FRONT-END TOOLS TO VIEW OLAP DATA

Dan Vlamis, Vlamis Software Solutions, Inc.
dvlamis@vlamis.com

INTRODUCTION

Discoverer release 10g uses BI Beans to present Oracle OLAP data. It gets its power from BI Beans. How does this strategy give Discoverer users the flexibility to analyze OLAP data? What features does this expose to the users over what has been available previously? Custom applications that use BI Beans have many of the capabilities that are in the Discoverer 10g application. When would you want to use Discoverer and when would you want to code an application directly in BI Beans? Finally, the Oracle BI Spreadsheet Add-in (again, using BI Beans) enables Microsoft Excel to access Oracle OLAP data, presenting the data in a familiar Microsoft Excel environment. This presentation will contrast these applications and discuss why you should consider these products.

BACKGROUND

BI Beans is the platform of choice for developing applications that can take full advantage of Oracle OLAP. Developers can use Java to develop any of the interface logic necessary for a business intelligence application. JDeveloper and BI Beans replaced Oracle Express Objects as the environment for building Oracle BI applications. While the switch from a proprietary language (Express Basic) with an industry standard (Java) was welcome, many companies stayed away from developing their own applications in JDeveloper. Many companies have been waiting for a pre-built application that exposes the power of BI Beans without having to code an application from scratch. A few consulting companies, including Vlamis Software Solutions, tried to fill this need by developing their own applications using BI Beans. Still, for the most part, Oracle clients have been waiting for "the technology stack to be complete" before adopting Oracle OLAP. In 2003, Oracle finally realized that it needed to develop an ad-hoc query tool based on BI Beans and extended Discoverer to directly access Oracle OLAP. This product was released in January, 2005, finally delivering a tool for displaying and analyzing Oracle OLAP data.

DISCOVERER

In its first release Discoverer 10g has two interfaces: the Discoverer "classic" interface for analyzing relational data, and Discoverer OLAP. Discoverer OLAP looks a lot like other applications built with BI Beans because it is built with BI Beans. Data is presented via the crosstab and graph BI Beans; queries are edited by using the Query Builder BI Bean. It is exactly this "reuse" of code that enabled Oracle to create a very feature-rich product in such a short period of time. Oracle already had beans that provided the raw components necessary; it just needed to package these together into an application and integrate it with the rest of Discoverer

DISCOVERER PORTLET

There are three main interfaces to Oracle Discoverer. Discoverer takes full advantage of Oracle Portal, and can run in a web portal. Discoverer runs as a "portlet", enabling a web portal to contain multiple sources of data. Since the portal framework allows any registered portlet provider to present data, there is a great deal of flexibility in building screens. Figure 1 shows Discoverer running in a portal. Notice that multiple presentations of the data are available on the screen. The right-most portion of the screen enables the user to navigate between various presentations. These portal screens are displayed very quickly, since the graphic images are cached as part of the Portal framework.

