

# SELECT<sup>®</sup> Online

The Journal of the International Oracle Users Group

Select Journal  
Home

Regular Columns

Past Issues

General Information

IOUG Home



## Select Magazine - October 1999

Volume 7, No. 1

### Bridging the gap between Oracle Financial Analyzer and Oracle General Ledger

By Mark Hoferle

With the proliferation of the Oracle line of applications across a variety of industries, there is an increasing number of companies using either the Oracle General Ledger (OGL) or Oracle Financial Analyzer (OFA), or both. This article hopes to describe the link between these two applications by describing "what's on the other side of the fence" for those people who are using only one of the two applications and don't have a solid understanding of how one works with the other. For those people who are currently using both applications and have linked the two together, this article will offer suggestions on how to take full advantage of the link and to describe some future enhancements to the process.

The link is a systematic, fully functioning out-of-the-box, supported process of transferring data back and forth between OGL and Financial Analyzer. OFA is primarily used as a Budgeting and Forecasting tool by offering more of a high-level data perspective with exceptional analytical capabilities. OGL, the transactional-based system, can supply Financial Analyzer with the values from the Chart of Accounts to define the OFA data sets, and can send the Actual results to OFA to be used for variance reporting. Working in the other direction, the Budget and Forecast results can be sent to the OGL from Financial Analyzer for variance reporting in the G/L.

Once the link is established, people often forget about the power of OFA. Tasks that traditionally were handled by the GL can easily, and sometimes more efficiently, be handled by Financial Analyzer. OFA can quickly and easily perform intricate allocation routines using the powerful features of its Express database engine. Financial Analyzer can also meet a company's formal monthly financial reporting requirements with its easy-to-use reporting features. New features involving the link are just now available (with more scheduled to be released) such as drill through to journal entries from Financial Analyzer and improved write-back features to OGL. These features promise to make the link even easier to use, and place more analytical power in the hands of the user.

#### What is Financial Analyzer?

Financial Analyzer is a single application which is part of Oracle's line of Data Warehouse/On-Line Analytical Processing (OLAP) application suite. OFA is a pre-built financial reporting, analysis, planning and forecasting tool designed for multiple users to create and analyze their Budget and Forecast results. OFA is a fully scalable product that can include hundreds of users across different locations within a company, and runs on either a UNIX or NT platform.

This distributed environment is what makes OFA such a powerful tool. The application uses a single shared database that serves as the "data repository" for the system to centrally store data. A single user is designated as the application administrator to define the data sets within the application, and to create and grant access to users. The administrator is also responsible for loading data into the system (such as Actual information) from either legacy systems or the OGL.

Using the functionality pre-built into the application, each user systematically communicates with this shared database to gain access to the data sets and to submit their Forecast and/or Budget data that they have created. Each user can be given their own personal database to store their data input, perform their own aggregations and apply their own models, and perform what-if analysis by studying the impact of their changes. Users also have the full power of the OLAP tool at their disposal within their personal workstation to

create and modify reports and analyze variances with drill-down capabilities. OFA is a true database application as it sits on top of an Express engine.

## **The Express Engine**

Express, Oracle's analytical database software, is the engine that powers OFA. Express is part of the group of On-Line Analytical Processing (OLAP) line of products that are designed to meet the analytical needs of an organization. Instead of storing data in a traditional table format, Express stores its data in a multi-dimensional format to allow for optimal response when users are analyzing the data.

Conceptually, the multi-dimensional database design stores data in cubes instead of a table. Each edge of the cube would represent a different perspective of the data. For example, a company might track its data along time periods, by different line items presented in the financial statements (Total Revenue, Office Supply Expenses, etc.), and by departments within the company. In Express we would create a cube called Actuals which would have three different edges, Time, Line Item, and Department. This cube called "Actual" would store the data in the Express database that represents the actual results for each combination of these three edges ("dimensions" in OLAP terminology).

The dimensions represent values at the lowest level of detail along with any sub-totals and totals that might exist. For example, the Time dimension would contain three "dimension values" called January, February and March. The same dimension would also contain dimension values called Quarter 1 and Full Year. By embedding the totals within the same dimension, this allows for quick retrieval when users are studying the totals and want to see the values that are making up that total. Using our Actuals example again, a user viewing the Full Year results on-line for a given Line Item and a given Department could quickly drill-down on that value with the click of a single button to view the quarters or months which comprise that Full Year value.

