New to this edition, Chapter 10 discusses the end-to-end process to set up a basic embedded SAP BPC planning scenario. Learn how to set it up and use MultiProviders, composite providers, DataStore objects, and local providers.

Peter Jones, Tim Soper
Implementing SAP Business Planning and Consolidation
900 Pages, 2015, $79.95/€79.95
ISBN 978-1-4932-1179-1

www.sap-press.com/3711
Before we move too far into our coverage of embedded SAP BPC, we should examine its architecture—the SAP BW tables, queries, and Integrated Planning objects that will be used moving forward.

10 Embedded SAP BPC Architecture

Embedded SAP BPC provides quite a bit of flexibility and options to meet customer planning and reporting requirements. Sometimes, we have so many options that half the challenge is figuring out when to use each one! As we go through this chapter, we'll emphasize a given business scenario first, discuss its architecture second, and then go through the configuration steps third.

In this chapter, we'll discuss the end-to-end process to set up a basic embedded SAP BPC planning scenario. We'll go through the buildup of the SAP Business Warehouse (SAP BW) InfoProviders, the Integrated Planning (IP) objects, the SAP BW Planning Query, and then how to use the query and a basic planning function in the SAP Enterprise Performance Management (EPM) add-in. After that, we'll talk about how MultiProviders, DataStore objects (DSOs), and local providers can be used in planning scenarios as well.

10.1 Setting Up the Embedded Planning Model

In this section, we'll go through the use cases and set up of a real-time InfoCube, the IP objects, the SAP BPC objects, and the EPM add-in report and action pane from start to finish. The main idea in this first pass is to give you a feel for how the components work together in an end-to-end scenario and do the deep dive later in Chapter 11 and Chapter 13 on reporting and planning and forecasting, respectively.

You need the following components for a planning scenario:

- Real-time InfoCube
- Aggregation level
10.1.1 Using Real-Time InfoCubes

InfoCubes represent a type of activity, whether it’s being used for reporting or planning. It has a group of homogeneous characteristics and key figures to meet the task at hand. Recall that real-time InfoCubes are write-optimized InfoCubes; they are the primary tables used in embedded SAP BPC 10.1.

As a point of reference, in standard SAP BPC, the real-time InfoCube is generated when you create the model; it can only use InfoObjects in the SAP BPC namespace, and only one key figure is allowed. So if we compare real-time InfoCubes used for the embedded version to real-time InfoCubes used for the standard version, four main differences stand out:

- Embedded InfoCubes use standard InfoObjects.
- Embedded InfoCubes can have multiple key figures.
- Embedded InfoCubes are created from SAP BW, not SAP BPC.
- Embedded InfoCubes use the normal SAP BW namespace.

As a result, using normal SAP BW objects means that you can share data more easily, resulting in less data redundancy, and you can use more than one key figure, which provides data modeling flexibility.

But using real-time InfoCubes in embedded SAP BPC involves a bit more discussion than it did in standard SAP BPC. With standard, when you create a model, the system generated the real-time InfoCube and associated MultiProvider automatically. Recall from Chapter 3 that each standard model always has one corresponding real-time InfoCube and one MultiProvider. In embedded, you create the real-time InfoCube first, and then you create the SAP BPC model. Furthermore, embedded SAP BPC models can use one or more real-time InfoCubes, DSOs, local providers, and VirtualProviders, so now it’s much easier to merge data from multiple sources.

Despite the fact that standard SAP BPC models can use different types of InfoProviders, real-time InfoCubes will remain the primary choice because DSOs, local providers, and virtual InfoCubes are intended for more specialized scenarios as we’ll explain shortly.

Before we start talking about how to create a real-time InfoCube we need to think about the planning activity first because that will drive the selection of InfoObjects of the InfoCube. In our scenario, we need to do product planning by material group by year for pricing data initially and then quantities revenue and costs by material and period later on. Because there isn’t a delivered InfoCube that meets this requirement exactly, we’ll create a custom real-time InfoCube.

To create an InfoCube, just go to Transaction RSA1, and right-click on an InfoArea, and choose Create InfoCube (Figure 10.1).

In Figure 10.1, InfoCube WSAN1_G00 is used as a template to bring in most of the required InfoObjects. Of course, the Real Time checkbox is checked, and because with SAP BPC 10.1 we’re using SAP HANA, this is by default an In-Memory InfoCube.
Choose Create, and the system brings in the InfoObjects from the template Info-Cube (WSAN1_G00) (see Figure 10.2).

Before SAP HANA, InfoCubes were star schemas with table joins from the surrogate ID (SID) table to the dimension table to the fact table, and characteristics were grouped into dimension tables (Chapter 3). With SAP BW on SAP HANA, the InfoCube is flattened out. The SID tables join directly into the fact table, and there is only one dimension table, which is used to keep track of uncompressed request IDs.

To see the dimension and fact table, review the structure for the InfoCube via EXTRAS • INFORMATION LOG/STATUS • DICTIONARY/DB STATUS.

As you can see in Figure 10.3, there are only two tables in this InfoCube: the fact table /BINFOCUBE/FPRODPLAN and the dimension table /BINFOCUBE/DPRODPLAN.

After you activate the InfoCube and return to Transaction RSA1, you can see the icons that signify the InfoCube as a Real-Time InfoCube that is an SAP HANA-Optimized InfoCube (see Figure 10.4)

At this point, you can create an aggregation level on the real-time InfoCube. Of course, you can actually create the aggregation level from the context menu (right-click) of an InfoArea, but let’s do it from the planning modeler because that’s where the other IP objects are maintained.

10.1.2 Using Aggregation Levels

You might recall that our real-time InfoCube has more InfoObjects than necessary for the initial planning phase. The real-time InfoCube has period, material, quantity, revenue, and cost InfoObjects that aren’t needed until later.

So you need to create a structure that only includes the InfoObjects required for this planning phase. This is called an aggregation level, and one of its main purposes is to provide a slice of the real-time InfoCube for your planning activities. After you create the aggregation level, you’ll also need IP planning functions, filters, and planning-enabled queries. All of these objects are created against the aggregation level.

Because planning-enabled queries are created against aggregation levels, the aggregation level is therefore an InfoProvider. It doesn’t contain any data, so it’s basically a view created against the real-time InfoCube.

To create an aggregation level, go into the planning modeler. From the SAP EASY ACCESS screen, open BUSINESS PLANNING AND SIMULATION, and choose RSPLAN – BI INTEGRATED PLANNING, or go to Transaction RSPLAN, as shown in Figure 10.5.
When you enter Transaction RSPLAN, you arrive at the InfoProvider screen that includes five buttons for the five objects used in RSPLAN:

- **InfoProvider**
  In this screen, you maintain characteristic relationships and data slices, and specify which planning sequences need to be executed as data is saved.

- **Aggregation Level**
  The aggregation level forms a conceptual slice of the real-time InfoCube for a planning scenario, and it owns other planning objects. They can be created against real-time InfoCubes, DSOs, MultiProviders, and local providers.

- **Filters**
  Filters are used primarily to define the data region for planning functions bundled into planning sequences.

- **Planning Functions**
  Planning functions are used to perform mass updates in planning scenarios such as copying actual to plan, deletions, and calculating revenue.

- **Planning Sequences**
  Planning sequences are used for testing and for bundling planning functions to be executed in real time or in batch mode.

The first thing to do in Transaction RSPLAN is select the real-time InfoCube InfoProvider (PRODPLAN, as shown in Figure 10.6) so that you can configure it for planning activities.

To configure the InfoProvider for planning, choose Edit. Figure 10.7 shows the metadata view in the InfoProvider tab. There are also tabs for the following:

- **Central Settings**
  You can select a planning sequence to be executed when data is saved and set a key date for time-dependent master data selections.

- **Characteristic Rels**
  You can provide valid characteristic combinations when planning and also perform derivations.

- **Data Slices**
  You can lock a data region to prevent data changes when a process has been completed. SAP BPC work status creates data slices automatically, so you don’t need to create them here.

To create the aggregation level, return to the main Transaction RSPLAN screen, proceed to the aggregation level screen, and choose Create. Then name the aggregation level, and select the PRODPLAN real-time InfoCube InfoProvider that it belongs to, as shown in Figure 10.8.

The resulting configuration screen looks the same as when you created the real-time InfoCube, except that the real-time InfoCube’s InfoObjects automatically appear on the left under Basis InfoProvider, as shown in Figure 10.9.
Based on the planning requirement, all you have to do is drag and drop the InfoObjects you need into the aggregation level on the right. As you can see in Figure 10.10, all of the InfoObjects in the real-time InfoCube have been selected except for Posting Period, Material, Sales Unit (Unit is referenced to Quantity so it needs to be included), Total Variable Costs, Sales Quantity, and Revenue. In other words, the planning activity is yearly planning so you don’t need Posting Period; you’re planning by material group only, so you don’t need Material; and so forth.

After the aggregation level is activated, you can use it to create planning functions.

### 10.1.3 Using a Copy Planning Function

IP provides quite a few planning functions, and being able to use them in embedded SAP BPC is a big advantage. We’ll be spending quite a bit of time going over these in Chapter 13.

To create a planning function, you leave the Aggregation Level screen and move into the Planning Function screen. When you choose Create, you can select the function type, provide a name and description, and select the aggregation level, as shown in Figure 10.11.
This planning function will be used in a scenario where you plan prices in version B1 and then copy them to version B2, where you can run a revaluation function to mark them up or down.

