: Select this option if the dimension is based on account information. You can have only one account dimension for each cube.
	Time: Select this option if the dimension is time-based. You can have only one time dimension for each cube.
	Currency Partition: This option is valid only for Oracle Essbase. Select this option if the dimension is based on currency. You can have only one currency dimension for each cube.
	Country: This option is valid only for Oracle Essbase. Select this option if the dimension is based on countries. You can have only one country dimension in each cube.
	See your Oracle Essbase documentation.
Rollup Name	The default rollup name is based on the dimension ID. You can change the rollup name, but the name cannot be identical to the dimension ID.
	The rollup name becomes the top node of the dimension if you specify more than one rollup in the dimension. If you have multiple rollups rolling up to a single top node, this name is used only for the alternate rollups. A node is created for each rollup as a child of the dimension's top node.
Rollup Options	Click to access the Rollup Options page, where you can name the blank members of a hierarchy so that you can see those blank members in your cube.
	See Chapter 4, "Designing Cubes," Defining Rollup Options, page 34.
Level Name (All Platforms)	This is a required field.
	To explicitly define level names in the rollup, specify them in the Rollup Levels section. Level names defined in this manner override the level names that might be returned from the tree or any names that Cube Manager creates as defaults. You must specify levels to be used in mapping wide queries or for defining aggregates.
	When you enter multiple level names, Cube Manager assigns a level number to the declared level name in the order in which the level name is entered. This number is also entered in the Dim Input Field page.

Designing a Dimension

The structure of a dimension is based on a PeopleSoft tree that you build, but dimensions can also be based on queries. You must create one dimension definition for each dimension of your cube.

Because dimensions are independent of cubes, you can use them in any number of cube definitions. Where possible, such as in star schemas, Cube Manager creates dimensions that can be shared across conforming dimensions (unless the underlying data relies on different business units). This method ensures consistent results across different data marts.

To design a dimension:

1. Select PeopleTools, Cube Builder, Cube Manager, Dimensions to access the Dimension page.

2. Select the Add a New Value tab.

3. Enter a value for the Dimension/Measure/Attribute field, and click the Add button.

   You must provide a unique name for each dimension that you define. The dimension ID uniquely
   identifies the dimension of the target OLAP database. After you enter a dimension ID, the Dimension
   page appears.

   Warning! After you create and save a dimension ID, you cannot rename or delete it. You can change the
   description, but you cannot remove the ID from the system.

4. Enter a description and specify the dimension type.

5. Set additional parameters for the dimension by clicking the Dimension Options link.

   The Dimension Options page appears.

6. Enter the dimension options, and click the OK button.

7. Define the rollups for the dimension.

8. Define the level name or names.

9. Define additional rollup parameters by clicking the Rollup Options link.

   The Rollup Options page appears.

10. Define the sources of the rollup structure and attributes by selecting the Rollup Inputs tab.

    The Rollup Inputs page appears.

    Note. The Rollups section on the Dimension page determines which rollup appears on the Rollup Inputs
    page.

Defining PowerPlay Time Dimensions

For a PowerPlay time dimension, you do not need to set a tree or query as an input source. Instead, you must
perform the following tasks:

• Identify the dimension as time.

• Define the levels.

• Set the corresponding date formats and date functions on the Dimension page.

• Select the earliest date and latest date for the time dimension by using the Cognos PowerPlay options.

See Chapter 5, "Building Cubes," Defining Cube Settings for Cognos PowerPlay, page 57.

See your Cognos PowerPlay Transformer documentation.

If you are creating a time dimension specifically for another platform, you should build the dimension as you would any other dimension, ignoring the Date Function and Format columns on the Dimension page. If you create a dimension that has both input source and the Cognos PowerPlay options, you can use it on any platform. Cube Manager ignores the unnecessary information for the specific platform and uses only what is required on that platform.

To define a PowerPlay time dimension:

1. Select PeopleTools, Cube Builder, Cube Manager, Dimensions.

2. Add a new dimension or search for an existing one.

3. On the Dimension page, specify *Time* as the dimension type.

   This value also sets the DimType in the .mdl file to *Date* when the cube is finally built, enabling the cube to use the Cognos PowerPlay date wizard.

   Note. When building PowerPlay time dimensions, you are not required to complete the fields on the Dimensions - Rollup Inputs page or the Rollup Options page (prefix and suffix information for node and details). However, to build a time definition for a platform-independent cube, you should still specify them. Cube Manager ignores the input source while building a Cognos PowerPlay cube, even if you specify the source.

4. Identify each level name with a date function.

When you create a Cognos PowerPlay cube, the levels in the hierarchy must be identified by one of the following date functions. The cube build process ignores this information if the cube is intended for another platform.

Note. Do not combine lunar calendar and standard calendar in the Date Function column when setting levels. To use both calendars, create two different time dimensions—one for the lunar calendar and one for the standard calendar.

Date functions include:

- None

- Year

- LunarYear

- Quarter

- LunarQuarter

- Month

- LunarMonth

- LunarMonth445

- LunarMonth454

- LunarMonth544

- Week

- Day

5. Identify each level name with a date format.

Specify the format—in which you want to see a year, quarter, month, and so on—from the following available formats:

Format	Description	Comments
YY	Two-digit year	01
YYYY	Four-digit year	2001
Q	One-digit quarter indicator	1
MM	Two-digit month	01
MMM	Abbreviated month name	Jan
MMMM	Full name of month	January
DD	Two-digit day	01
DDDD	Day of week	Sunday
/, −, or space character	Separator characters	01/01/2001, 01-01-2001
Any quoted string	The quoted string	*AD* displays the abbreviation AD

6. Select the earliest and latest date options on the Cognos PowerPlay Options page.

These settings are optional. The default values are *1901-01-01* for Earliest Date and *2100-12-31* for Latest Date. The Current Date option is controlled by settings in Cognos PowerPlay. The As of Date option instructs the system to use the as of date that is specified on the run control page.

7. Set up a fact query to return the lowest level of detail.

Defining Dimension Options

Use these steps to access the Dimension Options page:

1. Select PeopleTools, Cube Builder, Cube Manager, Dimensions.

2. Add a new dimension or search for an existing one.

3. Click the Dimension Options link on the Dimension page.

    Alternatively, select the Rollup Inputs tab, and then click the Dimension Options link on the Rollup Inputs page.

Dimension Options page

Not all data in the cube is associated with every dimension. For example, some data might be related to an account, but not to a product. In that case, the product dimension has blank members for that data. This means that an additional implied valid value of the product dimension exists that might not be on the product tree—blank (or No Product). You must provide a name to this blank member and make sure that it is on your dimension hierarchy because every valid member of a dimension must have a unique, nonblank name.

You can either add the unique, nonblank name to your tree or query and then specify it in the Dimension Options page, or just enter a name and let the Cube Manager add it to your dimension for you. Then, when the Cube Manager is building the cube, it creates a new node directly off the top node of the dimension with the blank member name that you specify. All blank members are grouped under this node name for this dimension. However, the name of the blank member does not accumulate cube data into a single-dimension member.

Blank Member	The default name is based on the dimension name. Enter a name for each dimension that might have data that is not already associated with a named hierarchy.
	The default value is a name based on the dimension name that you create. Change this name if necessary. However, remember that different dimensions within a single cube must have unique names for blank members. Thus, if a cube has a Department dimension and a Product dimension, and both of these dimensions have blank members, the blank member names must be different—for example, [No DEPARTMENT] and [No PRODUCT].
	Some dimensions, such as the Time dimension, may not have blank members. If you are sure that a dimension falls into this category, you can delete the default blank member value and leave the field blank. If you do this and a blank member is found in the dimension, it is excluded from the dimension.
	Note. Your blank member name can match a node or detail value elsewhere in the cube structure. This enables you to insert blank members at a level other than directly below the current dimension.

Label Prefix/Suffix	This is a required field. The default value is *None*.

