



**PowerOLAP<sup>®</sup>**

# ARCHITECTURE AND BENEFITS OF POWEROLAP<sup>®</sup>

WHITE PAPER



PARIS TECHNOLOGIES, INC.  
COPYRIGHT © 2011 ALL RIGHTS RESERVED



# ARCHITECTURE AND BENEFITS OF POWEROLAP®

PowerOLAP® represents a milestone in the evolution of OLAP (on-line analytical processing) technology. Like any important evolutionary event, PowerOLAP combines the most advanced features of what came before it with new capabilities. Most significantly, PowerOLAP enables users to *reach through seamlessly to access transactional data in a relational database for dynamic OLAP manipulations in a true multidimensional environment*. In addition, PowerOLAP employs Excel and the Web as a front end, connecting users throughout an organization with underlying data sources via the tools they know best, direct to their desktops.

This paper will examine the logic for the design of the PowerOLAP architecture. We will consider first, briefly, the necessity for OLAP in the organization and the development of the OLAP market; next, different OLAP product types, their strengths and shortcomings; and finally, in greater detail, how PowerOLAP supersedes predecessor products, the functionality that makes it unique, and the benefits that accrue to an organization that employs PowerOLAP as its OLAP tool.

## OLAP IN THE ORGANIZATION

The term OLAP, for *On-Line Analytical Processing*, is just over twelve years old, but has been adopted to describe a variety of decision support products. The individual who coined the term, E. F. Codd, is known as the father of the relational database. Codd noted that *relational database management system* (RDBMS) products do not have the functionality to perform sophisticated analysis on transactional data, the core operational data of a business. Relational systems were developed for capturing, validating, and storing large volumes of individual transactions: their strength is *on-line transaction processing* (OLTP). However, their static physical design—data is stored in two-dimensional tables, with each entry being a row in a table—stymies the fast, flexible modeling required for sophisticated business analysis.

The capability to perform sophisticated analysis—specifically, the multidimensional analysis provided by OLAP—is by now well-established as an organizational objective. Analysts need to view data along the multiple dimensions that define an organization: its products; regions where it operates; time-periods it organizes data by; measurements of business activity; the types of customers it sells its products to; versions (*Budget, Actual, Forecast*)—essentially, the number of dimensions necessary for the creation of effective business models.

Various products claim to provide OLAP functionality, which has led to questions over how OLAP should be defined, what product features an OLAP tool should include, what constitutes true OLAP functionality. *The OLAP Report*, an independent review of OLAP technology and products, has defined OLAP as “Fast Analysis of Shared Multidimensional Information”, also known as the FASMI test. Until now, with the advent of PowerOLAP, OLAP products so defined have tended to fall into two general categories (ROLAP and MDBs), discussed below. By examining the strengths and shortcomings of



these product types, we can appreciate how PowerOLAP goes beyond ordinary OLAP—in effect, how PowerOLAP, by seamlessly integrating both relational and multidimensional worlds, along with the Excel spreadsheet and the Web, combines different products' strengths and overcomes their limitations, raising the bar for optimal OLAP functionality and performance.