In the next step, shown in Figure 10.12, you can start to parameterize the function in the Edit Planning Function screen. Because you’ll be changing values only in version B2, select Version in the Fields to be changed column. Because the prices will be copied to version B2 for any currency, fiscal year, and material group, you don’t need to select any Fields for Conditions. We’ll spend more time on this in Chapter 13.

To specify that version B1 is the source and B2 is the target, open the Parameter screen (see Figure 10.13), and select B1 as the From value and B2 as the To value.

After you save the planning function, you can proceed with the planning query.

### 10.1.4 Using Planning-Enabled Queries

One of the main features of any planning system is the ability to perform manual input planning in Excel workbooks. In this example, this means the EPM Excel add-in. To do this, you need a planning-enabled query.

Planning-enabled queries must be created on an aggregation level and not on the real-time InfoCube. More could be said about how to use the basic and advanced planning related features of the BEx Query Designer, but for our purposes, we’ll stick to the basic query features and the fundamental planning settings so that you have a very clear understanding of the components and how they depend on each other. Also, this scenario uses an aggregation level on the real-time InfoCube, but we’ll cover queries on aggregation levels on MultiProviders, DSOs, and local providers later in this chapter.

To access the Query Designer, go to your Start menu. Choose All Programs → Business Explorer → Analyzer. In the Add-Ins tab, select Create New Query. In the Query Designer screen, choose New Query, and select the aggregation level.

When you create queries for SAP BW reporting scenarios, you can always pick and choose which InfoObjects to include in the query. In a reporting scenario, for example, if you don’t need to report by material group, you can simply choose not to include it in the query.

When creating queries for planning, however, the rules are a little different. In this case, you need to include every InfoObject in the query to allow for manual input planning because when doing manual planning, every cell must represent single characteristic values. In other words, any characteristic that isn’t in the
rows or columns must be included in the filter and restricted to single values. In this case, the business requirement calls for year and material group in the rows with three columns for prices in version B1, B2, and B. The three columns will include the price key figure that is restricted to version B1, B2, and B. The scenario for the three key figures is to allow the planner to test prices in B1 and B2, and, when they are ready, they will copy their final price to version B, which will then be used in the next part of the planning process.

Fiscal year variant will always be set to K4, so you can put that in the Characteristic Restrictions. Fiscal year will be selected by the user at runtime, so it’s placed in the Characteristic Relationships as well and restricted by a variable.

The Currency will initially be USD, but the planner is allowed to change it, so we’ll put that in the Default Values. Because Currency is in the Default Values, it’s available in Free Characteristics for the business user to add it into the rows or columns.

In Figure 10.14, you can see that every InfoObject is in the Rows, Columns, or Free Characteristics except for Version because that will be included in the columns as part of the three restricted key figures. Fiscal Year Variant is restricted to K4 (CAL Year, 4 Special Periods). To restrict it, in the context menu, select Filter • K4, and then click OK.

Currency was restricted to USD in the Default Values using the same technique. Fiscal Year is restricted by a variable so users can choose the year to plan on when they run the query.

To restrict the first column Sales Price per UM to version B1, simply double-click it, add the version to Details of the Selection, and restrict it to B1. Figure 10.15 also shows that the Description has been changed to B1 Price.

For this query to be plannable, you need to set at least one key figure to be input ready. Figure 10.16 shows the Planning tab and under Change Data, Input Ready (Relevant for Locking) has been selected. The three options under Change Data are used for the following purposes:

- **Not Input-Ready (Not Relevant for Locking)**
  Use this to read the data without blocking others from changing it while you’re viewing it.
Not Input-Ready (Relevant for Locking)
Use this to read the data but lock it so no else can change it while you’re viewing it.

Input-Ready (Relevant for Locking)
Use this to allow write access and block others from changing it while you’re viewing it or changing it.

Figure 10.16 Planning Settings for a Key Figure

To adopt these settings, click OK, copy B1 Price, rename it to B2 Price, and restrict it to B2. Then, copy B2 Price, rename it to B Price, and restrict it to B; however, in this case, you want this key figure to be written to only via the planning function, so Change Data will be set to Not Input-Ready.

When at least one key figure is set to allow input, the query will be set to open in change mode when initially executed. In the query Properties shown in Figure 10.17, the query is now set to Start Query in Change Mode.

The business requirement for this example calls for the rows to be available for input. Queries normally only display characteristic values if they have postings in the InfoProvider. Of course, because this is a planning scenario, and you haven’t created any data yet, you need to set the characteristics to display based on their master data. To do that, just select Fiscal Year/Material Group in the Rows, go to the Properties on the upper right, and in the Extended tab, choose Master Data, as shown in Figure 10.18.

Let’s summarize the planning-related settings in our query:

- Key figures are set to allow input.
- The query is set to run in plan mode.
- The characteristics are set to display based on master data.

So let’s test the planning-enabled query in the BEx Analyzer by choosing Query - Exit and Use.
In the result set in Figure 10.19, B1 Price and B2 Price are available for input based on the light blue border in the data cells, whereas B Price isn’t and therefore just has the black border.

In this case, the BEx Analyzer is only being used for testing because the ultimate UI will be the EPM add-in in Excel. To be able to use the EPM add-in, however, you first must create an embedded environment and a model.

10.1.5 Using Embedded SAP BPC Environments

If you have experience with standard SAP BPC, you know that when you’ve created environments, dimensions, and models in that version, the system generated the corresponding InfoArea, characteristic, and real-time InfoCubes in SAP BW—all in their own namespace.

In contrast, embedded SAP BPC environments don’t have a corresponding InfoArea, and the characteristic and real-time InfoCubes aren’t created from the environment’s administration UI because they already exist in SAP BW with the normal namespace. So why do you need an environment? In essence, the embedded SAP BPC environment provides a framework that allows you to use work status, BPFs, data audit, web reports/input forms, and the EPM add-in. After we’ve walked through the creation of an embedded environment, we’ll do some more comparisons to standard environments.

In the administration screen, standard SAP BPC environments are still created by copying from an existing environment. Embedded SAP BPC environments, on the other hand, can only be created from scratch by choosing Create.

Because embedded SAP BPC environments don’t generate any SAP BW InfoAreas, characteristic, and InfoCubes, it doesn’t take very long to create them. To create an embedded environment, go to the web client, choose the currently connected environment at the bottom of the screen, and then choose Manage • Manage all environments • Create.

In the resulting Create an Environment popup (see Figure 10.20), the Type defaults to Unified (remember: embedded was initially referred to as “unified”).

After you input the Environment ID and Description and then choose Create again, the new embedded environment is available for use.

In Figure 10.21, you can see the initial web client view with the Library screen selected. To do configuration in the web client, just go to Administration (see Figure 10.22). In Administration, you can use the hypertext links to configure anything from Models to Data Audit.

You can now create an embedded model.
Before we discuss the configuration of embedded SAP BPC models, let’s talk about what purpose they serve. Embedded SAP BPC models are used to select InfoProviders to report or plan on. In addition, these models and their linked InfoProviders contain the configuration for data profiles, work status, data audit, and BPFs.

So how many embedded models will you need? The answer always depends, but, in general, you’ll most often use one model for each InfoProvider. For example, in the simplest scenario, a real-time InfoCube is linked to one embedded SAP BPC model. If you’re using a MultiProvider, then you only need one embedded model for that as well. Conceptually, the model represents the type of reporting or planning activity. If you need to do CapEx and HR planning, you’ll use a model for CapEx and another model for HR, and so forth.

Now that we’ve discussed the use case for embedded SAP BPC models, we can go into the configuration steps. To create an embedded model, go to Administration and choose Models. Then the Create New Model dialog opens as shown in Figure 10.23.

After providing an ID and Description, select an InfoProvider, as shown in Figure 10.24. When you go through the MultiProvider scenario later in this chapter, you’ll select several InfoProviders.

By choosing Next and then Create, you attach the model to an existing real-time InfoCube. If you want to view the structure of the underlying real-time InfoCube from the web client, you can just select the model’s link; in the Structure tab, you can see the Characteristics and Key Figures as shown in Figure 10.25.
The Aggregation Levels and Related MultiProviders tabs display the structure of the related aggregation level and MultiProvider.

**Note**

Because the real-time InfoCube is compressed in SAP BW using process chains, the embedded web client doesn’t contain the model optimization option like the standard version does.

Now that the embedded model is complete, you can configure data access profiles, work status, and BPFs. It’s also available for web and EPM add-in planning and reporting.

### 10.1.7 Using the EPM Add-in Workbooks for Basic Planning Scenarios

The EPM add-in for Excel is, of course, the main user interface for planning and analysis of embedded SAP BPC data. Building workbooks that include multiple EPM reports, drop-downs, buttons, and VBA for before and after events, and so on will take up a large portion of the development in an implementation.

Chapter 4 describes all of the basic and advanced features of the EPM add-in for standard SAP BPC models. This chapter describes all of the basics when using EPM add-in on embedded connections. In Chapter 11, we’ll get into the finer points of how the EPM add-in works when connected to embedded models.

To access the Excel EPM add-in, you can get there from the web client and use a system-generated connection, or you can use the Start menu and create a new connection. For this discussion of how to create a new connection for the embedded version, launch Excel from the Start menu, and go to the EPM tab. Of course, after you create a connection, it will be available the next time you log in.

In the EPM tab, go to Report Actions, and choose Manage Connections. In the Connection Manager, create the connection with the parameters as shown in Figure 10.26.

If you’re familiar with standard SAP BPC, you’ll notice that there is a new connection type: SAP BW (INA Provider), with INA standing for Information Access. This new connection type is required for embedded and provides the following services:
Work with SAP BW queries with or without variables.
Retrieve data using EPM reports
Enter and save data using input forms
Execute IP functions using data processes

Let's walk through the parameters in Figure 10.26:

- **Connection Name**
  Enter a name of your choice, or use the Generate Connection Name button at the bottom of the screen.