Both Oracle Essbase and Cognos PowerPlay require unique member names and labels. You can apply a unique prefix or suffix to each member label to fulfill this requirement. To do so, select either the *Prefix* or *Suffix* option from the Label Prefix/Suffix drop-down list box. To bypass this option, select the *None* option. |

Defining Rollup Options

Use these steps to access the Rollup Options page:

1. Select PeopleTools, Cube Builder, Cube Manager, Dimensions.

2. Add a new dimension or search for an existing one.

3. Click the Rollup Options link on the Dimension page.

    Alternatively, select the Rollup Inputs tab, and then click the Rollup Options link on the Rollup Inputs page.

Rollup Options page

Type	In the Node Prefix/Suffix section, set prefix and suffix preferences at the node level. You can select *Prefix, Suffix,* or *None*. The prefix or suffix that you specify is applied to all members in the rollup, except the lowest-level members, providing a convenient mechanism for ensuring that all node members have unique names between rollups.

In the Detail Prefix/Suffix section, set prefix and suffix preferences at the detail level. As with nodes, select *Prefix, Suffix,* or *None.*. This value is valid only in the first rollup of the dimension. |

Style   Select to give your prefix or suffix no style, a custom style, or a dimension or level name. If you select *Dimension* or *Level,* the text is either the name of the dimension in the Description field on the Dimension page, or the level name that you specified in the Rollup Levels section on the Dimension page.

Text   Enter the text to be used for the prefix or suffix, if applicable. This field is editable only if you have given your prefix or suffix a custom style.

Specifying the Source of the Metadata for Each Rollup

Use these steps to access the Rollup Inputs page:

1. Select PeopleTools, Cube Builder, Cube Manager, Dimensions.

2. Add a new dimension or search for an existing one.

3. Select the Rollup Inputs tab.

Rollup Inputs page

Src Type (source type)	This is a required field. Select one option from the following options:
	Query: Specify how the hierarchical relationships, as well as member attributes, are defined.
	Tree: Specify how the hierarchical relationships, as well as member attributes, are defined.
	Note. Cube Manager currently supports only trees that are keyed by setID or no additional key. For example, Cube Manager does not support Business Unit (BU) or User Defined Node Key (UKV) trees.
Input Name	Enter the name of the tree or query to be used to define the dimension.
	If it is available, click the Lookup Input Name button to search for existing input names.
Top Node	Prepopulated with the top node from the tree based on the input name that you selected.
	If it is available, click the Lookup Top Node button to search for the existing top node.
Levels From Top	Specify the number of levels from the top node (including the top node) to the *stop level* that corresponds to the lowest level of the tree to be included in the rollup.
	If it is available, click the Lookup Levels From Top button to search for existing levels from the top node.
	See the Defining the Levels From Top Example section subsequently.
Field Map	Available only when the source type is a query.
	Click the Field Map link to access the Dim Input Field (Dimension Input Field) page, where you can map the field.
	See Chapter 4, "Designing Cubes," Mapping Fields to a Role in the Rollup, page 38.
Mapped	This option is selected when the input name is mapped; if the field is not mapped, this option is cleared.

Defining the Levels from Top Example

Use the following example for calculating the number of levels from top:

Definition	*Description*
Source Input	QE_PERS_DATA tree
Top Node	10100 Office of President

Definition	*Description*
Top Node Level	Company
Stop Level	Department

This is an example of Top Node Level:

Top Node Level

The Stop Level is *Department,* so the system calculates the Levels From Top value by counting the number of levels from Company (1) to Department. Using the following list of levels, the system calculates the Levels From Top value as three.

Example of organization levels for calculating Levels from Top

Adding Rollup Inputs

After you have defined the high-level dimensional metadata (identifying the dimension name, defining the number of different rollups, defining prefix and suffix options, and so on), you define the source of each rollup's metadata.

If multiple rollups exist for an Essbase dimension, only the total from the first rollup for the dimension is aggregated to the dimension total.

Note. Rollup level names that you enter on the Rollup Inputs page appear in an Essbase cube only if the cube contains a dimension with a multiple rollup. Rollup level names are always used in Cognos PowerPlay.

To add rollup inputs:

1. Select PeopleTools, Cube Builder, Cube Manager, Dimensions.

2. Add a new dimension or search for an existing one.

3. Select the Rollup Inputs tab.

    The Rollup Inputs page appears.

4. Define cube inputs, such as queries or trees, and their associated field mappings.

5. In the Src Type field, select the source input.

6. In the Input Name field, click the Lookup button to display and select from the valid values.

7. If you selected the *Tree* option as the source type, specify a value for the Top Node field by clicking the Lookup button and specifying a valid value.

8. If you selected the *Tree* option as the source type, select a value for the Levels from Top field by clicking the Lookup button and selecting a valid value from the list.

9. If you selected the *Query* option as the source type, define how each field in the result set is used in the rollup.

Mapping Fields to a Role in the Rollup

Use these steps to access the Dim Input Field (Dimension Input Field) page:

1. Select PeopleTools, Cube Builder, Cube Manager, Dimensions.

2. Add a new dimension or search for an existing one.

3. Select the Rollup Inputs tab.

4. Click the Field Map link on the Rollup Inputs page.

   The Dim Input Field page appears.

Dim Input Field page

Query Fld Name (query field name)   This value is based on the column names of the query.

Role	Select one of the following options:
	Attribute: If you select this option, you must also select the attribute name.
	Member: If all sources are members, select the level name.
	Parent: If a parent exists, you cannot select level names or numbers. The parent is the top of the dimension, and the member is the child of the parent.
Level Name	This field is available only if all roles are set to *Member.*
	Click the field to display a list of valid level names, and select one of the names that is specified on the Dimension page.
Seq (sequence)	Automatically assigned to the level name when the level name is declared on the Dimension page.
Attribute Name	This field is available only if the Role field is set to *Attribute.*
	Add to or change the values that are available in the drop-down list box on the Attribute Name page.

Mapping Fields to a Role

Now that you have created the high-level rollup information, you must map each of the fields in the query-based rollup inputs to a role.

To define the fields that you want to map from the input to the cube:

1. Select PeopleTools, Cube Builder, Cube Manager, Dimensions.

2. Add a new dimension or search for an existing one.

3. Select the Rollup Inputs tab.

4. Click the Field Map link on the Rollup Inputs page.

   The Dim Input Field page appears with the source query fields already completed.

5. In the Input Field Map section, specify the fields, roles, levels, and attributes of the cube dimension.

   Note. Use unique attribute names when configuring a dimension with multiple attributes. If you duplicate a name, Cube Manager populates only the first instance.

6. Click the OK button to save your changes.

   Alternatively, click the Cancel button to exit the page without saving changes.

Designing Cube Definitions

This section discusses how to:

• Map data sources to dimensions and select dimensions for the cube.

- Add cube inputs to a cube definition.

- Map source query fields to cube components.

Pages Used to Design Cube Definitions

Page Name	Definition Name	Navigation	Usage
Cube Definition	CUBE_DEF	PeopleTools, Cube Builder, Cube Manager, Cube Definitions	Map data sources to any dimension that you have defined using the Dimension page, and select which dimensions to use in the cube.
Cube Inputs	CUBE_DEF_INPUTS	PeopleTools, Cube Builder, Cube Manager, Cube Definitions Select the Cube Inputs tab.	Add cube inputs to your cube definition.
Cube Input Field	CUBE_INPUT_FLD	PeopleTools, Cube Builder, Cube Manager, Cube Definitions Select the Cube Inputs tab. Click the Field Map button on the Cube Inputs page.	Map the fields of the source query to the appropriate cube components to associate the data from the source query with a dimension. The cube inputs are the source values for the facts to be created in the OLAP database.

Mapping Data Sources to Dimensions and Selecting Dimensions for the Cube

Select PeopleTools, Cube Builder, Cube Manager, Cube Definitions to access the Cube Definition page.

