Exporting Data in PBCS With Business Rules

Introduction

If your environment is a cloud product, whether it be PBCS or ePBCS, one thing that is critical to understand is the backups produced in the Migration area, may not be what you think. Learning this after the fact may have negative consequences on your ability to restore data. In the migration, the Essbase Data section is a copy of the pag, ind, and otl files. When this is used to restore data, it restored the entire database. This includes data and metadata. This may be OK for many situation, but it won't help you if

- only specific data is required to be restored
- specific data has changed and needs to be excluded from the restore
- corruption exists in the database and all data is required to be restored
- The pag files that hold the data are not readable
- The size of the backup is quite large as it includes all data, and upper level data is normally exponentially larger than just level 0 data

Text Data Export

Business Rules can be written to export data to the Inbox/Outbox that is delimited with a few formatting options. The entire database can be included. With fix statements, specific data can be isolated. So, forecast could be exported to a file, plan another, and actuals a third. Specific accounts, entities, and/or products can be isolated in cases when specific data was inadvertently changed or deleted. This file is a text file that can be opened in any text editor, Microsoft Excel, a database, or any other application that you open text files to view or manipulate.

Example Business Rule

```
/* Set the export options */
 SET DATAEXPORTOPTIONS
 {
 DataExportLevel LEVEL0;
 DataExportDynamicCalc OFF;
 DataExportNonExistingBlocks OFF;
 DataExportDecimal 4;
 DataExportPrecision 16;
 DataExportColFormat ON;
 DataExportColHeader Period;
 DataExportDimHeader ON;
 DataExportRelationalFile ON;
 DataExportOverwriteFile ON;
 DataExportDryRun OFF;
 };

 FIX(@Relative("Account", 0),
     @Relative("Years", 0),
     @Relative("Scenario", 0),
     @Relative("Version", 0),
     @Relative("Entity", 0),
     @Relative("Period", 0),
     @Relative("custom_dim_name_1", 0),
     @Relative("custom_dim_name_1", 0),
     @Relative("custom_dim_name_1", 0))

  DATAEXPORT "File" "," "/u03/lcm/filename_xyz.txt" "";

 ENDFIX
```

Some Hints

There are a few things that you may encounter and be a little confused about, so the following are a few things that might help.

1. To see the data export, it must be exported to /u03/lcm/, which is the equivalent of your inbox. Any file name can be used.
2. Setting DataExportLevel to 0 will export the level 0 blocks, not the level 0 members. If there are any stored members in any of your dense dimensions, they will be exported unless the dimension is also in the fix to include ONLY level 0 members.
3. The fix statement works the same as a fix statement in any business rule, so the data to be exported can be easily defined.
4. My experience exporting dynamic calculated members drastically increases the time of the export.
5. The export options are all pretty logical. Some work in conjunction with each other and others are ignored depending on dependent setting values. These are documented for version 11.1.2.4 here.
6. This process can be automated with EPM Automate and include the download and time stamp of the backup for later use.

Conclusion

There are benefits to both types of backups. My preference is to either run both nightly, or run just the Business Rule. By having both, the administrator has the option of restoring the data as needed, in the way that is most effective. Having both provides the ultimate flexibility. If space is an issue, exclude the data option in the Migration and just run the business rule.

From Oracle’s Documentation

DataExportLevel ALL | LEVEL0 | INPUT

- **ALL**—(Default) All data, including consolidation and calculation results.
- **LEVEL0**—Data from level 0 data blocks only (blocks containing only level 0 sparse member combinations).
- **INPUT**—Input blocks only (blocks containing data from a previous data load or grid client data-update operation). This option excludes dynamically calculated data. See also the DataExportDynamicCalc option.

In specifying the value for the DataExportLevel option, use these guidelines:

- The values are case-insensitive. For example, you can specify LEVEL0 or level0.
- Enclosing the value in quotation marks is optional. For example, you can specify LEVEL0 or “LEVEL0”.
- If the value is not specified, Essbase uses the default value of ALL.
- If the value is incorrectly expressed (for example, LEVEL 0 or LEVEL2), Essbase uses the default value of ALL.

Description

Specifies the amount of data to export.

DataExportDynamicCalc ON | OFF

- **ON**—(Default) Dynamically calculated values are included in the export.
- **OFF**—No dynamically calculated values are included in the report.