- **Custom System**
  Provide the server name manually.

- **SAP Logon**
  Use the server from your SAP GUI logon.

- **System Name**
  Enter a name of your choice.

![Figure 10.26 Creating an Embedded Connection](image-url)

When using the new connection, you get the EPM – Query Selection screen, as shown in Figure 10.27. In the top part of figure, you can search for either queries or workbooks.

If you want to run a plannable query, you can enter a character string such as \texttt{prod} under Query Selection, and the system will return with a hit list of available queries to pick from. If you want to use a query more than once, you can make it a favorite by clicking the star icon under the Favorite column.

Notice that the Create Report From Query option is selected on the lower left of Figure 10.27. This will automatically create an EPM report based on the default query view. Choose OK, and the EPM-Set Variable prompt for fiscal year pops up. After you select 2015, the EPM report is displayed in Sheet1; you can then enter prices for B1, for example, and use the EPM Save Data button to store the data in memory (see Figure 10.28).

In Chapter 11, we'll go through the EPM ribbon, EPM Context area, and EPM pane on the right in more detail, but for now, we just want to mention that the Transfer Data option is only available for embedded SAP BPC connections. The Data Manager tab is only for standard SAP BPC connections.
To use a planning function, select the Data Processing tab on the lower right, right-click the EPM connection, and select a planning function. In this first pass, select the Copy function, and add it to the EPM connection, as shown in Figure 10.29.

Then choose Execute to copy the prices from version B1 to B2.

In Chapter 12, we’ll add a revaluation function to mark up or mark down prices, a delete function, and then a copy function that will take either B1 or B2 prices via a variable and copy to the final price data in version B.

For now, however, let’s save the workbook for future use by choosing Save as Input Form to Server Folder on the EPM tab. In the Save dialog box, save the new workbook to the Public Folder so others can access it.

That just about completes our initial goal of an end-to-end pass-through. However, there is usually a need to view the data in the database for debugging purposes, to clear up confusion over what a planning function is doing to the data, or to show how the system writes delta records when data is changed. We also need...
to discuss compression because that plays a slightly different role now that we’re on SAP HANA.

10.1.8 Browsing and Compressing Data in SAP BW

To take a comprehensive approach, we also need to take a look at the data first from SAP BW and then from the SAP HANA modeler.

But before we look at any data, let’s look at what data we’ve entered and saved in advance for this test example:

- Enter “25” for JUC.
- Enter “25” for WAT, and save the data.
- Enter “35” for JUC.
- Enter “0” for WAT, and save the data.

We then compressed the real-time InfoCube with zero suppression turned on. Compression, you might recall, is used to reduce the redundancy of the database by aggregating data over common key fields.

Now, you can use the SAP BW Transaction LISTCUBE to display the data in Figure 10.30. Enter the InfoProvider, press (F8), and choose Fld Selection for Output • Select All. Then press (F8) twice.

Before compression, the records in Figure 10.30 demonstrate that the real-time InfoCube works the way it always has: delta records are written to the database to get the net result (i.e., JUC has a delta of 10 to get a net of 35, and WAT has a delta of –25 to get a net of 0). After the zero-based compression, there is only one record (–35 for JUC); because the records for WAT net to zero, they were eliminated.

Compressing data in InfoCubes normally is carried out in a process chain. To do it manually for test purposes, follow these steps:

1. Go to Transaction RSA1.
2. From the context menu of the real-time InfoCube, choose Planning Specific InfoCube Properties • Change Real-Time Load Behavior.
3. Choose Load Mode to close the open request ID.
4. Choose Plan Mode • Continue.
5. Choose Manage, and then go to the Collapse (compress) tab shown in Figure 10.31.
6. Select the With Zero Elimination checkbox, and then click Release.

Here you can see that with SAP BW on SAP HANA, InfoCubes no longer have a Rollup tab because aggregates aren’t used with SAP HANA. In addition, there is no Performance tab either because the data statistics aren’t relevant when SAP HANA is being used.
10.1.9 Browsing Data and Statistics in SAP HANA

Because SAP HANA has modeling and ETL capability, there will be no doubt that some development work in SAP HANA in support of SAP BPC planning and analysis scenarios. You may also want to browse the data in SAP HANA as well to erase any doubts regarding what the application is doing to the data or to debug.

In Chapter 9, we discussed some of the basics of SAP HANA along with the important planning implications; and now we’ll browse the data in SAP HANA and view the memory consumption statistics.

Let’s take a look at the data using the SAP HANA Administration Console. To access the Administration Console, go to **Start / All Programs / SAP HANA / SAP HANA Studio / Open Administration Console.** Under **Systems**, right-click, choose **Add System**, and enter the information shown in Figure 10.32. The **Host Name** and **Instance Number** can be accessed from your SAP HANA administrator.

![Figure 10.32 Adding an SAP HANA System](image)

Choose **Next**, and then enter your **User Name** and **Password**, as shown in Figure 10.33.

Choose **Finish**. Expand the **Systems** tab on the upper left of the screen, open the **Catalog** folder, right-click on the table, and choose **Filter**. Enter your InfoCube name, and choose **OK**. Right-click the fact table (/BINFOCUBE/FPRODPLAN), and choose **Open Data Preview**.

![Figure 10.33 SAP HANA Connection Credentials](image)

Figure 10.34 shows basically the same data before compression as you saw in SAP BW: the two initial prices of 25, 10 for the change to 35, and –25 for the change from 25 to 0.
After zero-based compression, there is only one data record (Figure 10.35), as you saw previously. Because this data browser hits the fact table directly, the surrogate IDs (SIDs) are displayed. For example, the SID for version (SID_VERSION) is 4, which is an alias for version B1.

Embedded SAP BPC works the same as standard SAP BPC when it comes to writing delta records to the real-time InfoCube. Both embedded and standard applications compress the data as well. In addition, compression also triggers a merge of the data from delta storage to main memory in SAP HANA, which improves performance because subsequent reads won’t have to merge the data from delta and main memory.

To see this in the SAP HANA Administration Console, right-click the fact table, and choose Open Definition. Go to the Runtime Information tab shown in Figure 10.36.

Before compression, the Memory Consumption in Main Storage (KB) is 23 and Memory Consumption in Delta Storage is 153. After compression, Main increases to 56 and Delta goes down to 92 (see Figure 10.37).

Let’s review this end-to-end excursion using Figure 10.38. We started in SAP BW with the real-time InfoCube, and then created the aggregation level and copy planning function. In the Query Designer, we built a planning-enabled query on the aggregation level. In the web client (based on HTML5), we created an embedded environment and model. Finally, in the EPM Excel add-in, we created an EPM workbook. (We’ll work with web input forms in Chapter 11.)
In our scenario, we only used one real-time InfoCube, but in many cases, you’ll need to use multiple InfoProviders. For example, you might want to store granular actual data in one InfoProvider and less granular plan data in another. Or you might want to plan on product price data in one InfoProvider and at the same time plan quantities and revenue in another. You can do this with a MultiProvider.

**10.2 Using MultiProviders**

Just as a refresher, remember that MultiProviders can be used to union data in other InfoProviders. Although there is no data in a MultiProvider, it’s a structure that you can use in reporting and planning scenarios that involve multiple InfoProviders.

Of course, we could have used a MultiProvider in our previous scenario by adding an additional InfoProvider to it for the quantity and revenue planning. In reality, you should always use MultiProviders even if you only need one real-time InfoCube because they provide flexibility for modeling changes such as adding a new InfoProvider or switching to a different InfoProvider. Therefore, using MultiProviders provides a lower cost of maintenance for design changes.

Our new scenario calls for two real-time InfoCubes under a MultiProvider (Figure 10.39): PRICEPLAN and REVPLAN. The key difference between the two real-time InfoCubes is that PRICEPLAN has the price key figure, whereas REVPLAN has the revenue and quantity key figures.

The components you need for this scenario are essentially the same as in the previous example with one InfoCube except you need a MultiProvider:

- Real-time InfoCube
- MultiProvider
- Aggregation level
- Planning function
- Planning query
- Embedded environment
- Embedded model
- EPM workbook with an EPM report and planning functions

### 10.2.1 Creating the MultiProvider

The data model of the MultiProvider from Transaction RSA1 is shown in Figure 10.40. In this example, all InfoObjects from both real-time InfoCubes have been included in the MultiProvider, as well as the output from the characteristic and key figures.

Follow these steps to create a MultiProvider:

1. Go to Transaction RSA1.
2. In the InfoProvider tree, right-click on an InfoArea, and choose Create MultiProvider.
3. Input a Name and Description, and click Continue.
4. Select the relevant InfoProviders, and click Continue.
5. Select the desired InfoObjects, and choose Identify Characteristic • All • Continue • Continue • Identify Key Figures • All • Continue • Continue • Activate.
Before we head to Transaction RSPLAN to create the aggregation level, we should point out that because MultiProviders don’t store any data, you can’t use the characteristic relationships, use the IP data slice, or run planning sequences as data is saved. Those features, however, are inherited from the underlying Info-Providers, so you can only choose View in the InfoProvider screen of Transaction RSPLAN.

### 10.2.2 Creating Aggregation Levels for MultiProviders

In this scenario, the aggregation level is created on the MultiProvider not the real-time InfoCube directly. In Transaction RSPLAN, you can see the aggregation level, as shown in Figure 10.41.

If you have experience with MultiProviders, you know that the system provides a characteristic called InfoProvider. In Figure 10.41 you can see it in the Chars. folder. The InfoProvider characteristic can be used to select data in the underlying tables.