This concept of storing data in cubes instead of tables is ideal for the Budgeting and Forecasting community. Within a forecast environment, users are usually asked to input and track data at the monthly level of detail, not at the transactional (journal entry) level. In Express, users can create multiple cubes that store different types of data (a single cube for Forecast, another cube for Budget, etc.) which share the same dimensionality (Actuals, Forecast and Budget results are all tracked by Time, Line Item and Department). Users can easily compare the results between different cubes by creating standard formulas (Actuals minus Budget) in the system that can be quickly invoked and reused across multiple reports.

These features of a multi-dimensional database offer many advantages over the traditional method most companies use to gather and study their Budget and Forecast results-the spreadsheet application.

## **What's Wrong with Excel?**

Almost all of the OFA-OGL link projects which I have been involved called for implementing an OFA solution to replace a computer-based spreadsheet environment (such as Excel or Lotus 1-2-3) for gathering forecasted results. Although these spreadsheet applications are easy to use, growing companies quickly realize their limitations for generating a company's Budget or Forecast. Spreadsheets have a difficult time handling the Budget and Forecast process because of the Difficulty in Implementing Basic Changes, the lack of Built-In Data Collection or Aggregation Features, and the lack of Analytical Features.

### **Difficult to Implement Basic Changes**

If any sort of reorganization takes place within the company (a department is split into two, or a new series of revenue accounts are added) spreadsheet users must make certain that every spreadsheet file is correctly update with the new change. This can prove to be a huge burden if there are a large number of files, if there are a series of formulas impacted by this change, or if these organizational changes take place numerous times during a forecasting cycle. Not only an organizational change triggers this inconvenience, but any changes to any of the business model (a new way a company projects revenue) could result in a full-blown review of all of the spreadsheet files.

### **No Built-In Data Collection or Aggregation Features**

If more than one individual is responsible for generating the forecasted results, some sort of methodology must be created to accurately collect and aggregate the figures to arrive at the company totals. Collecting the results often involves physically moving files from one location to another, and if these files are large this can often take time. Aggregating the data means getting the figures from one file into another, either through some sort

of custom macro process or by simply referencing a cell from one file into another.

As data collection and aggregation must be set-up by the users, there is room for error as large groups of data must be moved and manipulated. Taking into account any organizational changes which might impact the process, the number of times this collection and aggregation must be done (multiple times during a single "cycle" of the Budget or Forecast process), and the limited amount of time the users have to finish the process, incorrect results can easily find their way into the spreadsheets.

### **No True Analytical Features**

The basic spreadsheet features offer no true analytical features. Ad-hoc queries can take time (and discourage users from analyzing the data) as they often involve manually reconfiguring the spreadsheets and pulling data from multiple sheets or files. A seemingly simple change to a report can take time to make to actually make the change and to ensure the data and formulas are correct.

OFA solves all of the issues involved with computer-based spreadsheets when it comes time to generate and analyze the results of a company's Budget and/or Forecast. Using the built-in Express technology, ad-hoc analysis is quick and powerful as users can quickly determine the factors driving variances, and users can easily change the layout of reports to highlight these factors. With OFA's built-in data collection features, users across multiple locations can quickly submit their piece of the data set. These results are consolidated in the shared database area within OFA by pressing just a few buttons. This automated, distributed environment also allows the system administrator to quickly distribute any changes to the system design (if new departments or accounts are added), and these changes are automatically updated in the users' database. By linking OFA's database design with OGL's chart of accounts, these organizational changes in the GL can be implemented even easier within OFA.

### **Implementing the OGL-OFA Link**

As I described before, establishing the link between the two applications allows OFA users to share the values and data created in the OGL. Structure needs to be created only one in the GL, and through a formal process OFA can use the same structure and data. Implementing the link between OGL and OFA not only requires knowing the steps to take to physically copy information from one application to another, but also requires an understanding of how the data model in the GL translates into and impacts the data model in OFA.

### **How the Chart of Accounts translates into OFA**

Understanding the concept behind the Chart of Accounts in OGL can help you understand the structure (and terminology) used within OFA, and vice versa.

Segments in the General Ledger (Department or Natural Account) become Dimensions in OFA

Segment Values (the Human Resources department or Payroll Expense) become Dimension Values

The GL Calendar is translated into the Time Dimension

GL Rollups can be used as Hierarchies to aggregate the data in OFA

The balances in the G/L (such as Actuals and Budget) become Financial Data Items in OFA

The power of the link is not only derived from the ease in which structure and data can be copied from the GL to OFA, but also the flexibility and control the administrator has when deciding exactly which parts of the Chart of Accounts are brought into OFA. An administrator can designate as many or as few of the Segments to be used as Dimensions as he or she deems necessary, and not all of the Segment Values within a Segment need to be brought over into OFA. Segments can even be combined into a single Dimension in OFA. Not all of the Rollups in OGL need to be used in OFA, and administrators can create their own Hierarchies in OFA on Dimensions brought over from the GL.

