

UNIVERSIDAD TECNOLÓGICA DE PANAMÁ
FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES

LICENCIATURA EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
BASE DE DATOS I

ESTUDIO DE CASO: DB2 UNIVERSAL DATABASE DE IBM

Profesora:
Ing. Maylin Chérigo

Integrantes:
Ho, Tommy 8-925-1168
Hurtado, Samuel 8-914-458
Luque, Julio 8-961-1454
Nelaton, Joy 8-902-1282

Grupo: VIL332

Fecha: 07 de julio de 2021

Índice del Trabajo

I. Introducción	4
II. Marco Teórico	5
1. Descripción general de DB2	6
2. Orígenes de DB2.....	7
3. Herramientas de diseño de bases de datos	8
4. Variaciones y extensiones de SQL.....	9
4.1. Características y funciones de XML	10
4.2. Soporte para tipos de datos	10
4.3. Funciones y métodos definidos por el usuario	11
4.4. Objetos de gran tamaño	12
4.5. Extensiones de índices y restricciones.....	12
4.6. Servicios Web	13
4.7. Colas de mensajes.....	14
5. Almacenamiento e indexación	14
5.1. Arquitectura de almacenamiento.....	14
5.2. Colas de memorias intermedias	15
5.3. Tablas, registros e índices	16
6. Agrupación multidimensional	17
6.1. Índices de bloque.....	19
6.2. Mapas de bloques	19
6.3. Consideraciones de diseño.....	19
6.4. Impacto sobre las técnicas existentes	20
7. Procesamiento y optimización de consultas.....	20
7.1. Métodos de acceso	22
7.2. Operaciones de reunión, agregación y de conjuntos.....	23
7.3. Soporte para el procesamiento de SQL complejo	24
7.4. Procesamiento de consultas en multiprocesadores	24
7.5. Optimización de consultas	24
8. Tablas de consultas materializadas	25
8.1. Encaminamiento de consultas a MQTs	25
8.2. Mantenimiento de MQTs.....	26

9. Características autónomas de DB2	27
9.1. Configuración	27
9.2. Optimización	28
10. Herramientas y utilidades	29
10.1. Utilidades	30
11. Control de concurrencia y recuperación	30
11.1. Concurrencia y aislamiento	30
11.2. Compromiso y retroceso	31
11.3 Registro histórico y recuperación	32
12. Arquitectura del sistema	32
13. Réplica, distribución y datos externos	33
14. Características de inteligencia de negocio	34
III. Conclusiones	35
IV. Referencias Bibliográficas	36
V. Anexos	37

Índice de Figuras

Figure 1 - Conjunto de productos DB2	7
Figure 2 - Creación y empleo de un tipo de dato distinto, nombrado "euro"	10
Figure 3 - Declaración de un tipo de dato estructurado nombrado "t_departamento"	11
Figure 4 - Ejemplo de una FDU.....	12
Figure 5 - Extensión de un índice espacial.....	13
Figure 6 - DB2 como un cliente en un servicio web nombrado "ObtenerCotización" que devuelve un valor de cotización para cada id_acción que aparece en la tabla cartera	13
Figure 7 - Ejemplo de grupo de nodos con Contenedores	15
Figure 8 - Vista lógica de la página y del índice	15
Figure 9 - Índice como páginas con registros índice y punteros a páginas hijas y hermanas	16
Figure 10 - Cubo de agrupación multidimensional	17
Figure 11 - Índice dimensional de bloques	18
Figure 12 - Mapa de bloques	19
Figure 13 - Árbol de consulta DB2	21
Figure 14 - Procesamiento de consultas DB2	23
Figure 15 - Función create tabla para consultas materializads en DB2	25
Figure 16 - Flujo de reencaminamiento y optimización.....	26
Figure 17 - Técnicas de Mantenimiento	27
Figure 18 - Centro de control de DB2.....	29
Figure 19 - Estructura del bloque, sus objetivos e interpretación	31
Figure 20 - Procesos o hebras en un servidor DB2	33

I. Introducción

Dentro del cumulo de sistemas de gestión de bases de datos relacionales podemos destacar el aporte de DB2, producto estrella de bases de datos de IBM, interesado desde 1990 en brindar un servidor de base de datos universal, capaz de ser ejecutado y mantenido en sistemas operativos basados en Linux, UNIX y Windows/NT.

A simple vista DB2 está diseñado para almacenar, analizar y recuperar los datos de manera eficiente. La capacidad de DB2 es amplia, incluyendo el soporte de características que se encuentran orientadas a multitud de objetos y estructuras no relacionales con XML. He aquí la razón de hacer hincapié en el desglose de los conceptos claves relacionados a DB2, sus orígenes, características y usos posibles que añaden valor a una organización o propósito específico.

II. Marco Teórico

Sistema de gestión de base de datos (SGBD) - Database management system (DBMS). es una agrupación de programas que sirven para definir, construir y manipular una base de datos.

Si el sistema soporta bases de datos relacionales se llama Sistema de gestion de bases de datos relacionales (SGBDR) - Relational Database Management System (RDBMS). Estos sistemas representan a la base de datos como una colección de tablas. Estas bases de datos suelen utilizar SQL como lenguaje de consultas de alto nivel.

Tareas llevadas a cabo por un SGDBR:

- Definir una base de datos: consiste en especificar los tipos de datos, estructuras y restricciones para los datos que se almacenarán.
- Construir una base de datos: es el proceso de almacenar los datos sobre algún medio de almacenamiento.
- Manipular una base de datos: incluye funciones como consulta, actualización, etc. de bases de datos.

1. Descripción general de DB2

DB2 es un conjunto de productos de servidores de base de datos relacionales desarrollado por IBM. DB2 es conocido ampliamente por su robustez y valores agregados tales como: inteligencia de negocio, integración de información, gestión de contenidos y universalidad. Siendo este último valor, la universalidad, el elemento de mayor utilidad al momento de realizar comparativas.

La universalidad se da gracias al motor *Cloudscape*, basado en Java, gracias a este, DB2 es capaz de ser incluido fácilmente en servidores de aplicaciones y otras aplicaciones. Estas aplicaciones pueden migrarse sin problemas de plataformas de gama baja a grandes servidores debido a la portabilidad de los interfaces y servicios de DB2. El soporte de DB2 incluye un gran número de plataformas de servidor y de sistema operativo, por consiguiente, el motor de base de datos DB2 consiste en cuatro bases de código diferentes: (1) Linux, Unix y Windows, (2) z/OS, (3) VM, y (4) OS/400.

Cabe señalar que todas las bases de código soportan un subconjunto común de lenguaje de definición de datos, de SQL y de las interfaces de administración. Sin embargo, los motores presentan características algo diversas debido a los orígenes de cada plataforma.