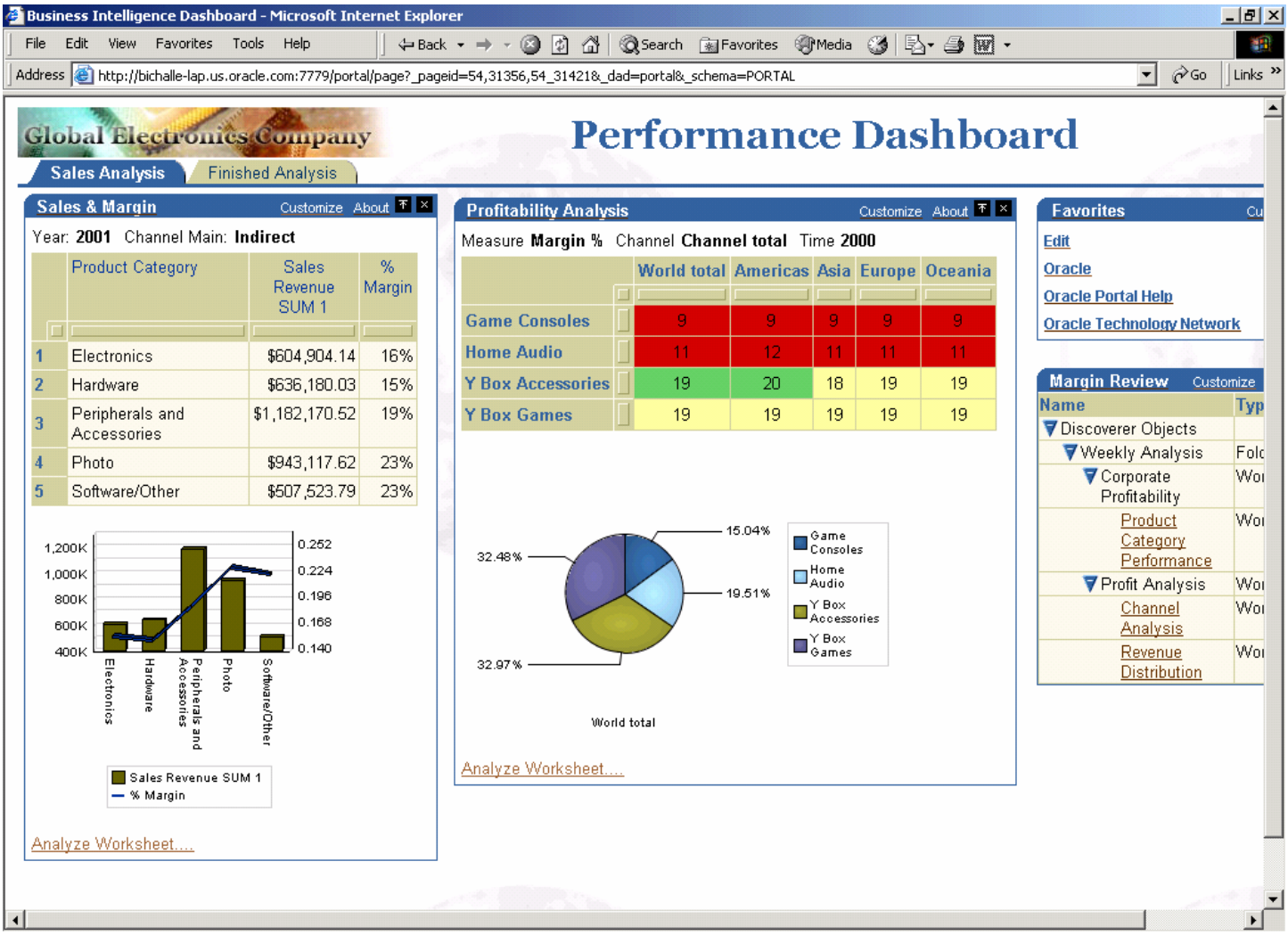

Figure 1 - Discoverer Running in Portal

DISCOVERER VIEWER

If a user wishes to navigate through the data, he clicks on the "Analyze Worksheet" link. This launches Discoverer Viewer. Discoverer Viewer, shown in Figure 2, allows the user to rotate the report, and drill down into lower levels of detail, but it does not let the user create brand new reports, create new custom calculations, or edit or create a new query from scratch. However, you can save the results of your drill down, rotation, etc. as a new report, so perhaps this is not as limiting as you would think. Discoverer Viewer is designed for a user that simply wants to view the data and maybe save a version of a report being viewed. The big advantage of the thin version is that it only requires HTML in a browser to run. There is no installation or even an applet to download.

