This tutorial introduces you to the basics of the IBM® DB2® 10.1 product editions, functionalities and tools, along with underlying concepts that describe different types of data applications such as OLTP, data warehousing / OLAP, non-relational concepts and more. It will briefly introduce you to many of the concepts you'll see in the other tutorials in this series, helping you to prepare for the DB2 10.1 Fundamentals certification test 610.

View more content in this series

Before you start

About this series

So you want to become an IBM Certified Database Associate? If so, you just found the right place. In this series of six DB2 certification tutorials, you'll find the basics on topics you'll need to understand to take the test 610 (DB2 Fundamentals 10.1). Although its main purpose is to help you to take the test, this series can also be used to discover some of the new exciting features of DB2 10.1 and help you learn about many of the features and functions available in DB2 10 for z/OS® and DB2 10 for Linux®, UNIX®, and Windows®.

You don't see the tutorial you're looking for yet? You can review the DB2 9 tutorials in the DB2 9 Fundamentals certification 730 prep series.

About this tutorial

This tutorial introduces you to the basics of the DB2 10.1 product editions, functionalities and tools, along with underlying concepts that describe different types of data applications such as OLTP, data warehousing / OLAP, non-relational concepts, and more. It briefly introduces many of the concepts you'll see in the other tutorials in this series, helping you to prepare for the DB2 10.1 Fundamentals test 610.

Objectives

At the end of this tutorial, you should be able to understand:
• What the different editions of DB2 and the various DB2 products are
• Which tools are included with DB2 10.1
• How to use Data Studio to manage your DB2 environment, access your data, and more
• What pureScale is and how it can help you get the most from your OLTP databases
• What data warehousing is and what DB2 products are available to leverage its success
• How DB2 stores and deals with non-conventional data such as large objects (LOBs) and XML documents

Prerequisites

As this is the first tutorial in the series, you are not expected to meet any prerequisites. But to benefit more from the tutorial, it’s recommended that you have some background in database administration and utilization. Having access to a DB2 server would also help you test many of the concepts you’ll see and enable you to practice the tasks presented throughout the series. Product installation is not covered here, but you can find help on that if you need it in the documentation.

System requirements

As mentioned in the prerequisites above, it is strongly recommended that you install DB2 to take better advantage of this tutorial, as well as the others in the series. As you’ll see soon, DB2 has a FREE, fully functional edition available that can be downloaded: DB2 Express-C. You can use it on your own computer without concerns about licensing, as it will certainly meet your learning needs. You can download DB2 Express-C through developerWorks.

Acknowledgements

Although the material in this tutorial covers much of what you’ll see in the exam, it was created based on the exam objectives, not on its questions. Therefore, it’s possible that when taking the exam, you’ll find questions that are not explicitly covered here. The best way to get prepared for the exam is to have experience with the topics listed in the exam objectives. This tutorial contains some content from its previous edition, “DB2 Fundamentals v9 preparation – Planning” (where the information has not changed), written by Paul Zikopoulos.

DB2 products

Stating that DB2 10.1 delivers the right data management solutions for any business is not a marketing speech. No other database management system can match the advanced performance, availability, scalability, and manageability features that are found in DB2 10.1. However, there are different editions of DB2 available, each suited to a different part of the marketplace. On the Fundamentals exam you are expected to understand the different DB2 products and editions, covered in this section.

Within IBM Information Management software, there are essentially two flavors of DB2: DB2 for z/OS and DB2 for Linux, UNIX, and Windows (sometimes referred to as DB2 for LUW or DB2 for distributed platforms). All the distributed editions of DB2 that are currently available are shown in Figure 1. If you examine this figure closely, you will see a progression — each edition displayed
includes all the functions, features, and benefits of the editions found below it (along with additional features and functionality) as you move up the stack.

**Figure 1. The different editions of DB2 for distributed platforms that are available**

It’s important to note that if you decide to move from one product edition to another, there’s no need to worry about incompatibilities due to product migration. For example, if you created a database using DB2 Express-C, and later decided to purchase DB2 Enterprise Server Edition, you can keep using the same server by simply upgrading DB2 using the DB2 Enterprise Server Edition (ESE) installation image. The end result will be a functioning database environment that now has many more features available. The only concern is whether or not features you desire are available in the target version. For instance, if you move from DB2 ESE to DB2 Express, you would no longer be able to use partitioned tables, MDC, storage optimization, multi-temperature storage and any other feature that is not present with DB2 Express.

Even if you want to move to another product such as InfoSphere® Warehouse (which you’ll see more about later), no changes to your existing databases are needed.

### DB2 LUW or DB2 UDB?

You saw the term LUW, and you know it stands for Linux, UNIX, and Windows. At one time, DB2 for Linux, UNIX, and Windows was called DB2 UDB (Universal Database). The term UDB is not in use anymore (since DB2 9), and you should avoid using it. If you find a paper referring to DB2 UDB, it’s probably referring to version 8 or earlier.

Across the Linux, UNIX, and Windows platforms, the DB2 code is about 90% common, with 10% of the code on each operating system reserved for tight integration into the underlying operating system (such as using huge pages on AIX® or the NTFS file system on Windows).

Throughout this tutorial, you will see several references to DB2 for z/OS and DB2 for IBM i. These are part of the DB2 family of products as well, however, they run on higher platforms (the
mainframe). While DB2 for z/OS and DB2 for IBM i databases run on specific hardware, operating system and platforms, their SQL is 95% portable to DB2 for Linux, UNIX, and Windows.

DB2 for Linux on System z® (also known as zLinux) should not be confused with DB2 for z/OS. In this case, the DB2 product that runs on zLinux is DB2 for Linux, UNIX, and Windows — and any DB2 Client or driver is able to connect to it, without need of DB2 Connect (which will be covered later).

Now let’s find out more about the editions of DB2 for Linux, UNIX, and Windows that are available.

**DB2 for Linux, UNIX, and Windows 10.1 Editions**

Different product editions of DB2 for Linux, UNIX, and Windows enable users to choose the specific flavor of DB2 that best fits their needs. The remainder of this section describes each product edition available, and provides a brief overview of the features and functionality that is offered with each product edition.

**DB2 Express-C**

As the entry-level edition, DB2 Express-C (also known as “DB2 Express-Community edition”) gives you all the core DB2 10.1 capabilities at no charge. Designed to be up and running in minutes, it includes self-management features as well as some of the functionality that was only found in paid editions in the past. Yes, it's FREE to develop, use and distribute!

Features included:

- **Self Tuning Memory Manager (STMM)**
  - STMM auto-configures several memory configuration parameters, simplifying the memory-management task.

- **pureXML**
  - Through pureXML, DB2 stores XML documents in their native format and enables the creation of indexes on XML columns (resulting in faster search and retrieval of data), query through XQuery and SQL over XML data, and much more.

- **Backup and archived logs compression**
  - Backups as well as archived logs are known as huge disk space consumers. Backups can be compressed by just using the COMPRESS clause with the DB2 backup utility, and log archive compression can be enabled through a database configuration (set LOGARCHCOMPR1 to ON). Archived log compression was introduced in DB2 10.1.

- **Oracle compatibility**
  - Since version 9.7, one of the great new features added to DB2 is its compatibility with Oracle databases (also called Oracle enablement). With this feature, you can use Oracle's data types, PL/SQL, functions and so forth, with a DB2 database. DB2 for Linux, UNIX, and Windows is 98% compatible with Oracle, which means that you would have to concern yourself with just 2% of your application’s code if you wanted to move from Oracle to DB2. That also means that Oracle professionals can now say they can work with DB2!

- **Time Travel Query**
Another new feature of DB2 10.1 makes it possible to issue queries that will find out what your data looked like at a specific date and time (using what are known as temporal tables, which are tables that have been prepared to hold temporal data). As temporal tables enable the use of system and application date/time values, it is possible to issue changes that will happen in the future! Confused? Curious? You can find out more in the third tutorial in this series: "Working with Databases and Database Objects".

Federation with DB2 for Linux, UNIX, and Windows and Informix®
- This feature allows you to access objects in other DB2 for Linux, UNIX, and Windows databases (Homogeneous Federation) as well as in Informix databases.