- Servidores de bases de datos
 - DB2 UDB para Linux, Unix, Windows
 - DB2 UDB para z/OS
 - DB2 UDB para OS/400
 - DB2 para VM/VSE
- Inteligencia de negocio
 - DB2 Data Warehouse Edition
 - DB2 OLAP Server
 - DB2 Alphablox
 - DB2 CubeViews
 - DB2 Intelligent Miner
 - DB2 Query Patroller
- Integración de datos
 - DB2 Information Integrator
 - DB2 Replication
 - DB2 Connect
 - Omnifind (para Enterprise Search)
- Gestión de contenidos
 - DB2 Content Manager
 - IBM Enterprise Content Manager
- Desarrollo de aplicaciones
 - IBM Rational Application Developer Studio
 - DB2 Forms para z/OS
 - QMF
- Herramientas de gestión de bases de datos
 - DB2 Control Center
 - DB2 Admin Tool para z/OS
 - DB2 Performance Expert
 - DB2 Query Patroller
 - DB2 Visual Explain
- Acceso móvil a datos
 - DB2 Cloudscape
 - DB2e (EveryPlace)

Figure 1 - Conjunto de productos DB2

2. Orígenes de DB2

El origen de DB2 se remonta al proyecto System R en el Centro de investigación de Almadén (Almaden Research Center) de IBM (entonces denominado Laboratorio de investigación de San José, IBM San Jose Research Laboratory).

La primera comercialización de DB2 se llevó a cabo en 1984 sobre la plataforma mainframe de IBM, seguido tiempo después por versiones para otras mainframes.

Desde el año 1990, IBM decidió desarrollar un servidor DB2 de base de datos universal (UDB), que puede ejecutarse en cualquier sistema operativo autorizado como Linux, UNIX y Windows.

A través del tiempo IBM ha ido perfeccionando cada vez más a DB2 en áreas tales como:

- Procesamiento de transacciones
- Procesamiento y optimización de consultas
- Soporte para bases de datos activas
- Técnicas avanzadas de consultas
- Almacenes de datos
- Soporte del modelo relacional orientado a objetos (ADTs, UDFs)

El motor de base de datos DB2 consiste de cuatro bases de código diferentes:

- 1) Linux, Unix, y Windows
- 2) z/OS
- 3) VM
- 4) OS/400

Cada una de estas le da soporte de administración de datos a las diversas plataformas de servidor y de sistemas operativos de IBM. Este trabajo escrito va a estar centrado en el motor DB2 Universal Database (UDB) para Linux, Unix y Windows. La versión 8.2 en particular de la base de datos universal trae con sí varias características que mejoran diversos aspectos del motor. Entre estas características están:

- **Ampliabilidad:** sus dos características importantes son las tablas de consultas materializadas y las agrupaciones multidimensionales.
- **Disponibilidad:** se describen las mejoras en las áreas de utilidades en línea y las réplicas.

Además de estas dos, hay varias otras características adicionales que se amplían, como las características autónomas asesor de diseño, el ajuste y supervisión automáticos de memoria, la robustez del motor, etc.

3. Herramientas de diseño de bases de datos

La base de datos DB2 puede ser diseñada por diversas herramientas de diseño. Las de modelado de datos, como *ERWin* y *Rational Rose*, permiten generar sintaxis LDD específicamente para DB2. Otra herramienta útil es el *UML Data Modeler* de Rational Rose para poder generar instrucciones ***create distinct type*** para definir tipos por el usuario y usarlos en definiciones de columnas. Otro aspecto relevante es que la mayoría de las herramientas de diseño de bases de datos soportan la ingeniería en inversa que lee las tablas del catálogo de DB2, así construyendo un diseño lógico para poder ser manipulado posteriormente.

DB2 proporciona diversos constructores SQL para dar soporte a las diferentes características lógicas en una base de datos. Aquellos constructores son:

- Restricciones
- Disparadores
- Recursión

Además de las mencionadas, DB2 también proporciona características físicas como:

- Espacios de tablas
- Colas de memoria intermedia
- Particionamiento (mediante instrucciones SQL)

Está la herramienta Centro de control de DB2 para emitir instrucciones LDD apropiadas que sean necesarias, y la herramienta *db2look* permite obtener un conjunto completo de instrucciones LDD para una base de datos (espacios de tablas, tablas, índices, restricciones, disparadores, funciones de reproducción).

También están incluidas herramientas de diseño y administración en el Centro de control de DB2. Se proporcionan la vista de árbol de un servidor, sus respectivas bases de datos, tablas vistas y demás objetos. Los usuarios pueden definir nuevos objetos y también pueden crear consultas SQL y visualizar los resultados de dichas consultas. Toda la familia de DB2 soporta el Centro de control para la definición de bases de datos y sus herramientas relacionadas. También se da la facilidad de utilizar módulos predeterminados para el desarrollo de aplicaciones en *IBM Rational Application Development* y *Microsoft Visual Studio*.

4. Variaciones y extensiones de SQL

DB2 es un sistema amplio de soporta de características SQL. Sus características y sintaxis han proporcionado la base para los estándares SQL-92 (la tercera revisión del

lenguaje SQL) o SQL-99 (la extensión de SQL.92). Se detalla a continuación las características XML del modelo relacional orientado a objetos.

4.1. Características y funciones de XML

- ***xmlelement***: se utiliza para construir una etiqueta elemento con nombre dado.
- ***xmlattributes***: construye el conjunto de atributos.
- ***xmlforest***: se construye una secuencia de elementos XML a partir de los argumentos.
- ***xmlconcat***: se utiliza para la concatenación de un número variable de argumentos XML.
- ***xmlserialize***: se da una versión del argumento serializado orientado a caracteres.
- ***xmlagg***: la concatenación de valores XML es devuelto.
- ***xml2clob***: se construye la representación XML del tipo objeto de gran tamaño de caracteres (conocido como clob); este se puede recuperar mediante aplicaciones SQL.

4.2. Soporte para tipos de datos

DB2 tiene soporte para tipos de datos según su naturaleza, los cuales se destacan:

- ***Distintos***: basados en tipos de datos incorporados en DB2 pero con semánticas adicionales o alternativas definidas por el usuario.

```
create distinct type euro as decimal(9,2)

select producto from ventas_Europa
where precio > euro(1000)
```

Figure 2 - Creación y empleo de un tipo de dato distinto, nombrado "euro"

- ***Estructurados***: se caracterizan por ser objetos complejos formados por dos o más atributos.

```

create type t_departamento as
  (nombredept varchar(32),
   directordept varchar(32),
   número integer)
mode db2/sql

create type t_punto as
  (coord_x float,
   coord_y float)
mode db2/sql

```

Figure 3 - Declaración de un tipo de dato estructurado nombrado "t_departamento"

4.3. Funciones y métodos definidos por el usuario

Una característica importante a identificar es la posibilidad que los usuarios pueden definir sus propias funciones y métodos. Estas funciones pueden ser incluidas posteriormente en instrucciones y consultas SQL. Las funciones pueden generar escalares (*único atributo*) o tablas (*fila multiatributo*) como resultado. Los usuarios pueden definir funciones (escalares o de tablas) mediante el uso de la instrucción *create function*. Pueden escribir las funciones en lenguajes de programación tales como C y Java o lenguajes de guiones tales como REXX y PERL.