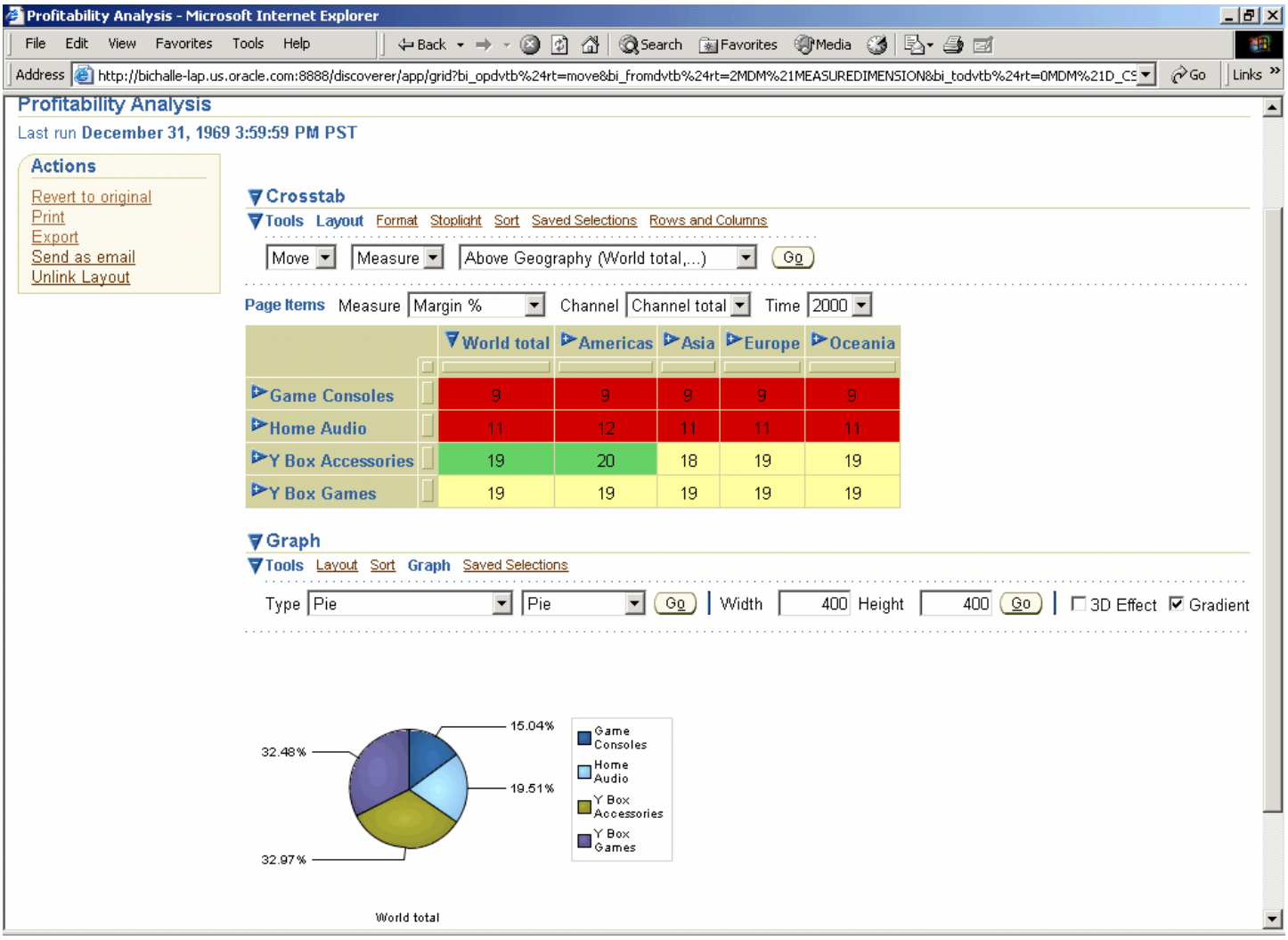

Figure 2 - Discover Viewer

DISCOVERER PLUS OLAP

For power users that really want to drill into the data and create their own presentations from scratch, there is Discoverer Plus. Figure 3 shows a typical Discoverer Plus OLAP screen. It has a standard menu bar, a toolbar for recently-accessed functions, a formatting toolbar that applies to graphs and crosstabs and a stoplight toolbar for applying stoplight formatting to crosstabs. Below this are two "navigation panes" on the left and a presentation space for crosstabs and/or graphs to present the data. Each crosstab or graph has a toolbar that applies to that crosstab or graph. The menu items let you open workbooks (crosstabs or graphs), edit the current query, insert new calculations and all of the features you would expect in navigating through Oracle OLAP data.

