



DB2 Information Management Software

**IBM DB2 Cube Views enables
information on demand**



Table of Contents

3	<i>Getting the answers your business needs with OLAP</i>
4	<i>Why OLAP matters</i>
4	<i>Teaching relational databases to understand OLAP</i>
6	<i>IBM DB2 delivers OLAP power</i>
7	<i>Meta data: The value of information about information</i>
10	<i>A data warehousing primer</i>
12	<i>DB2 Cube Views knows where its own data goes</i>
13	<i>Aggregation management speeds up query response</i>
15	<i>Web services for OLAP provide a lightweight interface for e-business data sharing</i>
16	<i>Virtual cubes save time, cost and storage space</i>
17	<i>Standards-based design leads to easier development, data access</i>
17	<i>Close partner relationships pay off</i>
18	<i>Summary</i>



Getting the answers your business needs with OLAP

Online analytical processing (OLAP) is a technique for summarizing and interrogating data along multiple dimensions. As such, it is a valuable tool for any organization that must make sense of a large volume of data quickly.

OLAP has been popular for decades in data-rich industries such as consumer packaged goods, transportation and retail, as well as in processes such as budgeting or measuring marketing campaigns analysis. One advantage of OLAP over simple spreadsheets is that it can nest an entire company's data (such as proposed budget figures for each project, department, division and the company as a whole) hierarchically in a single data structure. Whenever you want to know more, you can just click to zoom in on the details and then zoom back out to see the big picture, rather than having to patch together many individual files.

More importantly, OLAP can display different views of this data, according to the dimension that's of most interest—perhaps breaking down the companywide budget by projected versus actual figures, or overhead versus production costs; or checking how each department is allocating its employee bonuses. No longer does someone need to painstakingly go through an entire budget picking out the figures to be re-aggregated. Moreover, line-of-business staff members can obtain these multidimensional views of the data in just a few keystrokes.

This speed, flexibility and level of detail have made OLAP even more valuable in today's information-driven economy, when better knowledge often confers competitive advantage. Let's say a soft-drink bottler wants to become more profitable. Before the soft-drink company can make a decision on new ad campaigns, price cuts, flavor changes or anything else, it needs to analyze its current profitability. It can do so by having a data analyst look at sales of each of its soda flavors by quarter in each part of the U.S. and break down those quarterly sales figures into revenue, cost and profit.

This process gives an accurate, detailed view of which flavors are most popular in which parts of the country, how sales vary over time due to the season or new ad campaigns, and where the greatest profit is being realized. With better information, the company can see which ads are most effective, which time of year customers are ready to buy more soda and which flavors offer the widest profit margins. Based on this insight, the company can pick the approach that will yield the highest increase in profits.

**Why OLAP matters**

Several business trends have converged to make greater OLAP awareness in relational databases more important. First, a larger emphasis on speedy decision-making—to support capabilities such as just-in-time product delivery and fast reaction to changing market conditions—means accurate, timely information is a much higher priority.

Second, companies are starting to realize the high ROI of having a single, companywide mission, such as having a “six sigma” level of product quality or making customers wait no more than 20 minutes for a pizza. But this means every department of the company must contribute towards that goal—not just manufacturing, sales or delivery, but also procurement, marketing, accounting, HR and others. Therefore, corporate leaders need data from all these departments. And not just in raw form, but combined into time-saving summaries that provide a fast, easy-to-navigate, unified, single view of the truth.

Third (and paradoxically), within these unified enterprises, there is increasing pressure for lines of business and individual departments to do their own data analysis on demand, rather than hand off the job to a centralized cadre of analysts. This approach is faster and more efficient, since the lines of business generally know the most about the field in question and the exact questions that need to be asked.

Teaching relational databases to understand OLAP

Considering the benefits OLAP can bring to so many industries, it is no surprise that the OLAP market has grown an average of 40 percent per year from 1996 through 2003. Companies are finding new uses for OLAP every day, including clickstream analysis of e-commerce Web sites, airline passenger tracking for greater security, and compliance with new, stricter financial reporting requirements. But this increasing use of OLAP means that the underlying databases are bearing a heavier workload, storing and serving up ever larger volumes of data. And many of them are not keeping up with OLAP’s data access demands.