Las funciones definidas por el usuario (FDU) pueden operar en los siguientes modos:

- ***Separado (fenced)***: Las funciones se ejecutan mediante una hebra separada en su propio espacio de dirección.
- ***Compartido (unfenced)***: Se permite al agente de procesamiento de la base de datos ejecutar la función en el espacio de direcciones del servidor.

Las FDU pueden definir un área de trabajo donde pueden mantener variables locales y estáticas en invocaciones diferentes. Por tanto, las FDU pueden realizar manipulaciones complejas de las filas intermedias que son su entrada.

```

create function db2gse.GsegeFilterDist (
    operación integer, g1XMín double, g1XMáx double,
    g1YMín double, g1YMáx double, dist double,
    g2XMín double, g2XMáx double, g2YMín double,
    g2YMáx double )
returns integer
specific db2gse.GsegeFilterDist
external name 'db2gsefn!gsegeFilterDist'
language C
parameter style db2 sql
deterministic
not fenced
threadsafe
called on null input
no sql
no external action
no scratchpad
no final call
allow parallel
no dbinfo;

```

Figure 4 - Ejemplo de una FDU

4.4. Objetos de gran

tamaño

Las nuevas aplicaciones de las bases de datos requieren la manipulación de datos no estructurados (texto, imágenes, vídeo, etc). DB2 brinda soporte a estos proporcionando tres tipos de objetos de gran tamaño (*LOB, Large Object*) distintos.

Los objetos de gran tamaño en DB2 son:

- ***Objetos en binario*** (*Binary Large Objects, BLOBs*),
- ***Objetos de caracteres de un único byte*** (*Character Large Objects, CLOBs*)
- ***Objetos de caracteres de dos bytes*** (*Double Byte Character Large Objects, dbclobs*).

Ha de destacarse que cada LOB puede ocupar hasta 2 gigabytes.

DB2 organiza estos LOBs como objetos separados con cada fila en la tabla manteniendo punteros a sus LOBs correspondientes. Los usuarios pueden registrar FDUs que manipulen estos LOBs según los requisitos de la aplicación.

4.5. Extensiones de índices y restricciones

DB2 proporciona un constructor ***create index extension*** que ayuda a crear índices sobre atributos con tipos de datos estructurados mediante la generación de claves a partir de los tipos de datos estructurados.

```
create index extension db2gse.indice_espacial(
    gS1 double, gS2 double, gS3 double)
from source key(geometry db2gse.ST_Geometry)
generate key using
    db2gse.GseGridIdxKeyGen(geometry..srid,
    geometry..xMín, geometry..xMáx,
    geometry..yMín, geometry..yMáx,
    gS1, gS2, gS3)

with target key(srsId integer,
    lvl integer, gX integer, gY integer, xMín double,
    xMáx double, yMín double, yMáx double)
search methods <condiciones> <acciones>
```

Figure 5 - Extensión de un índice espacial

DB2 también proporciona un rico conjunto de características de verificación de restricciones para imponer la semántica de los objetos tales como *unicidad*, *validez* y *herencia*.

4.6. Servicios Web

DB2 permite integrar servicios Web como productor o consumidor. Es posible definir un servicio Web para invocar DB2 usando instrucciones SQL. La llamada al servicio Web resultante es procesada por un motor de servicios Web incorporado en DB2, que genera la correspondiente respuesta SOAP.

La siguiente instrucción SQL muestra cómo DB2 actúa como un cliente de un servicio Web.

```
select id_acción, ObtenerCotización(id_acción)
from cartera
```

Figure 6 - DB2 como un cliente en un servicio web nombrado "ObtenerCotización" que devuelve un valor de cotización para cada *id_acción* que aparece en la tabla *cartera*

4.7. Colas de mensajes

Mediante la definición de las FDU apropiadas DB2 también soporta el producto de IBM *Websphere MQ*, tanto para interfaces de lectura como de escritura. Dichas FDU pueden incluirse en instrucciones SQL para la lectura o escritura sobre colas de mensajes.

5. Almacenamiento e indexación

La arquitectura de almacenamiento e indexación de DB2 está formada por la capa de sistema de ficheros o gestión de disco, los servicios para gestionar las memorias intermedias, objetos de datos tales como tablas, LOBs, objetos índices y gestores de concurrencia y recuperación.

5.1. Arquitectura de almacenamiento

DB2 proporciona abstracciones de almacenamiento para gestionar tablas de base de datos lógicas en entornos multinodo (paralelo) y multidisco. Se pueden definir grupos de nodos para soportar la división de la tabla en conjuntos especificados de nodos en un sistema multinodo. Esto permite flexibilidad al asignar particiones de tabla a nodos diferentes en un sistema. Por ejemplo, las tablas de gran tamaño se pueden dividir entre todos los nodos en un sistema, mientras que las tablas pequeñas pueden residir en un único nodo. Dentro de un nodo, DB2 usa espacios de tablas para organizar las tablas. Un espacio de tablas consiste en uno o más contenedores que son referencias a directorios, dispositivos o archivos. Un espacio de tablas puede contener cero o más objetos de base de datos tales como tablas, índices o LOBs.

DB2 permite al administrador crear tantos espacios de tablas gestionados por el sistema como por el SGBD.

