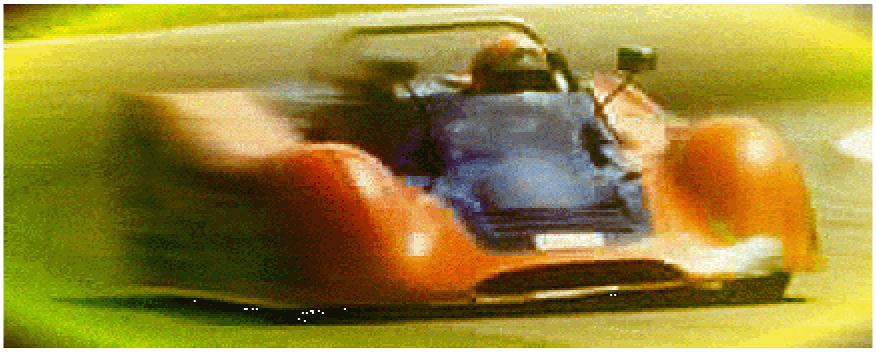
Alternative Technologies

IBM's Open Systems Database

Leveraging Robustness and Integration



David McGoveran
Alternative Technologies

13150 Highway 9, Suite 123
Boulder Creek, CA 95006
Telephone: 408/338-4621
FAX: 408/338-3113

Table of Contents

Executive Overview

Go back to the top

Introduction

This report is a summary of Alternative Technologies' comprehensive technical evaluation of DB2 Common Server. It also covers related products including:

IBM's DB2 Parallel Edition for data warehousing and DSS applications

DB2 World Wide Web Connection for transparent access to DB2 data from any Internet Web browser

- Distributed Database Connection Services (DDCS), DataPropagator, and DataJoiner for data integration (DB2 server interoperability, replication, and heterogeneous data access)

DataHub for centralized administration of both local and remote databases, including DB2, Oracle, and Sybase

Application development tools (each discussed briefly), including Approach, VisualAge, VisualGen, and TeamConnection

These products constitute a broad range of integrated database functionality. Unlike many offerings in the UNIX marketplace, they address robustness issues such as system administration and operational management problems in a manner that will be familiar to experienced mainframe personnel. Administrators increasingly are seeking solutions to these problems as workgroup, departmental, and divisional database applications are required to scale up to the enterprise. The challenges that such growth represents can hardly be overstated; far too many systems that work well when first deployed degrade as they are put under stress.

The causes of system degradation go beyond application design to issues that only can be the result of shortsightedness or lack of operational experience on the part of component vendors. In particular, components fail to integrate, to scale, and to have the management facilities necessary for larger operations. As a vendor with long experience in mission critical and rapid growth information systems, IBM's DB2 Common Server and related products bring a significant level of relational DBMS (database management system) product robustness and comprehensiveness to the open systems market.

Alternative Technologies' general assessment is that DB2 Common Server will become one of the premier open systems relational DBMS products. With the planned integration of DB2 Common Server and DB2 Parallel Edition, a product can be expected to emerge with one of the most complete feature and functionality sets available. The technical architecture of DB2 Common Server is well designed and should provide a foundation for scalability, robustness, performance, and many functional extensions over the coming years. IBM also provides a range of choice from a single-user full-function, SQL database for desktop business use to the massively parallel supercomputer for data warehousing.

IBM sees its products as addressing one of three categories of business users:

- *Functional Managers* -- these are users concerned with achieving specific goals for their department or function by purchasing a particular application
- *Asset Managers* -- these are users, such as database administrators, concerned with managing information and application assets
- *Solution Builders* -- these are users concerned with designing and developing applications that use the available information assets

Today's database products must support business applications and information integration. Servers for business applications must be robust and offer a complete suite of innovative functionality. Most applications are not, however, operated in isolation: tools are needed to move data, not only wherever and whenever it is needed, but in a desired form. DB2 Common Server is complemented by a rich array of tools that permit data movement and integration with other non-IBM client/server database systems as well as host systems.

The Business Environment for DB2 Common Server

Go back to the top

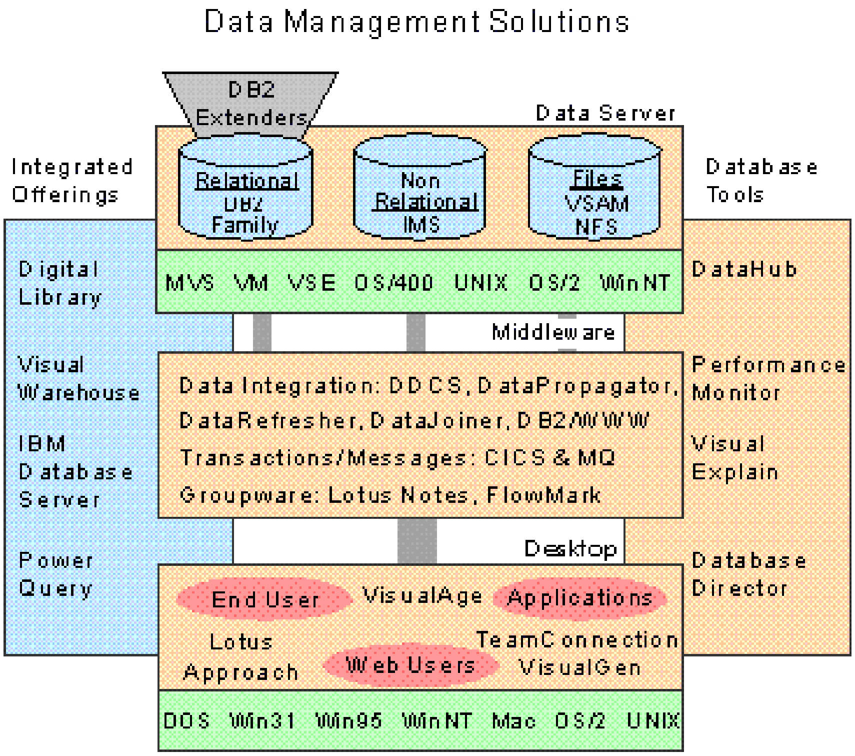
IBM has made tremendous strides in the UNIX marketplace with products such as DB2. With its current product lineup, it offers a broader spectrum of products and a more mature approach than most seasoned UNIX vendors. Given IBM's long standing as a supplier of robust, reliable, well-supported software, this will not surprise those familiar with IBM products.

IBM continues to work hard to integrate its products, make them more flexible and "open", and, at the same time, work with the technologies of the day. Its efforts to create one of the most sophisticated relational DBMSs, develop powerful middleware, and to incorporate object-oriented features in its tools and other products have begun to pay off.