For example, if you were to create planning functions on this aggregation level, you could use the InfoProvider characteristic to read prices from the price InfoCube and multiply it by quantity in the revenue InfoCube to calculate revenue.

### 10.2.3 Creating Queries in the MultiProvider Scenario

The InfoProvider characteristic is also needed in the query. Figure 10.42 shows that the Characteristic Restrictions for InfoProvider include the Price Planning and Revenue Planning real-time InfoCubes.

In addition, because you’ll be planning on key figures in both of the real-time InfoCubes, the key figures need to be restricted by the appropriate InfoProvider. For example, in Figure 10.43, you can see that the key figure Sales Qty CC1000 is restricted to the Revenue Planning InfoCube.

---

**Figure 10.40** MultiProvider Structure from Transaction RSA1

**Figure 10.41** Aggregation Level on a MultiProvider
In contrast, the Price key figure is restricted to the Price Planning real-time InfoCube, as shown in Figure 10.44.

To use the query in an EPM workbook, you need to create an embedded SAP BPC model.

**10.2.4 Creating Embedded Models in the MultiProvider Scenario**

To create an embedded model, go to the web client, choose Administration, and click the Models link. Then just choose New, enter the ID and Description of the model, and click Continue.

Next select the Price Planning and Revenue Planning InfoProviders shown in Figure 10.45. You may have expected that you would select the MultiProvider at this point, but the SAP BPC features such as data audit, security, and work status all relate to the InfoProviders themselves, which is why you select them and not the MultiProvider.

After selecting the InfoCubes, choose Next, and then Finish. Now you can use the planning-enabled query in the EPM add-in.
10.2.5  EPM Add-In for the MultiProvider Scenario

You’ll use the same process as before to open Excel and connect to the new model and then use the new query to generate an EPM report (see Figure 10.46). Notice the **Not Assigned** member for material group in row 3. This appears in the result set because the query filter is set to **Juice**, **Table Water**, and **#** (not assigned).

The **#** can be used in a planning scenarios when you want to plan at a higher level and distribute the unassigned values. For example, you could input a sales quantity of 300,000 in row 3 for 2015 and input **#** for material group to run a top-down distribution that would zero out the 300,000 for **#** and distribute it to 150,000 for juice and 150,000 for table water (assuming an even split). In Chapter 12, we’ll discuss this in more detail.

In summary, we’ve discussed how to create MultiProviders, aggregation levels, and planning-enabled queries on MultiProviders. In the web client, you created the model based on the two underlying InfoProviders as well.

Now that we’ve seen how to use a MultiProvider in embedded, let’s take a look at how embedded SAP BPC works with another VirtualProvider.

---

**10.3 Using Composite Providers**

Composite Providers are used for unions just like MultiProviders except that they can be maintained by business users in SAP BW Workspaces. Workspaces are easy-to-use Web Dynpro screens in which business users can model their own ETL solutions in a controlled environment.

Composite providers can also be used to perform joins and, because the join is executed on the SAP HANA database, it’s therefore suitable for mass amounts of data.

Composite providers can include a local provider, for example, and a real-time InfoCube to access global plan data. You can then create an aggregation level on top of the composite provider and, therefore, the touch point to the ABAP Planning Manager in IP, as shown in Figure 10.47.
Composite providers are a good example of how SAP HANA helps to provide data modeling options for planning by making data consumption so much easier.

Now that we've seen how to use a MultiProvider and composite provider in embedded SAP BPC, let's take a look at how to plan with DSOs.

10.4 Using DataStore Objects

Recall that real-time InfoCubes used with MultiProviders are the primary InfoProviders for embedded. DSOs can be used to fit more specialized scenarios such as for price planning and comments. In addition, planning-enabled DSOs handle data changes a bit differently, so we’ll discuss that as well.

To use a DSO in embedded SAP BPC, you’ll need to create the following objects:

- Planning-enabled DSO
- Aggregation level
- Planning functions
- Planning query

Of course, you would normally include the DSO in a MultiProvider and build the aggregation level on the MultiProvider in real-life scenarios.

10.4.1 Creating a Planning-Enabled DSO

To create a DSO, just go to Transaction RSA1, right-click on an InfoArea, and choose Create DSO. Input a name and description, and use a template (here, tsw370s00) if you want to copy in a set of InfoObjects (see Figure 10.48).

After choosing Create, you can change the DataStore type (use the pencil icon) to Direct Update and check the box next to Planning Mode to make it plannable. These settings can’t be changed after there is data in the DSO.

As shown in Figure 10.49, we’ve moved all characteristic into the Key Fields folder, which is required to make it plannable.

If ever you need to expose the DSO data via an analytic view in SAP HANA, just check the box next to External SAP HANA View for reporting. The package in SAP HANA where the view is generated is set up in Transaction SPRO, and the
SAP BW analysis authorizations are automatically created in the SAP HANA privileges.

When the DSO is activated, the system creates an active table that will be used to store the data (i.e., direct update DSOs don’t have three tables like standard DSOs do, and planning-enabled DSOs can’t be loaded via SAP BW ETL).

In Transaction RSPLAN, you can use the characteristic relationships, data slices, and central settings (i.e., run planning sequences as data is saved) for planning-enabled DSOs.

### 10.4.2 Creating an Aggregation Level for a DSO

An aggregation level for a planning-enabled DSO must contain all of the DSO InfoObjects, as shown in Figure 10.50.

The process to create a model for the DSO in the web client is the same as for a single real-time InfoCube, for example. Although you can’t use data audit for a planning-enabled DSO, all of the other features such as work status, BPFs, and data profiles are available.

![Figure 10.49 Creating a Planning-Enabled DSO](image)

**Figure 10.49** Creating a Planning-Enabled DSO

### 10.4.3 EPM Add-In for DSO Scenario

There are no unique settings in planning-enabled queries on DSOs, so we’ll go right to the output in the EPM add-in after running the query (see Figure 10.51).

![Figure 10.50 Aggregation Level for a DSO](image)

**Figure 10.50** Aggregation Level for a DSO

![Figure 10.51 EPM Add-In for a DSO Scenario](image)

**Figure 10.51** EPM Add-In for a DSO Scenario
In general, the EPM add-in functions the same for a DSO as it does for a real-time InfoCube or MultiProvider. However, here the delta buffer for a DSO works in overwrite mode.

### 10.4.4 DSO Delta Buffer

For example, if you change the CM1 contribution margin for IPHONE4 from 12 to 20, and IPHONE5 from 5 to 0, then the DSO only stores the latest data and not the original value of 12 and the incremental change. After saving 20 and 0 from the EPM add-in, Figure 10.52 shows the database values.

**Figure 10.52** The Data in a DSO after Saving Changes

### 10.4.5 DSO Price Planning

If you need to do price planning, consider using a planning-enabled DSO because it offers unique aggregation on parents. Generally speaking, you don’t usually want to aggregate prices because the total would be meaningless.

In plannable DSOs, if the children have different prices of 1.20, 1.30, and 1.50 EUR, the parent (Water) value will display as NONEX in Figure 10.53. On the other hand, if all of the children of a parent have the same prices of 1.20 EUR, then that price (1.20 EUR) is displayed for the parent.

Let’s look at another use case for DSO price planning. If the product prices are different, the planner can input a price for the parent (Water) and that price will be used for the individual products. For example, if you enter 25 for Water, then all three water products would have a price of 25. In addition, if one of the water products doesn’t have a price, then the price for water would display as a *.

**Figure 10.53** DSO Price Planning Example

### 10.4.6 DSO Comments

Aside from the price planning feature, you can also use plannable DSOs to store comments. Comments can be used to record planning assumptions in the database, and then other users can run reports on the comments.

The architecture is shown in Figure 10.54. A planning-enabled DSO is included in a MultiProvider along with a real-time InfoCube. The real-time InfoCube contains the transaction data, and the DSO is used to store the comments. The input form has a separate column for comments and that column is a key figure in the DSO with planning-enabled input.

The length of the comment can be up to 250 characters.
Using Local Providers

The main use case for local providers is to provide a quick ad hoc reporting and planning solution. Local providers are tables in SAP HANA that can be used without any corresponding SAP BW InfoObjects, so it’s potentially a low cost of development alternative that can be configured without any support from IT.

Local providers aren’t new with SAP BPC 10.1. They were originally offered as part of SAP BW Workspaces, which is a small sandbox for business users to perform ETL activities.

To use a local provider in a typical embedded SAP BPC planning scenario, you’ll need the following components:

- Local InfoProvider
- Embedded model

### 10.5.1 Creating a Local Provider and Model

The use case for a local provider is based on a planning user who has a flat file of actual data to upload into a database and then perform both manual input and automatic planning via planning functions to further develop the result set. In this particular example, assume that you don’t want to use any existing InfoObjects as well. The flat file for the example has a header row and two rows of actual data (see Figure 10.55).

To create the local provider, go to the web client, and choose InfoProviders → Local Providers → New in the Administration screen. Enter a name and description, and choose Next.

In the subsequent step (Upload Data File), you need to upload a flat file that is used to derive the structure of the resulting SAP HANA table and to provide the initial result set. The flat file needs to have a .csv extension. As you can see in Figure 10.56, these are the normal settings used in flat file loads to indicate if there is a header row, data separators, decimal indicators, and so forth.

In the next step, Map InfoObjects, you can turn on data audit to track who made changes to the data. This causes the system to add audit fields into the SAP HANA table.

