

June 2003



DB2 Information Management Software

Introduction to DB2[®] for Linux in Enterprise and Cluster Environments

*by, Rav Ahuja
DB2 Strategic Technologies
IBM Software Group*

Contents	
1. OVERVIEW	2
2. DB2 FOR LINUX FEATURES AND OFFERINGS	4
2.1 Features	4
2.2 Supported Platforms	5
2.3 DB2 Products and Packages	6
3. DB2 ENVIRONMENT	11
3.1 Deployment Topologies	11
3.2 DB2 Database Objects	12
4. PARALLELISM WITH DB2	15
4.1 SMP Environments	15
4.2 Database Clusters	16
4.3 Partitioned Databases	19
5. SUMMARY	23

1. OVERVIEW

It was not too long ago that Linux servers were being used primarily in academic and scientific domains. In just a few short years Linux has earned the designation of being the fastest growing server operating platform and is becoming increasingly pervasive in the enterprise. Its openness, flexibility and ability to lower cost of ownership are just some of the factors that have contributed to this operating system's phenomenal success in the commercial arena of e-business systems. Once relegated to running infrastructure tasks such as file and Web serving, Linux has now found its way into the enterprise datacenter. Many companies have moved beyond the initial phases of experimentation and testing and are now reaping the benefits of deploying mission-critical applications and databases with Linux and DB2.

IBM® DB2 Universal Database™ has long been known for its technology leadership. Therefore it was not surprising when IBM took the lead in bringing the proven performance, scalability, and ease-of-use features of DB2 to Linux. Over the years DB2 has kept up its lead by releasing the first database for clustered environments on Linux, showcasing the first commercial database for Intel® and AMD powered 64-bit platforms, and continuously being first to announce industry leading benchmarks on Linux. IBM's commitment to Linux is further reflected through its ongoing efforts to exploit and enhance the Linux kernel for database workloads.

As a result, IBM is the market leader on Linux for relational database systems¹. The reasons why major companies and governments around the globe choose to deploy DB2 for Linux in the enterprise setting are quite simple. The DB2 system's rich set of features have been running on Linux for almost four years and during this time, while the Linux kernel matured through the efforts of thousands of programmers and volunteers, the IBM teams were busy further hardening the kernel and DB2 on Linux for enterprise workloads.

¹ Based on: The RDBMS Top 10: License Sales Analysis and Market Forecast, 2001–2006
Analyst: Carl W. Olofson, IDC Research, Oct 2002

Highlights

The combination of DB2 and Linux is a robust database platform for a variety of solutions and vertical applications

Today DB2 is the most versatile and powerful information management system on Linux. DB2 for Linux is capable of effectively handling terabytes of data in both decision support and transactional environments. The combination of DB2 and Linux is a robust database platform for a variety of solutions and vertical applications, including:

- *Backend for Web and application servers*
- *Business intelligence and data warehousing*
- *Transactional enterprise systems*
- *Enterprise applications such as ERP, CRM, SCM*
- *Information integration and content management*
- *Gateway to mainframe and host data*
- *Financial, retail, public sector and manufacturing applications*
- *Life sciences and bio-informatics solutions*
- *Store for spatial and geographical systems*
- *High-performance computing applications including:*
 - *Financial modeling*
 - *Oil and gas exploration*
 - *Research and scientific*

Not only is DB2 known for lower initial cost , it also provides the lowest long-term total cost of ownership

Despite implementing a wide array of solutions in different industries, customers of DB2 for Linux generally talk about a few common themes regarding the benefits they derive. Foremost among them is the exceptional value that DB2 for Linux delivers. Not only is DB2 known for lower initial cost, it also provides the lowest long-term total cost of ownership (TCO) when compared to other databases from Microsoft® and Oracle². Running DB2 on the open Linux platform and Intel or AMD processor-based servers or blades delivers an even more compelling price/performance story. DB2 is renowned for critical self-tuning and self-healing features. The Self Managing and Resource Tuning (SMART) technologies make DB2 easy to use and maintain while minimizing administration costs. IBM's alliances with all the major Linux distributors and the ability to get 24 x 7 support for both DB2 and Linux directly from IBM provides added piece of mind.

² For details refer to <http://www.ibm.com/software/data/highlights/db2tcoreports.html>

Highlights

The following sections will provide an overview of the features and product offerings available on DB2 for Linux as well as the different architectures for deploying the product in the enterprise.

2. DB2 FOR LINUX FEATURES AND OFFERINGS

2.1 Features

DB2 UDB is ... strong enough to meet the demands of large corporations and flexible enough to serve medium-sized and small businesses

DB2 Universal Database is an open-standards, multi-platform, relational database system that is strong enough to meet the demands of large corporations and flexible enough to serve medium-sized and small businesses. Its features include:

- *Exploitation of SMP and cluster-based parallelism*
- *Federated support for a variety of data sources*
- *Advanced built-in OLAP and business intelligence functions*
- *Ability to manage multiple data types including Binary Large Objects*
- *Built on open industry standards: SQL, DRDA, CLI, ODBC, JDBC and more*
- *Leader in query optimization technology and performance*
- *Stored procedures, UDFs, data encryption and globalization*
- *Fully web-enabled and certified for leading enterprise applications*
- *High Availability features and Failover support*
- *J2EE certified and Java application interfaces for JDBC and SQLJ*
- *Support for open source interfaces such as PHP, Python and Perl*
- *DB2 XML Extender for storage and retrieval of XML documents*
- *High-speed, in-memory search capability with DB2 Net Search Extender*
- *A graphical, integrated toolset for managing local and remote databases*
- *Productivity tools for visual development and migration*
- *Support for the InfiniBand I/O high-speed interconnect*
- *Exploitation of the latest Linux kernel capabilities*

Highlights

DB2 has been commercially available on Linux since Version 6. However, most deployments today run on Version 7.2 or later. In Nov 2002, IBM released DB2 UDB Version 8.1, marking the next stage in the evolution of relational databases. This version introduced over 400 new features³, including:

- *Innovative self-managing technologies to reduce DBA intervention*
- *New tools and wizards such as Health Center, Memory Visualizer*
- *Simplified management of large-scale databases on clusters*
- *Federated Web Services for new levels of Information Integration*
- *Enhanced XML support and productivity tools*
- *Connection Concentrator for more user scalability*
- *Dynamic configuration (without the need for restarting database)*
- *Online tools and utilities such as reorg, load, storage and memory management*
- *Multidimensional clustering for improved performance of complex queries*
- *Null and default compression*

2.2 Supported Platforms

IBM is committed to Linux and supports it on a variety of hardware platforms

IBM is committed to Linux and supports it on a variety of hardware platforms. Besides the Intel 32-bit hardware, DB2 for Linux today runs on 64-bit Intel Itanium 2 based systems, PowerPC based IBM eServer iSeries™ and pSeries™ Linux platforms and IBM eServer zSeries® mainframe systems. IBM has also announced support for DB2 on 64-bit AMD Opteron powered systems.

On the 32-bit x86 platform, DB2 requires systems equipped with Intel Pentium, Intel Xeon, AMD Athlon or later processors. DB2 V7.2 is supported on any Linux distribution with 2.4 kernel, glibc 2.1.2 and libstdc++ 2.9.0 or later. With DB2 V8, the Linux Validation Program has been introduced to ensure better interoperability of DB2 on Linux platforms to enhance customer experience. IBM is working closely with the Linux community, Linux distribution vendors, and Independent Software and Hardware Vendors to test DB2 using various Linux

³ A more complete overview about the feature set of DB2 Universal Database Version 8 can be found at <http://ibm.com/db2/v8>

Highlights

kernels, distributions, third-party products, hardware and other components that interact with DB2. As a part of the validation program, before a product is supported with DB2, it undergoes rigorous testing, providing users the confidence to run DB2 with other products right out of the box.

The Linux distributions that have been successfully validated for use with DB2 include Red Hat Linux Professional 7.2, 7.3, 8.0, Advanced Server 2.1, SuSE Professional 7.3, 8.0, 8.1, SuSE Linux Enterprise Server 7, 8, SCO Linux 4.0, Turbolinux 7 Server, Turbolinux 8 Server, Turbolinux Enterprise Server 8.0, and other distributions powered by United Linux 1.0 with kernel and library levels specified at the following validation Web site: ibm.com/db2/linux/validate

To run Java applications and the graphical tools that come with DB2 (e.g., Control Center) IBM JDK 1.3.1 is required. For convenience it is shipped on DB2 v8.1 CDs, and in most cases is automatically installed with DB2 installation. Refer to the validation Web site for the most up-to-date list of supported products as the site is updated frequently.

2.3 Products and Packages

DB2 offerings on Linux span the spectrum, ranging all the way from products on handhelds to large clusters and mainframes. These products are discussed further under several categories:

- *DB2 packages for the production environment*
- *Products for accessing legacy and host data*
- *Additional DB2 features*
- *DB2 packages for Application Developers*
- *DB2 for pervasive platforms*

DB2 offerings on Linux span the spectrum, ranging all the way from products on handhelds to large clusters and mainframes

Highlights

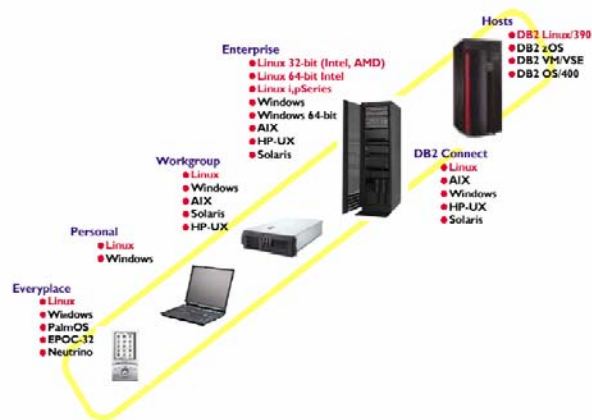


Figure 1: DB2 for Linux – from palmtop to teraflop

DB2 packages for the production environment

Personal Edition can be remotely managed and be used for managing other database servers on the network

DB2 Personal Edition (PE): PE provides a single user database engine ideal for deployment on desktops of PC-based users. PE can be remotely managed and be used for managing other database servers on the network, making it the perfect choice for deployment in occasionally connected or remote office implementations that don't require multi-user capability.