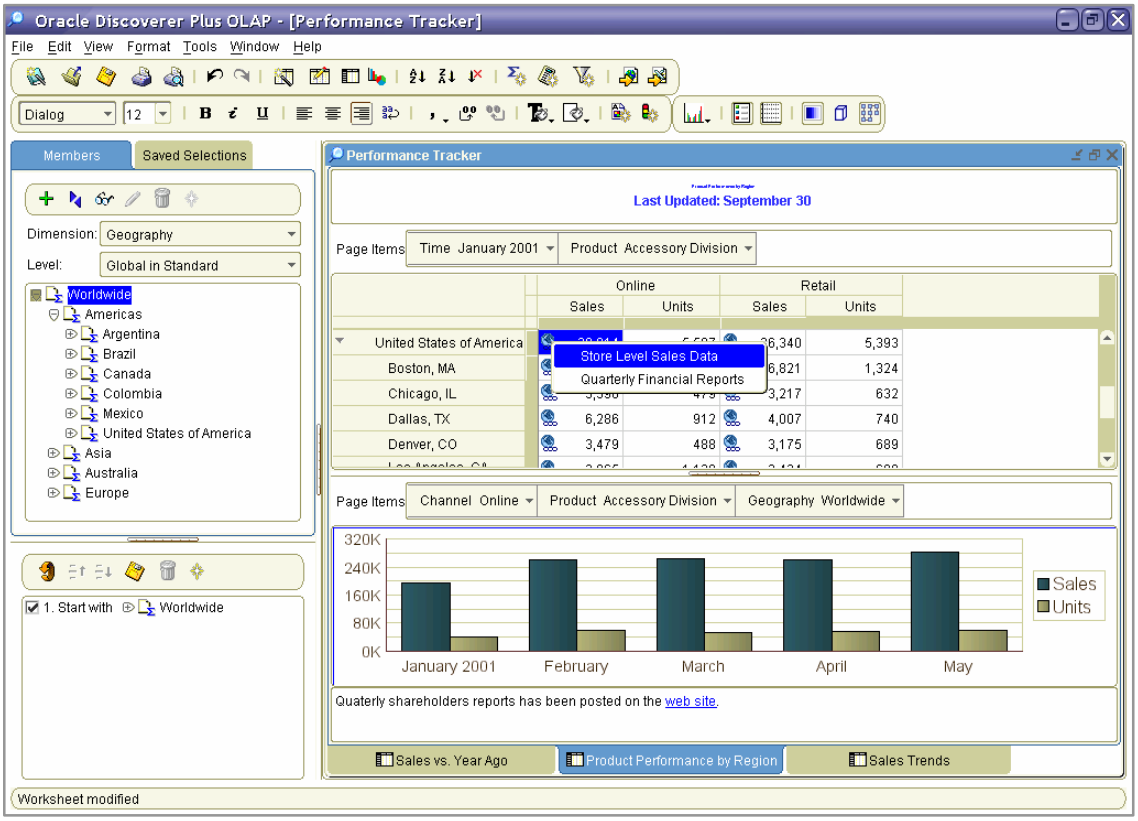

Figure 3 - Discoverer Plus OLAP

FEATURES UNIQUE TO DISCOVERER OLAP

Discoverer Plus OLAP does benefit from being in the Discoverer product line, in that Oracle has included some of the nice features in Discoverer in this product. Specifically, there is a summation tool that Discoverer users will recognize. It allows the user to add a "total" row or column to any crosstab. This allows a user to aggregate dimension values together to produce a total that is not part of the normal hierarchy. For example, a user can add a line to a report displaying data for DECEMBER, JANUARY, AND FEBRUARY that represents TOTAL WINTER. Of course, this totals all dimension values being displayed in the report, but it at least offers the user a way of calculating totals that were not anticipated by the person that designed the reporting hierarchies.

There is also a very nice "undo" feature that can works just as you would expect. This is a welcome enhancement and really helps in usability when exploring data.

Discoverer Plus OLAP also organizes crosstabs and graphs into "workbooks" that can combine multiple crosstabs or graphs into one workbook. The dimension selections and layouts of these individual elements can be linked or unlinked. This offers some additional flexibility when creating presentations since a crosstab and graph can be shown at the same time.

The navigation panes on the left let the user easily modify a query without having to specifically bring up the query editor. For example, to add a measure to a crosstab or graph, the user simply clicks and drags the measure to the crosstab or graph. Working in this way is very intuitive.

BUILDING WITH BI BEANS

If you want to change something about Discoverer, consider building your own application using BI Beans. Indeed, you could build your own "Discoverer" in JDeveloper using BI Beans (and a LOT of Java code!). BI Beans is an add-in to JDeveloper. When you purchase JDeveloper (also part of Oracle Developer Suite), you can download the BI Beans, and add the beans into the JDeveloper environment. Once the BI Beans are added to JDeveloper, you can use them just as any other Enterprise Java Bean when developing an application. Of course, you can choose to use a Java development environment other than JDeveloper, but some of the wizards are only available within the JDeveloper environment, and you will lose the ability to view data live while developing if you use a development environment other than JDeveloper.