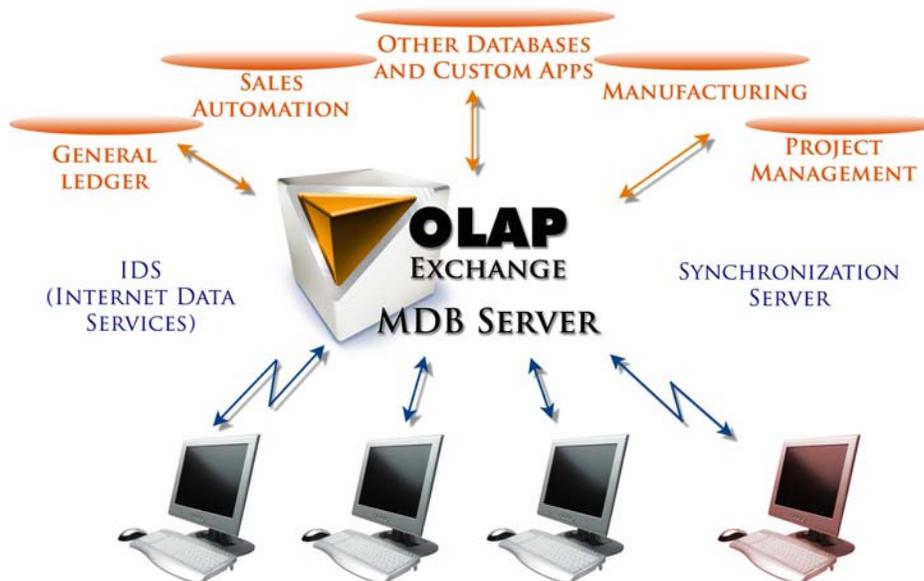
### DYNAMIC BI-DIRECTIONAL EXCHANGE WITH AN RDBMS VIA *OLAP EXCHANGE*®

OLAP Exchange® is the PowerOLAP component tool that establishes a dynamic bi-directional data exchange enabling users to “pull” or “push” data between relational databases and multidimensional models.

OLAP Exchange is one of the key component tools that set PowerOLAP apart as a next-generation decision support product. OLAP Exchange provides a seamless “missing link” OLAP technology to establish a dynamic, two-way connection between the previously incompatible worlds of relational and multidimensional databases.

The link from the PowerOLAP client to a relational database(s) is established via ODBC (open database connectivity), a standard database access method. OLAP Exchange then creates necessary tables within the relational database for both relational-to-multidimensional and multidimensional-to-relational exchange of data. Either data exchange can be set up to synchronize periodically, “on demand,” or live—provided the relational system supports “triggers”. Once the ODBC link is established and the synchronization period selected (in a few simple steps), live and periodic updates occur automatically; “on demand” updates require clicking a single button in a dialog box.

The following diagram shows the architecture of PowerOLAP with OLAP Exchange, and the bi-directional flow of data via OLAP Exchange between PowerOLAP clients and various relational databases.





### THE BENEFITS OF OLAP EXCHANGE ARE AS FOLLOWS:

- OLAP Exchange directly addresses the need to reach *up-to-the-instant* transactional data located in records databases, and to use that data to budget and forecast, analyze and report, *online, in real time*. PowerOLAP empowers end-users, giving them fast and accurate, current information from all organizational sources *direct to their desktops*.
- OLAP Exchange means an end to batching, importing or duplicating data between separate database applications; no longer do staff waste time—and cause errors—by manually inputting/re-keying data into various applications, including spreadsheets.
- By allowing incremental updates of data, OLAP Exchange eliminates the time-consuming need to entirely rebuild multidimensional cubes upon each change or update that occurs in the relational source.
- Through the use of OLAP Exchange’s Drill-Through feature, users can see the individual records in the source relational database that make up a point in a multidimensional cube.
- MIS groups use a single set of advanced tools, deploying OLAP Exchange with great speed and ease, to get data to end users, rather than utilizing various unsatisfactory methods to update disparate systems or to create and re-create business models.
- PowerOLAP supports an open-architecture environment, since it will integrate relational data sources that are ODBC-compliant via OLAP Exchange.