Administrators can pull in both Functional and Statistical balances in any of the currencies used in the GL. For a single OFA application, administrators can pull in balances from any of the sets of books used in the GL as long as they share the same Chart of Accounts.

## How Structure and Data flow into OFA

To get the structure and data into OFA from the GL, administrators need to first define the "Metadata" in the GL, run the necessary jobs in the GL to make the information available to OFA, and then request that OFA pull in the information.

### Defining the Metadata

After the administrator has created the OFA database design, he or she needs to determine what parts of the OFA database will be created by the GL. Once identified, the administrator navigates through a series of forms in the GL to define the segments and segment values that will be copied into OFA. These forms are also used to designate which Rollups will be copied over, and to define the cubes (the "Financial Data Items") that will be created by the GL for OFA. These cubes will hold the balances (both functional and statistical) that will be copied from the GL into OFA. The documentation provided by Oracle (Integrating Oracle Financial Analyzer with Oracle General Ledger) provides the step-by-step instructions on how to use these forms. This metadata needs to be identified only once, and will be used each time an administrator copies information from the GL into OFA.

Continuing our example from before, an administrator might identify the Natural Account and Department Segments from the GL to be used as Dimensions in OFA. The administrator may wish to pull in all of the segment values from the Natural Account Segment, but only a subset of the Department values. These two new GL Dimensions, along with Time (a dimension automatically created by the link using the GL calendar) are used to create a new Financial Data Item called Actuals.

### Running the Extract Programs in the GL

Once the Metadata has been defined, the administrator runs a series of extraction programs to format the information for OFA to read. These programs are identified as reports in the standard report section, and are run by simply submitting the program ("submitting the report ") from the Submit Requests form. Again, the Integrating Oracle Financial Analyzer with Oracle General Ledger guide describes the step-by-step process to follow to run the required programs.

Depending upon the type of information being copied from GL to OFA, the extraction programs will either create a text file with the information needed by OFA, or identify the relational table that OFA needs to read directly. Keep in mind these extraction programs are pre-defined and part of the standard functionality of the GL.

### Pulling the Information into OFA

After running the extraction programs in the GL, the administrator can then submit a request from OFA to pull in the information from the GL and automatically create the structure (Dimension, Dimension Values and Financial Data Items) and copy the data (balances). OFA will read the text files created by the extract programs, and read directly from relational tables identified by the programs.

All information pulled from the GL into OFA is pulled into the OFA Administrator's workstation and OFA's shared database. The administrator is then responsible for determining which OFA users should have access to the specific Dimension Values and data represented by these values. Perhaps each OFA user is only responsible for analyzing the information and creating the Budget for a select group of Departments. The administrator would only grant access to specific Departments for specific users.

Using the structure that originated in the GL, OFA users can generate Budget and Forecast figures using all of the features of the multidimensional database. The data in these separate Financial Data Items (data that originated in OFA using structure from the GL) can then be sent to the GL.

### Sending Data Back to the GL

Although not a requirement of the link, data can be sent back to the GL from the OFA. There may be a requirement that Budget and Forecast reporting must take place in the GL, which makes it necessary that the forecasted information reside in the GL. The rule that must be followed when sending data back to the GL is that the structure (Dimensions and Dimension Values) used to store the data must have originated from the GL.

Depending upon the version of OFA and OGL in use, there are two different ways data is copied to the GL. For older versions of OFA (pre-version 6.2), a text file was created by OFA to be loaded into OGL via the General Ledger Desktop Integrator. With OFA v. 6.2, administrators now send data back to the GL by having OFA write directly against the budget interface table in the GL. Both processes are described in detail in the integration manual.

### Taking Advantage of the Power of OFA

With the ease in which structure and data is transferred back and forth between the two applications, administrators often struggle with deciding where data processing (such as allocations) and reporting should take place. Usually most people choose the GL to meet these needs as they are more familiar with this application and feel more comfortable with its capabilities. Yet users should not forget about the power of OFA and the powerful multidimensional database it uses.