DB2 Text Search Extender
- With this feature, you can use the CONTAINS clause in your queries for full-text search in tables that have text search-compatible indexes.

Spatial Extender
- This feature makes it possible for you to store, retrieve, search, and manage spatial data that is represented by "geographic features" like a river, a forest and so on.

Resource Description Framework (RDF)
- Also known as NoSQL graphs storage, this new feature introduced in DB2 10.1 enables developers to work with information triples or quads, in huge volumes, and at a high velocity, using the SPARQL query language.

DB2 Express-C 10.1 is available for the following operating systems:

- Windows (32/64bit)
- Linux (on x86 32/64bit and POWER)
- Solaris (on x86-64)

DB2 Express-C is available in more than 16 languages and is much less restrictive than other free entry-level database products. As a free edition, DB2 Express-C limits are only applied to CPU (DB2 Express-C will use up to 2 cores) and memory (it can use up to 4 GB). That means, if your server has 16 cores and 20 GB of RAM, DB2 Express-C 10.1 will work, but it will only use 2 cores of your server's CPU and 4 GB of your RAM. If you were expecting more restrictions for a free/community edition database management product, that's not what happens with DB2 Express-C! Your databases will be able to grow a lot in volume, and you'll be able to have as many connected users as you need.

New releases for DB2 Express-C are made available as major updates are released for other editions. However, upgrades are only possible through the installation of new releases over previous ones, and once a new release is published, links for older releases are removed from the download website.

Should your database need more processing power or memory, or should you need more formal support, access to fix packs (product updates), or additional features like SQL Replication and High Availability Disaster Recovery (HADR), you can migrate Express-C to any of the other DB2 editions available.
DB2 Express Edition

Ideal for small and medium businesses (SMB), DB2 Express edition is a fully-functional edition of DB2 at an attractive entry-level price. DB2 Express Edition includes all the Express-C features, plus the following:

- **DB2 Advanced Copy Services (ACS)**
  - This feature enables you to use the fast copying technology available with some storage devices to perform backup and restore operations, which can dramatically speed-up backup and restore operations.

- **Online reorganization**
  - This feature allows you to issue REORGS (a command that reorganizes/rebuilds tables and indexes) online – that is, while the database is in use, and its objects are being accessed.

- **Label Based Access Control (LBAC)**
  - This makes it possible to protect data using labels and security policies.

- **Row and Column Access Control (RCAC)**
  - Introduced in DB2 10, RCAC complements the existing table privileges model by protecting access to a table at the row level, column level, or both.

- **Web services federation**
  - DB2 can have objects federated with web services, through web services wrappers, using Web Services Description Language (WSDL).

- **Homogeneous SQL replication**
  - DB2 Express can replicate data with other DB2 for Linux, UNIX, and Windows databases through capture and apply agents.

- **High Availability Disaster Recovery (HADR)**
  - This feature allows you to have a cluster of servers consisting of a primary database server and multiple standby database servers (multiple standby is a new feature in DB2 10.1). Standby database servers can take over to continue working and minimize impact for applications when problems occur with the primary database. Another offering with this feature is Read on Standby (ROS), which makes it possible to issue queries (SELECT statements) against a standby database.

- **Tivoli Service Automation for Multiplatforms (SA MP) support**
  - This support is used in conjunction with HADR to trigger automatic failover in a two-node HADR cluster when a failure occurs.

*To use the HADR feature, you must license the DB2 Express Edition product on both servers in the cluster. You can host several standby databases on the same server, in which case only one license is needed.

DB2 Express Edition 10.1 licenses are allowed to use up to 8 GB of memory (total), and can use up to 4 cores of a server's CPU. This is the only edition that lets you benefit from Fixed Term Licensing (FTL), a yearly subscription option offered as a low-cost alternative to permanent licensing. Being the entry-level charged edition (with a license that must be renewed annually), DB2 Express FTL is very well suited for users coming from Express-C. Other licensing methods are also available.
DB2 Workgroup Server Edition (WSE)

DB2 WSE is the perfect database solution for departmental, workgroup, or medium-sized business environments. It delivers all features present in DB2 Express Edition, and is the entry-level edition for DB2 pureScale functionality. DB2 WSE can be used on the following platforms:

- Linux (except Linux on System z)
- Windows
- AIX
- Solaris (SPARC and x64)
- HP-UX and Itanium

The main advantage of DB2 WSE over Express edition is that it allows you to use much more CPU and RAM.

DB2 Workgroup Server Edition 10.1 use is restricted to 16 cores per server and 64 GB of RAM. If the pureScale feature is in use, these limits will apply to the entire cluster. But the really good news is that pureScale is provided at no additional charge with DB2 WSE. (We'll look at the pureScale feature in more detail a little later.) It is important to note that installation requirements and platforms for DB2 pureScale are significantly different from those needed for regular DB2.

DB2 Enterprise Server Edition (ESE)

Ideal for high-performing, robust, on-demand enterprise solutions, DB2 ESE is designed to meet the data server needs of mid- to large-size businesses. It can be deployed on Linux, UNIX, and Windows servers of any size, from one to hundreds of processors, and on both physical and virtual servers.

DB2 ESE comes with all functionality of WSE, plus the following:

- **Connection Concentrator**
  - Allows DB2 to handle workloads for tens of thousands of users without dedicating database server resources to each one. This feature is part of Workload Manager, which is available with DB2 Advanced Enterprise Server Edition (AESE).

- **Query Tuner**
  - A utility that provides recommendations and analysis for tuning a single query.

- **Materialized Query Tables (MQTs)**
  - Structures that enable complex query results to be stored in regular tables. You can refresh those tables periodically, so access to the results is greatly improved. MQTs are key to solving complex query performance problems.

- **Multidimensional Clustering (MDC) Tables**
  - MDC provides an elegant method for clustering data in tables along multiple dimensions in a flexible, continuous, and automatic way. It can significantly improve query performance, and can significantly reduce the overhead of data maintenance, such as reorganization and index maintenance operations during insert, update, and delete operations. MDC is primarily intended for data warehousing and large database environments, but it can also be used in online transaction processing (OLTP) environments.
• **Multi-temperature data management**
  - This is a new feature of DB2 10.1 where, based on storage groups you define, DB2 distributes data among different device types, thereby placing data that is accessed frequently or constantly on faster storage, and data that is accessed infrequently or almost not at all on slower (and cheaper) storage devices.

• **Query parallelism**
  - This feature provides the ability to break a query in multiple parts and process them in parallel using intra-partition parallelism, thereby improving performance.

• **Table partitioning**
  - A data organization scheme in which table data is divided across multiple storage objects called data partitions or ranges according to values in one or more table columns. Each data partition is stored separately and can reside in different table spaces, in the same table space, or a combination of the two.

You can extend some of DB2 ESE's functionality with the purchase of any of the following additional extension packages:

  • **DB2 Storage Optimization feature**: Enables use of Adaptive Compression and classic row compression in DB2 ESE (offered free of charge in DB2 AESE).
  
  • **DB2 pureScale functionality**: Described later in the tutorial

The pricing model for DB2 ESE is available as Processor Value Unit (PVU) or per authorized user. Thus, as long as the license is dimensioned accordingly, a DB2 ESE database will be entitled to use all the resources available in a server.

**DB2 Advanced Enterprise Server Edition (AESE)**

The most complete DB2 edition available, DB2 AESE is a powerful database management solution that offers all of the functionality present with DB2 ESE, adding (at no additional cost) the following features and benefits:

  • **Adaptive Compression** and classic row (static) compression
    - Allows compression of data in tables using classic row compression (where data is compressed using table-level dictionaries) and extends it by compressing data dynamically using page-level dictionaries. Temporary tables are compressed when DB2 deems it necessary, and indexes used in compressed tables are compressed by default. (DB2 ESE users must purchase the DB2 Storage Optimization feature to obtain this functionality.)
  
  • **Workload Manager**
    - A utility that monitors the behavior of applications that run against a database, and changes the behavior depending on the rules that you specify in a configuration file. (For example, you can control system resources so that no one department or service class overwhelms a database server with requests.)
  