DB2 Workgroup Server Edition (WSE): WSE is the database server designed for deployment in a departmental or small business environment that involves a small number of internal users. WSE has a user-based licensing model designed to provide an attractive price point for smaller installations while still providing a full-function database server. WSE can be deployed on systems with up to four CPUs.

DB2 Workgroup Server Unlimited Edition (WSUE): WSUE offers a simplified per-processor licensing model for deployment in a departmental or small business environment that has Internet users or number of users that makes per processor licensing more attractive than the WSE licensing model. The WSUE is also for use on systems with up to four CPUs.

Highlights

Enterprise Server Edition with DPF allows customers to partition a database within a single system or across a cluster of systems

DB2 Enterprise Server Edition (ESE): ESE meets the database server needs of midsize to large businesses. ESE is the ideal foundation for building data warehouses, transaction processing, or Web-based solutions as well as a backend for packaged solutions like ERP, CRM, and SCM. Additionally, ESE offers connectivity and integration for other enterprise DB2 and Informix data sources.

In versions of DB2 prior to V8.1, the product was available in two enterprise packages: DB2 Enterprise Edition (EE) and DB2 Enterprise-Extended Edition (EEE), with the latter providing the ability to create partitioned databases or run on a cluster. In V8.1, EE and EEE capabilities have been merged into a single ESE product, but running a partitioned database requires DPF licenses.

DB2 Database Partitioning Feature (DPF): The Database Partitioning Feature is a licensing option that allows ESE customers to partition a database within a single system or across a cluster of systems. The DPF capability provides the customer with multiple benefits including scalability to support very large databases or complex workloads, and increased parallelism for administration tasks.

Products for accessing legacy and host data

DB2 Connect™ Personal Edition: DB2 Connect Personal Edition provides the application programming interface (API) drivers and connectivity infrastructure to enable direct connectivity from desktop applications to mainframe (zSeries, S/390®, VM/VSE) and iSeries (AS/400®) database servers. This product is specifically designed and is licensed for enabling two-tier client-server applications running on individual workstations and as such is not appropriate for use on servers.

DB2 Connect Enterprise Edition: DB2 Connect Enterprise Edition addresses the needs of organizations that require robust connectivity from a variety of desktop systems to mainframe and iSeries database servers. DB2 client software is deployed on desktop systems and provides API drivers that connect client-server applications running on these desktop systems to a DB2 Connect server (gateway) that accesses host data. The licensing model for this product is user based.

Highlights

DB2 Connect Application Server Edition: DB2 Connect Application Server Edition product is identical to the DB2 Connect Enterprise Server in its technology. However, its licensing terms and conditions are meant to address specific needs of multi-tier client-server applications as well as applications that utilize Web technologies. DB2 Connect Application Server Edition license charges are based on the number of processors available to the application servers where the application is running.

DB2 Connect Unlimited Edition: DB2 Connect Unlimited Edition product is ideal for organizations with extensive usage of DB2 Connect, especially where multiple applications are involved. This product provides program code of the DB2 Connect Personal Edition as well as program code identical to the DB2 Connect Application Server Edition for unlimited deployment throughout an organization.

Additional DB2 features

DB2 Spatial Extender: This product enables users to leverage the power of the relational database model for managing their location-based data just as easily as traditional business data, and use industry-standard SQL for spatial data analysis, which can add competitive advantages to existing and new applications. The DB2 Spatial Extender is available free of charge in the DB2 PE and DB2 WSE for up to five users. It is also available as a separate program for DB2 WSUE and DB2 ESE.

DB2 Net Search Extender... powerful in-memory search capability for text-based data

DB2 Net Search Extender: This product enables a solution to include a powerful in-memory search capability for text-based data. The integration of these advanced searching capabilities directly into the database saves the time and expense of integrating third-party, text-based solutions while providing the speed and flexibility in text searching demanded by e-commerce applications. The DB2 Net Search Extender is available free of charge in the DB2 PE and DB2 WSE for up to five users. It is also available as a separate program for DB2 WSUE and DB2 ESE.

Highlights

DB2 Intelligent Miner products provides the analysis capabilities needed to make insightful business decisions

DB2 Intelligent Miner™ Products: These products provide the analysis capabilities needed to make insightful business decisions. DB2 Intelligent Miner Scoring extends DB2 analysis capabilities and enables users to deploy mining in real-time applications with a simple SQL call. DB2 Intelligent Miner Modeling delivers DB2 Extenders™ (specifically stored procedures) for modeling operations like associations discovery, demographic clustering, and tree classification. DB2 Intelligent Miner Visualization provides Java™ visualizers, to interact and graphically present the results of modeling operations.

DB2 packages for Application Developers

DB2 Personal Developer's Edition (PDE): PDE enables a developer to design and build single user desktop applications. This offering comes with Linux and Windows versions of the DB2 PE products as well as the DB2 Extenders.

DB2 Universal Developers Edition (UDE): UDE offers a low-cost package for a single application developer to design, build, or prototype applications for deployment of any of the DB2 client or server platforms. It includes client and server DB2 editions for all Linux, UNIX® and Windows® supported platforms, DB2 Connect, DB2 Extenders, and Intelligent Miner. The software in this package cannot be used for production systems.