With the introduction of DB2 Common Server Version 2, the product has plenty of functions and features to make it competitive with other open systems RDBMSs such as Informix, Oracle, and Sybase. While other RDBMSs have recently announced plans for some form of object support, DB2 Common Server already integrates key object concepts including user-defined data types, user-defined functions, and large object storage and retrieval. These capabilities enable the multi-media applications that many organizations are developing to advance their competitive position in the marketplace, capitalizing on object technology without abandoning the relational database approach. DB2 stored procedures reduce network traffic and cost. Declarative integrity and database triggers provide automatic integrity checks on a centralized basis, ensuring information consistency across multiple applications. In addition, DB2's very large number of ISVs offer solutions to a broad set of application and administrative problems.

IBM is positioning DB2 Common Server as the application server RDBMS. This is exactly what is needed in today's evolving IS environments, in which a mix of departmental and enterprise databases must be managed. Mixed environments arose because early adopters of the client/server initiative contended with the immaturity of client/server applications, databases, and system management tools by implementing multiple, departmental solutions rather than integrated, enterprise solutions. Because these departmental solutions offer acceptable functionality, are unlikely to be replaced. Later adopters of client/server have focused on connectivity of application systems. As a result, information integration facilities are paramount for growing and consolidating application systems into a single enterprise solution.

System administrators constantly are being asked to do more with ever decreasing budgets. IBM's ability to bring key mainframe tools and strategies to the distributed, open systems environment should provide good options to system administrators seeking to invest in lower cost platforms. Products such as DataPropagator and DataHub (see later sections) will appeal greatly to administrators as a means of integrating and managing rather than redeveloping existing information assets.



IBM has brought its experience in (and long standing reputation for) delivering well-tested software products to bear on the UNIX market, specifically AIX. The result is that DB2 Common Server is one of the few open systems RDBMS products to undergo extensive robustness and stress testing. RDBMS vendors commonly confirm that design limitations are not restrictive in a typical customer environment. However, they rarely test to determine the effects of "pushing the envelope" in a large number of areas simultaneously. For example, testing the combination of high transaction rates, complex queries, and large, wide tables can sometimes reveal surprising limitations in today's products. By contrast, IBM addresses stress testing of DB2 directly.

1. Company Overview

Go back to the top

International Business Machines (IBM), based in Armonk, New York, is one of the world's largest manufacturers of computer equipment, from large mainframes to mini-computers to PCs and workstations. It is involved in every aspect of the computer industry, including hardware, software, and services. The company has a number of offices in the USA and most major countries around the world. IBM employs about 225,000 people worldwide.

The DB2 Common Server license base at the end of 1995 consisted of over 750,000 licenses for OS/2 and over 5000 licenses for AIX worldwide. Approximately 90% of the Fortune 500 run DB2. According to IDC, worldwide RDBMS and development tools revenues for 1995 were approximately $3.4 billion, giving IBM the largest share of the market by revenues (12.8%). Of this, approximately $110 million was associated with UNIX, NetWare, NT, and OS/2 environments with a compound annual growth rate of 115%. IBM Corporation has an impressive number of VARs and ISVs (between 1200 and 1500 supply products related to DB2 Common Server).

IBM supplies database products for use in a number of operating environments. Its principal database products include:

- IMS/ESA for the MVS/ESA operating system environment.
- DB2 for the MVS operating system environment.
- DB2 for the VM and VSE operating system environments, also known as SQL/DS.
- DB2 for the OS/400 operating system environment.
- DB2 Common Server (also known as DB2 Version 2), a server with 80-90% code shared across environments and the remainder designed to exploit specific environments. The following environments are supported:

- DB2 for the AIX operating system environment (previously known as DB2/6000).
- DB2 for the OS/2 operating system environment (previously known as DB2/2).
- DB2 for the HP-UX operating system environment.
- DB2 for the Sun Solaris operating system environment.
- DB2 for the Windows NT operating system environment.
- DB2 for the Siemens Nixdorf SINIX operating system environment

Database 2 Parallel Edition, a version of DB2 for IBM's SP2 AIX environment with parallel query and utility capabilities. Currently based on DB2/6000 Version 1, it will be merged with Version 2 in a future release.

2. DB2 Common Server

2.1 Overview

Go back to the top

DB2 Common Server is a relational DBMS for the UNIX (HP, SINIX, and Sun Solaris), micro (OS/2), and Windows/NT marketplaces. DB2 Common Server Version 2 was announced in September 1994. Separate products exist for IBM mainframe operating system environments including MVS, VM, and VSE. A version of the product is designed for the midrange AS/400 systems.

Access to DB2 data is supported from applications running in a variety of environments including AIX, HP-UX, DOS, OS/2, Windows 3.1, Windows 95, Windows NT, Solaris, etc. DB2 Common Server is a thirty-two (32) bit implementation of IBM's DB2 product, which improves performance, scalability, and reliability. The key features of DB2 Common Server are discussed in terms of relational model (structural, manipulative, integrity), architectural, and performance features. Strong relational model support is important to business users, in that it guarantees database correctness and flexibility while minimizing development and maintenance costs.

2.2 Relational Support: Structural Features

Go back to the top

Support for the structural component of the relational model is primarily indicated by a DBMS's support for relational tables (either base or derived) and domains. Logically, DB2 Common Server stores all data in relational tables; physically each table can be up to 64 gigabytes in size. The maximum number of columns in a table is 255. Tables can be placed in multiple tablespaces and tablespaces can be assigned to physical devices. This helps provide more granular data loading, backup, and recovery, and improves both parallel I/O and I/O balancing.

DB2 Common Server does not support relational domains, but does support ANSI SQL '92 data types (primarily a physical concept). Type compatibilities are checked automatically in the DBMS and violations result in the raising of errors. DB2 Common Server also supports user defined types (UDTs) implemented with strong typing. A UDT is a renamed built-in type which generally has a different behavior than the source type. Cast functions to and from the source type are created automatically. If created using the WITH COMPARISONS option, a set of comparison functions is also created. Consistency of data with its assigned type is enforced by the DBMS eliminating the need for this to be done in applications.

DB2 Common Server also supports the creation of views. General view updating algorithms are not supported at this time. However, the usual types of single table views can be updated, and update and delete are supported for UNION ALL views.