Description

Specifies whether a text data export excludes dynamically calculated data.

Notes:

- Text data exports only. If DataExportDynamicCalc ON is encountered with a binary export (DATAEXPORT BINFILE …) it is ignored. No dynamically calculated data is exported.
- The DataExportDynamicCalc option does not apply to attribute values.
- If DataExportLevel INPUT is also specified and the FIX statement range includes sparse Dynamic Calc members, the FIX statement is ignored.

DataExportNonExistingBlocks ON | OFF

- **ON**—Data from all possible data blocks, including all combinations in sparse dimensions, are exported.
- **OFF**—(Default) Only data from existing data blocks is exported.

Description

Specifies whether to export data from all possible data blocks. For large outlines with a large number of members in sparse dimensions, the number of potential data blocks can be very high. Exporting Dynamic Calc members from all possible blocks can significantly impact performance.

DataExportPrecision n

n (Optional; default 16)—A value that specifies the number of positions in exported numeric data. If n < 0, 16-position precision is used.

Description

Specifies that the DATAEXPORT calculation command will output numeric data with emphasis on precision (accuracy). Depending on the size of a data value and number of decimal positions, some numeric fields may be written in exponential format; for example, 678123e+008. You may consider using DataExportPrecision for export files intended as backup or when data ranges from very large to very small values. The output files typically are smaller and data values more accurate. For output data to be read by people or some external programs, you may consider specifying the DataExportDecimal option instead.

Notes:

- By default, Essbase supports 16 positions for numeric data, including decimal positions.
- The DataExportDecimal option has precedence over the DataExportPrecision option.

Example

```
SET DATAEXPORTOPTIONS
    {
    DataExportPrecision 6;
    DataExportLevel ALL;
    DataExportColHeader "Measures";
    DataExportDynamicCalc ON;
    };
    DATAEXPORT "File" "," "output1.out";
```

Initial Data Load Values

```
"Sales" "COGS" "Margin" "Marketing" "Payroll" "Misc" "Total
Expenses" "Profit" "Opening Inventory" "Additions" "Ending
Inventory" "Margin %" "Profit %"
"100-10" "New York"
"Jan" "Actual" 678123456.0 271123456.0 407123456.0
94123456789012345 6.0 51123456.0 0 145123456.0 262123456.0
2101123456.0 644123456.0 2067123456.0 60123456.029
38123456.6430
"Feb" "Actual" 645123 258123 3871234 9012345 5112345 112345678
14212345 24512345 2067123456 61912345 20411234 601234
37123456.98
"Mar" "Actual" 675 270 405 94 51 1 146 259 2041 742 2108 60
38.37037037037037
"Qtr1" "Actual" 1998 799 1199 278 153 2 433 766 2101 2005 2108
60.01001001001001 38.33833833833834
```

Exported Data Format

```
"Sales","COGS","Margin","Marketing","Payroll","Misc","Total Expenses","Profit","Opening Inventory","Additions","Ending Inventory","Margin %","Profit %","Profit per Ounce","100-10","New York"
"Jan","Actual",6.78123e+008,2.71123e+008,4.07e+008,9.41235e+017,5.11235e+007,0,9.41235e+017,-9.41235e+017,2.10112e+009,6.44123e+008,2.06712e+009,60.0186,-1.388e+011,-7.84362e+016
"Feb","Actual",645123,258123,387000,9.01235e+006,5.11235e+006,1.12346e+008,1.2647e+008,-1.26083e+008,2.06712e+009,6.19123e+007,2.04112e+007,59.9886,-19544.1,-1.05069e+007
"Mar","Actual",675,270,405,94,51,1,146,259,2041,742,2108,60,38.3704,21.5833
```

DataExportDecimal n

Where n is a value between 0 and 16.

If no value is provided, the number of decimal positions of the data to be exported is used, up to 16 positions, or a value determined by the DataExportPrecision option if that is specified.

Description

Specifies that the DATAEXPORT calculation command will output numeric data with emphasis on legibility; output data is in straight text format. Regardless of the number of decimal positions in the data, the specified number is output. It is possible the data can lose accuracy, particularly if the data ranges from very large values to very small values, above and below the decimal point.