### THE SPREADSHEET AS A FRONT END—THE DYNAMIC POWEROLAP/EXCEL CONNECTION

	January	February	March	1st-Qtr	April	May	June	2nd-Qtr	July	Aug.
1 Database:	LDV Corp									
2 Cube:	Financials									
3 Page:	version <b>Prior Year Actual</b>									
4	product <b>Desktop</b>									
5	region <b>Australia</b>									
8 OLAPTable										
9	<b>January</b>	<b>February</b>	<b>March</b>	<b>1st-Qtr</b>	<b>April</b>	<b>May</b>	<b>June</b>	<b>2nd-Qtr</b>	<b>July</b>	<b>Aug.</b>
10 <b>Before Tax Profit</b>	5301.89	-7991.54	-8460.26	-11149.9	9283.87	-6608.54	952.23	3627.56	-3371.85	-463
11 <b>Contribution</b>	5301.89	-7991.54	-8460.26	-11149.9	9283.87	-6608.54	952.23	3627.56	-3371.85	-463
12 <b>Net Margin</b>	7861.34	-4987.32	-5037.75	-2163.73	14241.8	-2305.96	4928.96	16864.8	111.79	2101
13 <b>Revenues</b>	38941.06	32052.98	44213.06	115207.1	57824.86	46395.65	49503.18	153723.7	43810.04	2825E
14 <b>Cost of Goods Sold</b>	31079.72	37040.3	49250.81	117370.8	43583.06	48701.61	44574.22	136858.9	43698.25	2615E
15 <b>Variable Costs</b>	2559.45	3004.22	3422.51	8986.18	4957.93	4302.58	3976.73	13237.24	3483.64	2564
16 <b>Warehousing</b>	1198.99	1206.04	1393.71	3798.74	2071.16	1754.11	1469.74	5295.01	1484.87	901
17 <b>Shipping</b>	590.72	764.53	1040.03	2395.28	1595.91	1273.56	1246.81	4116.28	1105.39	874
18 <b>Order Proc Staff</b>	682.82	924.8	845.73	2453.35	1148.02	1089.6	1097.38	3335	788.53	702
19 <b>Variable Benefits</b>	86.92	108.85	143.04	338.81	142.84	185.31	162.8	490.95	104.85	8E
20 <b>Fixed Costs</b>	0	0	0	0	0	0	0	0	0	0
21 <b>Net Margin</b>	7861.34	-4987.32	-5037.75	-2163.73	14241.8	-2305.96	4928.96	16864.8	111.79	2101
22 <b>Revenues</b>	38941.06	32052.98	44213.06	115207.1	57824.86	46395.65	49503.18	153723.7	43810.04	2825E
23 <b>Cost of Goods Sold</b>	31079.72	37040.3	49250.81	117370.8	43583.06	48701.61	44574.22	136858.9	43698.25	2615E
24 <b>Variable Costs</b>	2559.45	3004.22	3422.51	8986.18	4957.93	4302.58	3976.73	13237.24	3483.64	2564



PowerOLAP employs the everyday spreadsheet as a front end, one of its most “user-friendly” features, but also tremendously practical for enterprise tasks like performing roll-ups of departmental budgets. With a budget model, for example, located on a Server, multiple users can view reports or contribute figures via the tool they know best—Excel. It should also be noted that many users can work via a spreadsheet Client only (i.e., there is no need to install the full client); via the Web through an Excel spreadsheet; and via a spreadsheet that exists as an html page, which allows a virtually no-footprint access also through Excel.

With the click of a button, a parametric “slice” view of a multidimensional cube creates a spreadsheet—a fully functional spreadsheet, able to create graphs and employ formulas (both Excel- and PowerOLAP-based formulas). Then, from within the spreadsheet, it is possible to enter figures, change parameters of the view or perform higher-level analytic functions. Most important, the spreadsheet remains *dynamically connected*, via PowerOLAP, to underlying data stores. Thus the spreadsheet will update automatically as the underlying figures or structure changes, providing Excel—and end-users—with an unprecedented degree of empowerment, directly, at the desktop.

In addition to the Excel client, there is also a Web client existing as a Flash file that provides graphics and a grid—similar to the PowerOLAP slice grid—that allow “write back”. This Web access Client allows users to work “remotely”, i.e. using the Web, in real time, as the front end to PowerOLAP plan data.