This problem is particularly acute when performing OLAP on a relational database, which is called relational OLAP or ROLAP. The reason many relational databases struggle to support OLAP is that they're not compatible with the "star schema" logical design technique that most data warehouse and OLAP tools use (a semi-hierarchical technique that makes the databases easy to query but full of redundancies). Instead, relational databases are typically built according to the other major logical design technique, "entity-relation modeling." This technique minimizes redundant data but makes the databases hard to query—so they mostly end up as passive data repositories, contributing little in the way of OLAP functionality. (We will discuss competing logical design techniques in more depth later.) Business users certainly still appreciate the flexibility and economical data use of the classic relational database. But now they also require their relational databases to work smoothly with OLAP, allowing centralized, cohesive control of multidimensional data across the enterprise. This reduces the amount of time, skill and money needed to perform OLAP calculations across a wide range of data sources, and delivers valuable insights more cost-effectively and accurately.

As businesses express their needs for ROLAP to software companies, many relational database vendors are starting to accept the value of supporting OLAP. But few of them realize there is a fundamental disparity between the architecture of their flagship databases and the architecture OLAP requires for optimum performance. Even these companies mostly make their databases "speak the language of OLAP" only through awkward, bolt-on tools.

**IBM DB2® delivers OLAP power**

By contrast, IBM provides this kind of relational OLAP power in a new add-on package for IBM DB2 Universal Database™, a set of features for multidimensional data management called IBM DB2 Cube Views, which integrates seamlessly with the master database. DB2 Cube Views is built from the ground up to offer heightened awareness of OLAP in the underlying database catalogs, optimization for OLAP-style data access and new interfaces to support Web services. DB2 Cube Views enables faster deployment, easier management, cleaner source data (thanks to centralization of data marts) and better performance across all analytical applications. And because it's faster and easier for end users to query, it's also more cost-effective than competing products. Plus, it offers rapid, smooth integration with a broad range of OLAP add-ons and specialized analytic tools from IBM Business Partners such as Ascential, Brio, Business Objects, Cognos, Metadata Integration Technologies, Microstrategy, and other leading vendors.

This paper includes an overview of OLAP structure and techniques. It also discusses how IBM's OLAP Accelerators can offer quicker, easier and more cost-effective slicing and dicing of data, which can deliver numerous business benefits. Some of these benefits include increased efficiencies, improved service to customers and suppliers, better decision-making and faster response to business opportunities.

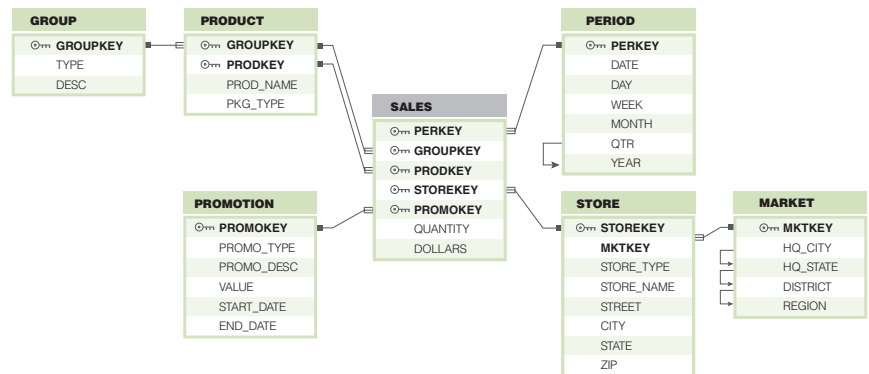
**Meta data: The value of information about information**

A problem for many businesses and government agencies today is too much information but not enough insight into what it all means. One enterprise's data might take up thousands or even millions of pages if printed out in a simple tabular format—far more than any one person could read through. ROLAP can provide a multidimensional view of this data so that users can quickly drill down to the relevant information they need for more high-level analysis.