Building your own application allows you to build the application any way you want. You may use BI Beans to display or manipulate some of the data, but by building the application yourself, you can more easily integrate OLAP data with other data, whether than data is stored in Oracle or some other location. By building your own application, you can make the application look like and do whatever you want and integrate non-OLAP features in a single application.

GENERAL FEATURES OF BI BEANS APPLICATIONS

BI Beans applications generally display data using crosstabs or graphs. The crosstabs support drilling, rotating, drill-through, color-highlighting, and other typical operations. Graphs support drilling and rotating. There are over 50 graph types to choose from. Selection of data is typically performed using the Query Editor bean. This offers the user a great deal of power in selecting data, with pre-built templates that allow for selections such as "top 10", "all children of a node", and many, many more types of selections. There is also a calculation bean that allows the user to create their own custom measures with templates supporting most of the common expressions that users need to perform such as ratios, year-to-dates, and many, many more.

FEATURES UNIQUE TO BI BEANS

By using BI Beans, you have complete control over how your application looks and operates. While some of the specifics of how the crosstab, graph, and query builder beans look and operate cannot be changed, you have a great deal of flexibility over how these can look and operate. Toolbars and individual buttons can be turned off and features can be locked down. If you want to add an action to a crosstab, you can write your own handlers to do something like drill to the detail of a given cell via a right-click on a cell. Or, add a button that computes a forecast or brings up a "what-if" screen. Indeed, BI Beans applications can look totally different from Discoverer. You can change many features of the crosstab and graph beans or even code your own screens that present the data in a totally different way. Figure 4 shows the "Executive Insight" sample application the Oracle uses to show how easily development departments can put together a custom application using BI Beans.