  • **Continuous data ingest**
    - A high-speed client-side DB2 utility that streams data from files or named pipes into DB2 target tables, usually to populate data warehouse databases.
• Federation with DB2 for Linux, UNIX, and Windows and Oracle data sources
  • A feature that allows you to query tables that reside in an Oracle database, as if they were local tables in your DB2 database.

• IBM InfoSphere Data Architect
  • A complete solution for designing, modeling, discovering, relating, and standardizing data assets. You can use it for data modeling, transformation, and DDL generation, and to build, debug, and manage database objects such as SQL stored procedures and functions.

• IBM InfoSphere Optim Configuration Manager
  • Provides advice on how to change database configurations, and stores states and changes in a repository. IBM InfoSphere Optim Configuration Manager makes it possible to compare current and historical data, helping to understand and resolve problems related to configuration changes.

• IBM InfoSphere Optim Performance Manager Extended Edition
  • Allows you to identify, diagnose, solve, and prevent performance problems in DB2 products and in associated applications including Java and DB2 Call Level Interface (CLI) applications.

• IBM InfoSphere Optim pureQuery Runtime
  • Lets you deploy advanced pureQuery applications that use static SQL for a wide range of benefits. It bridges the gap between data and Java technology by harnessing the power of SQL within an easy-to-use Java data access platform. It also increases security of Java applications helping to prevent threats like SQL injection.

• IBM InfoSphere Optim Query Workload Tuner
  • Enables all tuning features in both the IBM Data Studio full client and the IBM Data Studio administration client.

As with DB2 ESE, pureScale functionality is priced separately. Otherwise, the pricing model for DB2 AESE is the same as that used for DB2 ESE.

**DB2 Database Enterprise Developer Edition (DEDE)**

DB2 DEDE is a special offering tailored to provide developers almost all of the features that are present in other DB2 editions. With DB2 DEDE, a single application developer is able to design, build, and prototype applications, using advanced DB2 features, without having to spend unnecessary money for DB2 licenses. As the name implies, this edition is only meant to be used for development purposes and cannot be used in production.

DB2 DEDE further extends DB2 AESE by adding the following:

• Database Partitioning Feature (DPF)
• DB2 Connect functionality
• DB2 pureScale functionality

You must acquire a separate user license for each Authorized User of this product; PVU licensing is not available.
DB2 clients

No matter what edition you have running on your database server, any applications you use will have to connect to it. Such connection is done through DB2 clients and drivers — and there’s always a right client or driver suited for every application type. By knowing what each client product offers, you’ll be able to choose the one that best matches your application needs.

The IBM Data Server client and driver types available are as follows:

- IBM Data Server Driver Package
- IBM Data Server Driver for JDBC and SQLJ
- IBM Data Server Driver for ODBC and CLI
- IBM Data Server Runtime Client
- IBM Data Server Client

Figure 2 shows all features and capabilities that are offered with each DB2 client and driver available.

Figure 2. What’s inside each DB2 client and driver

For example, if you need to connect a Windows application to a DB2 database using ODBC, you will need the IBM Data Server Client Driver for ODBC/CLI. Or, if you want to administer DB2 using the Command Line Processor (CLP), the IBM Data Server Runtime Client should be enough.
Every DB2 Database Server edition includes the DB2 Data Server Client — which means that when working on the server where your database resides, you have all the connectivity to your databases that is available; this connectivity can also be used to establish a connection to databases residing on remote servers. (The only exemption is DB2 Express-C, which does not include Replication Center.)

It's also possible to connect to DB2 for System i or DB2 for System z databases by registering a DB2 Connect Personal Edition license to any DB2 client or driver. In fact, this is the simplest way of connecting an application to a DB2 database that resides on a mainframe.

DB2 data server drivers and clients are available for free — no additional licensing is required to use them, and the drivers can be embedded within applications. Thus, it is possible to redistribute DB2 drivers. For more details, access the IBM DB2 10.1 Information Center from the link in the Resources section.

### DB2 Connect

As mentioned earlier, DB2 is supported on both distributed systems and on platforms such as z/OS and IBM i — in fact, DB2 for Linux, UNIX, and Windows originally came from the mainframe.

But to be able to connect to such databases, just downloading and installing a DB2 client is not enough. Use of mainframe databases from DB2 clients requires another product known as DB2 Connect. If you want to connect directly to a database running on DB2 for z/OS or DB2 for IBM i, you must register a **DB2 Connect license** key within your DB2 client. As mentioned earlier, DB2 Connect licenses are applicable to both clients and drivers.

It is also possible to have a **DB2 Connect Server**, which acts as a gateway between DB2 on a mainframe and DB2 for Linux, UNIX, and Windows so that regular DB2 for Linux, UNIX, and Windows clients are able to connect to mainframe databases through a gateway. **Figure 3** illustrates how DB2 Connect works in a client and server environment.

**Figure 3. How DB2 Connect works**

It is important to understand that DB2 Connect and DB2 clients are different products that are used for different purposes. (The word "Connect" sometimes leads to confusion). Every time you see a reference to DB2 Connect, you must remember that, although it can be used to connect
to databases on DB2 for LUW, this product is mainly used to connect regular DB2 clients and drivers (or even database servers) to mainframe databases (that is, to DB2 for z/OS or DB2 for i databases). I’ve seen people download DB2 Connect and use it as regular client — which now you know is not what it is intended for.

Database workloads and DB2 10.1

What is data warehousing?

There are two main types of database application workloads: online transactional processing (OLTP), and data warehousing, which includes reporting, online analytical processing (OLAP), and data mining applications.

DB2 10.1 excels at both.

What differentiates an OLTP system from a business intelligence (BI) data warehousing system? The queries that are typically used to access the data. An OLTP system is typical of a web order system, where you perform transactions over the web (such as ordering a product). These applications are characterized by granular, single-row lookups with logic that likely updates a small number of records. In contrast, BI-type queries perform large table scans as they try to find data patterns in vast amounts of data. If you've ever been asked to summarize all of the sales for a particular region, that's an example of a warehousing query.

Quite simply, when you think of OLTP, think short and sweet. On the other hand, with BI, think of looking for needles in a haystack or aggregating a lot of data for reporting. Of course there's more to it than that, but you get the point.

Systems that contain operational data — data that runs the daily transactions of a business — are OLTP systems. However, these systems contain information that business analysts can use to better understand how a business is operating. For example, they can see what products were sold in which regions at which time of year. This helps identify anomalies or can be used to project future sales. However, several problems can present themselves if analysts access operational (OLTP) data directly for reporting and other BI activities:

- They might not have the expertise to query the operational database. In general, the programmers who have the expertise to query an operational database have a full-time job maintaining the database and its applications.
- Performance is critical for many operational databases (for example, a database used to process banking transactions). These systems can't handle users making ad-hoc queries on operational data stores. Consider, for example, the time you take to pay bills online. When you select OK, it usually takes only a few seconds to process a payment. Now, consider a bank analyst trying to figure out how to make more money from an existing customer base. The analyst runs a query that is so complex that your banking transaction now takes about 30 seconds to complete! Obviously that performance time is not acceptable (and neither are the new charges the analyst is dreaming up). For this reason, operational data stores and reporting data stores (including OLAP databases) are usually separated. Over the last few years, though, reporting data stores have tended to become pseudo-operational and
current. Such stores are called operation data stores (ODSs) or even active data warehouses. Consider the telecommunications industry, for example. ODSs are popular with these companies as they try to identify fraudulent charges as early as possible. DB2 is one of the few databases that is well-suited for both operational and reporting workloads.

- Operational data is not generally in the best format for use by business analysts. Sales data that is summarized by product, region, and season is much more useful to analysts than raw transaction data.

Data warehousing solves these problems. In data warehousing, you create stores of informational data — data that is extracted from operational data and then transformed and cleansed for end-user decision making. For example, a data warehousing tool might copy all the sales data from an operational database, perform calculations to summarize the data, and write the summarized data to a database that is separate from the operational data. End users can then query the separate database (the warehouse) without affecting the OLTP database.

**DB2 solutions for different workloads**

Now that you know the difference between OLTP and data warehousing, let's look at the solutions IBM has for both types of workloads. As already stated, DB2 delivers exceptional results when used to process both types of workloads — but as the volumes grow, you may have to expand your environment (one server might have to become a cluster of servers, more memory may be needed, more powerful processors may be required, and so on). DB2 can be expanded in same manner, through InfoSphere Warehouse (for data warehousing workloads) and DB2 pureScale (for OLTP workloads) — as you can see on Figure 4.