Cube Definition page

Description	Enter the description that you want to associate with the cube.
	The description is not transferred to the resulting cube definition; it is used for prompting within Cube Manager only.
Long Description	Enter the long description that you want to associate with the cube.
	This description is not transferred to the resulting cube definition; it is used for prompting within Cube Manager only.
Role	This is a required field.
	Select one of the following options:
	Cube Attribute: Select this option if the component is an attribute. A cube attribute is a field that can be included in the fact table for external processing. The Bookmark field on the fact table is an example of a cube attribute. You cannot use this bookmark in analysis, but Cube Manager uses it to manage facts.
	Dimension: Select this option if the component is a dimension.
	Measure: Select this option if the component is a measure or fact.
Dimension/Measure/Attribute	Specify the dimension, measure, or attribute that you want to associate with the cube.
📂	This icon is available only if you selected the *Dimension* option in the Dimension/Measure/Attribute field.
	Click the Open icon to display the Cube Input Field page.

Dim Type (dimension type)	This field is read-only.
	If you selected the *Dimension* option in the Dimension/Measure/Attribute field, Cube Manager displays the value that you entered for the Dim Type field on the Dimension page.
Status	This field is read-only.
	The value that Cube Manager displays is based on the information that you entered on the Dim Input Field page.

Designing a Cube Definition

After creating data source queries and defining the dimensions, you must define a cube definition by mapping fields from the data source queries to the dimensions you have defined. For each dimension, you must specify which field in the data source query is the source field for the dimension. You must define the dimensions, measures, and attributes that Cube Manager uses to pull data into the cube that you are defining. When you define a component, its dimension and rollup types appear, as well as the mapped status.

To design a cube definition:

1. Select PeopleTools, Cube Builder, Cube Manager, Cube Definitions.

2. Select the Add a New Value tab.

3. Enter a cube definition ID, and click the Add button.

    Provide a unique name for each cube definition that you define. After you enter a cube definition ID, the Cube Definition page appears.

4. Enter a description and a long description for the cube definition.

5. In the Components section, define the role of the cube definition—*Cube Attribute,Dimension,* or *Measure.*

6. Specify a dimension, measure, or attribute to be used in the cube.

    Because you can reuse the same dimension for various cubes, you can rename dimensions for use with this particular cube definition. By default, the label that you enter here corresponds to the name of the dimension that you have specified and is used as the dimension's alias or label in the resulting cube. You might want to put the word *ALL* in front of the dimension name—ALL PRODUCTS, ALL REGIONS, and so on.

    The number to the left of the Role column determines the order in which the dimensions and measures appear to the user in the OLAP tool. When you add dimensions and measures, this number increments automatically.

7. Click the Add button to add additional dimensions and measures.

    For each additional dimension or measure, repeat the previous two steps.

8. Click the Save button to save your cube definition and all of the components that you have added.

    Saving your new cube definition at this stage is important so that you can effectively add cube inputs.

Adding Cube Inputs to a Cube Definition

Use these steps to access the Cube Inputs page:

1. Select PeopleTools, Cube Builder, Cube Manager, Cube Definitions.

2. Add a new cube definition or search for an existing one.

3. Select the Cube Inputs tab.

Cube Inputs page

Input Name	Add the name of inputs by selecting one value from the list of valid values.
	Click the Add button to insert new rows for additional input names.
	Specify as many input sources as you want; however, you must completely map all of them to save the cube definition.
Field Map	Click to view how each field in the fact source data is mapped to a dimension.
Mapped	The system automatically selects the Mapped option after you map source query using the Cube Input Field page. This option indicates whether fields have been mapped for this cube input.

Mapping Source Query Fields to Cube Components

Use these steps to access the Cube Input Field page:

1. Select PeopleTools, Cube Builder, Cube Manager, Cube Definitions.

2. Add a new cube definition or search for an existing one.

3. Select the Cube Inputs tab.

4. Click the Field Map button on the Cube Inputs page.

Cube Input Field page

Query Fld Name (query field name)	This field is populated with data from the Cube Inputs page.
Dimension/Measure/Attribute	Specify the dimension, measure, or attribute to which you want to map the corresponding source field.
Role	Indicates whether a field has been mapped to a dimension, measure, or attribute.
OK	Click to save your changes. The system automatically selects the Mapped option on the Cube Inputs page. The Mapped option indicates whether fields have been mapped for this cube input.

Designing Attribute Names

You can specify member attributes by using the Attribute Name page. This feature enables you to specify user-defined attributes for members. Users can also set these attributes in Oracle Essbase and Star Schema platforms. You can then use calculation scripts to search for and manipulate members with particular user-defined attribute values. PeopleSoft workbenches use calculation scripts extensively.

See Also

PeopleTools 8.51 PeopleBook: PeopleSoft Cube Builder, "Designing Cube Metadata," Understanding Oracle Essbase Properties

Specifying User-Defined Attributes for Members

Select PeopleTools, Cube Builder, Cube Manager, Attribute Definitions to access the Attribute Name page (OLAP_ATTRIB_NAM).

Attribute Name page

Attribute Number	If you are working with large numbers of attributes, this field enables you to group the attributes into more manageable categories.
	For example, you might want to list human resource attributes in the 100–300 range, and finance attributes in the 400–600 range. The attribute number is not used anywhere else in the system.
Attribute Name	Enter a descriptive name for the attribute.

Chapter 5

Building Cubes

This chapter provides an overview of cube instance definitions and discusses how to:

*   Build cube instances.

*   Build cubes.

*   Use API and command-line specifications.

Understanding Cube Instance Definitions

Cube Manager provides a powerful template feature that enables you to set up cube instance definitions for use and reuse when building and updating cubes. Having a cube instance definition simplifies your work in managing updates to the cube structure, loading and reloading data, and setting platform-specific options when you create the cube. Without the benefit of Cube Manager—and the Cube Instance Definition feature in particular—building or updating a cube would require tedious and error-prone steps.

Each cube instance definition is associated with a single physical cube. You typically have one cube instance definition for each cube. At times you might find having more than one useful. For example, you might have one cube instance definition for Cognos PowerPlay and one for Oracle Essbase.

Up to this chapter, everything that this PeopleBook has described for creating a cube is more or less generic, and does not require you to specify the platform (Cognos PowerPlay, Oracle Essbase, or Star Schema) that you will ultimately use. Thus, if you decide to switch platforms, the majority of your cube design remains unchanged. When building the cube instance definition, however, you must designate the specific platform for which you are creating the cube.

Building Cube Instances

This section discusses how to:

*   Build a cube instance definition.

*   Define cube settings and select dimensions, measures, and attributes.

*   Create the cube filter.

*   Define cube settings for Oracle Essbase.

*   Define cube settings for Cognos PowerPlay.

• Add a prefix to field and record objects.

Note. The definition of a cube instance creates the link between the platform-independent definition of the cube and the actual physical storage mechanism of the cube. In essence, you set up a location that Cube Manager uses to create the cube that you have defined.

Pages Used to Build Cube Instances

Page Name	Definition Name	Navigation	Usage
Cube Inst Def (Cube Instance Definition)	ANALYSIS_DB	PeopleTools, Cube Builder, Cube Manager, Cube Instances Create a new cube instance or search for an existing one.	Define cube settings (such as platform), and select dimensions, measures, and attributes to be included as cube instance definition components.
Cube Filter Criteria	CUBE_FILTER	PeopleTools, Cube Builder, Cube Manager, Cube Instances Create a new cube instance or search for an existing one. On the Cube Inst Def page, select the Filter subtab, and click the Updt Filter (update filter) button.	Create the cube filter for use on the Filters tab of the Cube Inst Def page.
Essbase Advanced Options	ANALYSIS_DB_ESS	PeopleTools, Cube Builder, Cube Manager, Cube Instances Create a new cube instance or search for an existing one. On the Cube Inst Def page, select the *Oracle Essbase* option from the Platform list, and click the Platform Options link.	Define cube settings specifically for Essbase.
PowerPlay Options	ANALYSIS_DB_PPL	PeopleTools, Cube Builder, Cube Manager, Cube Instances Create a new cube instance or search for an existing one. On the Cube Inst Def page, select the *Cognos PowerPlay* option from the Platform list, and click the Platform Options link.	Define cube settings specifically for Cognos PowerPlay.