Large Objects

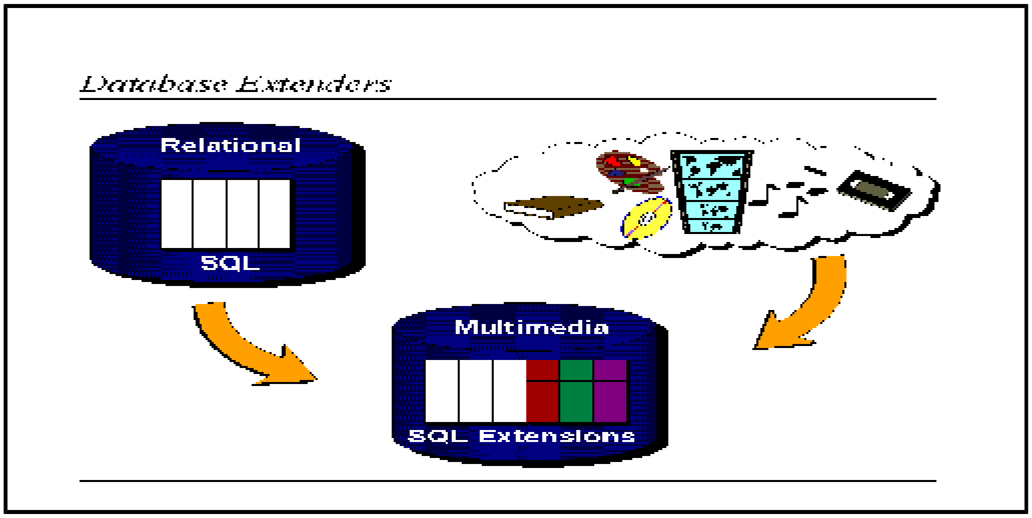
Go back to the top

Object definition, storage, and manipulation for multi-media, text search, and other applications is supported via UDTs, LOB (large object), and user defined functions (UDFs -- see below). A particular LOB may be either a CLOB (character LOB) or a BLOB (binary LOB). Both reference and manipulation of LOB data are made more efficient by storing a logical address, rather than the actual LOB data, in the rows. This means that queries that do not require LOB data do not incur the cost of accessing the data. Likewise, a LOB tablespace can be physically moved without the cost of reorganizing or even scanning the rest of the table. Comparisons and updates can be performed without bringing LOB data into the application. Retrieve, replace, LIKE, concatenation, SUBSTRING, POSITION, LENGTH, and UNION ALL operations are supported directly in SQL. Special LOB functions (GetSubstring, GetPosition, and GetLength) are supported in CLI (the call level interface). A number of LOB storage management features have been implemented, including:

- LOBs are maintained directly in the database
- many LOB columns are permissible per table
- a LOB column can be up to 2GB in size
- logging of LOB data is optional (it is specified at definition time)
- SQL can be used for direct transfer of LOB data to or from files

Relational Extenders

Go back to the top



DB2 Common Server features *relational extenders* that improve storage and manipulative flexibility. Relational extenders are UDTs and UDFs that support special access methods, physical storage, operators, etc., to enhance rapid development of special types of applications. IBM initially is offering image (with numerous common formats and Query By Image Content), text, video, audio, and fingerprint extenders. A kit of template code and design guidelines will be available for adding other relational extenders to DB2 Common Server.

2.3 Relational Support: Manipulative Features

Go back to the top

Manipulative support is documented here in terms of the SQL dialect and related capabilities such as stored procedures and data access authorizations. The more comprehensive the support, the easier it is to manipulate the data and, therefore, to develop applications. DB2 Common Server Version 2 manipulative support is a superset of that found in DB2/MVS Release 3.1. DB2 Common Server Version 2.1 is predominantly SQL92 Entry level compliant; full compliance will be supported with Version 3. Along with the usual ANSI SQL, DB2 Common Server supports a number of advanced SQL capabilities. Among these are recursive SQL (self-referencing expressions), common table expressions (named table expressions that can be treated like "local" temporary view definitions), table expressions in the FROM clause, and CASE expressions in either the SELECT list or WHERE clause. Each of these, especially recursive SQL, permits the use of a single SQL statement to achieve functionality that would ordinarily require multiple SQL statements.

Scalar full selects can be used as the source of updates, on the left hand side of expressions, and in SELECT lists. The VALUES clause of an INSERT can have multiple rows of data: These can be used for single row SELECTs, LOB retrieval, array inserts, and in-memory temporary "tables". These capabilities permit the writing of more flexible and faster performing queries.

DB2 supports extended stored procedures which IBM refers to simply as "stored procedures." DB2 stored procedures are accessed via either a call interface (Database Application Remote Interface, also known as DARI) or a simple CALL statement. Similar in some respects to the stored procedures supported by other relational DBMS products, these procedures are remote from the client and reside on the DB2 Common Server platform as application code external to the DBMS. As a result, they can be used to minimize network traffic between the client and the server. They are neither defined nor stored in the database (although the compiled SQL package is), but are written as DLLs in a host language (C or BASIC). This implementation permits work to be done outside the database such as sending mail, file I/O, complex computations, etc. DB2 stored procedures can be created to run in the same address space as the DBMS, minimizing the cost of invocation and data reference. If the user considers this unsafe, DB2 procedures can be run in their own address space so that severe procedure errors cannot damage the database. As well, when DB2 Common Server is teamed with DDCS, the client application can execute a DB2 stored procedure registered in DB2 for MVS Version 4.

UDFs (user defined functions) are external modules that extend and tailor the language. They can reference other pre-defined functions, can send mail, and so on. The use of pre-defined functions facilitates code reuse. UDFs can execute in either the same address space as the DBMS or in a separate address space (to protect the DBMS from catastrophic errors). UDFs can be invoked wherever an expression can be invoked, can return a single value, and can operate over all database types. They are typically written in C with C call conventions. Name overloading is permitted, the intent being to make application development easier.

DB2 Common Server security provides read, update, and utility access controls, supports SQL GRANT and REVOKE statements. User verification through external security products is not yet supported. Integration with DCE security is planned. The compiled SQL associated with an application is contained in "packages" which are stored in the database. DBMS authorization control over the execution of a package provides application level security. The user of the application only needs the privilege to run the package, rather than privileges for the individual statements in the application.

2.4 Relational Support: Integrity Features

Go back to the top

Integrity features provide automatic DBMS control of database consistency and correctness. DB2 Common Server supports both declarative constraints and database triggers to enforce integrity rules associated with data. DB2 Common Server supports, but does not require, primary keys. Foreign keys can be defined for referential integrity enforcement. Declarative referential integrity constraints can be defined with RESTRICT, NO ACTION, CASCADE, and SET NULL actions for DELETE and NO ACTION or RESTRICT actions for UPDATE. The DBMS automatically optimizes the execution of referential integrity constraints with the CASCADE and SET NULL actions.

CHECK CONSTRAINTS are defined via CREATE TABLE or ALTER TABLE. These are essentially WHERE clause constructs, including comparisons, BETWEEN, LIKE, IN, and user defined functions. No subqueries, column functions, host variables or special registers are permitted. Constraints can be turned on or off individually at the table level. If turned off, the table is automatically put in a check pending state after the first update.