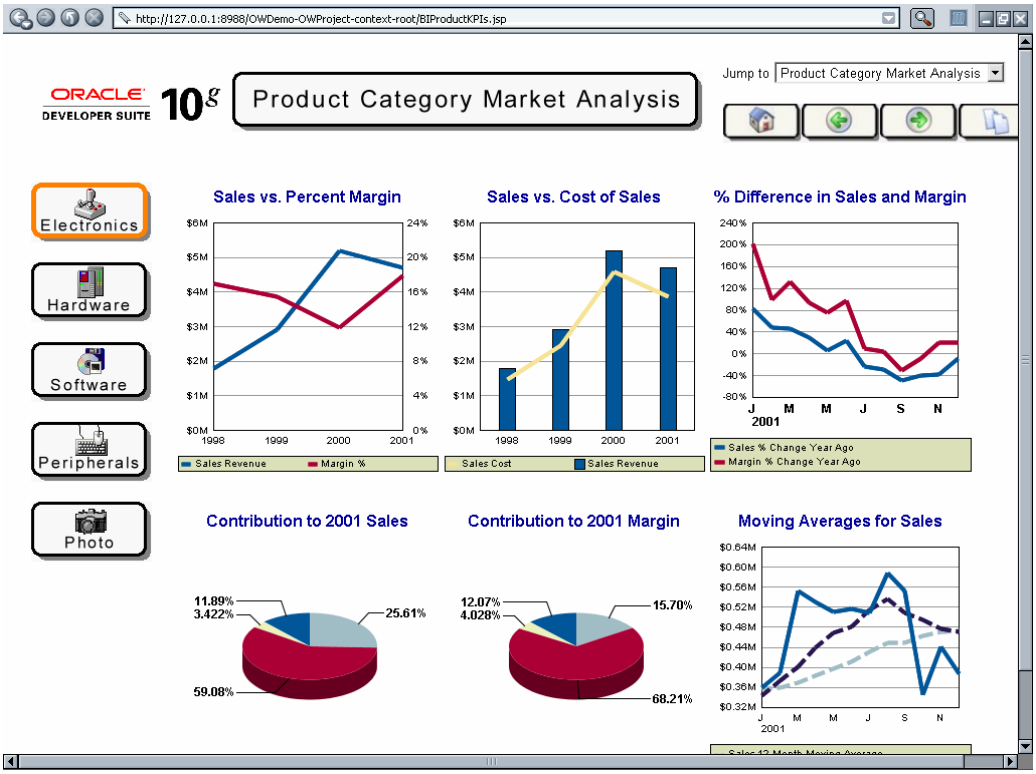

Figure 4 - BI Beans Custom Application

You should also be aware that the thin 10g BI Beans have far more power than is exposed via the Discoverer Viewer at the time of this writing. While the thin 10g BI Beans include a calculation builder, and a query editor that allows a user to change the query being displayed, these capabilities are not exposed in the Discoverer Viewer. In effect, you can build a more feature-rich viewer than the Discoverer Viewer. Just expect to spend some significant time and resources developing this. And, by the time you are ready to deploy your application and your users are enjoying the benefits, Oracle may have enhanced the capabilities of its Discoverer Viewer, negating this advantage.

SPREADSHEET ADD-IN

Perhaps your users simply want access to the data from their favorite tool, Microsoft Excel. The Oracle BI Spreadsheet Add-in enables Excel to have full access to Oracle OLAP data via a series of menu choices that run directly from Excel as an add-in. The user run a wizard from their Excel menu, and the data is presented using an Excel worksheet. Once the data is in the worksheet, the users can do anything they want—add totals, graphs, whatever. Again, these BI Beans are the same as those used in Discoverer or if you were to code your own BI Beans applications. Of course, the presentation beans—the crosstab, and graph beans are not used. Rather, Microsoft Excel presents the data. In effect, Microsoft Excel is acting as middleware in this case, providing the access to the Query Bean to go get the data in Oracle OLAP. In Figure 5, the user has already queried some OLAP data and is re-entering the OracleBI menu to make some changes, such as adding a calculation using the Calculation Bean.

Be aware that at the time of this writing, this Spreadsheet Add-in is only available for accessing data in Oracle OLAP. It cannot access data that is in relational tables that is not mapped into a cube using Oracle OLAP.