**Figure 4. Methods for growing and scaling a DB2 environment.**

**InfoSphere Warehouse**

IBM Infosphere Warehouse is a complete warehousing/OLAP/analytics solution that has DB2 Enterprise Edition at its core. This product also includes the Database Partitioning Feature (DPF); DPF is used to partition data among different databases, servers and storages, such that all servers process queries by retrieving their own (separate and different) portion of data, thereby achieving high levels of parallelism as more partitions are used. In the past, it was possible to add
DPF to DB2 ESE. But, beginning with version 10.1, if you want DPF, you must use InfoSphere Warehouse. That really makes sense, because DPF is suited for warehouse/OLAP workloads, and InfoSphere Warehouse is the IBM product that has been designed specifically for those types of environments.

InfoSphere Warehouse benefits from all the new features present in DB2 10.1, plus it offers several other functionalities to optimize your data warehouse needs. More information on InfoSphere Warehouse and its features and functionalities can be found on the InfoSphere Warehouse web site.

**DB2 pureScale**

To leverage the use of DB2 in critical environments that demand exceptional performance (usually made available through expensive and over-sized servers) where scaling processing power is often necessary, DB2 for Linux, UNIX, and Windows offers a new functionality that is based on the SYSPLEX coupling facility that has been in use for quite some time on DB2 for z/OS: pureScale. DB2 pureScale is an add-on feature that enables DB2 to better support transactional workload demands. Its use enables applications with big (or huge) online transaction processing (OLTP) volumes to obtain a high level of parallelism through a set of servers, working as a cluster, that access shared storage. Applications can connect to any member of a pureScale cluster, and each member processes transactions independently, delivering the desired performance, and enabling environment growth whenever necessary. For buffer coherency and global locking, a pureScale cluster relies on a component known as the Cluster Caching Facility (CF). Figure 5 illustrates a simple DB2 pureScale environment.

**Figure 5. A typical DB2 pureScale environment**

[Diagram of a typical DB2 pureScale environment]

Scalability a pureScale cluster is easy: you simply add a new “member” (as each server within a pureScale server is called), without any application outages. Removing member(s) also works in a similar fashion. Having such scalability available is really a differential, but that’s not the only thing that makes DB2 pureScale unique. By enabling use of servers working in a cluster environment, DB2 pureScale increases database availability—whenever a cluster member fails, its requests will be routed to another member in the cluster automatically, and this re-routing is transparent to applications that are accessing the database. The same behavior will be applied to the CF in case of failure (when multiple CF servers are in place).
As stated earlier, DB2 pureScale is a paid-separately, add-on product that can be used with DB2 WSE (free-of-charge), DB2 ESE and DB2 AESE. Its use is restricted to a specific hardware and can only be run on some versions of IBM AIX (AIX 6.1 and 7.1) and Linux (SUSE and Red Hat Enterprise Linux - RHEL). For more information, refer to the links provided at the end of this tutorial. Every member of a pureScale cluster requires a DB2 license, as well as a pureScale license. However, no additional license is needed for the CF server(s) used.

Before DB2 10.1, if you wanted to use DB2 pureScale, a special release of DB2 was necessary: DB2 9.8. (This version was the first implementation of DB2 pureScale.) Now, with DB2 10.1, pureScale has been integrated to DB2's core. It's important to note that no application changes are necessary when migrating from traditional DB2 to DB2 pureScale. In fact, to applications, it appears as if traditional DB2 is being used, but that performance has improved significantly.

**DB2 tools**

DB2 comes equipped with a set of tools that enable you to access, query, operate, and manage your DB2 database environment, either locally or remotely. Many of these tools are stand-alone with graphical user interfaces (like Data Studio and Replication Center) while others are command line tools; we'll look at both types in more detail in this section.

**Command line tools**

**DB2 Command Line Processor (CLP)**

The Command Line Processor (CLP) is a command-line interface that can be used to interact with DB2 instances and databases. You'll find it with every DB2 edition, as well as with each DB2 client (but not with driver packages).

The DB2 CLP can be run in interactive mode (by executing the command db2 once) or in non-interactive mode (by prefixing commands run from a system command prompt with the keyword "db2").

For example, in interactive mode, a DB2 command would be executed like this:

```
db2 => list applications
```

In non-interactive mode, however, the same command would be executed like this:

```
db2 list applications
```

When using non-interactive mode, you can execute operating system’s commands whenever you want. But if you need to execute an operating system command while running in interactive mode, you must prefix the command with an exclamation mark (!). For example:

```
db2 => !dir
```

On Windows, to enable use of DB2 CLP in interactive and non-interactive modes, you must click on the **DB2 Command Window** icon or run the db2cmd command. When the db2cmd command
is invoked from another command terminal, Windows will create a new window for DB2 CLP non-interactive use. **Figure 6** illustrates how the DB2 CLP is run in interactive mode.

**Figure 6. Running the DB2 CLP in interactive mode**

You must also use a Command Window to run CLP in interactive mode. When DB2 is installed on a Windows server, a menu item named “DB2 Command Line Processor” is created and when selected, this menu item launches CLP in interactive mode. **Figure 7** shows the menu item used to launch the CLP in interactive mode.

**Figure 7. The menu item used to start the DB2 Command Line Processor**

**Running CLP on Linux and UNIX**

On Linux and UNIX, you have to "source" the DB2 profile before you can use CLP. Sourcing the DB2 profile is done automatically when you log on with the DB2 instance owner user-id. If you are not an instance owner, you must do the following before you will be allowed to use the DB2 CLP:
1. Locate the home directory for the instance owner user (also called instance-home).
2. Within the instance home directory, you'll find a subdirectory called sqllib. Inside that subdirectory you will find a file named db2profile.
3. Execute the commands in the db2profile file as follows: (assuming the DB2 instance home directory is /home/db2inst1)

```
$ . /home/db2inst1/sqllib/db2profile
```

Don't forget the space between the point and the file location — otherwise the command won’t work. The execution won’t return any outputs and right after executing it, you will be in DB2 CLP non-interactive mode. Try it! You can add that command to the user’s profile (usually a hidden file called .profile or .bashrc within your user's home directory) so that it will be executed automatically whenever you log on.

Now that you know how to access DB2 CLP, let’s look at some of its basic functionality. I'll be showing commands using non-interactive mode, but you can use DB2 CLP in interactive or non-interactive mode as you wish. Want to switch from non-interactive mode to interactive mode? Just issue the command db2 at the command prompt. To get help at any time, enter db2 "?” (non-interactive mode) or ? (interactive mode). Figure 8 shows the output that is generated when help is invoked.

**Figure 8. The DB2 ? command**

```
$ db2 "?
```

To get more help on a specific command, run the command "? <DB2_Command>". For example:

```
db2 "? list utilities"
```

### Measuring your query execution times

On Linux and UNIX, you can take advantage of the `time` utility to measure the execution time of a given statement or command (when in non-interactive mode). For example:

```
$ time db2 "select count(*)
> from department"
1
1 record(s) selected.
real 0m0.033s
user 0m0.013s
sys 0m0.015s
```
You may have noticed that the DB2 command provided in this example is enclosed in double quotes. On UNIX platforms, you must enclose DB2 commands and SQL statements in double quotes any time they contain special characters — * ( ) \ & | < > ? ! — if you want to execute those commands/statements using DB2 CLP in non-interactive mode. That's because way in which the UNIX operating system interprets such characters can produce unexpected results.

The DB2 CLP uses DB2's directory files to establish database connectivity, which means that if you want to work with a remote database, its server has to be catalogued locally as a node, and the database itself must be catalogued in the system database directory. (You'll learn more about DB2's directory files in the third tutorial in this series: Working with Databases and Database Objects). Local databases are catalogued automatically, so you can easily access them through CLP.