DB2 for pervasive platforms

DB2 Everyplace®: DB2 Everyplace is a relational database and enterprise synchronization server that enables enterprise applications and enterprise data to be extended to mobile devices such as personal digital assistants (PDAs) and smart phones. DB2 Everyplace can also be embedded into devices and appliances to increase their functionality and market appeal. The product can be used as a local independent database on a mobile device, or query information on remote servers when a connection is available.

Highlights

3. DB2 ENVIRONMENT

3.1 Deployment Topologies

DB2 for Linux can be used with a wide range of applications, whether developed in-house or pre-packaged. The applications can be deployed with DB2 using a number of configurations:

Single-tier: In this configuration the application and the database reside on the same system. In enterprise environments, it may be rare to see such a configuration since remote access to a database server is typically required. Nonetheless this is quite common for developing applications that can later be deployed transparently in a multi-tier DB2 environment without any changes.

Client/Server or 2-tier: The application and the database reside on separate systems. The machines where the application runs typically have a DB2 client installed, which communicates over the network to a database server. For the application, the physical location of the data is transparent. The application communicates with the DB2 client using a standard interface (e.g., ODBC) and the DB2 client takes over the task of accessing the data over the network. In some cases, such as browser- or Java-based access, it is not necessary to have the DB2 client running on the same machine where the application executes.

DB2 provides exceptional flexibility for mixing and matching client and server platforms in a heterogeneous environment. DB2 client and server code is available for a wide variety of platforms. For example, the application can execute on a Windows-based machine with a DB2 client for Windows, which can then access a DB2 database on a Linux server. Likewise the Linux machine can act as a client and access data from UNIX servers or mainframes.

Multi-tier: In a multi-tier configuration, the application, the DB2 client and the data source typically reside on separate systems. Examples of such configurations include, but are not limited to, scenarios illustrated in the following table.

DB2 client and server code is available for a wide variety of platforms

Highlights

Client	Middle-Tier	Server
Web browser	Web Server DB2 Client	DB2 Database Server
Application Client	Application Server DB2 Client	DB2 Database Server 1 DB2 Database Server 2
Application DB2 Client	DB2 Connect gateway	OS/390, AS/400, VM/VSE
Application DB2 Client	DB2 Server	Secondary Data Sources (e.g., Mainframe, DB2 Non-DB2, non-relational)

Figure 2: Multi-tier configuration examples

IBM offers the most comprehensive business integration solution by allowing federated access to a variety of distributed data sources

IBM recognizes that in many cases there may be a need for accessing data from a variety of distributed data sources rather than one centralized database. The data sources can be from IBM products such as DB2 or Informix®, or non-IBM databases like Oracle, or even non-relational data such as files or spreadsheets. As illustrated in the last scenario in the figure above, IBM offers the most comprehensive business integration solution by allowing federated access to a variety of distributed data sources.

3.2 DB2 Database Objects

Instances

An instance (sometimes called a database manager) is DB2 code that manages data. It controls what can be done to the data, and manages system resources assigned to it. Each instance is a complete environment. It contains all the database partitions defined for a given parallel database system. An instance has its own databases (which other instances cannot access directly), and all its database partitions share the same system directories. It also has separate security from other instances on the same machine (system), allowing, for example, both production and development environments to be run on the same machine under separate DB2 instances without interfering with each other.