## DRILL THROUGH REFERENCE

Additionally, PowerOLAP has a Drill Through feature, also known as “drill down”, which allows users to see the database reference value which is being used to create their multidimensional reports. The Drill Through feature can be applied to any data value, but is particularly useful with Aggregates, opening a new worksheet in the current Excel workbook to reveal the underlying database information pertinent to that data value, like a relational table. The automatically created Excel worksheet shows the “source” values. If you are using you OLAP Exchange, users can drill through into the relational source of data—see the OLAP Exchange White Paper for more information).

There is also an Extended Drill Through feature, with which you can show information about the underlying relational database—drilling through for this information, from PowerOLAP or Excel at run time—to show data that isn’t part of the Cube logic. This truly extends reporting capabilities.

## IDS: INTERNET DATA SERVICES

PowerOLAP IDS Portal allows you to access, organize, manage, and display data all via the internet. Nowadays the amount of data available is so vast that it can be difficult to manage in an accurate and timely fashion. The IDS Portal makes business information available to those who need it when they want it. By business information, we mean the information that is relevant to your business, and not a mess of meaningless numbers in endless spreadsheets. PowerOLAP IDS features the ability to

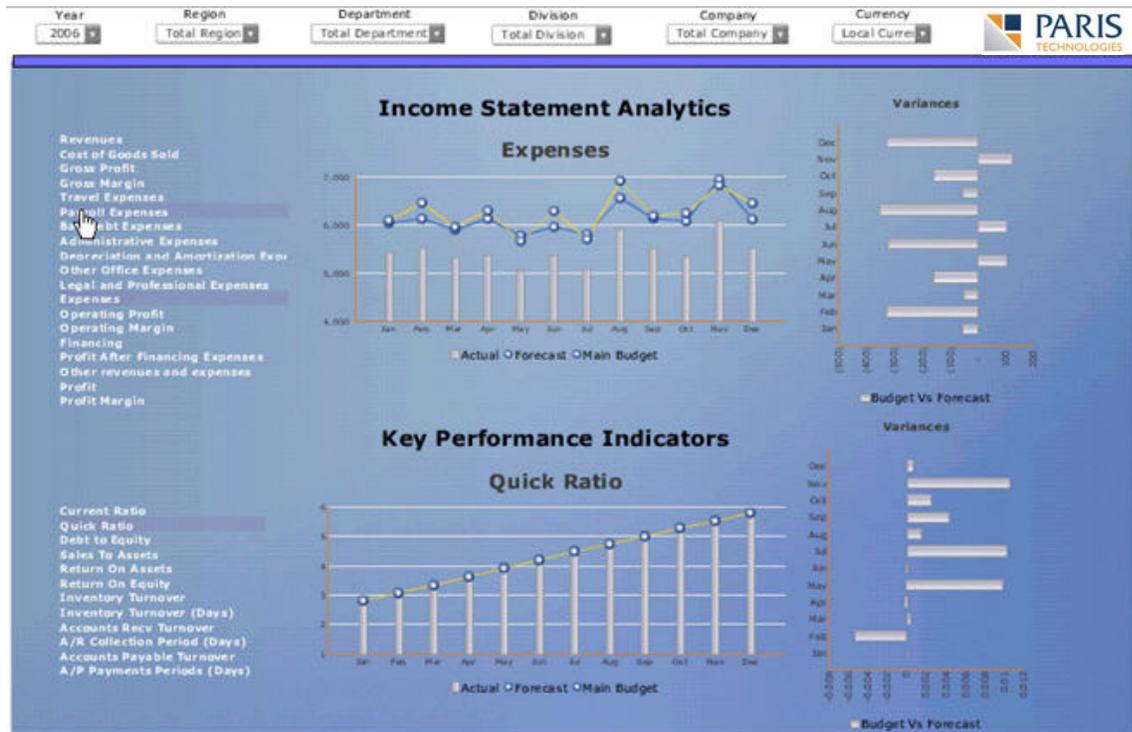


retrieve data from multiple sources via the internet—empowering users with the ability to transform and consolidate their data

Internet Data Services (IDS), broadly speaking, provides data services over the internet, and it allows data to be transferred between two or more databases using HTTP. IDS allows data to be transferred and synchronized between multiple databases that function independently or even reside in different locations. Large amounts of data can be requested or sent between databases; this data can be either fact data or meta data.