It's possible to issue any SQL query against a DB2 database using CLP. However, when updating data or changing the definition of objects in a database, CLP will automatically commit the changes after each statement is executed, by default. This behavior, as well as several other default behaviors of CLP can be changed. To check what options can be changed in CLP, run the following command:

$ db2 list command options

When this command is executed, you should see output that looks like this:

**Listing 1. Command options**

$ db2 list command options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
<th>Current Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>-a</td>
<td>Display SQLCA</td>
<td>OFF</td>
</tr>
<tr>
<td>-c</td>
<td>Auto-Commit</td>
<td>ON</td>
</tr>
<tr>
<td>-d</td>
<td>Retrieve and display XML declarations</td>
<td>OFF</td>
</tr>
<tr>
<td>-e</td>
<td>Display SQLCODE/SQLSTATE</td>
<td>OFF</td>
</tr>
<tr>
<td>-f</td>
<td>Read from input file</td>
<td>OFF</td>
</tr>
<tr>
<td>-i</td>
<td>Display XML data with indentation</td>
<td>OFF</td>
</tr>
<tr>
<td>-l</td>
<td>Log commands in history file</td>
<td>OFF</td>
</tr>
<tr>
<td>-m</td>
<td>Display the number of rows affected</td>
<td>OFF</td>
</tr>
<tr>
<td>-n</td>
<td>Remove new line character</td>
<td>OFF</td>
</tr>
<tr>
<td>-o</td>
<td>Display output</td>
<td>ON</td>
</tr>
<tr>
<td>-p</td>
<td>Display interactive input prompt</td>
<td>ON</td>
</tr>
<tr>
<td>-q</td>
<td>Preserve whitespaces &amp; linefeeds</td>
<td>OFF</td>
</tr>
<tr>
<td>-r</td>
<td>Save output to report file</td>
<td>OFF</td>
</tr>
<tr>
<td>-s</td>
<td>Stop execution on command error</td>
<td>OFF</td>
</tr>
<tr>
<td>-t</td>
<td>Set statement termination character</td>
<td>OFF</td>
</tr>
<tr>
<td>-v</td>
<td>Echo current command</td>
<td>OFF</td>
</tr>
<tr>
<td>-w</td>
<td>Display FETCH/SELECT warning messages</td>
<td>ON</td>
</tr>
<tr>
<td>-x</td>
<td>Suppress printing of column headings</td>
<td>OFF</td>
</tr>
<tr>
<td>-z</td>
<td>Save all output to output file</td>
<td>OFF</td>
</tr>
</tbody>
</table>
You can change any of the options available for each command you execute in non-interactive mode. In interactive mode, changing an option affects all subsequent commands executed in the CLP session.

The plus (+) and minus (-) characters can be used to turn an option ON or OFF. For instance, with the auto-commit option:

-c switches auto-commit to ON (which is the default) while +c or -c- (both do the same) do the opposite, that is, they switch the auto-commit option to OFF.

The following example shows several statements in which CLP options are modified as part of the statement execution.

To issue administrative commands like `db2start` (instance start) or `db2stop` (instance stop) on a Windows server, you have to open a DB2 Command Window in administrative mode (refer to Figure 6).

**Listing 2. CLP options modified as part of statement execution**

```
$ db2 "select sum(salary) from employee"
1
---------------------------------
  2686777.50
1 record(s) selected.

$ db2 +c "update employee set salary = salary*1.5"
DB20000I  The SQL command completed successfully.
$ db2 +c "select sum(salary) from employee"
1
---------------------------------
  4030166.25
1 record(s) selected.

$ db2 rollback
DB20000I  The SQL command completed successfully.
$ db2 "select sum(salary) from employee"
1
---------------------------------
  2686777.50
1 record(s) selected.
```

To issue administrative commands like `db2start` (instance start) or `db2stop` (instance stop) on a Windows server, you have to open a DB2 Command Window in administrative mode (refer to Figure 6).

**CLPPlus**

Like DB2 CLP, CLPPlus is a lightweight command line processor that is present on DB2 Data Server Driver packages and Clients. CLPPlus extends some features of DB2 CLP by adding the following functionality:
• Support for connection to databases given only the database name, port, user ID and password (no DB2 directory files needed)
• A text buffer that stores scripts, script fragments, SQL statements, SQL PL statements and PL/SQL statements
• Multiple options for formatting the output of scripts and queries

CLPPlus also provides a customizable way of showing query results — data is presented in a more friendly fashion on query executions and the way it is presented can be customized. It is possible, for example, to format numeric results, change column headers, and so on.

CLPPlus is also a great way to welcome users coming from other database management products. With CLPPlus, most of the options and functionality individuals coming from other RDMS products are used to remains available — coupled with the power of IBM DB2.

You can access CLPPlus from the operating system menu, or by typing the command clpplus in the command line terminal. It's also possible to specify a database connection when invoking CLPPlus from a command line terminal by entering a command that looks like this:

$ clpplus userid@hostname:port_number/dbname

For example, if you know you can access a database named SAMPLE through port 50005 and your user ID is idngf01, you can start CLPPlus and establish a connection to the SAMPLE database by executing a command that looks like this:

$ clpplus idngf01@localhost:50005/sample

Figure 9 shows the type of output that is generated if a database connection is successfully established when CLPPlus is invoked.

Figure 9. Starting CLPPlus with database connection information

Once CLPPlus has been started, you can use the following command to get an index of help topics that are available:

SQL> help index
**Figure 10** shows the output that is generated when the help index is displayed.

**Figure 10. Output from the CLPPlus HELP INDEX command**

![Output from the CLPPlus HELP INDEX command](image)

To get specific help on any of the commands available, simply type HELP, followed by the command.

For example:

```sql
SQL> help set
```

By changing parameters with the SET command, it is possible to change the way in which data is displayed when queries are executed. It is also possible to edit your commands prior to executing them (or after receiving an error message) through the EDIT command. After the EDIT command is called, a default editor is opened (Notepad on Windows, vi on Linux) and you are expected to create and save a script. After the script is edited and saved, an the editor is closed, you can run the edited command through the CLPPlus RUN command.

Commands can have multiple lines — the forward-slash symbol ("/"), provides a completion indicator to the editor — and thanks to the CLPPlus buffer, you can navigate through all of the commands that have been entered (since CLPPlus was started), by pressing the up and down arrow keys on your keyboard.

**db2pd — Monitor and troubleshoot tool**

The DB2 Problem Determination Tool (or db2pd) is a powerful tool with a command-line interface that can help you monitor and troubleshoot your databases with very little system interaction.
(db2pd attaches directly to DB2 shared memory sets to retrieve system and event monitor information; consequently, it does not acquiring latches or consume system resources.) The db2pd tool can be run in either interactive or non-interactive mode.