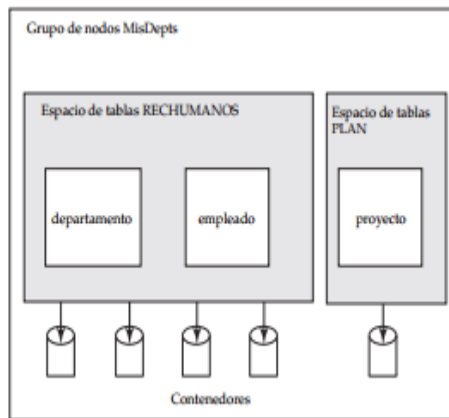


Figure 7 - Ejemplo de grupo de nodos con Contenedores

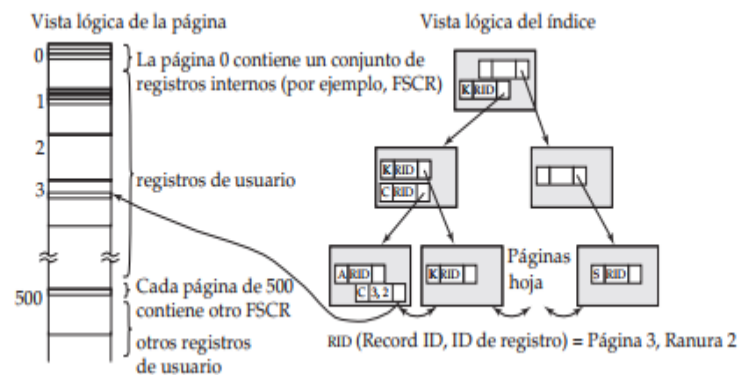


Figure 8 - Vista lógica de la página y del índice

5.2. Colas de memorias intermedias

Se puede asociar una o varias memorias intermedias con cada espacio de tablas para gestionar diferentes objetos tales como datos e índices. Una memoria intermedia es un área de datos compartida que mantiene copias de objetos en memoria. Estos objetos habitualmente están organizados en páginas para gestionar la memoria intermedia. DB2 permite la definición de memorias intermedias usando instrucciones SQL. La versión 8 de DB2 incluye la posibilidad de aumentar o disminuir el tamaño de las memorias intermedias de forma interactiva, o bien automáticamente si se selecciona la opción *automatic* en parámetro de configuración de memorias intermedias.

```
create          bufferpool          <cola-mem-intermedia>          ....  
alter bufferpool <cola-mem-intermedia> size <n>
```

DB2 también soporta la preextracción y escrituras asíncronas mediante el uso de hebras separadas.

5.3. Tablas, registros e índices

DB2 organiza los datos relacionales como registros en las páginas. Cada página consiste en un conjunto de registros (tanto registros de datos del usuario como registros especiales del sistema). La página cero de la tabla contiene registros del sistema especiales sobre la tabla y su estado. DB2 usa un registro del mapa de espacio denominado registro de control de espacio libre (*Free Space Control Record, FSCR*) para encontrar el espacio libre en la tabla. El registro FSCR normalmente contiene un mapa de espacio de 500 páginas. Una entrada FSCR consiste en unos pocos bits que proporcionan una indicación aproximada del porcentaje de espacio libre en la página. El algoritmo de inserción o actualización debe validar las entradas del FSCR realizando una comprobación física del espacio disponible en la página.

Los índices también se organizan como páginas que contienen registros índice y punteros a páginas hijas y hermanas. DB2 proporciona soporte para los mecanismos de índices de árbol B+. El índice de árbol B+ contiene páginas internas y páginas hoja. Los índices contienen punteros bidireccionales en el nivel hoja para soportar exploraciones hacia delante y hacia atrás.

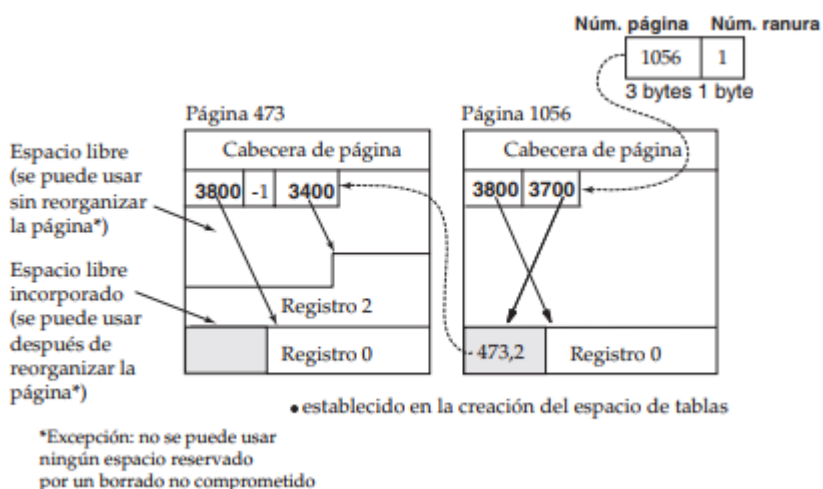


Figure 9 - Índice como páginas con registros índice y punteros a páginas hijas y hermanas

Las páginas hoja contienen entradas de índice que apuntan a los registros de la tabla. Cada registro de una tabla se puede identificar unívocamente usando su información de página y de ranura, que se denominan identificador de registro o RID

create unique index I1 on T1 (C1) include (C2)

Las columnas del índice incluidas permiten a DB2 extender el uso de las técnicas de procesamiento “sólo con índices” siempre que sea posible. Se pueden usar directivas adicionales tales como *minpctused* y *pctfree* para controlar la unión de páginas de índices y su asignación de espacio inicial.

DB2 soporta distintos tamaños de página tales como 4 KB, 8 KB, 16 KB y 32 KB. Sin embargo, cada página puede contener solamente 255 registros de usuario. Los tamaños de página mayores son útiles en aplicaciones tales como almacén de datos donde la tabla contiene muchas columnas. Los tamaños de página menores son útiles para datos operacionales con frecuentes actualizaciones.

6. Agrupación multidimensional

Con la agrupación multidimensional, una tabla DB2 puede crearse especificando una o varias claves como dimensiones para las que se agruparán los datos de la tabla. Existe una nueva cláusula denominada ***organize by dimensions*** para este cometido.

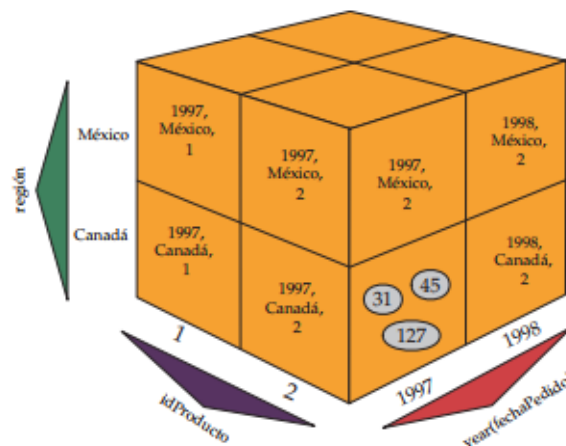


Figure 10 - Cubo de agrupación multidimensional

```

create table ventas(idAlmacén int,
  fechaPedido date,
  fechaEnvío date,
  fechaRecepción date,
  región int,
  idProducto int,
  precio float
  añoPedido int generated always as year(fechaPedido))
organized by dimensions (región, añoPedido, idProducto)

```

Cada combinación única de valores dimensionales forma una “celda” lógica que está físicamente organizada como bloques de páginas, donde un bloque es un conjunto de páginas consecutivas en disco.

El conjunto de los bloques que contienen páginas con datos que verifican un cierto valor de clave de uno de los índices de bloque de dimensión se denomina “banda”. Cada página de la tabla forma parte de exactamente un bloque, y todos los bloques de la tabla están formados por el mismo número de páginas, que se denomina tamaño de bloque. DB2 asocia el tamaño de bloque con el tamaño de extensión del espacio de tablas, de forma que los límites de los bloques estén alineados con los límites de las extensiones.