Highlights

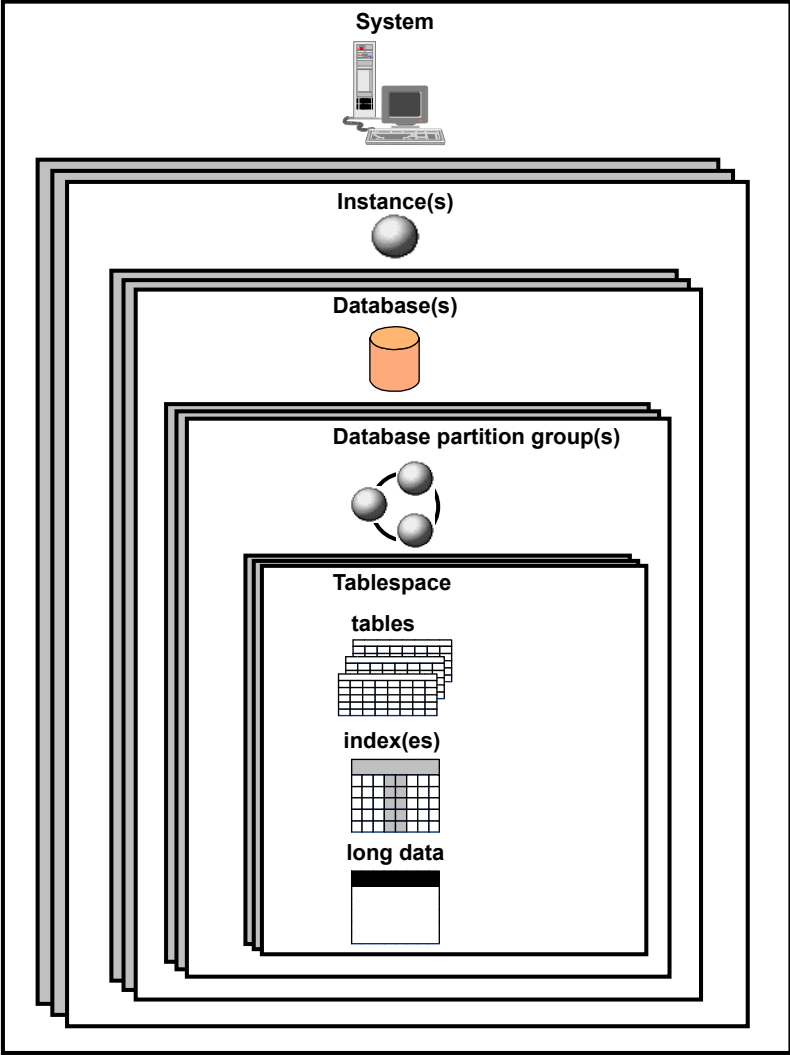


Figure 3: DB2 database objects

Databases

A relational database presents data as a collection of tables. A table consists of a defined number of columns and any number of rows. Each database includes a set of system catalog tables that describe the logical and physical structure of the data, a configuration file containing the parameter values allocated for the database, and a recovery log.

A relational database presents data as a collection of tables

Highlights

Database partition groups

A database partition group is a set of one or more database partitions. Before creating tables for the database, you first need to create the database partition group where the table space will be stored, and then create the table space where the tables will be stored. If a partition group is not specified, there is a default group where table spaces are allocated. In earlier versions of DB2, database partition groups were known as nodegroups. In a non-partitioned environment, all the data resides in a single partition, therefore it is not necessary to worry about partition groups.

Table spaces

A database is organized into parts called table spaces. A table space is a place to store tables. When creating a table, you can decide to have certain objects such as indexes and large object (LOB) data kept separately from the rest of the table data. A table space can also be spread over one or more physical storage devices.

When creating a table, you can decide to have certain objects kept separately from the rest of the table data

Table spaces reside in database partition groups. Table space definitions and attributes are recorded in the database system catalog. Containers are assigned to table spaces. A container is an allocation of physical storage (such as a file or a device). A table space can be either system managed space (SMS), or database managed space (DMS). For an SMS table space, each container is a directory in the file space of the operating system, and the operating system's file manager controls the storage space. For a DMS table space, each container is either a fixed-size preallocated file or a physical device such as a disk, and the database manager controls the storage space.

Tables

A relational database presents data as a collection of tables. A table consists of data logically arranged in columns and rows. All database and table data is assigned to table spaces. The data in the table is logically related, and relationships can be defined between tables. Data can be viewed and manipulated based on mathematical principles and operations called relations. Table data is accessed through Structured Query Language (SQL), a standardized language for defining and manipulating data in a relational database. A query is used in

Highlights

applications or by users to retrieve data from a database. The query uses SQL to create a statement in the form of *SELECT <data_name> FROM <table_name>*

Bufferpools

A bufferpool is the amount of main memory allocated to cache table and index data pages as they are being read from disk or being modified. The purpose of the buffer pool is to improve system performance. Data can be accessed much faster from memory than from disk; therefore, the fewer times the database manager needs to read from or write to a disk (I/O), the better the performance. (You can create more than one buffer pool, although for most situations only one is sufficient.) The configuration of the buffer pool is the single most important tuning area, because you can reduce the delay caused by slow I/O.

4. PARALLELISM WITH DB2

A system with a single processor and disk has limitations for the amount of data, users, and applications that it can handle. Parallel processing, which is key for enterprise workloads, involves spreading the load across several processors and disks, and allows for large volumes of data and high transaction rates to be processed. In many environments, the data volumes are growing at a phenomenal rate, therefore a database management system needs the ability to be easily scale to increased loads with the addition of more disks and CPUs.

Parallel processing, which is key for enterprise workloads, allows for large volumes of data and high transaction rates to be processed

The parallel technology of DB2 enables highly scalable performance on large databases by breaking the processing into separate execution components that can be run concurrently on multiple processors. Elapsed times for queries can be dramatically reduced by processing the individual queries in parallel. DB2 supports a scalable growth path to easily add more processing power to an existing system by “scaling up” (SMP) “scaling out” (MPP) or both.

Highlights

4.1 SMP Environments

In the past, complex enterprise workloads have typically run on high-end SMP (symmetric multi-processor) machines. A large number of processors running within the same machine and connected with a high bandwidth bus can deliver excellent performance. Since DB2 for Linux shares over 99% of its code with DB2 running on high-end UNIX systems, it inherits the ability to exploit SMP and NUMA architectures. The ability of DB2 to run on SMP systems is only restricted by today's Linux SMP scalability. The unprecedented results of a recent SAP benchmark⁴ on a 8-way Linux system are a testament to the effectiveness of DB2 for Linux in an SMP environment. A large number of developers at the IBM Linux Technology Center are engaged with the Linux community to extend the capabilities of Linux on 16-way and higher SMP machines.