IDS also allows specific data to be communicated, as opposed to all data, or allows for the structure of the databases to differ slightly. Subsets/supersets of members in a dimension can exist, and the dimension structure can also be slightly different. For example: a company with multiple locations may have a database with four dimensions at the head office, the fourth dimension being regions. At each of their other locations (Canada, USA, South America, or Europe), they would have a database with three dimensions, but no region dimension. That way, since the head office is the only one to use the regions dimension, the head office is the only one that receives that data. And, let's say each region distributes different sets of products, each region can have different members in the product dimension which reflect their location, and a superset containing all the products would be available at the head office. It's a very efficient way to pare down the amount of data that a given business handles. The data sent and requested between databases can be limited so only relevant data is communicated saving time and consequently money.

### POWEROLAP INTEGRATION WITH XCELSIUS®





PowerOLAP (version 7.0.14 and above) now supports a direct interface to Xcelsius®. Xcelsius developers can work with this interface to add Macromedia® Flash™ presentations to Excel models that dynamically draw data from PowerOLAP databases accessed via the web.

Now users can create custom web applications that can work dynamically with data from a PowerOLAP database that resides on a remote server. These applications have powerful read/write capabilities that maximize the effective presentation of business intelligence. After initial configuration, users will be able to review, update, and share data regardless of where they are working from.

Without additional programming knowledge, users can work with familiar Excel spreadsheets and clear, point-and-click configuration instructions; they can focus on visualizing the metrics they are concerned with the most, instead of being stymied by the common frustrations of building layered informational models using traditional tools. The users “slice” a worksheet from PowerOLAP to Excel and use this worksheet to create their model within Xcelsius. Once the model has been built, data moves dynamically from the PowerOLAP server to the Xcelsius model for stunning dashboards and executive-pleasing KPI reports.

## OLAP PRODUCT CATEGORIES — ROLAP AND MOLAP

### **ROLAP**

ROLAP products (for *Relational* OLAP) directly access data stored in relational databases. Consequently, they can readily retrieve and update transaction data. ROLAP products also enable organizations to leverage their existing investments in RDBMS (relational database management system) software. These are key strengths—strengths that, we shall see, PowerOLAP shares.

ROLAP products access a relational database by using SQL (structured query language), which is the standard language that is used to define and manipulate data in an RDBMS. Subsequent processing may occur in the RDBMS or within a mid-tier server, which accepts requests from clients, translates them into SQL statements, and passes them on to the RDBMS.

ROLAP products have made strides in providing GUIs (graphical user interfaces—drag-and-drop, point-and-click tools) and in generating SQL execution plans that remove end-users from the SQL writing process. However, this over-reliance on processing via SQL statements—including processing for multidimensional analysis—is a serious drawback. Whether it is generated “transparently” or not, SQL is the language of relational tables: it isn’t stretching the metaphor to say that SQL’s vocabulary is too limited and its grammar too inflexible to accommodate the most sophisticated modeling required for multidimensional analyses.

There are further drawbacks to structuring a multidimensional model solely within relational tables: Before end-users can submit requests, the relevant dimension data



must be extracted and reformatted in de-normalized structures known as star schema or snowflakes (so-called because of the way the tables are conjoined). These tabular structures are necessary to provide acceptable analytical performance. Sophisticated ROLAP applications also require that aggregate tables be pre-built and maintained, eliminating the need to process summary data at runtime.