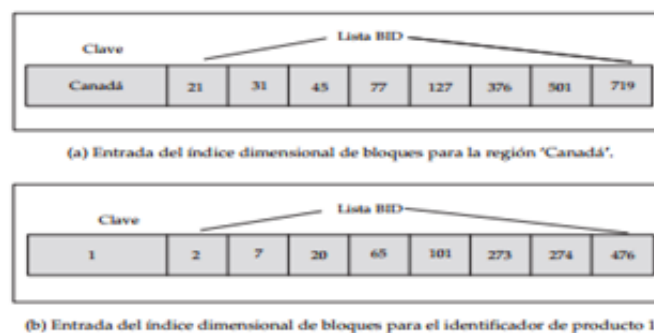


Figure 11 - Índice dimensional de bloques

6.1. Índices de bloque

Los índices de bloques son mucho más pequeños que los índices RID, y necesitan actualizarse solamente cuando se añade un nuevo bloque a la celda, o cuando se vacían y eliminan bloques existentes en la celda. Una banda, o el conjunto de bloques que contienen páginas con un valor clave particular en una dimensión, se representan en el índice dimensional de bloques asociado mediante una lista de BIDs para ese valor clave.

6.2. Mapas de bloques

La tabla también tiene asociado un mapa de bloques. Este mapa registra el estado de cada bloque de la tabla. Un bloque puede estar en diferentes estados tales como

in use, free, loaded y requiring constraint enforcement.

La capa de gestión de datos usa el estado del bloque para determinar diversas opciones de procesamiento. La *Figura 12* muestra un ejemplo de mapa de bloques de una tabla.

El elemento 0 en el mapa de bloques representa el bloque 0 en el diagrama MDC de la tabla. El estado de disponibilidad del bloque es “U”, que indica que está en uso.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
U	U	F	F	U	U	U	L	U	F	F	U	C	F	F	U	U	F	L	...

Figure 12 - Mapa de bloques

6.3. Consideraciones de diseño

Un aspecto crucial de las MDCs es la elección del conjunto correcto de dimensiones para agrupar una tabla y el tamaño correcto de bloque para minimizar el uso de espacio. Si las dimensiones y el tamaño de bloque se eligen apropiadamente, los beneficios de la agrupación se traducen en significativas ventajas de rendimiento y mantenimiento. Por

otra parte, si se eligen incorrectamente, el rendimiento se puede degradar y el uso de espacio puede ser sensiblemente peor. Hay muchas técnicas de ajuste que se pueden explotar para organizar la tabla.

6.4. Impacto sobre las técnicas existentes

Todas las características existentes tales como índices RID secundarios, restricciones, disparadores, vistas materializadas definidas y opciones de procesamiento de consultas están disponibles para las tablas MDC. Por consiguiente, las tablas MDC se comportan exactamente como las tablas clásicas excepto por sus aspectos mejorados de agrupación y procesamiento.

7. Procesamiento y optimización de consultas

El compilador de consultas de DB2 transforma las consultas en un árbol de operaciones. DB2 usa entonces el árbol de operadores de la consulta en tiempo de ejecución para el procesamiento. DB2 soporta un rico conjunto de operadores de consulta que permiten considerar mejores estrategias de procesamiento y proporcionan flexibilidad en la ejecución de consultas complejas.

-- 'Consulta TPCD de volumen de proveedor local (Q5)';
select *nac_nombre*, **sum**(*lp_precioextendido**(1-*lp_descuento*)) **as** *ingresos*
from *tpcd.cliente*, *tpcd.pedidos*, *tpcd.linea_pedido*,
tpcd.proveedor, *tpcd.nación*, *tpcd.región*
where *cli_clave_cliente* = *ped_clave_cliente* **and**
ped_clave_pedido = *lp_clave_pedido* **and**
lp_clave_proveedor = *prv_clave_proveedor* **and**
cli_clave_nación = *prv_clave_nación* **and**
prv_clave_nación = *nac_clave_nación* **and**
nac_clave_región = *reg_clave_región* **and**
reg_nombre = 'MIDDLE EAST' **and**
ped_fecha_pedido >= **date**('1995-01-01') **and**
ped_fecha_pedido < **date**('1995-01-01') + 1 **year**
group by *nac_nombre*
order by *ingresos* **dec**;

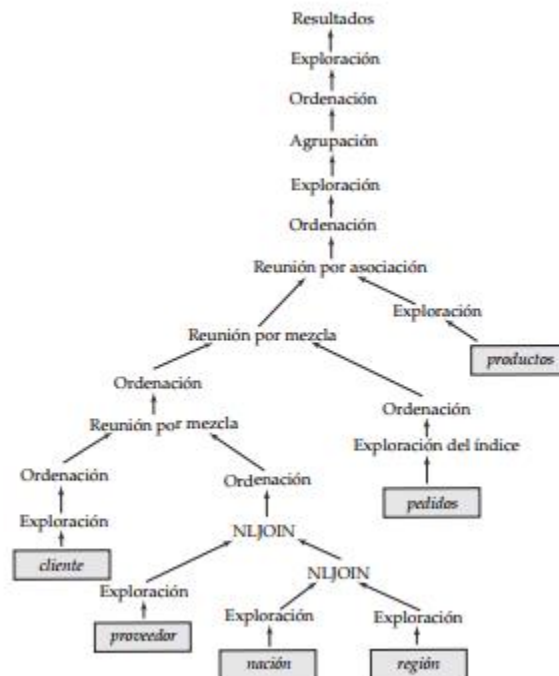


Figure 13 - Árbol de consulta DB2

DB2 transforma todas las consultas e instrucciones SQL, sin importar lo complejas que sean, en un árbol de consulta. La base u operadores hoja del árbol de consulta manipulan los registros en tablas de base de datos. Estas operaciones también se denominan métodos de acceso. Las operaciones intermedias del árbol incluyen operaciones del álgebra relacional tales como reuniones, operaciones de conjuntos y agregaciones. La raíz del árbol produce los resultados de la consulta o instrucción SQL.

7.1. Métodos de acceso

DB2 soporta un conjunto detallado de métodos de acceso sobre tablas relacionales:

- **Exploración de tabla:** Con este método, el más básico, se accede a todos los registros en la tabla página por página.
- **Exploración de índice:** DB2 usa un índice para seleccionar los registros específicos que satisfacen la consulta. Accede a los registros usando los RIDs en el índice.
- **Exploración de índice de bloques:** Es un nuevo método de acceso para tablas MDC. Se usa uno de los índices de bloque para explorar un conjunto específico de bloques de datos. DB2 accede y procesa los bloques seleccionados mediante operaciones de exploración de tablas de bloques.
- **Sólo con el índice:** Este tipo de exploración se usa cuando el índice contiene todos los atributos que requiere la consulta. Por ello es suficiente una exploración de las entradas de índice y no hay necesidad de extraer los registros.
- **Lista de preextracción:** Este método de acceso es una buena elección para una exploración de índices no agrupada con un número significativo de RIDs. DB2 recoge los RIDs de los registros relevantes usando una exploración de índices, después ordena los RIDs por el número de página y finalmente realiza una extracción de los registros de forma ordenada desde las páginas de datos.
- **Conjunción de índices de bloques y de registros:** DB2 usa este método cuando determina que se puede usar más de un índice para restringir el número de registros satisfactorios en una tabla base. Procesa el índice más selectivo para generar una lista de RIDs o RIDs.