DB2 Common Server supports business rules through DBMS triggers. Triggers are defined by the user, stored in the catalog as text, and compiled with the triggering SQL statement. Multiple triggers (BEFORE EACH ROW, AFTER EACH ROW, and AFTER EACH STATEMENT) are definable per INSERT, DELETE, or UPDATE. AFTER EACH STATEMENT triggers fire after all AFTER EACH ROW triggers have fired. Triggers of the same type fire in the order they are created. Triggers can contain INSERT, searched UPDATE, searched DELETE, or SELECT statements, and can cascade up to sixteen levels deep. They can co-exist with declarative referential integrity actions with no restrictions.

2.5 Performance Features

Go back to the top

DB2 Common Server implements a number of features intended to improve performance. Of these, buffer management, utilities, resource management, and the optimizer are probably the most significant.

Buffer management performance features include look-aside, deferred write, chained write, index prefetch, list prefetch (in which row IDs are sorted prior to fetching pages), parallel I/O, and sequential prefetch buffer management. The use of sequential prefetch is triggered by the automatic detection of sequential processing, even when singleton SELECT statements are used (instead of a cursor). DB2 Common Server buffer management also includes support for multiple buffers, big block reads (the size is associated with the tablespace), asynchronous page cleaners, and caching of the directory.

In the area of *utilities*, a high speed, restartable LOAD utility provides a faster option than IMPORT. LOAD supports index creation, statistics, exception tables, and direct media access (RAW devices); and is compatible with backup and restore parallelism. A simple query governor is provided to limit I/O consumption. A more complete *resource control* facility is planned for a subsequent release. An API is provided for monitoring of numerous performance-related statistics and configuration parameters. Most configuration parameters can be modified. Although modifications to most of these do not take place online, it is possible to fine tune most physical resources for best performance.

A strong *optimizer* is essential to performance, scalability, and data independence. The DB2 Common Server optimizer is one of the most sophisticated in the industry. Part of its power comes from the ability to rewrite a query in a form that is more efficient. DB2 selects a plan for executing this rewritten query by evaluating the performance cost associated with various algorithms and access methods. The selection is based on statistical information about data and resources (including the operating environment, such as disk and CPU speeds). Query rewrite is used to guarantee that a DBMS runs efficiently with a variety of front end tools, regardless of the SQL they produce. DB2 may have the best query rewrite facility of any relational DBMS on the market today.

DB2 Common Server keeps a variety of statistics regarding non-uniform data value distributions ("*n*" most frequent values), index clustering, UDFs, and tablespace cost parameters including overhead and transfer rate. Statistics are produced by running the RUNSTATS utility, which can produce a statistics report without updating the catalog. The optimizer uses both dynamic programming and greedy join enumeration to evaluate join algorithms. Either merge scan or nested loop join algorithms may be used. The optimizer is designed to support an unlimited number of tables in a join.

2.6 Server Architecture

Go back to the top

DB2 Common Server is written to consist of 80-90% common code shared across multiple environments, while the remainder is written to exploit each particular environment. The user interfaces, semantics, and APIs are the same across all platforms.

DB2 Common Server is designed to be a multi-threaded server. Units of processing, referred to generically as "agents", are implemented as either native OS threads or as OS processes. Native OS threads are used if supported by the platform. If native threads are not available across all supported versions of an operating system (such as AIX), processes are used instead. Some agents are created at initialization and additional agents are created at connect time as needed. The total number of agents that can be created is limited only by resources or OS limits.

Among the agents implemented are the Communication Manager, System Controller, Coordinator, Database Deadlock Detector, Database Logger, Prefetcher, Page Cleaner, and Resync Manager. The function of each of these agents is as follows:

- The *System Controller* agent handles DB2 initialization, shutdown, and housekeeping.
- The *Communication Manager* agent handles communication requests from clients, spawning an agent for each. It comes in various flavors, depending on the communications protocol (IPC, TPC/IP, SNA, or IPX).
- A *Coordinator Agent* handles all SQL processing for a particular connection and is allocated from a pool of available agents at connect time.
- *Prefetcher* agents handle read-ahead, big block I/O, and parallel I/O tasks and are allocated at connect time (the number depends on a configuration parameter).
- The *Database Deadlock Detector* detects and resolves deadlocks in the database.
- The *Database Logger* agent handles all log I/O for a particular database.
- *Page Cleaner* agents (there may be more than one) writes dirty buffer pool pages to disk asynchronously so that committed changes are written to disk. This procedure frees up space so that new pages can be read into the buffer pool. It also speeds crash recovery time since fewer transactions need to be read from the log and applied to the database.
- The *Resync Manager* handles recovery of in-doubt distributed transactions for DRDA.

Additional agents handle tasks such as spawning, utility operations, and abnormal termination.

Availability is a key architectural feature of DB2 Common Server. It provides backup and recovery at the database and tablespace levels. Backup is online, and the database can be online while an offline portion is recovered. Although tables cannot be partitioned, they can be assigned to multiple tablespaces potentially improving backup and recovery times over full table or database level operations. Dual logging (a robustness feature) is not supported, although the DBMS is compatible with mirroring supported by the environment. Log archiving is normally done automatically when the active log becomes full, but can also be initiated by the operator.

DB2 Common Server provides transaction management features including automatic transaction backout on errors and automatic recovery at system startup. Multi-user transaction isolation is provided via locking at the row level. Uncommitted read and committed read transaction isolation levels provide improved concurrency for tasks such as report generation. Repeatable read and serializable transaction levels are provided for higher levels of consistency enforcement.

---

3. DB2 Parallel Edition

Go back to the top

B2 Parallel Edition Version 1.2 is a shared nothing architecture version of DB2/6000 Version 1 (now called DB2 for UNIX). As such it does not have all of the enhancements of DB2 Common Server, but does include support for left and right outer join, CASE clauses in the select list, multi-page block writes during insert and create index operations, non-logged temporary tables (important for OLTP), and GUI database administration based on Database Director.

Because of its parallel architecture, DB2 Parallel Edition offers significant performance and scalability, categorized as follows:

parallel query processing: All relational operations can be performed in parallel, including joins. Three parallel join algorithms are supported: broadcast, re-distributed, and co-located.

parallel optimization: An N-way parallel join strategy, the degree of parallelism, and both horizontal and vertical parallelism are optimized.

parallel I/O: I/O can be performed in parallel across multiple nodes and I/O can be overlapped across nodes, but not across multiple storage controllers within a node.

data partitioning: Data rows can be assigned to user-defined node groups via a system-defined hash function. Data may be loaded into these partitions in parallel.

parallel utilities: Reorg, backup and restore, forward recovery, index build, and partition rebalance are all performed in parallel.