Page Name	Definition Name	Navigation	Usage
Star Schema Options	ANALYSIS_DB_SS	PeopleTools, Cube Builder, Cube Manager, Cube Instances Create a new cube instance or search for an existing one. On the Cube Inst Def page, select the *Generic Star Schema* option from the Platform list, and click the Platform Options link.	Identify the field and record objects generated by Cube Manager.

Building a Cube Instance Definition

To build a cube instance definition:

1. Design the cube definition.

   See Chapter 4, "Designing Cubes," page 25.

2. Select PeopleTools, Cube Builder, Cube Manager, Cube Instances.

3. Select the Add a New Value tab.

4. Enter a cube instance ID and click the Add button to define the cube platform and included components.

   You must provide a unique name for each cube instance definition that you define. After you enter a cube instance ID, the Cube Inst Def page appears.

5. Enter a description and long description.

6. Specify the platform.

7. Specify the criteria for this cube instance definition: either a cube definition or an existing star schema-based cube instance.

8. Remove any dimensions or measures that you do not want to be created in this cube instance.

   By default, Cube Manager adds all of the dimensions and measures to the cube instance definition. However, if you do not need them all, you can remove unwanted dimensions or measures by clicking the Remove (-) button to the right of the dimension name.

9. Click the Platform Options link to display platform-specific options, based on which platform you selected from the drop-down list box.

   Which page appear depends on which platform you select. Platform options are:

   • *Oracle Essbase*

     See Chapter 5, "Building Cubes," Defining Cube Settings for Oracle Essbase, page 53.

   • *Cognos PowerPlay*

     See Chapter 5, "Building Cubes," Defining Cube Settings for Cognos PowerPlay, page 57.

   • *Generic Star Schema*

     See Chapter 5, "Building Cubes," Adding a Prefix to Field and Record Objects, page 61.

Defining Cube Settings and Selecting Dimensions, Measures, and Attributes

Access the Cube Inst Def page by selecting PeopleTools, Cube Builder, Cube Manager, Cube Instances.

Cube Inst Def page: Dimension tab

Common Page Information

Cube Instance ID   Enter a generic name to identify the cube instance.

Description	Enter a description to be used for prompting within Cube Manager only.
	This description is not passed to the resulting cube.
Long Description	Enter a description to be used for prompting within Cube Manager only.
	This description is not passed to the resulting cube.
Platform	Select one of the following options: *Oracle Essbase, Cognos PowerPlay,* or *Generic Star Schema.*
Platform Options	Click to access platform pages, where you can define cube settings for Essbase, Cognos PowerPlay, or star schema.
	See Chapter 5, "Building Cubes," Defining Cube Settings for Oracle Essbase, page 53.
	See Chapter 5, "Building Cubes," Defining Cube Settings for Cognos PowerPlay, page 57.
	See Chapter 5, "Building Cubes," Adding a Prefix to Field and Record Objects, page 61.
Based On	Select one of the following options:
	Cube Definition: Refers to the cube definition ID that you specified when designing your cube definition.
	Cube Instance: Refers to a physical star schema that was created based on a particular cube definition. The value is the cube definition ID that was used to create the actual star schema. You can use this schema to create smaller data marts based on a previously created star schema.
	Click the Open button to open the selected cube definition.

Dimension Tab

Dimension ID	Enter the name of the dimension or measure to include in the cube.
Role	Displays the role of the dimension, measure, or attribute.
Dim Type (dimension type)	Displays the dimension type that is specified on the Dimension page.
	See Chapter 4, "Designing Cubes," Defining the Dimensional Structures of the Cube, page 27.
Status	Displays whether your dimension has been mapped.

Note. If your dimension has not been mapped, you must map it before you can use the dimension in a cube.

Sparsity   Appears specific to Essbase and refers to the Essbase option page.

See <u>Chapter 5, "Building Cubes," Defining Cube Settings for Oracle Essbase, page 53.</u>

Filter Tab

**Updt Filter** (update filter) Click to create the cube filter criteria.

Filter   Indicates whether a filter has been created for this dimension.

Label   Indicates how the dimensions label appears in the results.

Creating the Cube Filter

Use these steps to access the Cube Filter Criteria page:

1. Select PeopleTools, Cube Builder, Cube Manager, Cube Instance.

2. Create a new cube instance or search for an existing one.

3. Select the Filter subtab, and click the Updt Filter (update filter) button.
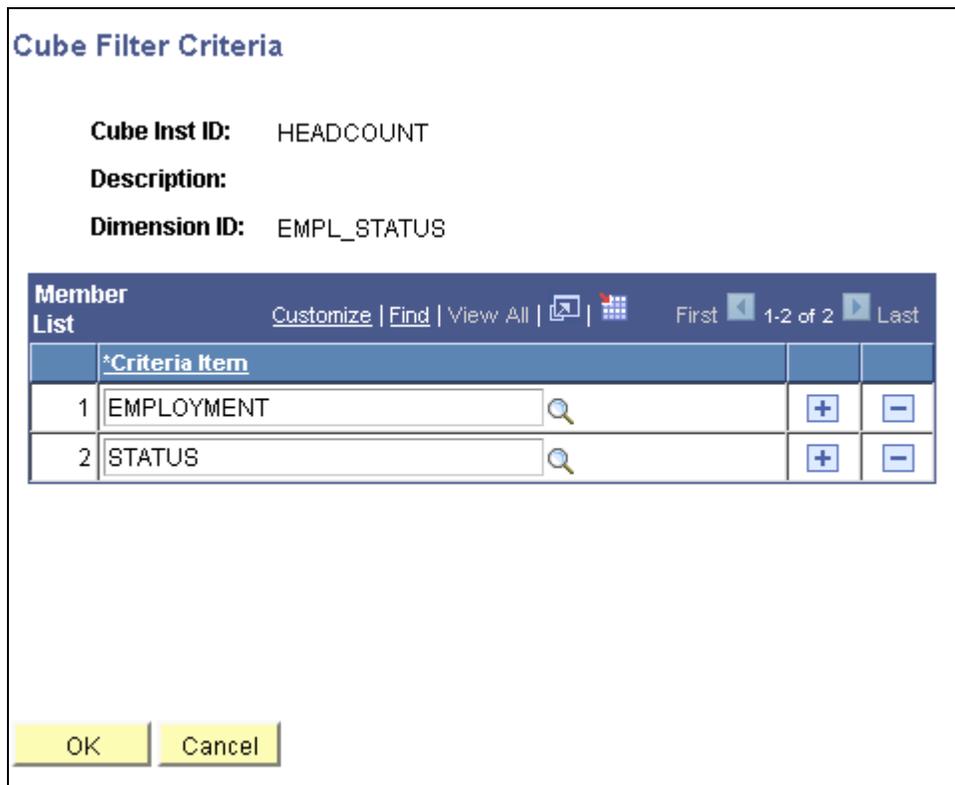
Cube Filter Criteria page

Member Value	Enter the value for the criteria.
	If your source dimension has more information than you want to report, you can create a filter to eliminate selected information. For example, if you have data from the United States, United Kingdom, and Japan, but want to view only the data for Japan, create a Japan filter to eliminate the United States and United Kingdom data from the cube dimension.