- **Ordenación de índices de bloque y de registro:** Esta estrategia es una buena elección si se pueden usar dos o más índices para satisfacer los predicados de la consulta que se combinan usando la operación OR. DB2 elimina los BIDs o RIDs duplicados realizando una ordenación y después extrae el conjunto de registros resultante. La disyunción de índices se ha extendido para considerar combinaciones de índices de bloque y RIDs.

7.2. Operaciones de reunión, agregación y de conjuntos

DB2 soporta una serie de técnicas para las operaciones de reunión, agregación y de conjuntos. Para la reunión DB2 puede elegir entre técnicas de bucles anidados, mezcla-ordenación y de asociación. Para describir las operaciones binarias de reuniones y de conjuntos se usará la notación de las tablas “externas” e “internas” para distinguir los dos flujos de entrada. La técnica de bucles anidados es útil si la tabla interna es muy pequeña o se puede acceder usando un índice sobre un predicado de reunión. Las técnicas de reunión de mezcla-ordenación y reunión por asociación son útiles para reuniones que involucren tablas internas y externas grandes.

DB2 implementa las operaciones de conjuntos mediante el uso de técnicas de ordenación y mezcla. La técnica de mezcla elimina los duplicados en el caso de la unión mientras que los no duplicados se eliminan en el caso de intersección. DB2 también soporta operaciones de reunión externa de todas las clases.

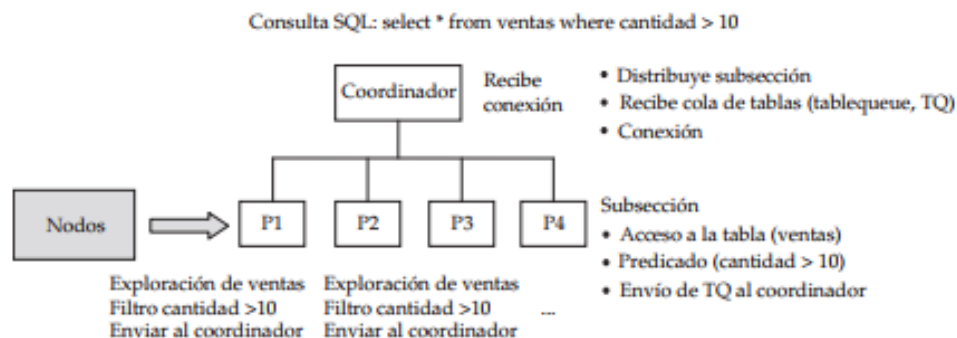


Figure 14 - Procesamiento de consultas DB2

7.3. Soporte para el procesamiento de SQL complejo

Uno de los aspectos más importantes de DB2 es que usa la infraestructura de procesamiento de la consulta de forma extensible para soportar operaciones SQL complejas. Las operaciones SQL complejas incluyen soporte para subconsultas profundamente anidadas y correlacionadas, así como restricciones, integridad referencial y disparadores. Como la mayor parte de estas acciones están incluidas dentro del plan de consulta, DB2 está preparado para soportar una gran cantidad de restricciones y acciones.

7.4. Procesamiento de consultas en multiprocesadores

DB2 extiende el conjunto base de operaciones de consulta con primitivas de intercambio de datos y control para soportar los modos SMP, MPP y SMP por agrupaciones del procesamiento de consultas. DB2 usa una abstracción tabla-cola para el intercambio de datos entre hebras sobre distintos nodos o sobre el mismo nodo. La tabla-cola es una memoria intermedia que redirige los datos a receptores apropiados mediante el uso de métodos de difusión, uno a uno o multidifusión dirigida. Las operaciones de control crean hebras y coordinan la operación de distintos procesos y hebras.

7.5. Optimización de consultas

El compilador de consultas de DB2 utiliza una representación interna de la consulta, denominada *Query Graph Model* (QGM, modelo de grafos de consultas) con el fin de ejecutar transformaciones y optimizaciones. Después de analizar la instrucción SQL, DB2 ejecuta transformaciones semánticas sobre el QGM para hacer cumplir las restricciones, integridad referencial y los disparadores. El resultado de estas transformaciones es un QGM mejorado. Seguidamente DB2 intenta ejecutar transformaciones de reescritura de la consulta que se consideran beneficiosas en la mayoría de las consultas.

8. Tablas de consultas materializadas

La utilidad de las tablas de consultas materializadas recalca en su habilidad de poder mantener una copia persistente de datos de la vista para poder agilizar el procesamiento de consultas. Estas vistas materializadas están disponibles a partir de la versión 8 de DB2 para los sistemas operativos de Linux, Unix y Windows, al igual que las plataformas z / OS. Cualquier definición que se proporcionado por el usuario sobre una vista en particular se puede materializar.

En DB2, estas vistas materializadas se denominan tablas de consultas materializadas (materialized query tables – MQTs). Las MQTs se especifican usando una instrucción **create table**. La *Figura 15* muestra esta característica.

```
create table empleado_departamento(id_departamento integer, id_empleado integer,  
                                     nombre_empleado varchar(100), id_jefe integer) as  
select id_departamento, id_empleado, nombre_empleado, id_jefe  
from empleado, departamento  
data initially deferred  
refresh immediate -- (o deferred)  
maintained by user -- (o system)
```

Figure 15 - Función create tabla para consultas materializads en DB2

En DB2 las MQTs pueden referirse a otras MQTs para crear un árbol o un bosque de vistas dependientes. Las MQTs se pueden ampliar y dividirse en un entorno MPP, lo que resulta en claves de agrupación MDC. Las MQTs resultan de máximo valor si el motor de la base de datos es capaz de encaminar perfectamente consultas hacia la MQT, y también cuando el motor de la base de datos puede mantenerlas eficientemente siempre que sea posible.

8.1. Encaminamiento de consultas a MQTs

Los MQTs benefician de la infraestructura del compilador de consultas DB2, y que es ideal para aprovecharlo al máximo. El modelo interno del QGM permite al compilador comparar la consulta con las definiciones de MQTs disponibles y elegir las MQTS apropiadas. Después de la comparación, el compilador considera varias opciones de

optimización. Entre ellas se encuentran tanto la consulta base como las versiones de reencaminamiento de MQTs apropiadas. El optimizador itera a través de estas opciones antes de elegir la versión de ejecución óptima. El flujo completo de reencaminamiento y optimización se muestra en la *Figura 16*.