DB2 Parallel Edition Version 1.2 also adds various query optimizer improvements and a query governor. The query governor monitors resources usage based on elapsed time, CPU consumption, rows accessed, and locks held. Limits on resource usage can be set based on time-of-day, user ID, or application and applied to the query, transaction, or package level. The query governor can respond to excessive resource consumption by lowering the execution priority automatically based on rules, can send email to designated personnel, execute a utility program, cancel the request, or halt process and wait for manual intervention.

MVS applications that use the DRDA protocol can now access DB2 Parallel Edition (acting as a database server) without changes to the "client" portion.

DB2 Parallel Edition runs on LAN connected RS/6000 and RS/6000 SP systems. One or more HACMP/6000 complexes may be used if high availability and node failover are desired. It has obvious advantages in complex query and query intensive applications such as data warehousing and decision support.

---

4. Data Distribution Facilities

Go back to the top

Information integration is supported through various data distribution facilities, including distributed transaction management. There are three environments that must be separately considered:

- DB2 Common Server to DB2 Common Server -- Two-phase commit for multi-site update supported without any required user action (that is, transparently).

DB2 Common Server to DB2 for MVS and DB2 for OS/400 -- Distributed transactions are supported via DRDA by the product Distributed Database Connection Services (DDCS). DRDA is a set of protocols that define how clients and servers interoperate and is supported by all DB2 servers. DDL (data definition language) and DCL (data control language) are supported in a remote unit of work. Remote unit of work access (including support for DML, DDL, and DCL operations) to remote DB2/ VM and DB2/VSE (SQL/DS) systems is also supported.

- DB2 Common Server to any XA compliant servers -- The XA interface provides distributed transaction support to any XA compliant server, including Encina, Tuxedo, and CICS.

Distributed Database Connection Services (DDCS)

Go back to the top

Distributed Database Connection Services (DDCS) is a gateway implementation of DRDA implementation for clients in the OS/2, AIX, HP-UX, and Solaris environments. DOS, Windows, OS/2, and UNIX applications can use DDCS to work with data stored on any DRDA enabled server. DDCS has demonstrated high performance transactional support in an OLTP environment. It is also provides efficient services for decision support system (DSS) applications by blocking data pages, grouping statements together (e.g., deferring PREPARE until an EXECUTE statement occurs) and caching catalog information from the remote server at the gateway.

In addition to these native facilities, a number of optional products provide important information integration functionality. These products are described briefly in the remainder of this section.

4.1 DB2 World Wide Web Connection

Go back to the top

DB2 World Wide Web (DB2 WWW) supports transparent access from a Web browser on any platform to DB2 data. DB2 WWW is a Web server CGI-BIN gateway which is installed on the Web server together with applications called macro files. The DB2 WWW application developer provides the macro files, which contain the hypertext markup language (HTML) definitions and SQL commands needed to query DB2 from the Web browser and to return the results in Web page format. DB2 WWW supports SELECT, INSERT, DELETE, and UPDATE capabilities, providing an efficient and inexpensive method to enable access to existing data for both Internet and intranet applications.

Each DB2 WWW Connection macro file supports connection to one database at a time and can contain multiple SQL commands. DB2 authentication can be accomplished by allowing login and password entry from the browser or by defining login and password as special variables in the macro file. DB2 WWW can access DB2 on the Web server or can be used for remote data access to the backend server. When used with DataJoiner (see below), DB2 WWW provides a solution to access heterogeneous data from a Web browser client. DB2 WWW Connection is part of IBM's Internet strategy and integrates with IBM's Internet servers and secure gateways. DB2 WWW Connection for OS/2 and AIX are downloadable at no charge from IBM's Software Homepage, and are also packaged with the Database Servers and the Internet Connection Servers for those platforms.

4.2 DataJoiner

Go back to the top

As its name suggests, DataJoiner supports multi-location queries against distributed multi-vendor databases. Thus it implements the well-known "distributed request" level of distributed database processing and uses sophisticated techniques for distributed optimization. DataJoiner supports transparent multi-site read and single-site update to multiple heterogeneous databases including DB2, IMS DB, VSAM, Oracle, Sybase, and MS SQL Server. Client Application Enabler (CAE) is a client interface that supports access to DataJoiner through both ODBC and its superset, X/OPEN CLI. Access through embedded SQL is also supported. DataJoiner is based on Version 1 of DB2 for UNIX (known as DB2/6000). The product globally optimizes multi-site read and write requests based on statistics collected from the catalogs or data from each foreign source. Global optimization considers a number of parameters in selecting the best strategy for executing a query, such as: relative communications, I/O and CPU speed, indexes available, cardinality in referenced tables, number of pages the table occupies, number of levels of an index, and index key cardinality.

Used in conjunction with DataPropagator, it permits replication to and from non-native databases. As a complete DBMS in its own right, it can be used as an extract or staging database. DataJoiner provides a single API for accessing (including joining) data in all connected databases using ANSI SQL, including join capability. DataJoiner has the ability to compensate for functions not contained in the semantics of the native SQL of a database. This is achieved by simulating functions that are lacking in the remote data source. For enterprises that have data stored in multiple sources and want to be able to take effective advantage of it without having to incur the costs of getting it into a common source. The ability to treat multiple sources as though they are a single, large database against which one writes queries is a cost-effective, manageable solution to an overwhelming problem. DataJoiner is a powerful facility for integrated data access.

4.3 DataRefresher

Go back to the top

DataRefresher is a full refresh copy facility running under MVS and VM with an OS/2-based GUI for defining sources and extracts. It can be used to extract data from DB2, IMS, VSAM, flat file, and other data sources, and automatically apply them to DB2 databases (if DB2 Common Server is used, transport and load of the extracted data require a manual step). The user registers the data source and then defines the extract via the GUI front end. DataRefresher can refresh DB2 databases from non-relational data sources including IMS, VSAM, flat files, and other. A variety of exit routines can be used to change data formats, date and time conversions, convert unsupported data types to supported data types, and track resource usage. DataRefresher is best used to copy larger amounts of data on an infrequent basis. DataRefresher extracts can be used as a source by DataPropagator Relational and its data definitions and mapping can be used by DataPropagator NonRelational.

4.4 DataPropagator

Go back to the top

For the many installations that require asynchronous updates to local data, DataPropagator supports this capability across the DB2 family. Used in conjunction with DataJoiner, it also supports replication to and from non-native databases (Oracle and Sybase). DataPropagator is available in two versions: Relational and NonRelational. It consists of DataPropagator Relational Capture (on MVS, OS/400, AIX, Windows NT, HP-UX, Sun Solaris, and OS/2) and Apply (on MVS, OS/400, AIX, HP-UX, OS/2, and Solaris). In addition, DataPropagator NonRelational supports bidirectional, asynchronous and synchronous replication (for IMS to or from DB2).