Defining Cube Settings for Oracle Essbase

Use these steps to access the Essbase Advanced Options page:

1. Select PeopleTools, Cube Builder, Cube Manager, Cube Instances.

2. Create a new cube instance or search for an existing one.

3. From the Cube Inst Def page, select the *Oracle Essbase* option from the Platform list.

4. Click the Platform Options link.

Essbase Advanced Options page

Server	Enter your server ID address or machine name.
	See <u>Chapter 5, "Building Cubes," Defining Cube Settings for Oracle Essbase, page 53.</u>
Application	This is a required field.
	Enter the application name. In Essbase, an application is a group of one or more databases.
Database	This is a required field.
	Enter the database name. In Essbase, a database is essentially an Online Analytical Processing (OLAP) cube.
User Access	This is a required field.
	Set the access to *Calculate, Design, Read-Only, Read-Write,* or *None.*
	This setting sets the default user security on the resulting cube if explicit security has not been granted in Essbase. It applies to everyone except the system administrator (to whom the User Name and Password options apply). Oracle recommends that you set at least *Read-Only* access.
User Name	Enter your user name.
Password	Enter your password.
	Note. This password should be same as the password that you use to set up the Essbase server.
Meta-Data Update Action	This is a required field. It enables you to specify the update action that should occur in Essbase when a cube structure has been redesigned in Cube Manager. The options are:
	Replace All: Refreshes the structure of the cube. Any members defined in the PeopleSoft application that don't exist in the cube are added to the cube. Any members in the cube that are no longer defined in the PeopleSoft application are deleted from the cube, and any associated data is lost. Members moved in the PeopleSoft application—from one node to another, for example—are moved in the cube with any associated data preserved.
	Incremental Update: Merges the metadata defined in your cube definition with the existing metadata of the cube. No members are deleted.

Rate Cube Action	If the cube definition contains the appropriate dimensions (account, time, and country), you can create a rate cube. This setting enables you to specify the following rate cube creation options:
	Create All: Regenerates the currency cube definition from the cube that is defined in Cube Manager. The system then populates the resulting currency rate cube with the data that is defined in the Rate Cube Mapping section, if any mapping exists.
	Rate Update Only: Replaces the existing data in the currency rate cube without regenerating the outline.
	No Action: Does not perform any action on the currency rate cube.
Run Default Calc (run default calculation)	Select to enable the system to run the default calculation script in Essbase when the cube is created. Essbase aggregates only the tree in the given rollups.
	Clear this option to create a custom calculation script.
Auto Config Sparsity (automatically configure sparsity)	Clear this option to manually set the sparsity levels on the Cube Instance Definition page. After you clear this option, you can set the sparsity of each dimension in the template to either *Sparse* or *Dense.*
	Select this option to enable Essbase to automatically set sparsity levels.
	Note. Dimensions marked as *Sparse* might take up significantly more space in the resulting cube than those marked as *Dense.* If you mark every dimension as sparse, you could end up with an enormous cube that is not practical or even possible to store on even the most powerful servers. That's why the Time and Account dimensions are *Dense* by default You can change these settings in Oracle Essbase after you create your cube. Whether you change the sparsity settings from Cube Manager or from Essbase, be careful when adjusting, and make sure to refer to the Oracle Essbase documentation.
Language Code	Select to build an Essbase cube in multiple languages. When the cube is then viewed in Essbase, it appears in the preferred language of the user. If the cube is not available in the language of the user, the default language (language of the developer) is used.
	To add one additional language, select a language from the drop-down list box.
	Note. You do not have to select your own language. The cube is automatically built in your own language.
Rate Cube Name	Enables you to supply a rate cube mapping. Cube Manager generates the actual currency rate cube by default.
	Specify the cube name that you want Cube Manager to create and map to the defined cube.
Query Name	Select the query that has your currency rate definition.
Time	Select the field name for time from the time dimension.
Account	Select the field name for currency category from the account dimension.

Country	Select the field name for currency name from the country dimension.
Type	Select the field name for currency type.
Rate	Select the field name for currency rate multiplier.
Default Currency Name	Enter the default currency name.
	If you do not specify a currency name for a member in your country dimension, the default currency name applies to the member.
Default Currency Category	Enter the default currency category.
	If you do not specify a currency category for a member in your account dimension, the default currency category applies to the member.

Multicurrency Rate Cubes

Oracle Essbase supports multicurrency cubes. An Oracle Essbase multicurrency cube contains two databases: the regular main database and the currency rate database. The currency rate database is created out of the main database.

Before you can generate the currency rate database from the main database, the main database *must* have the appropriate dimensions:

- Accounts dimension contains items that you want to measure, such as profit and inventory, and makes Essbase built-in accounting functionality available.

- Time dimension defines the time period for which you report and update data.

- Country dimension contains data about where business activities take place.

You can also specify the currency partition dimension (or *Currency Dimension)* in the main database for the Essbase multicurrency cube. The currency partition dimension is for separating local currency members from a base currency that are defined in the application. If the base currency for analysis is U.S. dollars, the local currency members contain values that are based on the currency type of the region. The currency partition dimension is used only in the main database for currency conversion, using Essbase script or other tools.

The following steps outline what you specify for the account dimension and the country dimension, specifically for Essbase multicurrency cubes. You create attribute queries for the Account and Country dimensions, and specify the queries as an input source on the Rollup Inputs page.

To create a multicurrency cube in Essbase:

1. Select PeopleTools, Cube Builder, Cube Manager, Dimensions, Rollup Inputs.

2. Click the Field Map link for the appropriate input name.

3. Specify the currency category in your accounts dimension, for example *P&L* (for profit and loss).

4. Specify the currency name in your country dimension, for example *$US.*

Defining Cube Settings for Cognos PowerPlay

Use these steps to access the PowerPlay Options page:

1. Select PeopleTools, Cube Builder, Cube Manager, Cube Instances.

2. Create a new cube instance or search for an existing one.

3. From the Cube Inst Def page, select the *Cognos PowerPlay* option from the Platform list.

4. Click the Platform Options link.

PowerPlay Options page

|	
PowerCube Path/File (.MDC)	Specify the path and file name.
	Cognos PowerPlay requires both a PowerCube file and a model file. The PowerCube file has an .mdc extension and equates to a cube.
	Note. Cognos PowerPlay refers to a multidimensional database as a PowerCube, so we use this term in place of *cube* when referring specifically to the resulting cube in Cognos PowerPlay.
Model Path/File (.MDL)	Specify the path and file name. The model file has an .mdl filename extension and represents the cube structure.
Data Working Directory	Specify where the system writes the .dat file.
Language Override	If you are implementing the PowerCube in a language other than English, select a language.
Override Windows Registry	Select to override the location of the Cognos binaries in registry.
Cognos Binaries	This section displays the locations of the registry location where the binaries folders are located.
Always Include	Select to retain the category in the model and include it in cubes, even if it—or any of its descendants—fails to appear in the data source.
	The category is also included if any of its descendants are included and if the category has not been excluded, summarized, or cloaked in a dimension view. If necessary, the transformer includes ancestors of the category, regardless of their inclusion settings.
	Note. The Always Include option is selected when you create a time dimension. If you exclude categories, special categories that use relative time concepts (such as Last Month) do not work properly.
Earliest Date	The fields in this section are optional.
	Use fields in this section when designing and creating time dimensions for Cognos PowerPlay.
	If these fields are not specified, the system supplies a default time of *1901-01-01*.
	Available options are:
	Use As of Date: Select to use the As of Date option that is provided on the run control page.
	Use Current Date: Select to use the current system date as controlled by settings in Cognos PowerPlay.
	Use Specific Date: Select to set a specific date. If this field is not specified, the default is *1901-01-01*.