Notes:

- By default, Essbase supports 16 positions for numeric data, including decimal positions.
- If both the DataExportDecimal option and the DataExportPrecision option are specified, the

DataExportPrecision option is ignored.

Example

```
SET DATAEXPORTOPTIONS
    {DataExportDecimal 4;
    DataExportLevel "ALL";
    DataExportColHeader "Measures";
    DataExportDynamicCalc ON;
    };
    DATAEXPORT "File" "," "output1.out";
```

Initial Data Load Values

```
"Sales" "COGS" "Margin" "Marketing" "Payroll" "Misc" "Total
Expenses" "Profit" "Opening Inventory" "Additions" "Ending
Inventory" "Margin %" "Profit %"
"100-10" "New York"
"Jan" "Actual" 678123456.0 271123456.0 407123456.0
941234567890123456.0 51123456.0 0 145123456.0 262123456.0
2101123456.0 644123456.0 2067123456.0 60123456.029
38123456.6430
"Feb" "Actual" 645123 258123 3871234 9012345 5112345 112345678
14212345 24512345 2067123456 61912345 20411234 601234
37123456.98
"Mar" "Actual" 675 270 405 94 51 1 146 259 2041 742 2108 60
38.37037037037037
"Qtr1" "Actual" 1998 799 1199 278 153 2 433 766 2101 2005 2108
60.01001001001001 38.33833833833834
```

Exported Data Format

```
"Sales","COGS","Margin","Marketing","Payroll","Misc","Total
Expenses","Profit","Opening Inventory","Additions","Ending
Inventory","Margin %","Profit %","Profit per Ounce"
"100-10","New York"
"Jan","Actual",678123456.0000,271123456.0000,407000000.0000,94
1234567890123520.0000,51123456.0000,0.0000,941234567941246980.
0000,-941234567534246910.0000,2101123456.0000,644123456.0000,2
067123456.0000,60.0186,-138799883591.4395,-78436213961187248.0
000
"Feb","Actual",645123.0000,258123.0000,387000.0000,9012345.000
0,5112345.0000,112345678.0000,126470368.0000,-126083368.0000,2
```

```
067123456.0000,61912345.0000,20411234.0000,59.9886,-19544.0820
,-10506947.3333
"Mar","Actual",675.0000,270.0000,405.0000,94.0000,51.0000,1.00
00,146.0000,259.0000,2041.0000,742.0000,2108.0000,60.0000,38.3
704,21.5833
```

Output Format Options

DataExportColFormat ON | OFF

- ON—The data is output in columnar format.
- OFF—Default. The data is output in non-columnar format.

Description

Specifies if data is output in columnar format. Columnar format displays a member name from every dimension; names can be repeated from row to row, enabling use by applications other than Essbase tools. In non-columnar format, sparse members identifying a data block are included only once for the block. Non-columnar export files are smaller, enabling faster loading to an Essbase database.

Notes

Do not use the DataExportColFormat option in combination with the DataExportRelationalFile option, which already assumes columnar format for files destined as input files to relational databases.

Example

```
SET DATAEXPORTOPTIONS
 {
 DATAEXPORTCOLFORMAT ON;
 };
 FIX("100-10", Sales, COGS, Jan, Feb, Mar, Actual, Budget)
 DATAEXPORT "File" "," "d:\temp\test2.txt" ;
ENDFIX;
```

DataExportColHeader dimensionName

Description

Specifies the name of the dense dimension that is the column header (the focus) around which other data is referenced in the export file. Use the DataExportColHeader option only when you export data to a text file. For example, if from Sample Basic the Year dimension is specified, the output data starts with data associated with the first member of the Year dimension: Year. After all data for Year is output, it continues with the second member: Qtr1, and so on.

Notes

MaxL, ESSCMD, and Essbase exports do not provide a similar capability. With these methods, Essbase determines the focal point of the output data.

Exporting through Report Writer enables you to specify the header in the report script.

Example

```
SET DATAEXPORTOPTIONS {DATAEXPORTCOLHEADER Scenario;};
```

Specifies Scenario as the page header in the export file. The Scenario dimension contains three members: Scenario, Actual, and Budget. All Scenario data is shown first, followed by all Actual data, then all Budget data.

DataExportDimHeader ON | OFF

- ON—The header record is included.
- OFF—Default. The header record is not included.