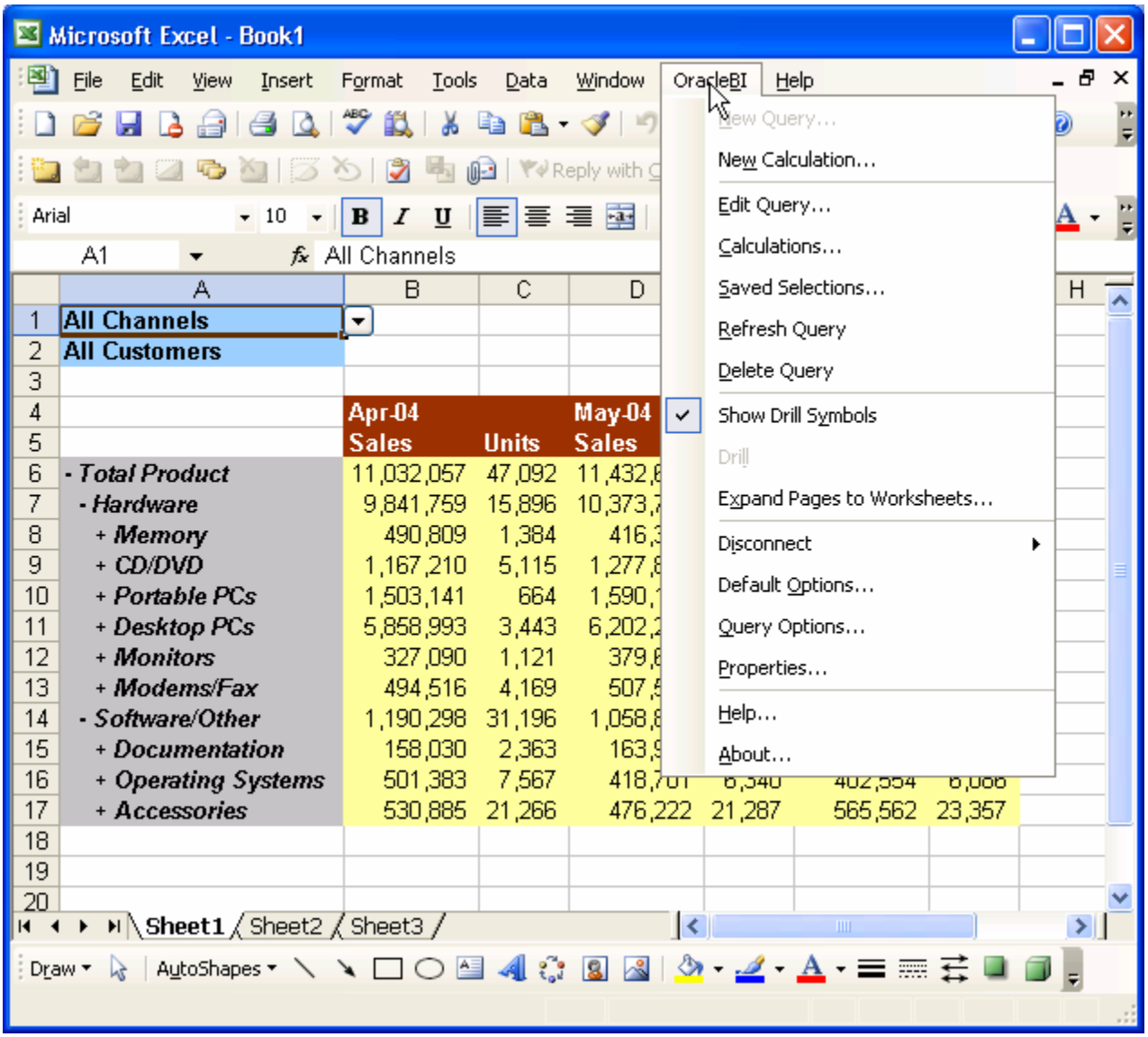

Figure 5 - Oracle BI Spreadsheet Add-in

WHICH IS RIGHT FOR YOU?

So which is right for your company? This really boils down to the classic "buy versus build" question so common in software selection processes. Discoverer is a complete application for your users, but don't expect to be able to change it significantly. With BI Beans, you can make your application do whatever you want. If all you want is access to OLAP data from Excel, look at the spreadsheet add-in.

WHY DISCOVERER?

If you select Discoverer OLAP, you will be able to install a tool and immediately use it against your OLAP data. Your end users will be able to enjoy end-user help, and training on a standard product, with installation routines developed by Oracle. You will save the time you would have to spend developing your own application.

WHY CUSTOM-BUILD WITH BI BEANS?

If you select BI Beans, you will need to create an application yourself. This is actually not that hard with the wizards built into JDeveloper. More importantly, you will be responsible for the installation, documentation, and quality assurance of your application. You will have more flexibility in how your application works and will be able to customize it to your exact needs, but expect to spend some significant time on deployment issues when developing your own application.

WHY SPREADSHEET ADD-IN?

If your users want to access the data from Excel, why make them learn the interface to another product? You can develop a few spreadsheets, and show them how to access the data for themselves directly from Excel. This is very powerful and flexible, but it does raise lots of questions about control over the data and security. Once the data is in Excel users can do anything to it—change the numbers email it all over, etc. They may not consider the ramifications of mundane activities such as keeping the data up-to-date, maintenance, etc.

HOW ABOUT A COMBINATION?

Of course, these decisions are not mutually exclusive. You may choose to use Discoverer for many of your users, since it may meet most of their needs. But, you may have an executive user community that needs specific screens that behave and present the data in a certain way. Finally, some users just want periodic access from Excel. The same Oracle OLAP data can be used in all three environments.

WHAT ABOUT ENTERPRISE PLANNING AND BUDGETING?

Enterprise Planning and Budgeting (EPB) is developed by the group responsible for Oracle E-Business Suite. As an Oracle Application, it has more integration with other Oracle applications. It is certainly worth looking into this as a possibility for analyzing Oracle OLAP data. It is really part of an overall solution for organizing the workflow associated with working with OLAP data. If you are looking for an enterprise-wide solution for organizing your OLAP data and controlling the flow of data and analyses, consider EPB.

MORE INFORMATION

More information can be found on all of these products on Oracle's web site. Starting with http://www.oracle.com/appserver/bi_home.html, you can navigate to all of these products. You will find self-running demos, white papers, tutorials, and even the software itself.

CONCLUSION

Finally, Oracle customers will have the choice of using an out-of-the-box application built by Oracle to analyze Oracle OLAP data, building their own applications using BI Beans, or easily accessing the data from Excel. There is a need for all these capabilities. Is it better to buy a pre-built application or build one yourself? Only you can decide.