Latest Date	The fields in this section are optional.
	Use fields in this section when designing and creating time dimensions for Cognos PowerPlay. If these fields are not specified, the system supplies a default time of *2100-12-31.* Available options are:
	Use As of Date: Select to use the As of Date option that is provided on the run control page.
	Use Current Date: Select to use the current system date as controlled by settings in Cognos PowerPlay.
	Use Specific Date: Select to set a specific date. If this field is not specified, the default is *2100-12-31.*
Enterprise Server	Select to allow Cube Manager to register your resulting cube automatically on the Cognos PowerPlay enterprise server.
	If you do not select the Enterprise Server option, Cube Manager creates the cube by using the transformer.
Service Port	The default value for the Service Port field is *8010.* This value should be the correct port for most machines.
	PowerPlay Enterprise Server is set to communicate on port number *8010* for both Windows and web connections.
	Note. Cube Manager assumes that the enterprise server is running locally, and it retrieves the local machine name from the system to connect to the enterprise server.
Logon User Name	Enter the logon user name to use to connect to PowerPlay server.
	Note. By default, the logon user name is Administrator.
Logon Password	Enter a password to use to authenticate the PSOLAP connection to Cognos Enterprise Server.
	Note. If this field is blank, Cognos Enterprise Server authenticates the connection as a blank-password logon.

Defining Cognos PowerPlay Cube Instance Definition Options

This diagram shows both the optional and the required steps that enable Cube Manager to build a Cognos PowerPlay cube.
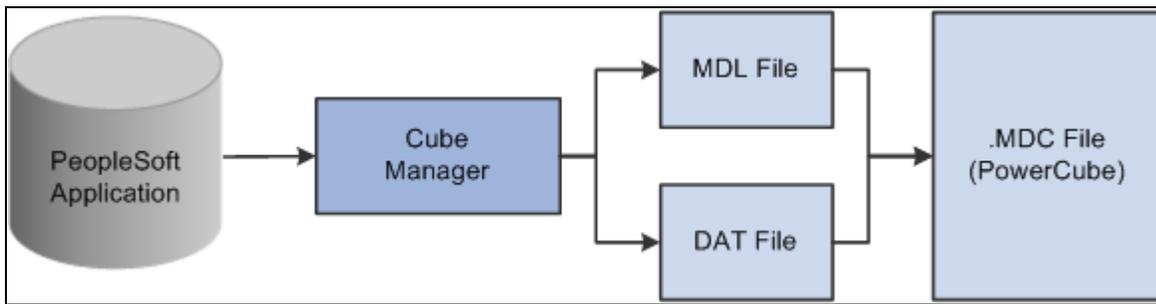
The .mdl and .dat files merge to create the PowerCube (.mdc file)

Note. Cube Manager and the enterprise server must be set up to run on the same machine.

To define Cognos PowerPlay cube instance definition options:

1. Select PeopleTools, Cube Builder, Cube Manager, Cube Instances.

2. Create a new cube instance or search for an existing one.

3. From the Cube Inst Def page, select the *Cognos PowerPlay* option from the Platform list.

4. Click the Platform Options link.

   The PowerPlay Options page appears.

5. On the PowerPlay Options page, specify the file names to be created when the cube is built.

6. Enter the data working directory.

   When building a PowerCube, Cube Manager sends output from each query used in the PowerCube to a flat data file before sending this data to the PowerCube (.mdc) file. This flat data file has a .dat extension, and is written to the data working directory that you specify.

   Note. The data file created in the data working directory is not automatically deleted or overwritten when Cube Manager builds a PowerCube. You must manually delete the files.

7. If necessary, select a language override.

8. Click the OK button if you are not using the enterprise server; otherwise, continue to the next step.

9. Select the Enterprise Server option to use the enterprise server functionality.

   The Cube Instance Definitions page appears.

Note. Cube Manager assumes that the enterprise server is running locally, and that it retrieves the local machine name from the system to connect to the enterprise server.

Cube Manager uses Ppadmtool.exe to communicate with the local enterprise server to add or remove cubes. Cube Manager first attempts to remove the cube from the enterprise server. If the cube already exists, Cube Manager removes the cube from the enterprise server. This action prevents other users from trying to access the cube through the enterprise server while Cube Manager is actually creating it. After removing the cube from the enterprise server, Cube Manager uses the transformer to generate the .mdc file.

If the cube is not listed in the enterprise server, Cube Manager uses the transformer to generate the .mdc file.

After creating the .mdc file, Cube Manager uses the Ppadmtool.exe to add the .mdc file to the enterprise server. After the cube has been added to the enterprise server, any user connected to it from a local or remote machine can view the newly added cube.

If you do not select the Enterprise Server option on the PowerPlay Options page, Cube Manager creates the cube by using the transformer.

Note. Cognos PowerPlay does not allow duplicate details in the same hierarchy, but it does allow duplicate details across hierarchies for the same dimension.

See Also

Chapter 2, "Introducing PeopleSoft Cube Manager," Cognos PowerPlay, page 10

Adding a Prefix to Field and Record Objects

Use these steps to access the Star Schema Options page:

1. Select PeopleTools, Cube Builder, Cube Manager, Cube Instances.

2. Create a new cube instance or search for an existing one.

3. From the Cube Inst Def page, select the *Generic Star Schema* option from the Platform list.

4. Click the Platform Options link.

Star Schema Options page

PeopleTools Object Name Prefix	Enter the prefix to be applied to each record and field definition that is created by Cube Manager. The prefix provides some control over the record and field names.
	The default is *DW_*.

Building Cubes

This section provides an overview of building cubes and discusses how to:

- Define the cube build specifications and bind variables.

- Define bind variables.

- Define a cube build process using Process Scheduler Manager.

- Run the process to build the cube.

Understanding Cube Creation

When you are ready to build a cube, you need a place to specify the runtime parameters to be passed during the creation or update of the data cube using Oracle PeopleSoft Process Scheduler. The Build Cube page in Cube Manager provides an example of how to pass this information from a PeopleSoft page to the resulting cube by using a PeopleSoft Process Scheduler run control. Most likely, you are building cubes using pages that were either supplied with your PeopleSoft applications or pages that you have built yourself.

You can configure the process in which a cube is built by creating your own run control pages and process definitions using the Cube Manager process type. Certain PeopleSoft applications deliver their own pages for building cubes that leverage product-specific metadata.

Note. If you are building an Essbase cube and it contains time, account, and country dimension types, Cube Manager automatically builds a currency cube, if one is defined. The calculation for the currency cube always has a default value of *Division.*

Note. After you have built an Essbase cube, Oracle does not support your changing the dimension name by changing the cube instance definition. The work-around is to access Essbase and manually change the dimension name to match the one in the cube instance. Another option is to re-create the cube with the Metadata Create and Data Create options.

Pages Used to Build Cubes

Page Name	Definition Name	Navigation	Usage
Build Cube	RUN_MAKE_ANL_DB	PeopleTools, Cube Builder, Cube Manager, Build Cubes	Define the cube build specifications and bind variables.
Bind Variables	CUBE_BUILD_BIND	PeopleTools, Cube Builder, Cube Manager, Build Cubes Create a new run control ID or search for an existing one. Select the Bind Variables tab.	Define bind variables that limit the data that is used to build a cube by prompting users for information when they start the build process.
Build Ledger Cubes	GL_OLAP_LED_REQ	Financial Application, Build Ledger Cubes	Run the background process that creates ledger cubes.

Page Name	Definition Name	Navigation	Usage
Process Definition	PRCSDEFN	PeopleTools, Process Scheduler, Processes	Define a cube build process request to run on the system.
Process Scheduler Request	PRCSRQSTDLG	PeopleTools, Cube Builder, Cube Manager, Build Cubes Create a new run control ID or search for an existing one. Enter the appropriate value in the Build Cube page, and click the Run button.	Run the process that builds the cube.

Defining the Cube Build Specifications and Bind Variables

Access the Build Cube page by selecting PeopleTools, Cube Builder, Cube Manager, Build Cubes.

Build Cube page

 Click the Open Cube Instance Definition icon to access the Cube Inst Def page, where you can view the cube instance definition with the instance definition that you specify.

See Chapter 5, "Building Cubes," Defining Cube Settings and Selecting Dimensions, Measures, and Attributes, page 50.