Figure 10.57 shows QTY and REV selected as the key figures. SAP HANA will create these as data fields.
In the Type column, select the data type for each field. For characteristics, you have the following Type options:

- **Character String with Leading Zeros**
- **Date (saved as yyyymmdd)**
- **Time (saved as hhmmss)**
- **InfoObject**

If you select InfoObject, you can select a characteristic and also choose whether to use its conversion routine or not.

For the Key Figure fields, you have the following Type options:

- **Integer**
- **Decimal**
- **Floating Point**
- **InfoObject**

If you select InfoObject here, you can select a key figure.

In this example scenario, use Character String for characteristics and Integer for key figures. While you’re creating the local provider, you can also create the model on the fly. In the next step (Create Model), enter a model ID and description (here, “LMODEL” as shown in Figure 10.58).

Finally, the system displays, among other things, the name of the aggregation level and query that it created automatically (see Figure 10.59). At the time of writing (fall 2014), the query must be created manually.
10.5.2 An Aggregation Level for a Local Provider

You can view the generated aggregation level in Transaction RSPLAN. Figure 10.60 shows the generated field names beginning with @3B. Although they appear under the InfoObjects folder, they don’t exist in the normal SAP BW Data Dictionary but only in SAP HANA. Note that the aggregation level is considered to be an InfoProvider and can be viewed in Transaction RSA1.

![Figure 10.60 Aggregation Level for a Local Provider](image)

**Note**

The description Workspace Aggregation Level of Local Provider serves as a reminder that local providers first came on the scene as a component of workspaces as mentioned earlier.

10.5.3 Query and Workbook for a Local Provider

In the query for the local provider scenario, all of the fields of the aggregation level have been placed into the query. The characteristic restrictions are set to single values, the characteristics in the rows are set to display on master data, and the key figures are set to allow input, as shown in Figure 10.61.

When the query is rendered in the Excel EPM add-in, the data can be manually changed, and planning functions can be executed if desired. Updated values for IPAD have been entered, as you can see in Figure 10.62. QTY was changed from 1000 to 1100, and REV was changed from 10 to 15.

![Figure 10.61 Planning-Enabled Query for the Local Provider Scenario](image)

![Figure 10.62 EPM Excel Add-In for a Local Provider Scenario](image)

10.5.4 View Data in SAP HANA for a Local Provider

When you view the data in SAP HANA, you can see in Figure 10.63 that the data changes of 100 for QTY and 5 for REV are stored in memory.
Because you turned on data audit, the SAP HANA table also includes four audit fields:

- User
- Data mode – PLAN
- Timestamp
- Source

See Chapter 13 for more information on data audit.

**Summary**

In this chapter, we touched on most of the major components for embedded SAP BPC architecture: real-time InfoCubes, MultiProviders, DSOs, composite providers, and local providers. You now know the positioning to use for each one. We moved into the IP planning modeler in Transaction RSPLAN, and you saw how to create aggregation levels and planning functions. We created several planning-enabled queries and described the key planning settings. We then ran the queries in the Excel EPM add-in, and you saw how to run IP functions from the data processing panel.

We’re now ready to take a look at how to perform reporting in embedded SAP BPC, starting with the BEx query in plan mode and moving into the EPM add-in in Excel and the web client.
Contents

Acknowledgments ............................................................................................ 17
Introduction ..................................................................................................... 21

1 Planning and Consolidation at a Glance .................................. 37
   1.1 Planning Basics ................................................................. 38
   1.2 Consolidation Basics ....................................................... 41
   1.3 Summary ........................................................................ 44

PART I Standard SAP PBC

2 The Fundamentals of Standard SAP BPC ................................. 47
   2.1 System Architecture .......................................................... 47
      2.1.1 Microsoft Version ....................................................... 48
      2.1.2 SAP NetWeaver Version ............................................. 50
   2.2 Accessing the System ......................................................... 52
   2.3 SAP Business Planning and Consolidation System ............... 60
      2.3.1 Library Objects ........................................................... 61
      2.3.2 Component Toolbar .................................................... 62
      2.3.3 Support Toolbar .......................................................... 66
      2.3.4 SAP Enterprise Performance Management (EPM) Add-In .................. 69
   2.4 SAP BPC Starter Kit Information ......................................... 77
      2.4.1 IFRS Starter Kit .......................................................... 79
      2.4.2 Other Starter Kits ......................................................... 80
   2.5 Summary ........................................................................ 81

3 Standard SAP BPC Architecture ................................................... 83
   3.1 Terminology and Objects in SAP BPC ................................. 83
   3.2 Basic Data Modeling for SAP BPC ....................................... 87
      3.2.1 Key Data Modeling Questions ...................................... 88
      3.2.2 EnvironmentShell ....................................................... 94
      3.2.3 Configuration/Copy of the EnvironmentShell ................. 97
   3.3 Dimensions and Properties ................................................. 100
<table>
<thead>
<tr>
<th>Contents</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.1 Definition of Dimensions</td>
<td>101</td>
</tr>
<tr>
<td>3.3.2 Dimension Types</td>
<td>103</td>
</tr>
<tr>
<td>3.4 Creating a Dimension</td>
<td>105</td>
</tr>
<tr>
<td>3.4.1 Time Dimension</td>
<td>110</td>
</tr>
<tr>
<td>3.4.2 Category Dimension</td>
<td>111</td>
</tr>
<tr>
<td>3.4.3 Account Dimension</td>
<td>113</td>
</tr>
<tr>
<td>3.4.4 Entity Dimension</td>
<td>116</td>
</tr>
<tr>
<td>3.4.5 Intercompany Dimension</td>
<td>118</td>
</tr>
<tr>
<td>3.4.6 Subtable/Flow Dimension</td>
<td>119</td>
</tr>
<tr>
<td>3.4.7 DataSource/AuditTrail Dimension</td>
<td>120</td>
</tr>
<tr>
<td>3.4.8 Scope Dimension</td>
<td>121</td>
</tr>
<tr>
<td>3.4.9 Rptcurrency Dimension</td>
<td>123</td>
</tr>
<tr>
<td>3.4.10 Inputcurrency Dimension</td>
<td>124</td>
</tr>
<tr>
<td>3.4.11 Additional Dimensions and Properties</td>
<td>126</td>
</tr>
<tr>
<td>3.5 Developing Components of the Dimensions</td>
<td>126</td>
</tr>
<tr>
<td>3.5.1 Dimension Security</td>
<td>126</td>
</tr>
<tr>
<td>3.5.2 Hierarchies in Dimensions</td>
<td>127</td>
</tr>
<tr>
<td>3.5.3 Custom Measure Formulas</td>
<td>128</td>
</tr>
<tr>
<td>3.5.4 Dimension Member Formulas</td>
<td>131</td>
</tr>
<tr>
<td>3.5.5 Owner Properties</td>
<td>136</td>
</tr>
<tr>
<td>3.5.6 Reviewer Properties</td>
<td>137</td>
</tr>
<tr>
<td>3.6 Models</td>
<td>138</td>
</tr>
<tr>
<td>3.6.1 Developing Models</td>
<td>141</td>
</tr>
<tr>
<td>3.6.2 Creating Models</td>
<td>141</td>
</tr>
<tr>
<td>3.7 SAP BW Objects</td>
<td>147</td>
</tr>
<tr>
<td>3.7.1 Architecture of SAP BW Objects</td>
<td>148</td>
</tr>
<tr>
<td>3.7.2 SAP BW Objects Support for SAP BPC for SAP NetWeaver</td>
<td>148</td>
</tr>
<tr>
<td>3.7.3 SAP BW/SAP BPC InfoObjects</td>
<td>152</td>
</tr>
<tr>
<td>3.7.4 SAP BW/SAP BPC InfoCubes</td>
<td>153</td>
</tr>
<tr>
<td>3.8 Summary</td>
<td>156</td>
</tr>
<tr>
<td>4 Reporting in Standard SAP BPC</td>
<td>159</td>
</tr>
<tr>
<td>4.1 Report Considerations</td>
<td>160</td>
</tr>
<tr>
<td>4.2 Introducing the EPM Add-In</td>
<td>170</td>
</tr>
<tr>
<td>4.2.1 Navigating in the EPM Add-In</td>
<td>176</td>
</tr>
<tr>
<td>4.2.2 Using the Context and Current Report Panes</td>
<td>178</td>
</tr>
<tr>
<td>4.2.3 Build a Basic Report</td>
<td>180</td>
</tr>
<tr>
<td>4.3 Using the EPM Ribbon</td>
<td>187</td>
</tr>
<tr>
<td>4.3.1 Reports</td>
<td>188</td>
</tr>
<tr>
<td>4.3.2 Data Analysis</td>
<td>194</td>
</tr>
<tr>
<td>4.3.3 Undo</td>
<td>197</td>
</tr>
<tr>
<td>4.3.4 Data Input</td>
<td>198</td>
</tr>
<tr>
<td>4.3.5 Collaboration</td>
<td>207</td>
</tr>
<tr>
<td>4.3.6 Tools</td>
<td>220</td>
</tr>
<tr>
<td>4.4 Using the EPM Add-In</td>
<td>245</td>
</tr>
<tr>
<td>4.4.1 Report Editor and Report Selection</td>
<td>246</td>
</tr>
<tr>
<td>4.4.2 Local Members</td>
<td>254</td>
</tr>
<tr>
<td>4.4.3 Formatting of Reports</td>
<td>258</td>
</tr>
<tr>
<td>4.4.4 Multi-Reports in a Workbook</td>
<td>262</td>
</tr>
<tr>
<td>4.5 Excel Add-In Advanced Features</td>
<td>265</td>
</tr>
<tr>
<td>4.5.1 Asymmetric Reports</td>
<td>265</td>
</tr>
<tr>
<td>4.5.2 EPM and FPMLX Functions</td>
<td>267</td>
</tr>
<tr>
<td>4.6 PowerPoint and Word Documents in the EPM Add-In</td>
<td>270</td>
</tr>
<tr>
<td>4.7 Summary</td>
<td>273</td>
</tr>
<tr>
<td>5 Data Loading in Standard SAP BPC</td>
<td>275</td>
</tr>
<tr>
<td>5.1 Basic Data Loading with SAP BPC</td>
<td>275</td>
</tr>
<tr>
<td>5.1.1 Data Manager</td>
<td>276</td>
</tr>
<tr>
<td>5.1.2 SAP BW</td>
<td>285</td>
</tr>
<tr>
<td>5.1.3 Flat Files</td>
<td>289</td>
</tr>
<tr>
<td>5.2 Loading Data in SAP BW</td>
<td>293</td>
</tr>
<tr>
<td>5.2.1 Master Data, Text, and Hierarchies</td>
<td>294</td>
</tr>
<tr>
<td>5.2.2 Transactional Data</td>
<td>304</td>
</tr>
<tr>
<td>5.3 Loading Data from SAP BW to SAP BPC</td>
<td>307</td>
</tr>
<tr>
<td>5.3.1 Master Data and Text</td>
<td>308</td>
</tr>
<tr>
<td>5.3.2 Hierarchies</td>
<td>319</td>
</tr>
<tr>
<td>5.3.3 Transactional Data</td>
<td>329</td>
</tr>
<tr>
<td>5.4 Loading Data from a Flat File to SAP BPC</td>
<td>340</td>
</tr>
<tr>
<td>5.4.1 Master Data and Hierarchy</td>
<td>341</td>
</tr>
<tr>
<td>5.4.2 Text</td>
<td>346</td>
</tr>
<tr>
<td>5.4.3 Transactional Data</td>
<td>348</td>
</tr>
<tr>
<td>5.5 Using the Data Manager</td>
<td>355</td>
</tr>
<tr>
<td>5.5.1 Data Manager Configuration</td>
<td>356</td>
</tr>
<tr>
<td>5.5.2 Process Chains</td>
<td>362</td>
</tr>
<tr>
<td>5.6 Uploading Currency Exchange Rates for SAP BPC</td>
<td>366</td>
</tr>
<tr>
<td>5.6.1</td>
<td>Data Flow of Currency Exchange Rates to the Rate Model</td>
</tr>
<tr>
<td>5.6.2</td>
<td>Configuring SAP BW and SAP BPC for Exchange Rate Uploads</td>
</tr>
<tr>
<td>5.7</td>
<td>Advanced Data Loading for SAP BW for SAP BPC</td>
</tr>
<tr>
<td>5.7.1</td>
<td>Analysis of the DELTA INIT Process</td>
</tr>
<tr>
<td>5.7.2</td>
<td>Setup of the SAP BW Process Chain to Execute the SAP BPC Data Manager Package</td>
</tr>
<tr>
<td>5.8</td>
<td>Summary</td>
</tr>
</tbody>
</table>