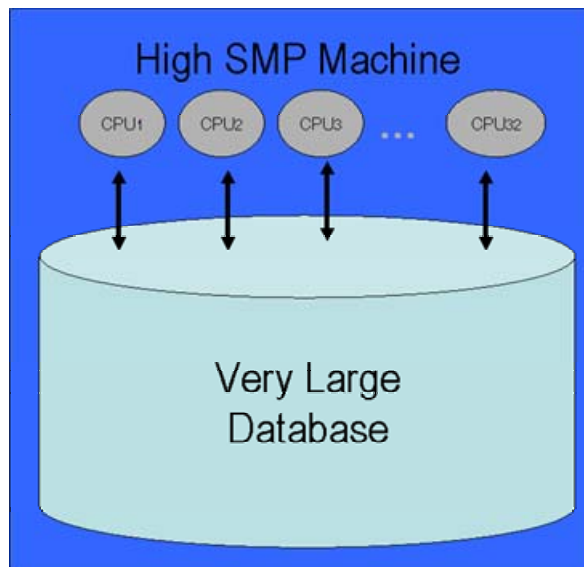


Figure 4: Database on a large SMP system

The driving force behind using Linux clusters is that by distributing the load over several low-cost servers running an open-source operating system, a larger task can be accomplished faster, more reliably, and a lot more economically

4.2 Database Clusters

The driving force behind using Linux clusters is that by distributing the load over several low-cost servers running an open-source operating system, a larger task can be accomplished faster, more reliably, and a lot more economically. And if the load increases, the cluster can be

⁴ Refer to <http://www.sap.com/benchmark/pdf/cert5702.pdf>

Highlights

DB2 [ESE] can be deployed across a Linux cluster right out of the box, without the need for additional clustering software.

extended for managing the additional demand without compromising on performance. DB2 was the first commercial database on Linux to provide built-in capabilities for clusters. That is, DB2 can be deployed across a Linux cluster right out of the box, without the need for additional clustering software. DB2 clusters on Linux are ideal for running demanding transactional applications and warehouses that involve large volumes of data while providing the following benefits:

- *Faster processing time*
- *Very large databases*
- *Excellent scalability*
- *Increased availability*
- *Reduced cost*

Combining the clustered approach with support of DB2 for 64-bit Linux environments provides additional performance and scalability advantages.

Database Clustering Architectures: There are two primary database clustering architectures: shared-disk and shared-nothing. Shared-disk is used by Oracle RAC. DB2 for Linux employs shared-nothing.

Shared-disk:

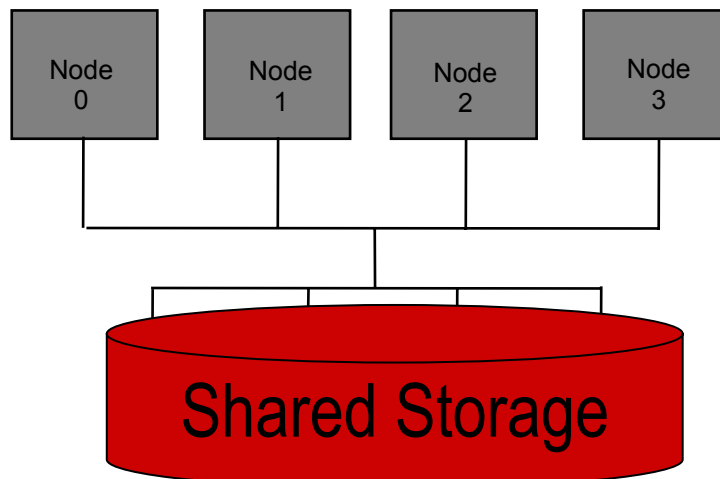


Figure 5: representation of a shared-disk system

Highlights

In a shared-disk environment, all the processors can access all the disks in the cluster

In a shared-disk environment, each database node has its own processors but shares the disks with other nodes. Therefore, all the processors can access all the disks in the cluster. This introduces additional overhead for coordinating resources and locks between nodes. For example, if Node 1 wants to access data on a certain disk that has been locked for update by another node in the cluster, Node 1 must wait for other nodes to complete their operations. While this works well when there are few nodes in the cluster (e.g., mainframe environments), the overhead for distributed lock management and cache coherency issues can severely limit scalability and introduce performance degradation for four or more nodes, making it impractical to exploit economies of large clusters on Linux. Furthermore, this approach involves specialized hardware and software for shared disk and cache management, making it a lot more expensive than shared-nothing.

Each partition has its own private resources, lending itself to virtually unlimited scalability

Shared-nothing: As the name implies, partitions (nodes) in a shared-nothing environment do not share processors, memory or disks with partitions on other machines. Each partition acts on its own subset of data. Since each partition has its own private resources, this approach does not involve any resource contention with other servers, lending itself to virtually unlimited scalability. This is one reason DB2 for Linux can support up to 1000 partitions. And since there is no coordination overhead for accessing resources, additional machines can easily be added to the cluster with linearly scalable performance, which implies, if doubling of the data volume is matched by the doubling of the cluster resources, the high performance of the database will be maintained at the same level. If one partition in a cluster fails, its resources can dynamically be transferred to another machine, ensuring high availability. Another benefit of the shared-nothing approach used by DB2 for Linux is that it does not require specialized hardware, making the solution a lot simpler, less expensive, and suitable for Linux-based “commodity” hardware.