**Figure 11** shows the use of db2pd to retrieve information about DB2 memory usage in interactive mode.

**Figure 11. Running db2pd in interactive mode**

```
$ db2pd
db2pd> You are running db2pd in interactive mode.
db2pd> If you want command line mode, rerun db2pd with valid options.
db2pd> Type -h or -help for help.
db2pd> Type q to quit.
db2pd> -dbptnmem
Database Member 0 -- Active -- Up 0 days 00:00:12 -- Date 10/03/2012
Database Member Memory Controller Statistics
Controller Automatic: Y
Memory Limit: 4194304 KB
Current usage: 154944 KB
HWM usage: 154944 KB
Cached memory: 72512 KB
Individual Memory Consumers:

<table>
<thead>
<tr>
<th>Name</th>
<th>Mem Used (KB)</th>
<th>HWM Used (KB)</th>
<th>Cached (KB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DMG_f01</td>
<td>169216</td>
<td>169216</td>
<td>72512</td>
</tr>
<tr>
<td>DMG_R</td>
<td>22528</td>
<td>22528</td>
<td>0</td>
</tr>
<tr>
<td>DMG_PRIVATE</td>
<td>3260</td>
<td>3260</td>
<td>0</td>
</tr>
</tbody>
</table>

To run db2pd non-interactive mode, just issue the command db2pd, followed by the appropriate option.

For example: db2pd -dbptnmem

**Data replication tools**

DB2 provides two different solutions you can use to replicate data to and from relational databases (including some non-IBM databases): SQL replication and Q Replication. Both solutions can be configured and maintained by both the Replication Center (a stand-alone GUI for replication setup and maintenance) and the replication configuration command-line processor (ASNCLP). Such tools are not available on DB2 10 Express-C and Express editions.

While SQL replication is a bit simpler to configure and doesn’t depend on additional products for homogeneous replication, Q replication tends to be used most often when big portions of a database need to be replicated, or when significant amounts of data need to be transferred to data marts/data warehouses. Q replication requires installation of InfoSphere Replication server as well as WebSphere® MQ (both are priced separately). As it divides a workload across these two products, Q Replication is able to replicate large volumes of data with a low level of latency.
Other tools

Other useful tools that can be found with almost any DB2 product edition are:

• **Visual Explain** — Provides a visual depiction of the access plan that was chosen for a query (either SQL or XQuery), thereby helping you solve query performance issues more easily. Visual Explain is part of IBM Data Studio — which will be covered a little later. Two additional tools that can be used to format and display Explain data are db2explain and db2explainfmt.

• **db2look** — Use this tool to generate DDL statements for an entire database, for specific schemas, or for a list of tables. db2look is a command-line tool; however, its functionality is also present with IBM Data Studio, through the option "Generate DDL", which is available for most objects that can be accessed in with this tool.

• **db2level** — Use this tool to display information related to the current level (version and release) of the DB2 product you have installed.

• **db2licm** — Use this tool to determine whether to apply a license or to display licensing information for your installed products.

• **db2cfexp / db2cfimp** — With these utilities, you can export and import connectivity information as well as configuration information for databases, database manager instances, and servers, all in one shot.

• **db2advis** — With this extremely useful utility, you can improve performance, by having DB2 advise you on which indexes and MQTs to create (or drop).

• **db2top** — Use this tool to monitor your database in real-time (not available in GUI / Windows).

• **db2diag** — Use this diagnostics tool to understand what’s happening right now with your DB2 environment.

There are many other tools that you can learn and take advantage of. I encourage you to learn more about them from the [Information Center for DB2 10.1](https://www.ibm.com/support/knowledgecenter/SSEKQY_10.1.0/com.ibm.db2.luw.admin.sqlref.doc/doc/cng_index.html).

**IBM Data Studio**

Having powerful and productive command line tools to administer database servers is something indispensable for every experienced DBA. But what about newcomers? Graphical user interface (GUI) tools are great for users who are just starting with any product — especially when it comes to database management systems. Usually, GUI tools are of great help even for the most skilled professionals — it’s just not possible to memorize all the commands you need!

IBM Data Studio is the new standard GUI tool for DB2 databases administration, management and development. Through an integrated and modular environment, Data Studio provides collaborative database development tools for DB2 for Linux, UNIX, and Windows, DB2 for z/OS, DB2 for i, Informix and other non-IBM database products, with support for several programming languages.

You can download Data Studio at no-charge, or can you can get it together with DB2 when downloading the product (by selecting Data Studio as well through use of Download Director). It is also possible to download Data Studio when installing DB2 through the DB2 installation launchpad. The version of Data Studio that is currently available (at the time this tutorial was written) is 3.1.1.
Figure 12 shows what the start-up panel for this version looks like.

**Figure 12. The start-up panel for Data Studio 3.1.1**

![Data Studio Start-up Panel](image)

**Data Studio components**

There are 3 different components to choose from when downloading IBM Data Studio (all are available at no-charge):

- IBM Data Studio Administration client
- IBM Data Studio Full client
- IBM Data Studio Web console

The Administration client is a lightweight tool for administering databases and meets most of the basic development needs for DB2 for Linux, UNIX, and Windows and DB2 for z/OS. The Full client expands the functionality of the administration client to support development of Java, SQL PL and PL/SQL routines, XML, and other technologies. You must decide which version (administration or full client) will better meet your needs. Both full and administration client components can be installed on Linux and Windows platforms.

Web console complements Data Studio by adding monitoring and job management, both of which can be accessed through a web browser (you’ll see more about this product later).

**Getting started with IBM Data Studio**

If you are already used to Eclipse-based applications, you'll very likely feel comfortable with the Data Studio interface. After spending some time with the tool, users coming from the DB2 Control Center will find Data Studio very intuitive and easy to use. **Figure 13** shows the basic interface for IBM Data Studio.
One key concept in Data Studio is that almost all functionality is accessed through context menus, which are presented whenever you right-click on the object you want to work with. To find out what options are available for any object, simply highlight that object and put your mouse’s right button to work!

When you first start Data Studio, you are presented with the Task Launcher view, which welcomes you to Data Studio and provides you with a list of some of the tasks you can perform. The Task Launcher view can be seen in Figure 14.

To connect to a database from the Task Launcher, select **Administer**, followed by **Connect and browse a database**. Then, when the **Select Connection** dialog is presented, click **New**. (You can do the same thing by choosing the Administration Explorer view and clicking the button **New**.)
Figure 15 depicts the menu items that must be selected to establish a new connection in this manner.

**Figure 15. Establishing a new database connection**

When the New Connection dialog appears, select **DB2 for Linux, UNIX and Windows** in the left-most box (to connect to a DB2 for Linux, UNIX, and Windows database), and enter the appropriate information such as database name, host name, port, user ID, and password. **Figure 16** shows a New Connection dialog whose input fields have been populated with this type of information.

**Figure 16. The New Connection dialog**

(Before you can connect to any database, the database must first be created; DB2 ships with a SAMPLE database that can be created at any time by executing the command `db2sampl`, or by making the appropriate selection from the First Steps utility.)
DB2 Control Center (deprecated)

In versions prior to DB2 10.1, there was a GUI tool called the Control Center that offered Wizards that could be used to guide database administrators through the steps needed to complete specific tasks (such as backing up a Db2 database). The Control Center was deprecated in DB2 9.7 and was discontinued in DB2 10.1.

IBM Data Studio and the Data Studio web console replace (and in some cases, extend) most of the functionality that was available with the Control Center; both offer task assistants in place of the Control Center's Wizards, to help you to perform database administration/development tasks (and to help you better understand what's required to do each task). For a complete mapping of features and functionality between the Control Center and Data Studio, refer to this developerWorks article: Migrating from DB2 Control Center to IBM Data Studio

The Test Connection button on the New Connection dialog can be used to check the validity of the data entered; when this button is pressed an attempt is made to connect to the database specified using the credentials and connectivity information provided.

It is important to note that you can use most of Data Studio's functionality without having to install a DB2 client/driver package. That's because Data Studio comes equipped with drivers that offer connectivity to several database products (usually through JDBC). Therefore, if you plan to administer DB2 remotely, or you plan to develop using a remote database server, having only IBM Data Studio installed locally should be enough.

Now that you know how to establish a database connection in Data Studio, let's take a look at Data Studio's interface.

As an Eclipse-based application, IBM Data Studio makes it possible for you to choose between different perspectives, making its use more directed to what you intend to do. For example, when you choose the Database administration perspective, Data Studio's interface changes to show you the view elements that are relevant to database administration like the Administration Explorer, the Job Manager, and so on. Figure 17 illustrates how you can select the Database Administration, Data and IBM Query Tuning perspectives.

Figure 17. Changing between perspectives

Perspectives, as the name implies, influence only the way in which you see what you are doing in Data Studio: when you switch between perspectives, nothing is lost — only the way in which you view what you have done is affected. A variety of perspectives are available. Figure 18 shows all the perspectives that exist in Data Studio full client.