DataPropagator Relational can be used to select, transform, and copy data asynchronously between DB2 databases. Data can be copied on scheduled intervals ranging from minutes to weeks. Both refresh (replacement of the target) and update (changes only) copy requests are supported, although update replication requests are not supported for VM and VSE environments. The Capture component is used to read log or journal data and write changed rows to staging tables. The Apply component is used to move data from source staging tables to point-in-time, history, or target staging tables. Point-in-time tables represent source data at a point-in-time. History tables contain a record of data over time. Staging tables are used for further copies, so that data can be unique in content and structure. The Apply component can also move data directly from the source tables themselves to the targets in the case of refresh-style data replication.

Tables must be registered as sources (registration), copy definitions must be created and maintained (subscriptions), and registration and subscription privileges must be granted. Any SQL view can be registered, and subsetting can be performed by the Capture component. SQL statements can be executed on the data either before and/or after the data is replicated to the target. SQL can be used with the Apply component to subset data by columns and rows, create new columns, and summarize data. Subscribers also specify registered source tables, copy frequency, and select columns and rows. DataHub for OS/2 provides a graphical user interface for replication administration. DataPropagator logs its behavior in trace and audit control tables, which can be analyzed with user-written SQL programs.

DataPropagator NonRelational supports asynchronous propagation from IMS to DB2,

and synchronous two-way propagation between IMS and DB2 (assuming DB2 updates are issued via the IMS attachment facility). For asynchronous propagation, the Selector component is used to capture changed data from IMS log data sets. The Receiver component reads captured data from PRDS and calls a program to issue static SQL DB2 updates. With synchronous propagation, the Selector and Receiver components are eliminated and changes are "captured" via a call exit. As a result, both source and target updates occur in the same unit of work.

A mapping definition is used to specify (1) the relationship between key fields and table keys, fields and columns, and automatic data type conversions, (2) one of three methods of mapping IMS data segments to DB2 tables, and (3) one of two mapping options to control conditions for copying and nonkey field mappings. Mappings are done using a GUI OS/2-based tool and mappings are shared between DataPropagator NonRelational and DataRefresher. Segment, field, and propagation exit routines permit means of modifying segments, fields, and propagation methods, respectively.

DataPropagator products should be considered whenever the enterprise has requirements for data to be replicated on a scheduled or event-driven basis. These products can help keep multiple copies of data synchronized with the source copy and access privileges to each copy can be independently managed as needed, improving overall system security. Users choose what data is to be distributed, where, in what format, and how often. By automating the execution of these choices, replication products like DataPropagator can decrease systems management costs and reduce network overhead.

---

5. Administrative Facilities

Go back to the top

5.1 Utilities

In Version 2 of DB2 Common Server, IBM has added the Database Director. The main administrative utilities provided by the Database Director and command line interface include:

BACKUP -- online and offline full and partial backup of the database

RESTORE -- restore the database offline, or restore individual tablespaces while the rest of the database is online

RECOVER -- roll forward the database, including point- in-time capability, and roll forward a tablespace to the end of the log

LOAD -- load the database to operating system files

REORG -- physical reorganization after many updates, inserts, deletes. ***NOTE:*** Space reclamation and reorganization are not dynamic, and REORG must be performed offline.

RUNSTATS -- update database statistics.

EXPLAIN -- a facility for examining query plans, EXPLAIN provides information not only as to what the query plan is, but why the plan was selected, and why other plans were not selected.

Most DB2 administrative utilities are restartable, thereby improving availability and robustness. As well, the database supports SNMP subagents for database monitoring from a central network console like NetView. The database can be registered in DCE directories for a central server lookup.

5.2 DB2 Visual Explain

Go back to the top

Of particular interest to application builders (SQL specialists in particular), DB2 Visual Explain produces a graphical display and analysis of query plans. It also produces detailed optimizer information such as cardinalities, catalog statistics, predicate information, I/O costs, and CPU costs, as well as what-if modeling. Statistics can be modified and optimization classes can be set. This tool is designed to help tune the queries used in applications for optimal performance.

5.3 DB2 Performance Monitor

Go back to the top

DB2 Performance Monitor reports on the various statistical information collected by DB2. It provides periodic, exception, and event driven data capture, problem analysis and determination, alert notification, logging, real-time viewing, and performance tuning capabilities. Information about locks, deadlocks, buffer pools, sorting, communications, agent, and logging can be acquired. Derived performance measurements can be defined based on calculations on existing performance measurements. For each measurement, threshold values can be defined which invoke notification, audible alarms, logging, and/or execution of a command or program. The facility is designed to help pinpoint system performance problems so they can be corrected before they become serious.

5.4 DataHub

Go back to the top

DataHub is a database systems management tool providing single point of control management of objects, operations, thresholds, authorizations, storage, and configurations. Versions run on OS/2 and UNIX (AIX, HP-UX, Solaris). It can be used to manage multi-vendor geographically dispersed databases, facilitate expansion into client/server, reduce staffing requirements, and provide unattended remote management. The OS/2 version uses SNA and can be used to manage DB2 running under VM, MVS, OS/400, OS/2, and AIX. The UNIX version uses TCP/IP and can be used to manage DB2 for UNIX, Oracle, Sybase, CA-OpenIngres, and, in the future, Informix RDBMS servers.

DataHub can be used to monitor, create, destroy, copy, and manage objects on both remote and local RDBMS servers. Transaction status can be monitored. Administrative and other tasks, such as backup and recovery, can be scheduled for automatic remote execution. The facility includes support for events and alarms. Checks and rules (many are provided with the product) provide DataHub's threshold monitoring and automated management capabilities.

Because each database managed by DataHub does not require a separate administrator, the burden of managing a distributed, possibly geographically dispersed, environment is greatly eased. From a single screen, the DBA can see all of the databases and objects in those databases (such as tables) and can manage them from that screen.

---

6. Development Tools

Go back to the top

Most businesses need to develop and enable new types of applications and technologies that quickly respond to today's changing business requirements if they are to gain a competitive edge. These new applications have to fit new business processes and work flows that have been re-engineered to enhance customer service. Developers benefit most if they can leverage their existing skills and tool sets while adopting new technologies at a reasonable pace. IBM's application development business provides a set of tools and an integrated development environment intended to address these new requirements.

In this section we provide an overview of IBM's key DB2 Common Server application development tools. These tools help solution builders provide the applications needed to meet the needs of functional managers.

6.1 Overview

Go back to the top

DB2 applications can be written using a traditional fourth-generation language (for example, VisualGen) or new tools such as VisualAge, which combine the power of object oriented technology with the ease of visual programming. Additionally, familiar third-generation programming languages (3GLs) precompilers including BASIC, COBOL, PL/I, REXX, FORTRAN, C, and C++ can be used. Other languages can be used to develop non-SQL applications or modules including ADA, APL2, Pascal, Prolog, RPG, Assembler, or LISP. In addition, a call level interface to DB2 exists for the C language.