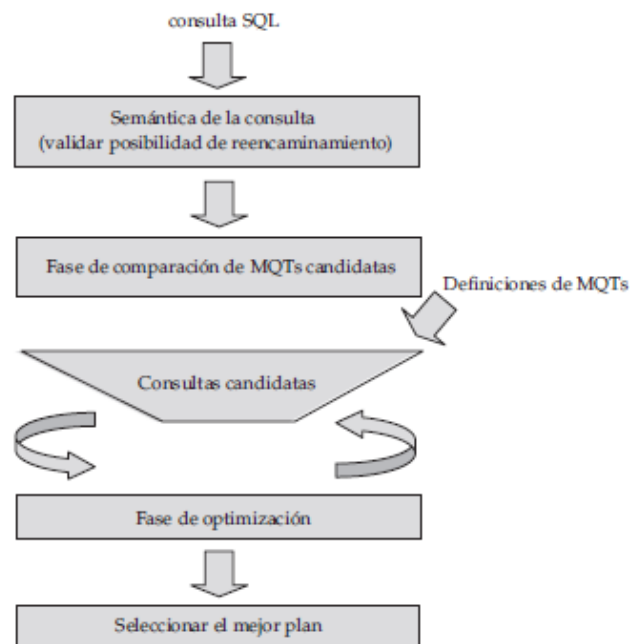


Figure 16 - Flujo de reencaminamiento y optimización

8.2. Mantenimiento de MQTs

El motor de la base de datos es lo que va a determinar la utilidad de las MQTs en base a las técnicas de mantenimiento que proporcione. Las dos dimensiones que determinan la eficiencia de estas técnicas de mantenimiento son el tiempo y el costo.

Con respecto al tiempo, están las opciones *inmediato* y *diferido*. DB2 soporta ambas opciones. La opción *inmediato* crea disparadores internos que se compilan en las instrucciones de inserción, actualización o borrado de los objetos fuente para procesar las actualizaciones de las MQTs dependientes. Con la opción *diferido*, las tablas actualizadas pasan al modo integridad y se ejecutan con la instrucción **refresh** de manera obligatoria para realizar la actualización.

Con respecto a la dimensión del costo, se puede elegir entre las opciones *incremental* o *completo*. El mantenimiento *incremental* sólo se utiliza para mantenimiento de filas que han sido actualizadas recientemente. En cambio, el mantenimiento *completo* actualizar toda la MQT desde las fuentes. En la *Figura 17* se muestra la relación de las opciones de mantenimiento y sus opciones correspondientes.

load from datosnuevos.txt of type del
insert into empleado;

refresh table empleado_departamento

<i>Elecciones</i>	<i>Incremental</i>	<i>Completa</i>
Inmediata	Sí, Después de insert/update/delete	Normalmente no
Diferida	Sí, Después de la carga	Sí

Figure 17 - Técnicas de Mantenimiento

9. Características autónomas de DB2

La versión 8.2 en particular de DB2 UDB proporciona diversas características para apoyar en el la simplificación de diseño y administración de bases de datos. Tiene un conjunto de técnicas que permiten la facilidad de administración de la misma y la reducción de dependencias externas con respecto a la seguridad tanto externa como interna con la carga del sistema y sus elementos. Las opciones que se explicarán son la de configuración y optimización.

9.1. Configuración

DB2 proporciona soporte de ajuste automático a diversos parámetros de configuración de memoria y del sistema. Los parámetros de tamaño de la memoria intermedia y del

montículo de ordenación se pueden especificar como automáticas. De tal manera, DB2 supervisa al sistema para poder aumentar o disminuir lentamente el tamaño de estas funciones con respecto a las características de la carga de trabajo.

9.2. Optimización

Las estructuras de datos auxiliares y las características de organización de datos son aspectos importantes para mejorar el rendimiento de las bases de datos en DB2. Anteriormente, el administrador de la base de datos tenía que basarse en su propia experiencia y pautas conocidas para elegir los índices correspondientes (MQTs), claves de división y claves de agrupación significativos.

La versión 8.2 de DB2 proporciona un Asesor de diseño (*Design Advisor*) que da sugerencias a las características que se desean configurar. La sintaxis para invocar al asistente es:

db2advls -d <nombre BD> -i <fichero carga de trabajo> -m MICP

El parámetro “-m” permite al usuario especificar lo siguiente:

- M – Tablas de consultas materializadas
- I – Índices
- C – Agrupaciones (MDC)
- P – Selección de claves de división

El asesor de diseño utilizar el poder del marco de optimización de consultas de DB2 para recomendarle al usuario las mejores opciones. Se utiliza una carga de trabajo y restricciones de tamaño y tiempo como parámetro de entrada. Al utilizar el marco de optimización de DB2, se está usando por completo el esquema y las estadísticas de los datos subyacentes.

10. Herramientas y utilidades

DB2 proporcionar una serie de herramientas para facilitar su uso y administración. El núcleo del conjunto de herramientas ha sido aumentado y mejorado mediante un número de herramientas de parte de fabricantes externos.

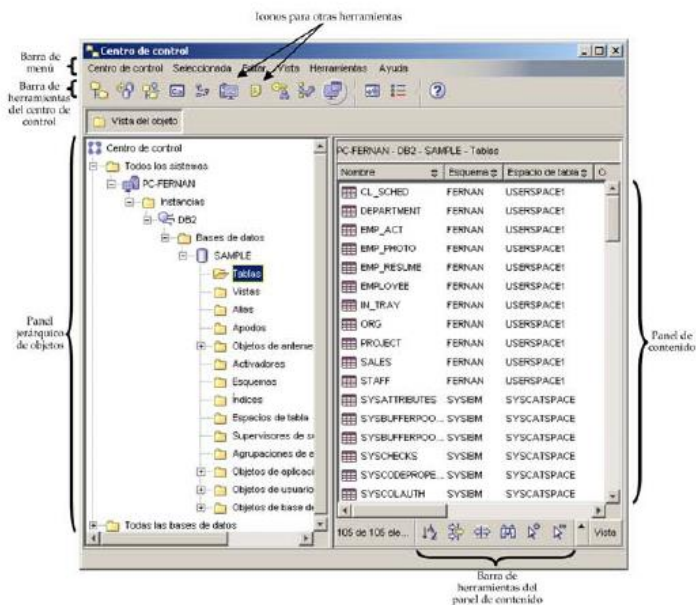


Figure 18 - Centro de control de DB2

El Centro de control de DB2 es la herramienta primaria para el uso y administración de bases de datos en DB2. El Centro de control se ejecuta sobre diversas plataformas de estación de trabajo (PC). En él se encuentran objetos de datos como servidores, bases de datos, tablas e índices. Es una interfaz orientada a tareas para ejecutar comandos y permite que los usuarios generen guiones SQL. El administrador puede utilizar el menú para invocar un conjunto de herramientas componentes.

Los componentes principales del Centro de control son:

- El centro de comandos,
- El centro de guiones,
- Diario,
- Centro de licencias,
- Centro de alertas,

- Supervisor del rendimiento,
- Explicación visual,
- Administración de bases de datos remotas,
- Gestión de almacenamiento y
- Soporte para la réplica.