However, as noted previously, many relational databases don't do much to support the ROLAP process. In contrast to hierarchical databases, relational databases are built on the entity-relation model, which minimizes redundant data in a set of records by separating all the variables into multiple tables. Each table is linked to all the others by a complex network of data joins that resembles a spiderweb. The internal catalogs in a typical relational database provide basic information about the entities managed by the database (e.g., lists of the tables and columns into which data is arranged along various dimensions). But the database's non-hierarchical network of connections holds little "meta data"—that is, labels for the information in the database—so it doesn't deliver much knowledge about the relationships between tables or how tables and columns are related to the business entities. (Users tolerate this low level of meta data because compared to hierarchical databases, relational databases are easier to build and add categories to after the fact. They also offer more flexibility to examine data according to any dimension.)



DB2 Cube Views offers a better foundation for data aggregation and analysis because in addition to the basic entity-relation architecture, it offers support for “dimensional modeling” architecture, the same logical design that is the industry standard for data warehouses. The dimensional modeling technique that produces “star schemas” combines relational and hierarchical techniques for arranging data in ways that provide easy access and navigation along each dimension. (It’s called a “star” schema because each dimensional model consists of a central table with a multipart key, called a fact table, and a set of smaller tables, called dimensional tables, each with a single-part primary key that corresponds exactly to one of the components of the multipart key in the fact table. These connections between the central fact table and its subsidiary dimensional tables make a characteristic star-like shape—hence the name.)



Star schemas, produced with the dimensional modeling technique, combine relational and hierarchical architecture to let data nest in layers according to related values, which yields faster analysis. The connection between the central fact table and the subsidiary or dimensional tables create a star-like shape that gives this schema its name.



The star schemas produced by dimensional modeling let data nest in layers. For example, the product category “footwear” could include supergroups “boots,” “shoes” and “sandals”; the “shoes” supergroup could be broken down into “sports shoes,” “walking shoes” and “fashion shoes,” before you get to individual product lines, each with variables of “size,” “width” and “color.” More importantly, databases and analytical structures built on star schemas generate meta data that helps explain relationships among and within various data dimensions. This capacity lets users navigate through data marts and data warehouses more quickly and easily. For example, the DB2 database understands that “quarter” is a subset of “year,” and “month” is a subset of “quarter,” so that it can use the “month” results to assemble the “quarter” and “year” results, rather than building the latter two from scratch.

Many specialized OLAP tool vendors offer multidimensional meta data capacities, but their products are more effective (not to mention simpler to configure and use) when the underlying database helps share the burden of organizing the relationships of various data sets in the underlying database. The meta data-aware approach of DB2 Cube Views delivers useful insights sooner. It also cuts your operating costs, since data analysis takes less time, can be freely shared among multiple databases and other vendors’ tools, and is easy enough to be handled by line-of-business staff rather than expensive specialists. The next section of this paper explains how these benefits are delivered.



A data warehousing primer

Before delving more deeply into the subject of OLAP and data warehousing, let's look at a few more definitions. A “data warehouse” is a set of data structures that are integrated and optimized for distribution and analysis. The warehouse may include an integrated database, such as DB2 Universal Database, or a data staging area. However, it must include the meta data that describes and identifies the various data and its characteristics.

A data warehouse generally feeds one or more “data marts” or “OLAP cubes”—data structures that are optimized for end-user access to and analysis of the information. The distinguishing feature of a data mart or cube is that it offers a multidimensional view of data that lets users look at a combination of factors or dimensions simultaneously. A physical cube has three dimensions—depth, width and height—but a data cube may be built with more (for example, department, employee, time period, position, and location).

Also, cubes or data marts let you choose the dimension to search—for example, to look at employee salaries companywide, to tally the number of administrative assistants or programmers in each department or calculate the average rate of employee turnover from year to year. This search process is often called “dicing” the cubes and the results are called “slices.”

“Drilling down” is the term for examining a particular slice of data in closer detail—for instance, looking at the company's entire payroll, then payroll by region, then by individual store, then at individual employees in that store. An individual piece of information in a slice, say the amount of a particular store employee's February 1st through 15th paycheck, is a “cell.” The opposite of drilling down is “rolling up”—that is, starting at the cellular level and compiling all the individual figures into summaries, and collapsing these summaries into higher-level summaries, until you get the most high-level view of how the enterprise operates.