Accessing objects and data

Usually, the one thing users want to see in a database management tool is the ability to examine a database's tables and views.

To view the tables within a database you can navigate through Administration Explorer view (under Database Administration perspective) and after successfully connecting to a database, the Tables folder will be shown in the tree that's presented on the left-hand side of the screen. By clicking the Tables folder, you will be able to get a list of all the tables available in the Editors area. Accessing tables through the Data perspective has a different path: you must go through the Data Source Explorer view and open the Schemas folder to obtain a list of the tables that have been defined within the schema chosen.

Figure 19 shows how to access tables in the SAMPLE database using the Database Administration perspective.
As you can see, when using the Database Administration perspective, every folder in the Administration Explorer view shows you all existing objects of a particular type (for example, all tables, all schemas, all sequences, and so forth). But when you switch to the Data perspective, navigation follows a top-down hierarchy. Within a database, you can see the schemas, and inside each schemas, you can see tables, views, sequences and so on that have been defined within that schema). You get to choose the way you want to navigate.

Right-clicking on a table will show the context menu with all the options that can be performed on that table. And by using this context menu, you can operate the table specified as you wish.

Data Studio also has a diagram editor, which can be used to better visualize a database’s objects and the relationships between them. Figure 20 shows you the menu items that must be selected to invoke the diagram editor.

Figure 20. Invoking the diagram editor

You'll be prompted to choose additional tables as you wish, and then Data Studio will present the tables you have selected, arranged in a diagram as requested. Figure 21 illustrates how a diagram produced by the diagram editor might look.
Figure 21. The diagram editor interface

![Diagram Editor Interface]

Using a diagram like the one produced by the diagram editor makes it easier to understand structures and object relationships.

Changing data

Using context menus, it's also possible to browse and edit data. (Simply right click on a selected table and choose the **Browse Data** or **Edit Data** menu option). You can edit data in line and when you save, Data Studio will show you the SQL statement(s) that were used to make the changes. Data Studio highlights changes in yellow; Figure 22 illustrates an example where the data for a row and column has been changed. Figure 23 shows the SQL that was used to make the change.

Figure 22. How modified values are presented

![Modified Values]

Figure 23. The SQL that was used to make the change shown in Figure 22

![SQL Results]

Issuing queries against a database

Data Studio assists you when writing SQL or XQuery queries. To create a new script (that is, a script that does not contain any statements), simply highlight a database object, right-click on it to bring up a context menu, and select the **New SQL Script** option. (The same thing can be done by right-clicking on an SQL Scripts folder in any project you select under **Data Project Explorer** — the context menu will be shown as "New SQL or XQuery Script").
All SQL or XQuery scripts you create will be shown on the Data Project Explorer, within the project you chose, when it is saved.

As with other Integrated Development Environments (IDEs), Data Studio can assist you as you enter queries in a script, by auto-completing object names. This functionality is provided via the Content Assist context menu item. Figure 24 illustrates how this menu item looks.

**Figure 24. Using Content Assist**

![Content Assist Example](image)

Data Studio can also help you format any unformatted SQL. Figure 25 illustrates how the “Format SQL” menu item can be used to apply the proper formatting to an unformatted SELECT statement.

**Figure 25. Using Format SQL**

![Format SQL Example](image)

There’s also the Content Tip function, which assists you with syntax tips as you write your scripts. Figure 26 illustrates how Content Tips are presented.
Figure 26. Content Tips

Once you are done and are ready to submit your script/query to DB2 for execution, you simply just press the Run SQL button (the green circle with a triangle in it that looks like a Play button). This can also be done by pressing the F5 key. And if your script contains multiple queries, you can execute an individual query by selecting just the text for the query that you want executed, before pressing the Run SQL button or the F5 key. Figure 27 illustrates how the Run SQL button looks (on the Data Studio tool bar).

Figure 27. The Run SQL button

With this toolbar, it is also possible to show the access plan for a query (Open Visual Explain button), tune a query (Start tuning button), import or export the SQL script (In-Box and Out-Box buttons), or schedule a query to be executed later (Clock button).

Retrieving results

When you execute queries, Data Studio shows the results in either text form or in a grid. Figure 28 illustrates how query results are returned in a grid; Figure 29 shows query results being returned as text.

Figure 28. Query results shown in grid mode

Figure 29. Query results shown in text mode

Tuning queries inside Data Studio

In a perfect world, DB2 would always return the results for a particular query “in the blink of an eye.” But in the real world, you are required to create appropriate indexes and materialized query tables (MQTs), keep statistics up to date, and eliminate any fragmentation in your data (through reorgs) before DB2 can deliver the best query response times possible. The DB2 Optimizer does an exceptional job in choosing the access plans that are used to retrieve the data that is needed to resolve a query (and is being continually improved as new versions are released), but there
are times when some queries simply do not behave as expected. For these types of queries, Data Studio can help you identify problems and solve performance issues by advising on the appropriate action(s) to take.

Data Studio offers basic tuning features (Statistics advisor, Query formatting, Access Plan graph and Reports) free of charge and many of these features can be extended with the purchase of InfoSphere Optim Query Tuner for Linux, UNIX and Windows.

As with most of the Data Studio functionality, it is possible for you to invoke the query tuner from many places. One possible way of calling it is through the SQL script editor (by pressing the **Start tuning** graphical button.) After pressing this button and interacting with the query tuner assistant, you should get a screen like the one that's shown in **Figure 30**.

**Figure 30. The Query Tuner Workflow Assistant**

If you look closely at **Figure 30**, you will see a data access plan that is being displayed for a particular query. By analyzing access plans, you can determine whether or not a query is using indexes efficiently. Data Studio can also suggest running statistics on tables that a query frequently accesses. You'll have more tuning alternatives available when you use InfoSphere Optim Query Workload Tuner (paid-separately), or alternatively, the DB2 Advisor, which can be invoked by executing the `db2advis` command.

**Database administration with Data Studio**

Earlier, we saw how to switch to the Database Administration perspective to facilitate database administration work inside Data Studio. After changing to this perspective, you'll be presented with the Administration Explorer view.

Inside the Administration Explorer view, there are a lot of objects shown that you can choose to operate with. It is possible to change objects such as tables, columns, indexes, and for every change you make, Data Studio creates a change plan that analyses impacts and can be used to implement your changes — or schedule them to be done later. Change plans are saved in
the “Change Plans” folder on the Administration Explorer view. And objects that have a pending change plan have their icons changed to a delta sign (like a triangle) to visually indicate that they have changes that are pending.

**Figure 31** illustrates how to change a column in a table, and the menu items that must be selected to review the resulting change plan that gets generated.

**Figure 31. Altering a table's structure, and reviewing it through a Change Plan**

As you can see in **Figure 31**, there's also an option called **Review Undo Script** which you can use to revert all changes, if necessary. When the **Review and Deploy** menu item is selected, the change plan that was generated will be displayed in the Review and Deploy dialog and you will be given the opportunity to run the plan immediately or schedule it to be executed at a later date and time. **Figure 32** shows what this dialog looks like.
If you elect to run the change plan right away, the results of the execution will be displayed in the SQL Results window. **Figure 33** shows what this window looks like.

**Figure 33. The SQL Results window after changes to an object have been made**

To execute administrative tasks, Data Studio connects to the database server specified using Secure Shell (SSH). DB2 10.1 includes a lightweight SSH server with Windows installs; on Linux and UNIX servers the SSH server that is available with the operating system is used instead. When connecting to older versions of DB2 on Windows, Data Studio can also make use of the Database Administration Server (DAS) if there’s no SSH support available. (DB2 8.2 and earlier only uses DAS; the DAS was deprecated in DB2 9.7 and may be discontinued in a later version.)

**Monitoring and managing jobs through Web Console**

Data Studio web console is available as a separate download, and can be used to monitor database health and availability, as well as manage scheduled jobs through a web browser. Data Studio web console contains a web server which can be accessed from anywhere — any device that has a web browser and can connect to the web console can be used to monitor a DB2 database. Thus, you can keep an eye on a database using a smartphone or tablet — you’re not limited to having to use a computer. Data Studio web console is available on Linux, AIX, HP-UX, Solaris, and Windows platforms. **Figure 34** shows what the opening screen of the Data Studio web console looks like. **Figure 35** shows how the Data Studio web console can be used to monitor the health of a database.
Non-relational data concepts in DB2