| 6 | Forecasting, Planning, and Budgeting in Standard SAP BPC | 395 |
| 6.1 | Basic Planning, Forecasting, and Budgeting | 395 |
| 6.1.1 | Driver-Based Approach | 402 |
| 6.1.2 | Top-Down and Bottom-Up Approaches | 402 |
| 6.2 | Script Logic in SAP BPC | 403 |
| 6.2.1 | Worksheet Logic | 405 |
| 6.2.2 | Dimension Member Logic | 406 |
| 6.2.3 | Standard Script Logic Prompts for SAP BPC | 406 |
| 6.2.4 | Use of Script Logic in the Planning Process | 420 |
| 6.2.5 | Allocation Script Logic | 434 |
| 6.2.6 | Script Logic in the Automation of Data Loading | 435 |
| 6.2.7 | BAdIs and the ABAP Program | 438 |
| 6.3 | Generic Planning Process | 442 |
| 6.3.1 | Initial Planned Data | 442 |
| 6.3.2 | Copy Process | 443 |
| 6.3.3 | Executing Calculations | 444 |
| 6.3.4 | Top-Down Planning | 445 |
| 6.3.5 | Bottom-Up Planning | 454 |
| 6.3.6 | Data Transfer Process | 454 |
| 6.4 | Summary | 455 |

| 7 | Consolidation in Standard SAP BPC | 457 |
| 7.1 | Basic Consolidation | 457 |
| 7.1.1 | What Consolidation Is All About | 457 |
| 7.1.2 | Performing Consolidations with SAP BPC | 459 |
| 7.1.3 | Introducing Business Rules | 461 |

| 7.2 | Ownership Data and Elimination Methods | 464 |
| 7.2.1 | The Ownership Process | 464 |
| 7.2.2 | Setting Up Ownership Data | 466 |
| 7.2.3 | The Purchase Method Concept | 471 |
| 7.2.4 | The Proportional and Equity Method Concepts | 472 |
| 7.3 | Setting Up Journal Entries | 473 |
| 7.3.1 | The Business Scenario for Using Journal Entries | 473 |
| 7.3.2 | Creating Journal Templates | 473 |
| 7.3.3 | Controlling Journal Activity | 475 |
| 7.3.4 | Creating Journal Entries | 476 |
| 7.4 | Setting Up and Executing Consolidation Tasks | 477 |
| 7.4.1 | Balance Carry Forward | 478 |
| 7.4.2 | Reclassification | 483 |
| 7.4.3 | Currency Translation for Consolidation | 487 |
| 7.4.4 | Purchase Method Elimination | 496 |
| 7.4.5 | Proportional Method Elimination | 507 |
| 7.4.6 | Equity Method Elimination | 511 |
| 7.4.7 | Ownership Elimination | 513 |
| 7.4.8 | Intercompany Matching | 520 |
| 7.4.9 | Intercompany Elimination | 525 |
| 7.4.10 | US Eliminations | 528 |
| 7.5 | Setting Up Balancing Script Logic for Consolidation | 532 |
| 7.6 | Setting Up and Executing Equity Pick Up | 536 |
| 7.6.1 | Prerequisites for Using Equity Pick Up for Banking | 536 |
| 7.6.2 | Equity Pick Up Business Rule | 537 |
| 7.6.3 | Equity Pick Up Monitor | 541 |
| 7.7 | The Consolidation Monitor | 543 |
| 7.7.1 | Prerequisites to Use the Consolidation Monitor | 543 |
| 7.7.2 | Using the Consolidation Monitor | 543 |
| 7.8 | Summary | 548 |

| 8 | Managing the Standard SAP BPC Process | 549 |
| 8.1 | Security | 549 |
| 8.1.1 | Users | 552 |
| 8.1.2 | Teams | 555 |
| 8.1.3 | Task Profiles | 557 |
| 8.1.4 | Data Access Profiles | 559 |
| 8.1.5 | Security in the EPM Add-In | 563 |
PART II Embedded SAP BPC

9 The Fundamentals of Embedded SAP BPC ......................... 609
9.1 The Evolution of SAP BPC .................................................. 609
9.2 A Primer on SAP HANA ...................................................... 618
9.3 Comparing Standard and Embedded Features .................... 624
9.4 Strategy for Conversion ..................................................... 631
  9.4.1 From IP with PAK Turned On ....................................... 631
  9.4.2 From SAP Business Planning and Simulation ................. 632
  9.4.3 From SAP BPC 7.x ...................................................... 633
  9.4.4 From SAP BPC 10.0 without SAP HANA ....................... 634
  9.4.5 Hybrid/Mixed Scenarios ............................................. 635
  9.4.6 New Customer with Business Ownership Focus .............. 635
  9.4.7 New Customer with EDW Integration Focus ................. 636
9.5 Deployment Options ......................................................... 636
9.6 Summary ........................................................................ 638

10 Embedded SAP BPC Architecture ..................................... 639
  10.1 Setting Up the Embedded Planning Model ....................... 639
    10.1.1 Using Real-Time InfoCubes ...................................... 640
    10.1.2 Using Aggregation Levels ....................................... 643
    10.1.3 Using a Copy Planning Function ............................... 647
    10.1.4 Using Planning-Enabled Queries ............................... 649
    10.1.5 Using Embedded SAP BPC Environments ..................... 654
    10.1.6 Using SAP BPC Models ........................................... 657
    10.1.7 Using the EPM Add-in Workbooks for Basic Planning 
      Scenarios ....................................................................... 659
  10.1.8 Browsing and Compressing Data in SAP BW .................. 664
  10.1.9 Browsing Data and Statistics in SAP HANA ................. 666
  10.2 Using MultiProviders .................................................... 670
    10.2.1 Creating the MultiProvider ...................................... 671
    10.2.2 Creating Aggregation Levels for MultiProviders .......... 672
    10.2.3 Creating Queries in the MultiProvider Scenario .......... 673
    10.2.4 Creating Embedded Models in the MultiProvider 
      Scenario ....................................................................... 675
    10.2.5 EPM Add-In for the MultiProvider Scenario ............... 676
  10.3 Using Composite Providers ............................................ 677
  10.4 Using DataStore Objects ................................................. 678
    10.4.1 Creating a Planning-Enabled DSO ............................. 679
    10.4.2 Creating an Aggregation Level for a DSO ................. 680
    10.4.3 EPM Add-In for DSO Scenario ............................... 681
    10.4.4 DSO Delta Buffer .................................................. 682
    10.4.5 DSO Price Planning ............................................... 682
    10.4.6 DSO Comments .................................................... 683
  10.5 Using Local Providers ................................................... 684
    10.5.1 Creating a Local Provider and Model ....................... 685
    10.5.2 An Aggregation Level for a Local Provider ............... 688
    10.5.3 Query and Workbook for a Local Provider ............... 688
    10.5.4 View Data in SAP HANA for a Local Provider ........... 689
  10.6 Summary ...................................................................... 690