The necessity to reformat data and pre-build tables within the relational database in advance of the actual querying points up two issues:

1. Since it is unlikely that end-users have the skills to do this advance work, MIS professionals set the parameters for analytical processing.
2. Once a multidimensional model is created in a relational format, it becomes difficult to alter the model by then adding new optimization schemes. Consequently, the ability to alter/add dimensions for modeling is limited.

In summary, ROLAP products show deficiencies in providing fast and flexible modeling capabilities. In spite of this, they have become increasingly popular as an OLAP product choice because of their access to relational databases.

### ***Multidimensional Database — MOLAP***

Rather than provide a multidimensional view of relational data, as ROLAP products do, multidimensional databases, known as MOLAP products, enable end-users to model data in a multidimensional environment. This is significant product strength, as it provides for the fastest, most flexible method to process multidimensional requests.

The structure of a multidimensional model is not a series of tables (as exists in a relational database) but what is generally referred to as a *cube*. Cubes modeled in a multidimensional database extend the concept associated with spreadsheets: just as a cell in a spreadsheet represents the intersection of two dimensions (sales of product by region), a cell in a cube represents the intersection of an infinite number of dimension members (e.g., *Products, Customers, Regions, Months ...nth dimension*). As in a spreadsheet, a cell might be calculated by formulas involving other cells.

In short, multidimensional databases allow users to add extra dimensions to the cube, rather than additional tables, as in a relational model.

The cube structure allows for particularly fast, flexible data-modeling and calculations. For one, locating cells is vastly simplified—an application can identify a cell location by name (at the intersection of dimension members) rather than by searching an index or the entire model (via SQL SELECT statements), as in a relational database.

Further, multidimensional models incorporate advanced array-processing techniques and algorithms for managing data and calculations. As a result, multidimensional databases can store data very efficiently and process calculations in a fraction of the time required of relational-based products.

What, then, are the drawbacks of multidimensional databases?



1. Relevant data must be transferred from relational systems—a potentially “redundant” re-creation of data in another, albeit multidimensional, database. Once data has been transferred, there may be no simple means for updating the MOLAP “engine” as individual transactions are recorded by the RDBMS.
2. MOLAP products may have limited storage capabilities—generally no more than 1 GB.
3. MOLAP products are proprietary systems—unlike the relational model, there is no agreed-upon multidimensional model, nor any standard access method (such as SQL) or APIs.

In summary, MOLAP products’ shortcomings correspond to ROLAP products’ strengths—although MOLAP products provide the fastest means for developing flexible multidimensional models, their chief drawback is their *lack of access* to transactional data stored in relational stores.

Now, in the context of the ROLAP/MOLAP dichotomy, we can situate PowerOLAP and explain how it supersedes these two predecessor OLAP products.

## POWEROLAP—MODELING IN AN OPTIMAL MULTIDIMENSIONAL ENGINE

The screenshot shows the PowerOLAP application window titled "PowerOLAP - [Financials : Accounts by Month]". The interface includes a menu bar (File, Edit, View, Model, Slice, Data, Tools, Window, Help), a toolbar with various icons, and a main data table. On the left side, there are panels for "Page:" (version: Prior Year Actual, product: Desktop, region: Australia), "Columns:" (month: All), and "Rows:" (account: Subset). The data table displays financial metrics for January, February, and March, with a "1st-Q" column. The table is scrollable, and the status bar at the bottom indicates "Local" and "LDV Corp".