As mentioned previously, when OLAP is performed on relational databases, that technique is called ROLAP. ROLAP generally relies on SQL (Structured Query Language) to convey queries to and from the database. As is typical of relational databases, the most commonly used queries are stored for reuse, to save time. Its counterpart is MOLAP, or “multidimensional OLAP”—that is, OLAP performed on a multidimensional database.

ROLAP allows multidimensional views of data (e.g. by date, by product group or by geography), but only by accessing and processing tables for each dimension of a data aggregate. The advantage is that ROLAP supports larger user groups and greater amounts of data. MOLAP, on the other hand, works on data that is already stored in a multidimensional array in which all possible combinations of data are reflected, each in a cell that can be accessed directly. This means MOLAP is generally faster for queries but less flexible when organizational needs change because the structure and combinations are predefined. Also, its multidimensional architecture means it often can be done only in specialized data marts, not directly on a relational database.

Hybrid OLAP, or HOLAP, combines the best of both ROLAP and MOLAP by using either a relational or a multidimensional approach to OLAP, depending on where the data is most efficiently stored.

Whatever the flavor of OLAP, the result is “business intelligence” (BI)—knowledge that’s useful in a specific competitive context, such as how well various flavors of soda are selling in different parts of the country. Business intelligence systems are often the material for more complex queries known as data mining, which can spot patterns and predict trends based on large amounts of apparently random data. An example of data mining might be searching for the most common combinations of products bought by grocery shoppers, and discovering that disposable aluminum cake pans are most often bought in conjunction with cake mix and frosting. Therefore, a smart grocery manager will move the foil cake pans from the kitchen-supplies aisle over to the cake mix shelves, to appeal to the most likely buyers. For more information about data mining, please refer to the white paper entitled *DB2 Embedded Analytics for Information on Demand*.



Generally, a relational database only permits the most basic OLAP functions; the user must buy a separate tool for sophisticated OLAP queries. However, the new DB2 Cube Views add-on product incorporates OLAP functionality in order to make DB2 a more powerful platform for managing and deploying information across the enterprise. IBM and many leading OLAP vendors have worked together to create a set of application programming interfaces (APIs) and tools that allow unprecedented integration of the relational database and business intelligence tools through DB2 Cube Views.

With powerful ROLAP and MOLAP functionality, DB2 Cube Views can make DB2 a superb platform for an entire BI infrastructure; users can import pre-built OLAP cubes, build custom data warehouses and OLAP structures, and integrate highly specialized analytical tools for business intelligence. This can give businesses a cost-effective, speedy, easy-to-use system to deliver a single version of the truth by coordinating all the add-on tools with the underlying database.

DB2 Cube Views knows where its own data goes

Because of its understanding of dimensional models, DB2 Cube Views has a unique capacity to understand its own higher-level data organization, which is required for OLAP and BI. The way it works is that the database catalogs within DB2 Cube Views track not only the base data, but also create a new set of higher-level meta data, including facts, attributes, relationships, hierarchies and dimensions. Once these high-level labels are defined, they can be stored and managed in the catalogs and made available to the DB2 database engine and optimizer, so that DB2 Cube Views is consciously “aware” of how all the various pieces of information it contains fit together. (For example, if it were tracking shoe sales it would know that “6, 7, 8 and 9” are attributes within the “size” dimension, and it would also know that if a shoe is size 7, by definition it cannot also be size 8.)

Moreover, DB2 Cube Views inventories and creates its meta data not only when it is first deployed, but also any time the schema is changed (perhaps shoe styles are renamed) or a new one is introduced (say, the manufacturing department wants to start tracking product quality against customer complaints). This awareness makes for faster, smoother querying and less frequent needs to update and maintain the data warehouse.



In addition, DB2 Cube Views makes this meta data about dimensions, hierarchies, attributes and facts available to all tools and applications working with the database. This means external BI applications, ranging from simple reporting interfaces to data mining and spatial analysis tools, can now learn all they need to know about the data model by querying the database. Otherwise, the knowledge about how all the various attributes relate to one another would remain locked up in the data warehouse modeling tool or the model designer's head, requiring a great deal of explanation or even trial and error after the data structure is turned over to its end users. The result of this greater openness about meta data within DB2 Cube Views is faster deployment and easier management of OLAP applications. And that means lower operating costs and faster delivery of useful information.