11 Reporting with Embedded SAP BPC ................................. 693
  11.1 System Architecture Review ......................................... 694
  11.2 Web Client Reporting .................................................. 695
  11.3 EPM Add-In in the Embedded Model ............................... 702
    11.3.1 EPM Features ...................................................... 704
    11.3.2 Using EPM ........................................................ 706
  11.4 Comparing the EPM Add-In for Standard and Embedded .... 711
  11.5 EPM Add-In for SAP HANA Data .................................... 712
  11.6 SAP BusinessObjects Analysis, Edition for Microsoft Office 716
  11.7 Summary .................................................................... 720
12 Forecasting, Planning, and Budgeting in Embedded SAP BPC

12.1 Standard IP Planning Functions ........................................................ 722
12.1.1 The Copy Function ........................................................................ 722
12.1.2 The Revaluation Function ................................................................ 724
12.1.3 The Delete Function ......................................................................... 725
12.1.4 The Repost Function ......................................................................... 725
12.2 Executing Planning Functions from Excel .......................................... 726
12.2.1 Data Processing Tab ......................................................................... 726
12.2.2 Pushbuttons .................................................................................... 728
12.3 Disaggregation and Inverse Formulas ..................................................... 730
12.3.1 Disaggregation ................................................................................ 731
12.3.2 Inverse Formulas ........................................................................... 733
12.4 Planning Filters and Planning Sequences .............................................. 736
12.4.1 Planning Filters ................................................................................ 736
12.4.2 Planning Sequences ......................................................................... 738
12.5 FOX Formulas ...................................................................................... 741
12.5.1 Create a Driver-Based Formula ......................................................... 742
12.5.2 Create a FOX Formula with a Condition .......................................... 746
12.5.3 Create a Formula to Calculate Revenue Based on the Prior Year ........ 747
12.5.4 Create a Formula with a Loop ........................................................ 749
12.6 Characteristic Relationships ................................................................. 751
12.6.1 Combination Checking ..................................................................... 751
12.6.2 Characteristic Derivation ................................................................. 754
12.7 Advanced Planning Functions ............................................................... 756
12.7.1 Currency Translation ........................................................................ 756
12.7.2 Unit of Measure Conversion ............................................................ 761
12.7.3 Top-Down Distribution .................................................................... 765
12.7.4 Distribution with Keys ..................................................................... 769
12.7.5 Deletion of Invalid Combinations .................................................... 770
12.7.6 Repost on Basis of Characteristic Relationships .............................. 772
12.7.7 Generate Combinations .................................................................. 773
12.7.8 Custom Planning Functions .............................................................. 775
12.8 Summary ............................................................................................. 779

13 Managing the Embedded SAP BPC Process ..................................... 781

13.1 Security ............................................................................................... 781
13.1.1 Users ............................................................................................ 782
13.1.2 Teams ............................................................................................ 784
13.1.3 Data Access Profiles ...................................................................... 784
13.1.4 Matrix Security .............................................................................. 791
13.1.5 Authorization Objects ..................................................................... 792
13.2 Work Status ....................................................................................... 794
13.2.1 Configuration ................................................................................ 795
13.2.2 Using Work Status ......................................................................... 800
13.2.3 Reporting .................................................................................... 801
13.3 Data Audit .......................................................................................... 804
13.3.1 Configuration ................................................................................. 804
13.3.2 Using Embedded Data Audit ......................................................... 807
13.4 Transports .......................................................................................... 811
13.5 Summary ............................................................................................ 813

14 Migrating to SAP BPC 10.1 ................................................................. 815

14.1 SAP BPC 7.5 to Standard SAP BPC 10.1 ........................................... 817
14.2 SAP BPC 10.0 to Standard SAP BPC 10.1 ........................................... 822
14.3 Integration Planning to Embedded SAP BPC 10.1 .............................. 824
14.4 SAP BPC 10.0 to Embedded SAP BPC ................................................. 825
14.5 SAP BPC 10.0 to Mixed Standard/Embedded SAP BPC 10.1 ............ 826
14.6 Summary ............................................................................................ 826

The Authors ............................................................................................ 827
Index ......................................................................................................... 829
Index

$INFOCUBE, 75
%APPLICATION%, 432
%APPSET%, 432
%ENTITY_DIM%, 362
%TIME_SET%, 418
%YEAR%, 432
*ADD */ENDADD, 432
*ADD/*ENDADD, 410
*COLX, 349
*COMMIT, 411, 417, 423
*DESTINATION_APP, 415
*ENDALLOCATION, 435
*ENDWHEN, 414
*INCLUDE, 411
*LOOKUP, 416
*LOOKUP (MODEL), 416
*MVAL, 331
*NEWCOL, 331
*NEWCOLX, 349
*REC, 412, 428, 429
*RUNALLOCATION, 434
*SELECT, 413, 431
*WHEN, 413
*WHEN/*ENDWHEN, 413
*WHEN/*IS/*ENDWHEN, 429
*XDIM_ADDMEMBERSET, 416, 428
*XDIM_FILTER, 428
*XDIM_MEMBERSET, 412, 415, 422, 427
.NET, 48, 50
/CMPB, 55, 154, 364
/CMPB/LOAD_DELTA_IP, 336
0GL_ACCOUNT, 300, 319
0GLACCEXT, 377
0TCAACTVT, 786
0TCAIPROV, 786
0TCAVALID, 786
501, 281

A

ABAP code, 293
ABAP program, 438
ABAP/BAdI, 420
ACCOUNT, 108
Account and key figure-based InfoCubes, 615
Account dimension, 113
Account-based calculations, 463
Account-based view, 85
ACCTYPES, 105, 115
Activate Local Member Recognition option, 230, 254, 255
Activate Member Recognition checkbox, 180
Activate Member Recognition option, 230
Activate Metadata Cache option, 238
Active cell, 204
Activities, 571, 585
audit, 581, 582
monitor, 569
report, 585
Add Comment option, 204
Add Conversion Sheet option, 284
Add Data Link option, 222, 223
Add dimension, 164
Add Member Link option, 222
Add Report Link option, 222
Add/Remove button, 576
Adding comments, 422
Admin console, 816
Admin Package option, 352
ADMINISTRATION, 64
Administration, 436
Administration activity audit, 582
Adobe Flex web, 612
Advanced, 190
Aggregate overwrite option, 337
Aggregation level, 616, 639, 643, 647, 649, 669, 670, 672, 677, 678, 680, 685, 686, 688, 693, 694, 710
create, 643, 646
Allocation, 395, 434, 456, 621
Allocation Script Logic, 434
ALLOCATION.LGF, 409
Allow Member Formulas checkbox, 109
AMOUNTDECIMALPOINT, 311