Now, more than ever, relational database management systems are being used to store not so conventional types of data such as audio clips, video clips, images, graphs, binary files and practically any other type of data that you can imagine. Such data can be stored in DB2 databases through the use of Large Object (LOB), Extensive Markup Language (XML), and structured data types. Although you’ll learn about data types in more detail in other parts of this tutorial series, we’ll explore the LOB, XML, and structured data types here.

Large Objects (LOBs)

DB2 has several built-in data types that can be used to store various types of “traditional” information, such as numbers, characters and character strings, dates, times, and timestamps. But
there may be times when you want or need to store large chunks of data that is "non-traditional" or too large to be stored in one of the more conventional data types. For these situations, DB2 provides the following large object (LOB) data types:

- Binary Large Object (BLOB), which is used to store files and any kind of varying length binary data;
- Character Large Object (CLOB), which is used to store large text data values;
- Double-Byte Character Large Object (DBCLOB), which is used to store large graphic/double-byte character data;
- National Character Large Object (NCLOB), which is essentially the same as DBCLOB, and is provided for compatibility with Oracle applications. NCLOB is only available in DB2 10.1 for LUW. (Not present in DB2 10 for z/OS.)

From an application perspective, accessing large object data is done a little differently, depending on the type of LOB data being retrieved. CLOB data can be retrieved and stored methods that would be used if the data was stored in a more traditional character data type. Consequently, locating, retrieving, inserting, updating, and deleting data is imperceptible to applications.

For BLOB data, as the binary values usually don’t make sense to our eyes, applications have to be programmed to deal with such data. Some programming languages use statements like SELECTBLOB and UPDATEBLOB to specially fetch and update binary data.

**LOB locators**

LOB and XML values can be very large (up to 2 GB in size), and the transfer of such values from the database server to a client application program host variable can be time consuming. Because of this, most application programs prefer to process LOB values in pieces, rather than as a whole. C/C++ and embedded SQL applications can reference a LOB value by using what is known as a large object locator. LOB locators represent a value for a LOB resource that is stored in a DB2 database and enable applications to operate on small chunks of that resource at a time, instead of having to retrieve the entire LOB data value. They behave as a snapshot of a piece of an LOB value, and not as a pointer to a row or a location in the database.

**XML data**

Earlier, we saw that DB2 offers native XML manipulation and storage through the pureXML feature, available in all DB2 editions (including DB2 for z/OS) at no-charge. One of the components of pureXML is the XML data type, which enables DB2 to store XML documents in their native hierarchical format, rather than as text or mapped as a different data model.

While other database management products store XML data as plain text or through shredding, DB2 enables efficient search, retrieval and updates to well-formed XML documents that are natively stored in DB2 databases through pureXML. The access and manipulation of such XML data can be done through XQuery, SQL, or a combination of the two.

**Structured data types**

Structured types are user-defined types available on DB2 LUW that contain one or more attributes - which can be mapped to any supported data type, including LOBs and data types other than
those that are available with DB2 for Linux, Unix and Windows. (You can create new user-defined structured types and use them as columns in a table.) Structured types can be compared to objects / classes in an Object-Oriented model. It is possible to instantiate structured data type objects, use inheritance, create/use methods (actions), and much more. Structured data types are stored much like LOBs and XML documents are stored.

Storage for XML, LOB and structured data types

At one time, LOB, XML and structured data types were stored in a separate object within their tables and in the case of LOB data, LOB descriptors were used to reference the location of every value stored. (Figure 36 shows how LOB data has traditionally been stored on DB2 for Linux, Unix and Windows - as DB2 for z/OS uses a different mechanism, based on columnar tables.)

Figure 36. How LOB data is stored

Starting with DB2 LUW 9.7, is has been possible to store such data values inline. That means that, for smaller values, it is possible to have LOB, XML and structured data be stored together with other data in a table. Inlining is enabled by specifying the INLINE LENGTH clause when defining LOB, XML, and structured data type columns with CREATE and ALTER TABLE statements. Figure 37 shows how inlined LOB data values are stored.

Figure 37. How inlined LOB data values are stored

LOB and XML data can also be inlined in DB2 10 for z/OS.

More about DB2 Linux, UNIX, and Windows

DB2 Information Center

No description about the tools that are available with DB2 would be complete without a comment about the DB2 Information Center (also known as DB2 Infocenter.) Whenever you need specific information about a particular DB2 command, SQL statement, configuration parameter, registry variable, or you just want to know more about what's new in the latest DB2 release, how to migrate databases from older versions, or how to get started with a specific DB2 feature/functionality, you can usually find it in the DB2 Information Center. The DB2 Information Center is the one-stop shop
for any information you need about DB2. Its contents go far beyond the product documentation, and it’s a must-read for just about any situation you might encounter.

The DB2 Information Center can be accessed via the Internet, but it can also be installed locally during the DB2 installation process. Because DB2 exists in several different versions (which correspond to major releases), when you access the Information Center from the web, you need to make sure that the Information Center you are looking at corresponds to version/release of DB2 that you are using. (For example, make sure you are using Information Center 10.1 when working with DB2 10.1 databases). If you need to access the DB2 Information Center via the Internet but can't remember the link (and don't have the link saved as a bookmark) just search for the words “db2 Infocenter 10.1” with your favorite search engine. (Always include the version number).

There's an Information Center available for Data Studio as well; the link to that Information Center is: [http://publib.boulder.ibm.com/infocenter/dstudio/v3r1/index.jsp](http://publib.boulder.ibm.com/infocenter/dstudio/v3r1/index.jsp)

**First steps**

DB2 First Steps opens right after you install your DB2 product, and can help you get started using DB2 by:

- Guiding you to the DB2 documentation website (Information Center)
- Checking availability of new updates for your recently installed product
- Creating a sample database (populated with sample data) through db2samp
- Pointing you to the IBM Data Studio download website

**Figure 38** shows how the First Steps dialog looks immediately after it has been invoked.

**Figure 38. First Steps**
DB2 First Steps can also be accessed through the operating system’s menu, or by executing the db2fs command.

There’s also a great independent tool called “Technology Explorer for DB2” which can help you learning DB2 with command samples and more. It can be accessed through a link in the Resources section.

**Summary**

The goal of this tutorial was to introduce you to DB2 for Linux, UNIX, and Windows 10.1, its editions, tools and more. Now that you have reached the end of it, you should be able to:

- Describe the similarities and differences between the different DB2 product editions that are available and know which would be more suited for situations/scenarios you may encounter.
- Identify the different workloads that are typically seen in DB2 database environments, and know which product / functionality should be used to scale an OLTP or OLAP/data warehouse environment.
- Use the command-line tools that are available with DB2.
- Use IBM Data Studio to administer your databases and access/manipulate the data stored in them.
- Choose how to store non-traditional data in a DB2 database.
- Know where to go when you need help.

Since the DB2 10.1 Fundamentals certification exam (Exam 610) is focused on DB2 10.1, this tutorial is based on this specific version. As DB2 is evolving constantly, should you have questions regarding future releases of DB2, it’s recommended that you access the DB2 Information Center for the desired information.
Related topics

- Read the Preparation Guide for DB2 10.1 Fundamentals Exam 610 to learn in-depth information about each of the concepts presented in this tutorial. This guide is a compilation of topics from the DB2 10.1 Information Center documentation.
- IBM DB2 10.1 for LUW - Information Center is the official documentation for DB2 Linux, UNIX, and Windows 10.1. Use it to improve your experience with DB2, and get help whenever needed.
- IBM DB2 10 for z/OS - Information Center is the official documentation for DB2 10 for z/OS.
- IBM Data Studio 3.1.1 Information Center, the one-stop for help on Data Studio, to learn on how to do more with it.
- "Which distributed edition of DB2 10.1 is right for you?" (developerWorks, May 2012) — Learn more about the different DB2 editions, their features and licensing models.
- Learn more about specifics of DB2 LUW through the DB2 10.1 for LUW Manuals available in English and many other languages.
- "Migrating from DB2 Control Center to IBM Data Studio" (developerWorks, Jan 2012) — Understand what are the differences between Data Studio and Control Center, and how to map functionalities between them.
- "DB2 Express-C 10: Even more value" (developerWorks, Apr 2012) — Discover the new features brought to you on DB2 Express-C 10.
- Stay current with IDUG (International DB2 Users Group), the worldwide DB2 user community and join the IDUG DB2 Tech Conferences.
- Join the DB2Night Show community, watch the live episodes or see the replays, and get valuable information from those who are on top of DB2 technology.
- Read the "Oracle to DB2 Conversion Guide: Compatibility Made Easy" RedBook and see how smoothly you can convert Oracle databases to DB2.
- Learn from the experts on PlanetDB2 blog aggregator.
- Download DB2 Express-C 10, the most powerful and least restrictive free database product.

© Copyright IBM Corporation 2012
Trademarks
(www.ibm.com/developerworks/ibm/trademarks/)