Highlights

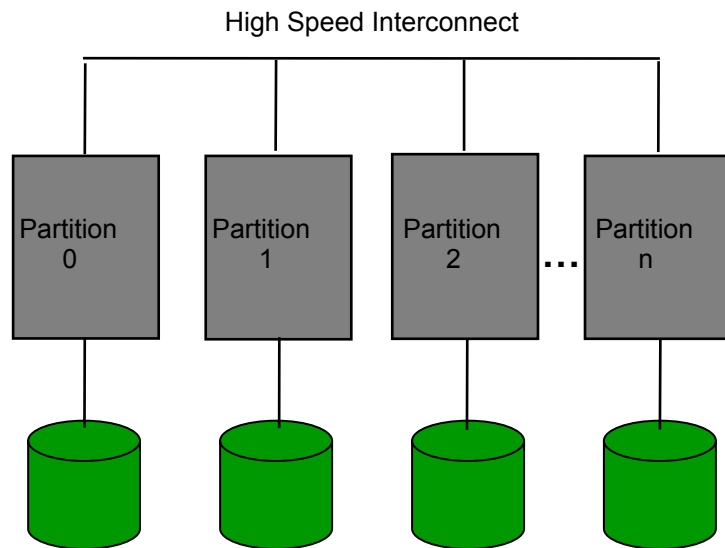


Figure 6: Representation of a shared-nothing system

4.3 Partitioned Databases

DB2 exploits the power of Linux clusters by employing database partitioning. In a partitioned environment, a database is distributed across multiple partitions, usually residing on different machines. Each partition is responsible for a portion of a database's total data. A database partition is sometimes also called a node or a database node. Because data is divided across database partitions, you can use the power of multiple processors on multiple physical nodes to satisfy requests for information. Data retrieval and update requests are decomposed automatically into sub-requests, and executed in parallel among the applicable database partitions.

As an illustration of the power of processing in a partitioned database system, assume that you have 100 000 000 records that you want to scan in a single-partition database. This scan would require that a single database manager search 100 000 000 records. Now suppose that these records are spread evenly over 20 database partitions; each partition only has to scan 5 000 000 records. If each database partition

You can use the power of multiple processors on multiple physical nodes to satisfy requests for information

Highlights

Any database partition can be used as a coordinator node.

server scans in parallel with the same speed, the time required to do the scan should be approximately 20 times faster than a single-partition system handling the entire task.

The fact that databases are partitioned across several database partitions is transparent to users and applications. User interaction occurs through one database partition, known as the coordinator node for that user. Any database partition can be used as a coordinator node. The database partition that a client or application connects to become the coordinator node. You should consider spreading out users across database partition servers to distribute the coordinator function.

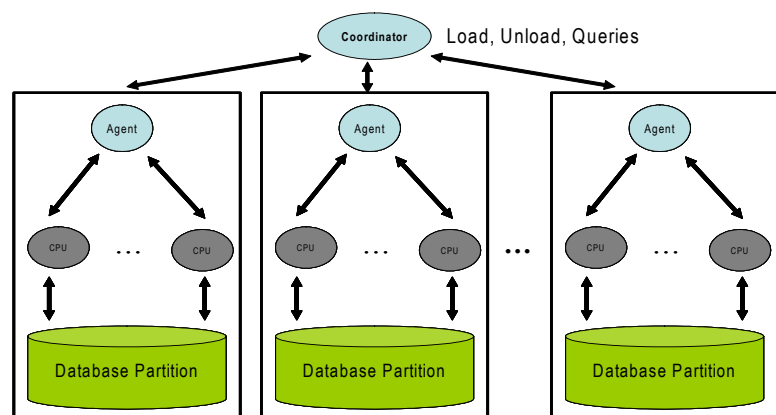


Figure 7: Coordination in a partitioned database

The data is distributed among database partitions using an updatable partitioning map and a hashing algorithm, which determine the placement and retrieval of each row of data. For example, if a row is being added to a table, the coordinator node checks a partitioning map, which specifies the database partition where the row is to be stored. The row is only sent to that database partition server, with the result that only the interested database partition servers take part in the insert. This keeps communications and coordination overhead between nodes as low as possible.

Highlights

INSERT ... (000120, "Joe Manager", "Reg")

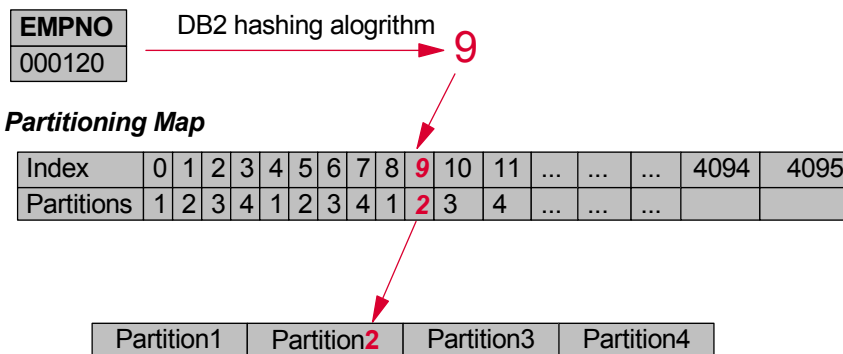


Figure 8 : An example of intelligent data distribution across DB2 partitions

The data, while physically split, is used and managed as a logical whole. Users can choose how to partition their data by declaring partitioning keys. Tables can be located in one or more database partitions. As a result, you can spread the workload across a partitioned database for large tables, while allowing smaller tables to be stored on one or more database partitions. Each database partition has local indexes on the data it stores, resulting in increased performance for local data access.