All access to DB2 data is done using SQL. 3GL support for SQL is primarily through precompilers (as contrasted with preprocessors). A precompiler is more tightly integrated with the compiler, making SQL symbols accessible to debugging facilities. The SQL (ALL) precompiler option allows the precompiler to accept non-native (for example, SQL/DS) SQL syntax.. The products include support for packages and "hold" cursors.

6.2 VisualAge

Go back to the top

IBM VisualAge is an integrated set of tools for rapid development of object-oriented client/server applications running under OS/2, Windows, and AIX. VisualAge is designed for teams of programmers requiring multi-user programming, library management, version control, and configuration management. The user can develop applications that integrate text, audio, image, and video. DB2, Sybase, SQL Server, and Oracle, as well as other relational DBMS products (accessible via ODBC) are supported.

IBM VisualAge was first released with a Smalltalk development environment. It now also supports C++ and COBOL (with object oriented extensions). VisualAge offers developers increased flexibility and productivity with an integrated set of tools for rapid development of object oriented client/server applications. VisualAge applications can be developed and deployed on OS/2, Windows, or UNIX operating systems.

IBM intends to adopt the VisualAge visual programming paradigm across a variety of its application development products. By adopting a single language and platform independent visual development environment, both individual and team productivity should be enhanced. The uniformity of the visual interface is also intended to protect investments in systems, application code, and development skills.

VisualAge for Smalltalk

Go back to the top

VisualAge for Smalltalk is a set of object oriented tools for developing mission critical applications. It consists of both client and server components. Applications deployed on the workstation employ a GUI (graphical user interface). VisualAge for Smalltalk programmers use ready-built components to build applications via a construction-from-parts approach. This approach is intended to improve productivity, without requiring object oriented skills. As a result, it can be expected to ease the migration to object oriented technology.

There are separate client products for OS/2, Windows (including Windows 3.11, Windows NT, and Windows 95), and AIX. One server product that supports all three client platforms. VisualAge for Smalltalk provides native support for both the DB2 family and Oracle. In addition, an ODBC-compliant call level interface provides access to other databases.

VisualAge for C++

Go back to the top

IBM's VisualAge for C++ combines a traditional C++ compiler with the power of a visual application builder. VisualAge includes Data Access Builder, the IBM Open Class Library, and a compound document facility. Data Access Builder visually maps DB2 relational database tables into "objects" with a single mouse click. The IBM Open Class Library is a large set of C++ class libraries which serve as building blocks for creating applications. It handles the complexity of low-level APIs (application programming interfaces) and provides platform independence. The compound document facility simplifies OLE programming. VisualAge for C++ is currently available on OS/2, Windows NT, and Windows 95.

VisualAge for COBOL

Go back to the top

VisualAge for COBOL adds object oriented extensions to COBOL in a visual development environment. OS/2 developers can create graphical user interface (GUI) applications in hours rather than days. It is intended for the creation of advanced client/server applications on the desktop and the deployment of objects across various platforms. "Assistants" eliminate a great deal of tedious, error-prone development work and come in two flavors: data assistants and transaction assistants. VisualAge for COBOL supports IBM's SOM (System Object Model), including a DSOM (Direct-to-SOM) feature, making it easier to create SOM objects. VisualAge for COBOL productivity benefits include object oriented programming, visual development, and enhanced database access. Traditional COBOL developers should be able to achieve these benefits with little or no retraining.

6.3 VisualAge Generator

Go back to the top

VisualAge Generator (formerly known as VisualGen) is a 4GL workstation-based, visual programming tool for building business critical applications rapidly. It is used to develop and generate both event-driven GUI client/server applications and COBOL server applications. Although it runs on OS/2, applications can be targeted for most DB2 environments.

The product contains the same visual construction environment as IBM's VisualAge development tools. From an OS/2 development platform, VisualAge Generator can generate C++ and COBOL applications that run on OS/2, AIX, Windows, OS/400, MVS, and VSE/ESA. VisualAge Generator also offers extensive testing and debugging features.

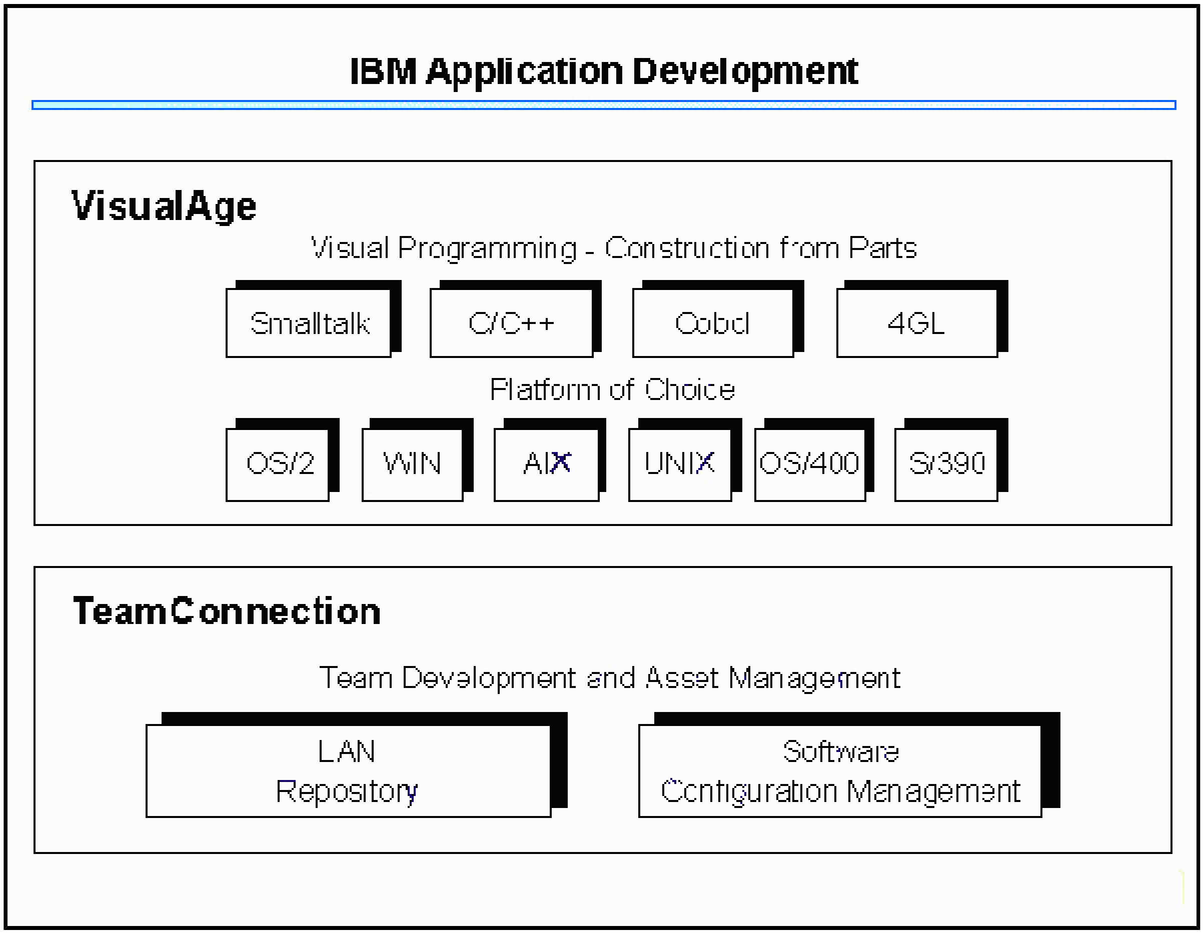
VisualAge Generator (currently sold as VisualGen Version 2.0) is available for both OS/2 and Windows client platforms. It supports a number of IBM server environments including OS/2, CICS/6000, OS/400, AIX, MVS, VSE/ESA, and CICS OS/2.