Description

Use the DataExportDimHeader option to insert the optional header record at the beginning of the export data file. The header record contains all dimension names in the order as they are used in the file. Specifying this command always writes the data in “column format”.

Example

```
SET DATAEXPORTOPTIONS
 {
 DATAEXPORTLEVEL "ALL";
 DATAEXPORTDIMHEADER ON;
 };
FIX("100-10", "New York", "Actual")
 DATAEXPORT "File" "," "E:\temp\2222.txt" ;
ENDFIX;
```

Specifying the DataExporttDimHeader ON option while exporting Sample Basic writes the data in column format, with common members repeated in each row. The data begins with a dimension header, as shown in the first two rows of the example file below:

```
"Product","Market","Year","Scenario","Measures"
"Sales","COGS","Marketing","Payroll","Misc","Opening
Inventory","Additions","Ending Inventory"
"100-10","New
York","Jan","Actual",678,271,94,51,0,2101,644,2067
"100-10","New
York","Feb","Actual",645,258,90,51,1,2067,619,2041
"100-10","New
York","Mar","Actual",675,270,94,51,1,2041,742,2108
"100-10","New
York","Apr","Actual",712,284,99,53,0,2108,854,2250
"100-10","New
York","May","Actual",756,302,105,53,1,2250,982,2476
"100-10","New
York","Jun","Actual",890,356,124,53,0,2476,1068,2654
"100-10","New
York","Jul","Actual",912,364,127,51,0,2654,875,2617
"100-10","New
York","Aug","Actual",910,364,127,51,0,2617,873,2580
"100-10","New
```

```
York","Sep","Actual",790,316,110,51,1,2580,758,2548
"100-10","New
York","Oct","Actual",650,260,91,51,1,2548,682,2580
"100-10","New
York","Nov","Actual",623,249,87,51,0,2580,685,2642
"100-10","New
York","Dec","Actual",699,279,97,51,1,2642,671,2614
```

DataExportRelationalFile ON | OFF

- ON—The output text export file is formatted for import to a relational database.
    - Data is in column format; sparse member names are repeated. (The DataExportColFormat option is ignored.)
    - The first record in the export file is data; no column heading or dimension header is included, even if specified. (The DataExportColHeader and DataExportDimHeader options are ignored.)
    - Missing and invalid data is skipped, resulting in consecutive delimiters (commas) in the output. The optional “missing_char” parameter for DATAEXPORT is ignored
- OFF—Default. The data is not explicitly formatted for use as input to a relational database.

Description

Using the DataExportRelationalFile option with DATAEXPORT enables you to format the text export file to be used directly as an input file for a relational database.

Example

```
SET DATAEXPORTOPTIONS {
 DataExportLevel "ALL";
 DataExportRelationalFile ON;
};

FIX (Jan)
```

```
 DATAEXPORT "File" "," c:\monthly\jan.txt
ENDFIX;
```

Processing Options

DataExportOverwriteFile ON | OFF

- ON—The existing file with the same name and location is replaced.
- OFF—Default. If a file with the same name and location already exists, no file is output.

Description

Manages whether an existing file with the same name and location is replaced.

DataExportDryRun ON | OFF

- ON—DATAEXPORT and associated commands are run, without exporting data.
- OFF—Default. Data is exported

Description

Enables running the calculation script data export commands to see information about the coded export, without exporting the data. When the DataExportDryRun option value is ON, the following information is written to the output file specified in the DATAEXPORT command:

- Summary of data export settings
- Info, Warning, and Error messages
- Exact number of blocks to be exported
- Estimated time, excluding I/O time.

Notes

- The DataExportDryRun option does not work with exports to relational databases.

- If you modify the script for reuse for the actual export, besides removing the DataExportDryRun option from the script you may want to change the name of the export file.

Example

```
SET DATAEXPORTOPTIONS
 {
 DataExportLevel "ALL";
 DataExportColHeader "Measures";
 DataExportColFormat ON;
 DataExportDimHeader ON;
 DataExportDynamicCalc OFF;
 DataExportDecimal 0;
 DataExportDryRun ON;
 DataExportOverwriteFile ON;
 };

FIX("Qtr1")
 DATAEXPORT "File" "," "E:\temp\log.txt" ;
ENDFIX;
```