You are not restricted to having all tables divided across all database partitions in the database. DB2 supports partial declustering, which means that you can divide tables and their table spaces across a subset of database partitions in the system. An alternative to consider when you want tables to be positioned on each database partition is to use materialized query tables and then replicate those tables. You can create a materialized query table containing the information that you need, and then replicate it to each node.

When using SMP machines with several processors, it is possible to create more than one partition on the same system.

Partitioning on SMP clusters: In the most simplistic scenario, each physical machine in the cluster has a single database partition on it. In this type of a configuration, each partition is a *physical database partition* and has access to all of the resources in the machine. One partition per machine is typical for systems having 1 or 2 processors. When using SMP machines with several processors, it is possible to create more than one partition on the same system. These are called *logical*

Highlights

database partitions. A logical database partition differs from a physical partition in that it is not given control of an entire machine. Although the machine has shared resources, database partitions do not share all resources. Processors are shared but disks and memory are not.

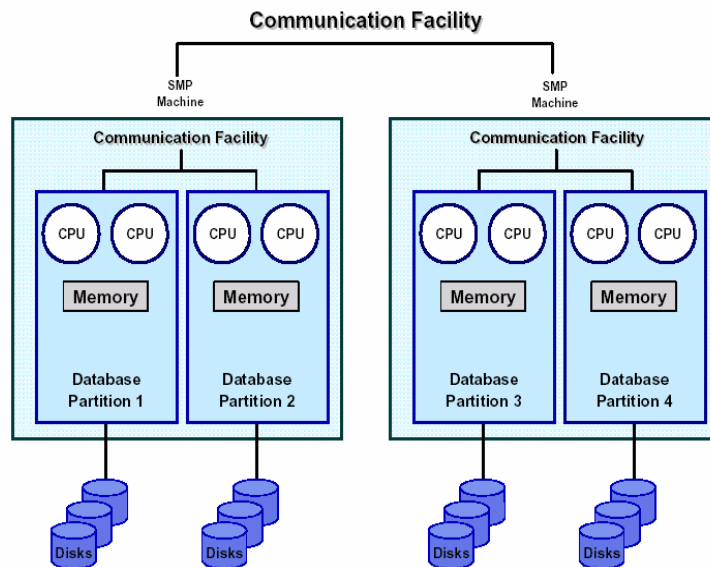


Figure 9: A partitioned database across a cluster of SMP machines

Logical database partitions can provide better scalability.

There are advantages to using logical partitions on high-SMP systems in both clustered and non-clustered environments. Logical database partitions can provide better scalability. Multiple database managers running on multiple logical partitions may make fuller use of available resources than a single database manager could. The ability to have two or more partitions coexist on the same machine (regardless of the number of processors) allows greater flexibility in designing high availability configurations and failover strategies. Upon machine failure, a database partition can be automatically moved and restarted on a second machine that already contains another partition of the same database.

SUMMARY

With a wide spectrum of offerings and capabilities, DB2 for Linux allows you to access, manage, and analyze all forms of information across the enterprise. DB2 gives you a robust, easy-to-manage database that offers high performance to complement the stability and reliability of Linux. Industry-leading parallel technologies in DB2 make it the most scalable and powerful database in production today, and when combined with Linux clusters, it allows you to manage mission-critical data at a cost lower than any other enterprise class database.

To test drive DB2 for Linux today, download the Version 8.1 Enterprise Server Edition of the database from: ibm.com/db2/v8

For further information visit: ibm.com/db2/linux

Or email: DB2forLinux@ca.ibm.com



© Copyright IBM Corporation 2002
IBM Canada Ltd.
8200 Warden Avenue
Markham, ON
L6G 1C7
Canada

Printed in United States of America
05-03
All Rights Reserved.

Neither this documentation nor any part of it may be copied or reproduced in any form or by any means or translated into another language, without the prior consent of the IBM Corporation.

IBM makes no warranties or representations with respect to the content hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. IBM assumes no responsibility for any errors that may appear in this document. The information contained in this document is subject to change without any notice. IBM reserves the right to make any such changes without obligation to notify any person of such revision or changes. IBM makes no commitment to keep the information contained herein up to date.

The information in this document concerning non-IBM products was obtained from the supplier(s) of those products. IBM has not tested such products and cannot confirm the accuracy of the performance, compatibility or any other claims related to non-IBM products. Questions about the capabilities of non-IBM products should be addressed to the supplier(s) of those.

AIX, AS/400, DB2, DB2 Connect, DB2 Extenders, DB2 Universal Database, Everyplace, Informix, Intelligent Miner, iSeries, OS/400, PowerPC, S/390, VM/ESA, VSE/ESA, zSeries, zOS, the e-business logo, IBM, and the IBM logo are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft and Windows are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product and 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.