	January	February	March	1st-Q
<b>Before Tax Profit</b>	5301.89	-7991.54	-8460.26	-1
<b>Contribution</b>	5301.89	-7991.54	-8460.26	-1
<b>Net Margin</b>	7861.34	-4987.32	-5037.75	.
Revenues	38941.06	32052.98	44213.06	11
Cost of Goods Sold	31079.72	37040.30	49250.81	11
<b>Variable Costs</b>	2559.45	3004.22	3422.51	
Warehousing	1198.99	1206.04	1393.71	
Shipping	590.72	764.53	1040.03	
Order Proc Staff	682.82	924.80	845.73	
Variable Benefits	86.92	108.85	143.04	
<b>Fixed Costs</b>	0.00	0.00	0.00	
<b>Net Margin</b>	7861.34	-4987.32	-5037.75	.
Revenues	38941.06	32052.98	44213.06	11
Cost of Goods Sold	31079.72	37040.30	49250.81	11
<b>Variable Costs</b>	2559.45	3004.22	3422.51	



PowerOLAP does incorporate a multidimensional database engine as the optimal environment for sophisticated modeling. However, PowerOLAP, by seamlessly accessing relational data, overcomes the shortcomings of predecessor MOLAP products. The architecture of PowerOLAP's component tools will be fully described in the next section; for now, it is worthwhile to address the points listed previously concerning MOLAP "drawbacks":

1. In PowerOLAP, data is not "re-created redundantly" from the relational database; rather, it is available for true multidimensional analysis *in real time, as transactions are recorded*.
2. The limit on storage that hampers predecessor MOLAP products is not relevant to PowerOLAP. PowerOLAP can access data that is necessary for immediate modeling; and PowerOLAP has the capability to store models to the relational database to which it is linked, with no limit as to the size.
3. Because of its seamless integration with relational databases, PowerOLAP loses the "proprietary" label that attaches to predecessor MOLAP engines and can neither be labeled as ROLAP or MOLAP. The data processed in the PowerOLAP engine is as accessible as all data is in an RDBMS. PowerOLAP is effectively an *open-architecture* multidimensional database.

Another product difference between PowerOLAP and other MOLAP products should be noted: PowerOLAP does *not* pre-calculate all derived values in its multidimensional cubes. Most other MOLAP products follow a pre-calculation strategy because, in theory, it can provide faster results: if many users typically ask for the same information, the calculated value is already available. The drawback to this strategy is that the most sophisticated models contain many calculated results, so as input values raise; derived values rise exponentially making the data very difficult to maintain (known as "data explosion"). Further, when changes to the data are made—quite often, in most analysts' offices—a user must request that the database be calculated again, and this can take valuable time.

The PowerOLAP multidimensional engine *calculates values on demand*, the optimal approach for a database that receives dynamic updates from a transaction processing system. PowerOLAP maintains a fast response time by storing data in memory—as an in-memory database—and by allowing calculations to be stored on disc, as well as by using incredibly fast algorithms that are designed to handle the sparse models characteristic of business analysis scenarios.

## SUMMARY

PowerOLAP enables organizations to create dynamic solutions quickly—in days rather weeks—between relational and multidimensional databases. With OLAP Exchange, users have *online* access to transactional data—as *that data is recorded in real time*—within a multidimensional format. OLAP Exchange empowers PowerOLAP as a "rapid application deployment" solution for true, online analytical/ reporting/ budgeting capabilities. In all, PowerOLAP represents the "next-generation" advance of OLAP technology.



PowerOLAP leverages investments and knowledge in existing database systems, providing unprecedented flexibility and speed for the back-end integration of transaction and analytical systems. By employing Excel as a front end, PowerOLAP connects users throughout an organization with underlying data sources via the tool they know best, the familiar spreadsheet, direct to their desktops—another example of how PowerOLAP leverages company investments and knowledge. The use of the Web as a front end completes the “full familiarity” and “real-time reach” of a PowerOLAP system.

With real-time transaction data at their fingertips—and with the ability to contribute budget and forecast numbers via “write back” from Excel and across the Web—analysts are far more productive: the business models they create and participate in contain up-to-the-instant information; they no longer waste time keying in figures or trying to reconcile data from various sources, including individual spreadsheets.

PowerOLAP is one single solution—*finally*—that closes the gap between relational and multidimensional worlds, and which fully leverages Excel and the Web, making advanced technology available to everyday analysts for powerful, real-time business benefits.