6.4 TeamConnection

Go back to the top

TeamConnection is a combination of software configuration management and object oriented information services, which allows team development across most products in the VisualAge family. These services are combined via a semantic model for tool integration. The product is intended for both large and small development teams. TeamConnection helps businesses manage their LAN-based software development allowing an integrated collaborative approach. Data and objects can be shared and managed throughout the development cycle, from business modeling through application deployment.

Features include configurable process support for defect tracking and change management, project management across the development life cycle, security and asset management, and multiple production platform support.


IBM Application Development
VisualAge
Visual Programming - Construction from Parts
Smalltalk
C/C++
Cobol
4GL
Platform of Choice
OS/2
WIN
AIX
UNIX
OS/400
S/390
TeamConnection
Team Development and Asset Management
LAN Repository
Software Configuration Management

6.5 DB2 DataBasic

Go back to the top

DB2 DataBasic is an integrated development environment with tools for building and registering DB2 extended stored procedures, applications, and UDFs (user-defined functions) in BASIC. The product also includes a project manager, a code editor, a BASIC interpreter, a debugger, and a set of forms for testing of stored procedures and UDFs. This permits stored procedures to be completely unit tested and debugged prior to implementation.

DB2 DataBasic provides a GUI development environment for Windows 3.1, OS/2, and AIX. Client applications developed in COBOL, C, C++, or Visual Basic can access the DB2 stored procedures and UDFs developed via DataBasic using the DB2 CLI (Call Level Interface), ODBC, DB2 DARI (Distributed Application Remote Interface), or OLE2 via an IBM-supplied VBX (Visual Basic clients only). Server environments supported by DataBasic include OS/2 and AIX. DB2 DataBasic is for those customers that do not wish to use a compiled 3GL for building DB2 stored procedures or UDFs. While this product has competitors, it is distinguished in supporting extended stored procedure build and registration within an interpretive language environment (BASIC).

7. End User Tools

Go back to the top

7.1 Intelligent Decision Server (IDS)

Go back to the top

Intelligent Decision Server (IDS) is a facility for discovering business trends and answering decision support questions using a graphical approach. It consists of a set of OS/2-based end-user tools for access, analysis and presentation of DB2 data. Data from a variety of sources can be correlated and manipulated. Examples include Query Tool, Spreadsheet Tool, and Capsule Tool, all of which employ a common iconic, windows-based user interface. Internet (WWW) access to IDS objects is a key new capability. The product also includes tools for filing, storage, printing, mailing, text editing, and screen capture. IDS, available on OS/2 in third quarter of 1996, evolved from the Data Interpretation System (DIS).

7.2 Approach

Go back to the top

Approach is IBM's end-user database reporting and analysis tool. Approach supports forms creation, query-by-forms, Lotus Notes integration, drill down in charts or crosstabs, and the creation of Notes reports, analyses, and groupware applications. Database integration capabilities include the ability to query, manage, and report on SQL data from a variety of source including DB2, dBASE, Paradox, FoxPro, and any ODBC-compliant database. Approach for Windows 95 contains a Find Assistant for performing sophisticated database searches and a SQL Assistant to help users with SQL development. In addition, queries can be stored and reused. Plug and play network support permits the sharing of files across a network with no additional set up. Approach reports, forms, worksheets, mailings, and crosstabs can be distributed via any VIM- or MAPI-compliant mail system. OLE 2.0 is supported and OLE custom controls (OCX) can be added to the tools palette. Internet Web sites can be stored and categorized, and Internet functions such as extracting and charting information from Internet pages can be programmed.

8. Conclusions

Go back to the top

IBM's DB2 product dominates the MVS relational DBMS marketplace, along with its aging hierarchical DBMS, IMS. With the introduction of DB2 Common Server, it will make its mark on the open systems relational DBMS marketplace as well. DB2 Common Server is a powerful and flexible product that should serve users well.

DB2's key strengths lie in DBMS architectural capabilities such as high performance, strong backup and recovery capabilities, in its security and integrity features, and in its enhanced SQL manipulative capabilities. Its fidelity to the relational model is comparable to other products. Its relational optimizer is strong and has been significantly enhanced with this release of Version 2. DB2 Parallel Edition is a strong entry in the parallel database server market, especially for complex query and intensive query applications. Although it currently lacks some of the state-of-the-art features introduced by DB2 Common Server Version 2, IBM plans to merge the parallelism in DB2 Parallel Edition with the expanded features of DB2 Common Server Version 2 in the near future.

Data integration support through DataPropagator and DataJoiner is strong. Some awareness is needed by users of the difficulties of keeping replicated data sources in synchrony and maintaining global integrity. DataJoiner is also a strong product. IBM has stated an intent to add the state of the art features that were introduced by DB2 Common Server Version 2 to DataJoiner.

DB2 Common Server system management facilities are above average, especially DataHub and Visual Explain. Few DBMS vendors have products to compete in this area.

The DB2 Common Server suite of products is particularly strong with respect to system management and administration. IBM's client/server development tools are expected to support most needs, whether for decision support, OLTP, or data warehouse applications. IBM's end-user tools such as Approach are very powerful and yet demonstrate ease of use.

Alternative Technologies has been providing strategic, design, and development consulting services as well as training in the relational database and client/server areas for twenty years. Standard reports in The Database Product Evaluation Report Series are published semi-annually, custom reports by arrangement, and special topic reports are published from time-to-time. For further information about Alternative Technologies' Database Product Evaluation Reports or consulting services contact:

Alternative Technologies

13150 Highway 9, Suite 123
Boulder Creek, CA 95006
Telephone: 408/338-4621
FAX: 408/338-3113

Report Number 9605