829
Custom Member Default Format option, 258
Custom planning functions, 775
Customer dimension with properties, 342

D
Data, 315
Data access profile, 550, 551, 552, 555, 556, 559, 561, 587, 659, 784, 788
Data analysis, 194
Data audit, 581, 582, 590, 616, 631, 655, 687, 804, 822, 824, 825
Dataflow, 583
Data information, 581
Data process, 582
Data purge, 584
Data changes, 588, 589, 807
Data collection component, 362
Data connection, 263, 264
Data flow, 276, 286
Data flow of currency exchange, 368
Data input options, 198, 230
Data Link Symbol Color option, 239
Data loading in SAP BW, 293
Data Manager, 270, 276, 277, 291, 310, 355, 356, 408, 705, 711, 720, 819
Data configuration, 356
Data upload dialog, 279
Data Manager Group, 279, 342
Data Manager Import, 592
Data Manager package, 379, 423, 440
Data mapping, 93
Data modeling, 93
Data preview, 291
Data Preview option, 279, 342
Data processing, 705, 711, 717, 726, 727, 728, 761
Data Processing tab, 712, 726
Data profile, 618, 631
Data slice, 645
Data transfer, 395
Data transfer process (DTP), 304, 598
Data type, 314, 333, 749
Data validation, 460
Database migration, 825
Database procedure, 778
DatabaseSource, 286, 287, 294, 296, 297, 298, 304, 379, 598
DatabaseObject, 639, 678, 680, 682, 683, 702, 712
DatabaseSource/AuditTrail dimension, 120
Database store object (DSO), 287, 304, 368
Default values, 650
DEFAULT.LGF, 407, 408
Defer Layout Update option, 184, 186
Delete function, 722, 725
Delete invalid characteristic combinations, 751
Delete invalid combinations, 770
Delete reports, 191
Delimited or Fixed Width options, 291
Delimiter, 311
Delta buffer, 682
DELTA INIT package, 379
Delta initialization, 379
Delta loading process, 281
Delta Update option, 305
Delta updating, 296
Denied, 560, 561
Deployment options, 631
Depreciation calculation, 444
Derivation, 752
Descendant option, 195, 230
DETAILS, 339
Dimension, 84, 143, 211, 275, 294, 397
SAP BPC, 281
Dimension formulas, 131
Dimension member formula, 404, 623
Dimension member logic, 406
Dimension Member/Property Formatting section, 258, 262
Dimension security, 126
Dimension type, 103
Dimension/field, 168, 260
Disaggregation, 619, 721, 731, 733, 765, 769
Display component, 355
Display Disaggregation Warning option, 240
Display EPM Context Bar option, 237
Display input template, 729
Display Local Connections option, 239
Display MDX option, 243
Distributions, 207, 211, 721, 731, 768
by key, 765
Collect option, 208
Distribute option, 207
non-assigned, 766
Now option, 215
option, 211
template, 214
with keys, 769, 770
with reference, 765, 766, 769
Do Not Load Members at Connection, 76
Do Not Load Members at Connection checkbox, 174
Do not Store Connection option, 169
Document type, 209
Documents feature, 66
Documents option, 97
Download Center option, 68
Download Data option, 279
Drill Through option, 221, 222, 227
Drill-through, 227, 705, 711, 720
Driver, 396
Driver-based formula, 742
Driver-based planning, 402
Drivers and rates, 138
Dynamic Package Script, 357
Embedded environment, 640, 654, 670
Embedded planning model, 639, 640, 670, 679
Embedded web client, 659
Embedded web client administration, 656
Enable Control by Model parameters, 572
Enable Double-Click option, 182, 237
Enable work status, 795
Enable Work Status by Model section, 565
ENABLE_FIXED_CUBENAME, 59
Enhancement spot, 439
UJ_CUSTOM_LOGIC, 439
Enterprise Cloud, 636
Entity dimension, 116
Environment, 86, 88, 99, 661
connection, 106
offline, 603
option, 68
EnvironmentShell, 53, 57, 83, 84, 89, 94, 97, 408
alterations, 95
EPM – Distribution Template screen, 212
EPM 10.1, 69, 170
EPM add-in, 159, 160, 163, 166, 170, 172, 176, 183, 188, 265, 267, 404, 405, 436,
563, 568, 592, 611, 612, 613, 618, 632,
633, 639, 654, 659, 676, 677, 681, 682, 688
description, 270
formatting parameters, 261
formatting sheet, 168, 260
frontend, 551
functions, 225, 274
reporting tool, 172
ribbon, 187, 188, 190, 193, 201, 661
EPM context menu, 178, 179, 180, 185
EPM Office add-in for Excel, 277, 310
EPM Office Add-in options, 68
EPM SAP Template, 168
EPM Workbook, 640, 670, 679
EPM Worksheet option, 231
EPMAddInAutomation, 730
EPMComparison, 269
EPMFormattingSheet, 260
EPMMemberDesc, 269
Equal distribution, 732, 767
Equity method, 42  
\textit{business scenario}, 511  
elimination, 511  
Equity Pick Up, 536  
\textit{business rule}, 537  
for banking, 536  
Equity Pick Up Monitor, 541  
Exposed, 183  
EvDRE, 169, 183, 221, 822  
Migration option, 245  
templates, 634  
Excel add-in advanced features, 265  
Exchange rate, 289, 290  
Excel add-in advanced features, 265  
Excel add-in advanced features, 265  
Fixed data format, 298  
Flash Objects option, 243  
Flat file, 275, 291, 317, 340  
Flow, 119  
FOREACH, 750  
Forecasting, 396  
FORMAT, 311  
Format, 164, 315  
Formatting of reports, 258  
Formatting sheet, 260  
Formatting tab, 232  
Formula column, 283, 284, 331  
Formula name field, 129  
FOX, 621, 721  
FOX formula, 741  
FOX formula with a condition, 746  
FPMXL, 267  
FPMXLClient, 168, 728  
FPMXLClient function, 225  
FPMXLClient.TechnicalCategory.EPM, 168  
function, 265, 268, 274  
Free characteristics, 650  
Freeze Data Refresh option, 238, 244  
Full optimization, 152  
Function arguments, 269  
Function type, 621  
Functional area, 330  
FX restatement, 361  
FX translation, 362  
FX.TRANS.LGF, 409  
EXTRANS, 436  
EXTRANS logic script, 437  
First consolidation entries, 472  
Fixed and variable fields, 214  
Fixed data format, 298  
Flash Objects option, 243  
Flat file, 275, 291, 317, 340  
Flow, 119  
FOREACH, 750  
Forecasting, 396  
FORMAT, 311  
Format, 164, 315  
Formatting of reports, 258  
Formatting sheet, 260  
Formatting tab, 232  
Formula column, 283, 284, 331  
Formula name field, 129  
FOX, 621, 721  
FOX formula, 741  
FOX formula with a condition, 746  
FPMXL, 267  
FPMXLClient, 168, 728  
FPMXLClient function, 225  
FPMXLClient.TechnicalCategory.EPM, 168  
function, 265, 268, 274  
Free characteristics, 650  
Freeze Data Refresh option, 238, 244  
Full optimization, 152  
Function arguments, 269  
Function type, 621  
Functional area, 330  
FX restatement, 361  
FX translation, 362  
FX.TRANS.LGF, 409  
EXTRANS, 436  
EXTRANS logic script, 437  
Field to be changed, 724  
Field for conditions, 723  
Filter, 644  
Filtering, 252  
Filtering tab, 247, 250  
Filters, 721, 726  
Finance model, 167  
Financial expenses, 397  
Financial Information Management, 69  
Financial model, 400  
Find Comments option, 204, 205  
Finish editing, 166  
FireFly, 695, 702, 706  
First consolidation entries, 472  
Fixed and variable fields, 214  
Fixed data format, 298  
Flash Objects option, 243  
Flat file, 275, 291, 317, 340  
Flow, 119  
FOREACH, 750  
Forecasting, 396  
FORMAT, 311  
Format, 164, 315  
Formatting of reports, 258  
Formatting sheet, 260  
Formatting tab, 232  
Formula column, 283, 284, 331  
Formula name field, 129  
FOX, 621, 721  
FOX formula, 741  
FOX formula with a condition, 746  
FPMXL, 267  
FPMXLClient, 168, 728  
FPMXLClient function, 225  
FPMXLClient.TechnicalCategory.EPM, 168  
function, 265, 268, 274  
Free characteristics, 650  
Freeze Data Refresh option, 238, 244  
Full optimization, 152  
Function arguments, 269  
Function type, 621  
Functional area, 330  
FX restatement, 361  
FX translation, 362  
FX.TRANS.LGF, 409  
EXTRANS, 436  
EXTRANS logic script, 437  
First consolidation entries, 472  
Fixed and variable fields, 214  
Fixed data format, 298  
Flash Objects option, 243  
Flat file, 275, 291, 317, 340  
Flow, 119  
FOREACH, 750  
Forecasting, 396  
FORMAT, 311  
Format, 164, 315  
Formatting of reports, 258  
Formatting sheet, 260  
Formatting tab, 232  
Formula column, 283, 284, 331  
Formula name field, 129  
FOX, 621, 721  
FOX formula, 741  
FOX formula with a condition, 746  
FPMXL, 267  
FPMXLClient, 168, 728  
FPMXLClient function, 225  
FPMXLClient.TechnicalCategory.EPM, 168  
function, 265, 268, 274  
Free characteristics, 650  
Freeze Data Refresh option, 238, 244  
Full optimization, 152  
Function arguments, 269  
Function type, 621  
Functional area, 330  
FX restatement, 361  
FX translation, 362  
FX.TRANS.LGF, 409  
EXTRANS, 436  
EXTRANS logic script, 437  
Field to be changed, 724  
Field for conditions, 723  
Filter, 644  
Filtering, 252  
Filtering tab, 247, 250  
Filters, 721, 726  
Finance model, 167  
Financial expenses, 397  
Financial Information Management, 69  
Financial model, 400  
Find Comments option, 204, 205  
Finish editing, 166  
FireFly, 695, 702, 706
Index

VBA, 274, 659
Versioning, 395, 443
Vertical line, 169
View Format option, 260
View Schedule Status, 277
Virtual InfoCube, 163, 173
Virtual model, 695
VMWare, 53
Volume, 397

W

Warehousing, 398
Warn If Navigation Will Break Dynamic Selection option, 238
Web Administration link, 52
Web Administration screen, 60
Web client start page, 60
Web Documents option, 97
Web services, 171, 175, 563
Weight, 200
Wizard, 555
Word document, 270, 271
Work status, 329, 564, 566, 570, 588, 616, 631, 794, 824, 825
reporting, 801

Work status (Cont.)
settings, 566
Workbook Location Local option, 239
Workbook Location Planning and Consolidation Server option, 239
Worksheet generation, 191
Worksheet Generation option, 191
Worksheet logic, 404, 405
Worksheet Type options, 231
Workspace, 67, 166, 575
Write, 561
WRITEMODE, 390

X

XDIM_ADDMEMBERSET, 416
XDIM_MEMBERSET, 423

Y

YTD, 131, 288
YTD format, 290

Z

ZBPC_LOAD_DATASRC_MDATTR, 364
Peter Jones is a BI/BPC Sr. Application Consultant with MI6 Solutions LLC specializing in the areas of CO (Controlling), Enterprise Controlling (EC), Business Intelligence/Business Objects (BI/BOBJ), Strategic Enterprise Management (SEM) and Business Planning and Consolidation (BPC). He has over 16 years of consulting and educational experience in a variety of strategic and leadership roles, focused on global architecture and strategies for BI, ERP, and BPC.

Tim Soper is a senior SAP Educational Consultant and has been with SAP for over 15 years. His areas of expertise are FI, CO, BI, SEM, and BPC. He has been involved with the development of numerous courses, including CO, BI, SEM and BPC and is a course owner for several of them. As an instructor, he has taught courses at all levels from novice to advanced workshops.

Implementing SAP Business Planning and Consolidation

We hope you have enjoyed this reading sample. You may recommend or pass it on to others, but only in its entirety, including all pages. This reading sample and all its parts are protected by copyright law. All usage and exploitation rights are reserved by the author and the publisher.