The end-to-end flow of dimensional meta data within the application and to and from specialized BI tools is also aided by the use of XML within DB2 Cube Views. (XML is the open-standards language for describing data attributes—for example, distinguishing between the year 1999, \$19.99, and 1,999 widgets.) For one thing, the XML interface gives partner OLAP tool vendors a shortcut for tighter, more effective integration with DB2 Cube Views meta data. In fact, DB2 Cube Views is built on open standards from end to end, shortening the learning curve and enabling faster, smoother and easier integration. IBM's open-standards approach will be discussed in more detail later in this paper.

Aggregation management speeds up query response

DB2 Cube Views also boosts OLAP performance through more effective creation and management of aggregates. Aggregates represent all the possible combinations of data under consideration—say, weekly sales figures for the past three years, for 200 separate products sold to 50 different customers through two sales channels. Multiply 156 weeks by 200 products by 50 customers by 2 channels, and the resulting aggregate contains 2.3 million cells in a multidimensional array. Alternative views of these figures by month, quarter or year could add another couple of million cells.



You can imagine how time-consuming and expensive (in terms of processing and staff time) it can become to manage a data mart to perform such large computations. Building whole or partial aggregations ahead of time and storing them is one way around the problem. But if you overdo the latter strategy, you could end up with too many large aggregations in storage or “cube explosions” as MOLAP cubes are added and expanded, which consume space and processing power.

DB2 Cube Views helps to defuse cube explosions because it is shipped with a graphical wizard that helps decide the best aggregates to build based on the dimensional meta data stored in DB2. DB2 can transparently rewrite queries to take advantage of the available aggregates to deliver the precise answers you requested. It can even consider partial as well as exact matches. For example, when asked for sales by product for January, February and March, DB2 Cube Views can recognize the Quarter 1 cube that contains those monthly figures as an exact match. And when asked to sum up sales by product across February and March, it can add up the existing aggregates for those months. You won't even know the queries have been rewritten—all you'll see is faster performance.

This means query response is much faster than starting from scratch. (Cube loads were up to 70 percent faster for DB2 OLAP Server™ and over 90 percent faster for analysis of hybrid data sources, according to IBM testing.) Instead of drumming their fingers waiting for answers, OLAP users are already steering better decisions. Bolt-on OLAP tools and BI applications working with DB2 Cube Views also benefit from the speed boost, without any need to be aware of the existing aggregates or make any adjustments themselves.

One of the technical underpinnings for this feature is the recently introduced ability within DB2 to perform a unique method of multidimensional indexing called “multidimensional clustering.” With multidimensional clustering, data that is to be indexed across multiple dimensions is stored in an arrangement that clusters rows with the same values (say, if all sizes and colors of a particular shoe style have the same manufacturer) physically together; therefore, only one index entry is required for that set of rows. The result is more efficient indexing across multiple dimensions, which in turn speeds up the process of aggregating similar types of data.

**Web services for OLAP provide a lightweight interface for e-business data sharing**

Web services are gradually catching on as the basis of the next generation of e-business applications and the new model for application development. Much of the promise of Web services is in open service-based interfaces to which applications can dynamically bind. OLAP functionality that delivers data cubes, slices or cells from a multidimensional model on the fly would be a very valuable foundation service, providing “raw material” for these applications.

Specialized OLAP tools are unlikely to adopt Web services as their interface for slicing, dicing and drilling, because they need the high-speed service they get from their native interfaces. But other, more lightweight analytic applications that use Web services will also need access to multidimensional data as they assemble information from a variety of sources to drive business processes. DB2 Cube Views can give them what they need, quickly and accurately, while minimizing development time, which is often the limiting factor in creating new Web services applications.