Meta-Data	Select an option from the following options:
	Create: Create the cube. If the cube already exists, Cube Manager re-creates it, overwriting any existing dimensions and data.
	Update: This option updates the metadata in Generic Star and the structure of the cube according to the setting for the Meta-Data Update Action field on the Cube Instance Definition page in Essbase. It has a same effect as *Create* in Cognos PowerPlay.
	None: Do not make any changes to the structure of the cube or its individual dimensions. (You may want to update the data only.)
Data	Define the data action to take place when you update the cube:
	Create: Reload the data and overwrite any existing data.
	Update: Update the existing data in the cube. This option is linked to the Data Update Action option that is selected on the Cube Instance Definition page of Essbase.
	None: Do not make any changes to the data in the cube. (You may want to update the structure only.)
Aggregation	Select one of the following options:
	Create: Reload the aggregates and overwrite any existing data.
	None: Do not make any changes to the aggregates in the cube. (You may want to update the structure only.)
Business Unit	Enables you to get the appropriate setIDs for tree metadata.
	Leave this field blank if your trees do not use set IDs.
As of Date	Enables you to get the appropriate effective dates for tree metadata.
Bookmark	Bookmarks identify a set of facts in the Generic Star Schema cube. These facts are used to determine whether facts are loaded for the first time or reloaded. The first time that you create a build cube definition, the bookmark is set to *Initial.* You can also add a build script.
Script	Refers to platform-specific, post-build scripts, such as the .mdl file for Cognos PowerPlay or any command line for Windows.
	This feature enables you to extend the capabilities of Cube Manager. For example, you can specify an Esscmnd script that sets up security. Then, rather than having to configure security manually every time that you build a cube, you can write a script once and specify it in the Script field to carry it out for each build.
Report Manager	Displays the report list in Report Manager.
	See *PeopleTools 8.51 PeopleBook: Using PeopleSoft Applications*, "Working With Processes and Reports."

Process Monitor	Displays the process list in Process Monitor.
	See *PeopleTools 8.51 PeopleBook: PeopleSoft Process Scheduler*, "Using Process Monitor."
Run	Click to display the Process Scheduler Request page.

To build a cube:

1. Build the cube instance definition.

   See Chapter 5, "Building Cubes," Building Cube Instances, page 47.

2. Select PeopleTools, Cube Builder, Cube Manager, Build Cubes.

3. If you add a new cube definition, enter a run control ID and click the Add button.

   Alternatively, if you update and display an existing cube definition, click the OK button.

4. In the Cube Specification section of the Build Cube page, select a cube instance ID.

5. Define the run control information in the Run Time Parameters section.

   Note. If you specified the default calculation script on the Cube Instance Definition page of Essbase, the default calculation script runs first. The command line is then invoked to run a subsequent script.

6. Define the metadata action that should occur during the build.

7. Define the data action that you want to occur during the build.

8. Define the aggregate action that you want to occur during the build.

9. Set your Process Scheduler parameters and save the page.

   See *PeopleTools 8.51 PeopleBook: PeopleSoft Process Scheduler.*

   Note. When working in a three-tier environment, you *must* build your cubes on a Windows Process Scheduler server, not on the client.

Defining Bind Variables

Use these steps to access the Bind Variables page:

1. Select PeopleTools, Cube Builder, Cube Manager, Build Cubes.

2. Create a new run control ID or search for an existing one.

3. Select the Bind Variables tab.

Bind Variables page

To use dynamic queries, you must create a query that joins the run control with the desired record and joins the operator ID and the run control ID.

You can make any Cube Manager query—dimension query, data source query, or attribute query—dynamic by joining in runtime parameters. The parameter value must be stored in the database on a Cube Manager run control record or on a record to which the run control record can join. Then, in PeopleSoft Query, join the run control record and set prompts for the operator ID and the run control ID. These prompts are completed at runtime with the operator ID and run control ID of the run control that is used to launch the Cube Manager process. The Description and Bind Value fields on the sample Build Cube run control page provide one way to join the parameters.

Adding Bind Variables

To add bind variables to a run control page:

1. Identify the run control record that is used to launch the Cube Manager process.

    If the sample Build Cube page is delivered in the PeopleTools Demo database (PTDMO), this record is MAKE_ANL_DB_REQ.

2. Identify the fields that you want to use for parameters for runtime queries.

    For example, values for From and To for time periods.

3. Ensure that the fields identified in the previous step are available on the run control record, or that they are on a record that can be joined with the run control record.

    See *PeopleTools 8.51 PeopleBook: PeopleSoft Application Designer Developer's Guide*, "Planning Records, Control Tables, and TableSets," Record Definition Planning.

4. Ensure that these same fields are available on a page so that users can enter values for them.

    If these fields exist on the run control record, they most likely also exist on the run control page.

    See *PeopleTools 8.51 PeopleBook: PeopleSoft Application Designer Developer's Guide*, "Creating Field Definitions."

5. In PeopleSoft Query, join the records that contain the fields mentioned previously to the query that you want to make dynamic.

    For example, assume that you want to bind in values for the From and To periods and that these fields exist on the run control record. In the data source query, join the From and To periods on the run control record to the From and To fields in the main data record (the primary record in the query that contains the data that you want to return).

    See *PeopleTools 8.51 PeopleBook: PeopleSoft Query*, "Working with Advanced Query Options," Joining Records.

6. In PeopleSoft Query, set up prompts for the operator ID (first) and the run control ID (second).

    Instead of prompting for these values at runtime, Cube Manager automatically supplies the bind values that the user entered on the run control page.

7. Open the Bind Variables page by selecting the Bind Variables tab in the Build Cubes component.

    Alternatively, select PeopleTools, Cube Builder, Cube Manager, Build Cubes, Build Cube; enter a cube instance ID; and select the Bind Variables tab.

8. Add variables and save your changes.

9. Enter a short text description and give value of the variable.

See Also

Oracle PeopleSoft 9.1 PeopleBook: General Ledger

Defining a Cube Build Process Using Process Scheduler Manager

Access the Process Definition page by selecting PeopleTools, Process Scheduler, Processes.

This is an example of the Process Definition page:

Process Definition page

Priority	Select a relative priority that is used by the Process Scheduler Server Agent to determine which process to initiate first, if multiple processes are queued to run on a given server.
API Aware (application programming interface aware)	Select this option if this process is application programming interface (API) aware.
	If this option is selected for any process that is not API-aware, Process Scheduler includes it in the concurrent task count. This inclusion can result in improper server load balancing. Selecting this option does not mean that your process becomes API-aware. You must still add API code to your process.

Running the Process to Build the Cube

Use these steps to access the Process Scheduler Request (PRCSRQSTDLG) page:

1. Select PeopleTools, Cube Builder, Cube Manager, Build Cubes.

2. Add a new run control ID or search for an existing one.

3. Enter the appropriate values in the Build Cube page, and click the Run button.

Process Scheduler Request page

Server Name	Select the name of the server on which you want the process to run.
	Note. *PSNT* is a Microsoft Windows server. Cube Manager runs only on a Windows server.
Recurrence	Select the recurring time intervals for a process request to run. Select either *Daily Purge* or *M-F at 5pm.*
	See *PeopleTools 8.51 PeopleBook: PeopleSoft Process Scheduler.*
Time Zone	Select the time zone in which your process runs.
	For example, you could be in Eastern Standard Time (EST) and schedule a process to run in Pacific Standard Time (PST).
Run Date	Select the date on which your process runs.
Run Time	Select the time when your process runs.
Reset to Current Date/Time	Click to set the run date and runtime to the present date and time.
Select	Specify the cube to be built.
	If you select this option, the process is included in the build process.
Description	Displays a description that uniquely identifies the process.
Process Name	Displays the name of the process as it appears in the definition.
Type	Select the type of output. Select *None.*
Format	Select the format of the output. Select *None.*

[icon]   Click the Distribution button to view a list of people who have the correct security to receive the output of this process.