### The Power of the Express Engine

OFA can quickly and easily perform intricate allocation routines that may have traditionally been performed in the OGL. Base-level information required by any allocation can be brought into OFA via the link, and OFA can perform the necessary processing and allocation reporting. OFA can use any non-GL structure or data sources (perhaps allocation tables supplied by another source) to generate the allocations. With Express' own 4th generation programming language, intricate models can be created to handle any variety of allocation routines.

### OFA's Reporting Capabilities

When companies study their formal monthly financial reporting requirements, they often first turn to the OGL to meet their needs. OFA can also meet most of these requirements while offering additional features such as: unlimited report orientations by dragging and dropping dimension tiles, easily changing report selections using the selector tool, and analyzing data via the drill-down features. With all of the required information (Actual and forecasted results) and all of the power of OFA in the hands of all of its users, analysts throughout the organization can generate a variety of reports.

When determining whether to use OFA or the GL for allocations or to meet specific reporting requirements, you must understand:

Who will need access to the information? Are users already familiar with the application, or will additional training be required? Will we need to purchase additional software licenses, or do users already have access to the needed application?

How long will it take to build the allocation routines or to create the report templates (OFA vs. OGL)? How long (processing time) will it take each application to run the routines or reports? Do we have any in-house expertise in either application to accomplish the task?

### Tips/Future Enhancements of the Link

For those people already using the link, below are some tips based upon my experience of implementing OFA and reviewing the link other people have implemented:

#### **Don't pull in every segment into OFA**

Unless needed, you don't need to identify every Segment in the GL as a Dimension in OFA. Increasing the number of Dimensions for a given Financial Data Item only increases the calculation time for the given data item, and adds complexity to any report using this data item. Also remember that you can combine two or more Segments into one Dimension (on an as needed basis).

#### **Don't feel confined by the Link**

Although the link allows you to easily create structure and pull-in data into OFA, don't feel that your OFA design is limited to what the GL has to offer. Remember that you can add non-GL dimension values to a GL dimension, you can create a Financial Data Item using GL and non-GL

dimensions, and you can apply non-GL hierarchies to GL dimensions.

### **Understand how Hierarchies are Created by the Link**

OFA requires that within a given hierarchy, a child can have only one parent. If you define a Rollup in the GL, and want to use this to create a hierarchy in OFA, the link will only recognize the first child-parent association it encounters. Make note of the GL Rollups where a child rolls to multiple parents as you have no control how this will be carried over into OFA.

### **Dependent Segment Values**

The link will correctly bring over the values and data associated with dependent Segment Values (be certain to combine them with their associated independent Segment Value into the same Dimension) but it will not bring over the correct description (OFA label) of the value.

### **Linking OFA to Multiple GL Instances**

Although the instructions discourage this, it is possible to redirect a single OFA application from one GL instance to another. This might be required when a GL is still being developed, and a test instance is created which will eventually be converted into a production instance. OFA relies upon specific GL tables and OFA catalogs to be in sync when moving structure and data. Be certain these tables and catalogs always stay in sync. There is an Oracle document, Linking an OFA Instance with an Alternate Instance of the OGL, which covers this in detail.

With each new release of OGL and OFA, there are always new features added involving the GL link. Listed below are some of the more notable new features, and some features scheduled to be available sometime soon:

#### **Summary Balances**

Using the link, you always had to pull in the detail balances and run Solves to calculate the summary balances in OFA. With OFA v. 6.2, you are now able to pull in summary balances directly from OGL.

#### **Different Currency Types within the same Financial Data Item**

If you wanted to include Functional balances (revenue, expenses, etc.) and Statistical balances (such as Headcount) within the same Financial Data Item, you need create two separate FDI's to pull both sets of data over into OFA. You then had to copy the data from one of the FDI's (such as the Statistical balances) into the other within OFA. With OFA v. 6.2 and OGL Release 11, you are now able to bring both types of data over within the same FDI.

#### **Scheduling Extraction Programs and pulling information into OFA**

With OFA v. 6.2 you can now schedule the Extraction Programs to run at a specific time in the future or on a specific schedule. You can then also set-up your Task Processor to look for these scheduled tasks and process them when the GL has finished the extraction process. You can also schedule any Solves to run after the load is complete.

#### **Loading Data back to the GL**

As mentioned before, OFA now (with v. 6.2) loads data directly into the General Ledger budget interface table.

#### **Drill-Through back to OGL**

Scheduled to be included with OFA v. 6.3 (not yet available) is the ability to drill-through from OFA back to OGL down to the journal entry level.

### About this paper

This paper was originally presented during IOUG-A Live! 99 in April.

[Download Acrobat Reader](#)

Copyright 2003 by the International Oracle Users Group