A good example of Web services in action is an inventory management application that balances stock against demand by using Web services to check internal merchandise levels and communicate with suppliers and shipping partners. This kind of application needs access to multidimensional data such as planned and actual inventory levels by product and location, over time. The developers building this application need to focus their time and expertise on the algorithms and business models that provide the best service at the lowest cost. These developers will certainly be familiar with Web services development tools and languages, such as XML, SOAP and XPath—but they are not OLAP specialists. And they don't have to be if they're working with DB2 Cube Views, because its OLAP awareness helps it identify the right data quickly to “feed” the inventory management application through a simple Web services interface. Therefore, developers don't have to spend time mastering OLAP on top of their Web services skills—reducing overall IT budgets and shortening time to delivery. Web Services is a technology preview in the first release of DB2 Cube Views, intended only to demonstrate capabilities of DB2 and Web Services.

In the following sections, we will take a look “under the hood” at some of the architectural components of DB2 Cube Views that enable many of the performance advantages discussed earlier.



Virtual cubes save time, cost and storage space

DB2 Cube Views speeds up the creation and management of data cubes and limits the problem of “cube explosion” because it creates only *virtual* cubes—that is, star schemas pointing to a logical collection of data summaries that are the ingredients of the cube. It lets you skip the process of manually building summary tables that physically assemble all the necessary sets of data going into a cube—say, sales broken down by product, region, or time period—by delivering virtualized shortcuts called Materialized Query Tables or MQTs.

DB2 Cube Views cuts down on processing time because it doesn’t literally build cubes and it minimizes the number of MQTs it generates. DB2 Cube Views has a dedicated wizard to build optimal MQTs (called super-MQTs) based on the underlying schema. When running a query, the DB2 optimizer checks for the best combination of MQTs to resolve the query.

This super-MQT is more efficient than a collection of small, manually built, cross-referenced query tables because it has been optimized after the meta data descriptions are in place and therefore contains far fewer gaps or redundancies. DB2 Cube Views uses the super-MQT to deliver whatever virtual cube views OLAP users want. This approach saves precious time and reduces confusion and wasted space as well. Using super-MQTs may be compared to doing a Google search on various topics, which assembles information on the fly, as compared to photocopying articles and putting them in physical files. After a while, much of your printed material would be invalid or redundant, and in the meantime it would take up valuable office space.

Also, with DB2 Cube Views the original data stays where it is, so that all queries run off the *same* data structures in the *same* database on the *same* warehouse tier. This reduces the chance of discrepancies, confusion and wasted time reconciling conflicting information views. Plus, there’s less risk of loss, theft or “pollution” of the data that can be an enterprise’s most valuable asset.

DB2 Cube Views can help build special MQTs that decrease the load time of OLAP cubes created by IBM Business Partners’ OLAP tools. In addition, the effective use of MQTs pays off in performance boosts for *any* SQL-based query tool used on top of DB2 Cube Views. The meta data definitions underlying the MQTs guide, and therefore speed up, all SQL summarization queries into the database.

**Standards-based design leads to easier development, data access**

The SQL-based performance boost discussed above is just one example of how DB2 Cube Views provides easier data access because it is built on open standards. The use of industry-standard SQL throughout DB2 Cube Views makes life simpler for users and developers. They don't have to learn a complex, proprietary language, and a wide range of SQL tools and tutorials exist.

The open-standards approach also makes it easier to integrate reporting, data mining and other specialized BI tools from multiple vendors. Any third-party vendor's tool, even Microsoft Excel, will work effectively in connection with DB2 Cube Views because they share an understanding of SQL.

However, IBM encourages its certified IBM Business Partners—including leading analytic tool vendors such as Ascential, Brio, Business Objects, Cognos, and Microstrategy—to develop tools and applications that offer a higher level of integration than the SQL link discussed above with DB2 and IBM's other powerful information management products.

Close partner relationships pay off

With this open-standards policy, IBM opens up the APIs of DB2 Cube Views to these vendors so they can use XML to synchronize their meta data with DB2, making MQT generation and other query functions much faster and, once again, delivering answers sooner and with less labor and expense. These partners use the Cube Views APIs to create XML documents that say in effect, "I am a label; here's everything you need to know about my contents and what they're supposed to connect to." This open-standards approach to integration reduces the need to re-enter meta data into the client tools, which speeds up application development from days to a matter of minutes.