OK   Click to run the process.

Using API and Command-line Specifications

PSOLAP (PeopleSoft Online analytical processor) can be run from the Process Scheduler component or from the command line.

This table lists the parameters that PSOLAP uses when PSOLAP runs from the command line:

Parameter	Description
A	Cube Name.
B	Business Unit (for example, *B -US008*).
R	RUN Control.
I	Process Instance ID number (for example, *I42*) usually passed by process scheduler.
m	Metadata action. • C - Create. • R - Replace. • N - None. For example, *-mC.*
D	Data Action. • N - None. • R - Create. • I - Incremental. • D - All Upper Level Data. • A - Non Input Data.
CT	Database type (for example, *-CTORACLE*).
CD	Database name (for example, *-CDPT820RC1*).
CO	Operator ID (for example, *-COPTDMO*).

Parameter	Description
CP	Password (for example, *-CPPTDMO*).
t	As of date (for example, *-t1997-07-21*).
L	Trace level (for example, *L31*).
X	Action. • F - Fact. • C - Complete. • D - Dimension.
Y	This string contains the single factory or the single dimension that is specified in the X parameter. This string is not valid if X=C (Complete).

Example: By setting the following command line, you cause the processing for dimension DEPARTMENT-FIN to be executed:

```
PSOLAP  -CX -CTORACLE -CDPT820RC1 -COVP1 -CPVP1 -I279 -RTEST_SAN_GLACCT_CII -aFIN_
GLACCT_CII -mC -dR -b SHARE -kINITIAL -t2001-03-01 -s -X D -Y "DEPARTMENT-FIN"
```

Logging

You can pass the log level with the *L* option to the command line of psolap.exe; for example, L2 or L6, and so on.

This table lists the log level and what is recorded in each log level:

Log Level	What is recorded
1	Basic logging
2	Detail logging
4	Dimension member logging
8	Fact map logging
16	API logging

For example, to obtain the Basic, Detail, Member, and Fact map logging in the Parameter List field on the Type Definition panel, we specified −*L15,* where 15 = 1 (Basic) + 2 (Detail) + 4 (Member) + 8 (Fact):

```
psolap.exe -CX -CTMICROSFT -CDPT820RC1 -COPTDMO -CPPTDMO -I42 -REMPLOYMENT -a
EMPLOYMENT -mC -dR -b -kINI IAL -t1997-07-21 -s -L31

psolap.exe -CX -CTMICROSFT -CDPT820RC1 -COPTDMO -CPPTDMO -I42 -REMPLOYMENT -a
EMPLOYMENT -mC -dR -b -kINITIAL -t1997-07-21 -s -L31
```

```
PSOLAP.EXE -CX -CTMICROSFT -CDPT820RC1 -COPTDMO -CPPTDMO -A EMPLOYMENT -I43 -B  -
L31 -X F -Y "OLAP_EMP_DATA_QRY" -t1997-07-21
```

See Also

PeopleTools 8.51 PeopleBook: PeopleSoft Cube Builder, "Securing Cubes"

Index

A

account dimensions 28
ad hoc queries 15
API aware 68
Attribute Definitions - Attribute Name page 45
attribute queries 20
    Essbase attributes 21
    general attributes 20
    PowerPlay attributes 22
    setting on detail members 20
    setting on node members 20
attributes
    designing names 44
    naming 39
    selecting 50
    specifying user defined 21, 45

B

base currency 56
bind variables
    adding 66
    defining 63, 65
blank members 33
bookmarks 64
Build Cubes - Bind Variables page 62, 65
Build Cubes - Build Cube page 62, 63
Build Ledger Cubes page 62

C

calendars
    lunar 31
    standard 31
Cognos PowerPlay 2, 10
    applying cube design attributes to 7
    attributes 22
    date functions 31
    defining cube instance definition options 59
    defining cube settings for 57
    defining time dimensions for 29
    installing 2
    PowerCube 10
    Transformer 10
country dimensions 28
cube components
    mapping source query fields to 43
    Overview 12
cube definitions
    adding cube inputs to 43
Cube Definitions - Cube Definition page 40
Cube Definitions - Cube Inputs page 40, 43
cube design 12
Cube Filter Criteria page 48, 52
Cube Input Field page 40, 43
cube inputs, adding to cube definitions 43

cube instance definitions 12, 47
    naming 49
    platform specific 47
    specifying 13
cube instances
    building 47
    overview definitions 47
Cube Instances - Cube Inst Def page 48, 50
Cube Manager *See* PeopleSoft Cube Manager
    implementing: 2
Cube Manager process 13
Cube Manager process type 8, 62
cube outlines 12
cubes
    adding inputs to definitions 43
    building 8, 9, 61, 65
    creating 62
    creating filters for 52
    creating instance definitions 49
    creating new 13
    defining build process in Process Scheduler
        Manager 67
    defining build specifications for 63
    defining settings for 50
    designing 9
    designing definitions 39, 42
    filters 52
    instance definitions 12, 47
    instances 13
    mapping source query fields to components
        of 43
    multicurrency rate 56
    naming instance definitions 49
    outlines 12
    running the build process for 68
    selecting dimensions for 40
    updating existing 13
currency, base 56
currency category (Essbase attribute) 21
currency conversion type (Essbase attribute) 21
currency cubes, Oracle Essbase 62
currency members, local 56
currency name (Essbase attribute) 21
currency partition dimensions 28, 56
currency rate database 56

D

database agent queries 15
data source queries 23
data sources
    mapping to dimensions 40
data storage (Essbase attribute) 21
data store, persistent 9
dates
    Cognos PowerPlay functions 31
    formats 32
dat file 58
default calculations 55
Dimension Options page 26, 32

dimension queries 16
  narrow 16
  using level specification 16
  using parent/child relationship 16
  wide 19
dimensions 12
  account type 28
  country type 28
  currency partition 28, 56
  defining options for 32
  defining structures of 27
  designing 28
  IDs 27, 29
  mapping data sources to 40
  naming 29
  selecting 50
  selecting for cubes 40
  sharing 28
  standard type 28
  time type 28
  types 28
Dimensions - Dimension page 25, 27
Dimensions - Rollup Inputs page 26, 35, 37
Dim Input Field page 26, 38
dynamic queries 66

E

Earliest Date field 58
Enterprise Server 10
  and Cube Manager 60
  and PeopleSoft Cube Manager 11
  running locally 60
Enterprise Server field 59
EP Server *See* Enterprise Server
Essbase *See* Oracle Essbase
Essbase Advanced Options page 48, 53
expense item (Essbase attribute) 21

F

fact data 6
field objects
  identifying 61
fields
  mapping 36
  mapping source query fields to cube
    components 43
  mapping to roles 38, 39
filters 52
  creating 52
flip sign (general attribute) 20

G

general attributes 20

H

HOLAP 6
hybrid OLAP *See* HOLAP

L

label (general attribute) 20
label prefixes 34
  setting preferences 34
  setting styles 35
label suffixes 34
  setting preferences 34
  setting styles 35
Language Override field 58
languages
  adding 55
  default 55
  multiple 55
  preferred 55
Latest Date field 59
levels
  names 28, 39
  numbers 28
levels from top
  calculating 36
  nodes 36
local currency members 56
long description (PowerPlay attribute) 22
lunar calendars 31

M

mapping
  data sources to dimensions 40
  fields 36
  fields to roles 38, 39
  source query fields to cube components 43
mdc files 58
mdl files 58
measures
  selecting 50
member properties 12
members
  blank 33
  label prefixes 34
  label suffixes 34
  naming 33
metadata 8
  defining source for rollups 35
  designing 15
  updating 54
metadata update action 54
model file 58
MOLAP 6
multicurrency rate cubes
  creating 56
  overview 56
multidimensional analysis 6
multidimensional OLAP *See* MOLAP

N

narrow queries 16