The use of XML to share meta data also allows hybrid OLAP, or HOLAP, to be done on the relational database. HOLAP typically relies on drawing summary data about aggregations from both the database and the OLAP query tool (hence the "hybrid"). But all too often the vast size of the underlying database forces the client tool to store too much data for optimum performance. But the XML-based sharing of meta data that Cube Views offers IBM Business Partners' query tools means that DB2 with MQTs can carry more of the data weight itself; this leaves only the most commonly requested data to be stored in the query tool, which can then operate faster.



A third advantage from closer integration using the APIs of Cube Views is quicker, more convenient reporting and OLAP cube navigation directly against the underlying DB2 database. For example, IBM OfficeConnect, an add-in for Microsoft Excel, lets you plug into a DB2 database, select a DB2 Cube Views cube, and slice, dice or drill down into the data using Excel spreadsheets, an interface that is familiar to most professionals. Users familiar with Excel's "pivot table" feature can start doing OLAP navigation sooner than they ever thought possible, without having to learn SQL query methodology (as they would if they used Excel without the OfficeConnect interface). This accelerated Excel interface puts the power of OLAP analysis directly in the hands of line-of-business professionals as well as specialized data analysts or DBAs, so that enterprises can get the fast answers they need to respond to changing market conditions.

IBM has designed DB2 Cube Views to make it more flexible and accessible to a wide variety of users, so that its benefits can be applied on demand. The CPU-based licensing structure lets users add more occasional users as needed, just by requesting the appropriate number of new CPUs. You aren't forced into a fixed number of seats for a fixed term. This saves the company the expense of unused seats while still allowing increases when needed.

Summary

DB2 Cube Views makes the relational database a more effective platform for OLAP by giving it a more OLAP-ready structure and by using meta data to label and index the data it stores. This delivers more powerful and cost-effective analysis and reporting on demand, both because the database can perform many OLAP tasks on its own and because it speeds up data sharing with OLAP tools. For example, a wizard in DB2 Cube Views makes short work of the process of aggregating and summarizing large blocks of data.



And because DB2 Cube Views is based on open standards, including SQL, XML and Web services, data sharing and integration with other vendors' tools is no longer a challenge, especially with query tools from certified IBM Business Partners. This means information can flow freely between departments, divisions or even whole companies, letting businesses make quicker and more accurate decisions based on better, richer, more up-to-date knowledge of market conditions. In fact, DB2 Cube Views is more than just a useful product in its own right—it is an ideal platform for creating a centralized BI architecture. Its flexible, scalable architecture and pricing are easy to expand or redeploy as company needs change.

DB2 Cube Views has another advantage as part of a BI platform—it is part of the larger family of DB2 information management tools:

- DB2 Universal Database Warehouse Edition, which includes DB2 Cube Views
- IBM DB2 OLAP Server, an integrated set of powerful OLAP and reporting tools
- The IBM DB2 Intelligent Miner™ data-mining suite
- A wide range of industry-specific BI solutions for banking, government, healthcare, retail, telecommunications and more.

IBM also provides a wide range of storage management software products within its Tivoli® software portfolio, as well as consulting and IT services to deliver the promise of e-business on demand.

With its emphasis on making data analysis and reporting faster, easier, more powerful and more accessible to a wide range of business users, IBM has distinguished itself as an enabler of realtime, cost-efficient information management. IBM solutions can give businesses an unsurpassed framework for collecting, storing, managing and deriving more insight from information, a powerful competitive advantage for any enterprise.

For more information

Please contact your IBM marketing representative or an IBM Business Partner, or call 1-800 IBM CALL within the U.S. Also, visit our Web site at:

ibm.com/software/data/db2/db2md



© Copyright IBM Corporation 2003

IBM Corporation
Software Group
Route 100
Somers, NY 10589
U.S.A.

Printed in the United States of America

06-03

All Rights Reserved

DB2, DB2 OLAP Server, DB2 Universal Database, the e-business logo, IBM, the IBM logo, Informix, Intelligent Miner, QMF and Tivoli are trademarks of International Business Machines Corporation in the United States, other countries or both.

Other company, product or service names may be trademarks or service marks of others.

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates.



Printed in the United States on recycled paper containing 10% recovered post-consumer fiber.