10.1. Utilidades

Para los administradores, DB2 proporciona un soporte completo para la carga, importación, exportación, reorganización, redistribución y otras utilidades relacionadas con los datos. Además, también soporta varias herramientas, tales como:

- Auditoría para el mantenimiento de la traza de auditoría de las acciones sobre la base de datos.
- Regulador para controlar la prioridad y tiempos de ejecución en distintas aplicaciones.
- Supervisor de consultas para gestionar los trabajos de consulta en el sistema.
- Características de traza y diagnósticos para la depuración.
- Supervisión de eventos para seguir los recursos y eventos durante la ejecución del sistema.

11. Control de concurrencia y recuperación

DB2 soporta un completo conjunto de técnicas de control de concurrencia, aislamiento y recuperación.

11.1. Concurrencia y aislamiento

Para el aislamiento DB2 soporta los modos lectura repetible (*Repeatable Read, RR*), estabilidad en lectura (*Read Stability, RS*), estabilidad del cursor (*Cursor Stability, CS*) y lectura no comprometida (*Uncommitted Read, UR*). Los modos RR, CS y UR no necesitan mayor explicación. El modo de aislamiento RS bloquea solamente las filas que recupera una aplicación en una unidad de trabajo. En una exploración posterior la aplicación tiene garantizado ver todas estas filas (como RR) pero podría no ver nuevas filas que debería ver. Sin embargo, esto podría ser un compromiso aceptable para algunas aplicaciones con respecto al aislamiento RR estricto. Normalmente el nivel de aislamiento predeterminado es CS. Las aplicaciones pueden elegir el nivel de aislamiento en la fase de enlace. La mayoría de las aplicaciones comerciales disponibles soportan los distintos niveles de aislamiento y los usuarios pueden elegir la versión correcta de la aplicación para sus requisitos.

Modo de bloqueo	Objetos	Interpretación
IN (intent none, sin intención)	Espacios de tablas, tablas	Lectura sin bloqueos de filas
IS (intent share, intentar compartir)	Espacios de tablas, tablas	Lectura con bloqueos de filas
NS (next key share, siguiente clave compartido)	Filas	Bloqueos de lectura para los niveles de aislamiento RS o CS
S (share, compartido)	Filas, tablas	Bloqueo de lectura
IX (intent exclusive, intencional exclusivo)	Espacios de tablas, tablas	Intención de actualizar filas
SIX (share with intent exclusive, compartido intencional exclusivo)	Tablas	Sin bloqueos de lectura en las filas pero con bloqueos X en las filas actualizadas
U (Update, actualización)	Filas, tablas	Bloqueo de actualización pero permitiendo leer a otros
NX (next-key exclusive, siguiente clave exclusivo)	Filas	Bloqueo de la siguiente clave para inserciones y borrados para prevenir las lecturas fantasma durante las exploraciones de índice RR
X (exclusive, exclusivo)	Filas, tablas	Sólo se permiten lectores no comprometidos
Z (superexclusive, superexclusivo)	Espacios de tablas, tablas	Acceso completo exclusivo

Figure 19 - Estructura del bloque, sus objetivos e interpretación

11.2. Compromiso y retroceso

Las aplicaciones pueden comprometerse o retrocederse mediante el uso de las instrucciones explícitas *commit* y *rollback*. Las aplicaciones también pueden emitir instrucciones *begin transaction* y *end transaction* para controlar el ámbito de las

transacciones. No se soportan las transacciones anidadas. Normalmente DB2 libera todos los bloqueos que se mantienen por una transacción en *commit* o *rollback*. Sin embargo, si se ha declarado una instrucción de cursor mediante la cláusula *with hold* entonces se mantienen algunos bloqueos durante los compromisos.

11.3 Registro histórico y recuperación

DB2 implementa estrictamente el registro histórico y los esquemas de recuperación ARIES. Emplea el registro histórico de escritura anticipada para enviar registros del registro histórico al archivo de registro histórico persistente antes de que las páginas de datos se escriban al compromiso. En el registro histórico circular, se utiliza un conjunto predefinido de archivos de registro histórico primario y secundario. El registro histórico circular es útil para la recuperación de caídas o la recuperación de un fallo de la aplicación. En el registro histórico de archivo, DB2 crea nuevos archivos de registro histórico y debe guardar los archivos de registro históricos antiguos con el fin de liberar espacio en el sistema de archivos. Los registros históricos de archivo son necesarios para la recuperación hacia delante de una copia de seguridad de archivo.

12. Arquitectura del sistema

La *Figura 20* muestra algunos de los distintos procesos o hebras en un servidor DB2. Las aplicaciones remotas cliente se conectan al servidor de la base de datos empleando agentes de comunicación tales como db2tccpm. Se asigna un agente a cada aplicación (agente coordinador en entornos MPP o SMP) denominado hebra db2agent. Este agente y sus agentes subordinados ejecutan las tareas relacionadas con la aplicación. Cada base de datos tiene un conjunto de procesos o hebras que ejecutan tareas tales como preextracción, limpieza de páginas de la cola de la memoria intermedia, archivo histórico y detección de interbloqueos. Finalmente, se encuentra disponible un conjunto de agentes en el entorno del servidor para ejecutar tareas tales como detección de caídas, servicios de licencia, creación de procesos y control de recursos del sistema. DB2

[illegible]

13. Réplica, distribución y datos externos

33

usando condiciones de filtro además de agregaciones. Las filas resultantes se pueden aplicar a una o más tablas destino. Los medios de administración controlan todas estas acciones.

14. Características de inteligencia de negocio

DB2 Data Warehouse Edition es un producto de la familia DB2 que incorpora características de inteligencia de negocio. Esta edición tiene como base el motor de DB2, y lo mejora con características para ETL, OLAP, minería y generación interactiva de informes. El motor de DB2 proporciona dimensionabilidad mediante sus características MPP. En el modo MPP, DB2 puede soportar configuraciones dimensionables a varios cientos de nodos para bases de datos de gran tamaño (terabytes). Adicionalmente, las características como MDC y MQT proporcionan soporte para los requisitos de procesamiento de consultas complejas de la inteligencia de negocio. Otro aspecto de la inteligencia de negocio es el procesamiento analítico interactivo (*On-Line Analytical Processing, OLAP*). La familia DB2 incluye un componente denominado vista de cubos que proporciona un mecanismo para construir estructuras de datos apropiadas para MQTs dentro de DB2 que se puedan usar para procesamiento OLAP relacional. La vista de cubos proporciona soporte para el modelado de cubos multidimensionales y proporciona un mecanismo de correspondencia con un esquema relacional en estrella.

III. Conclusiones

En este trabajo se pudo desarrollar ampliamente diversas facetas de la herramienta que es DB2 Universal Database. Esta herramienta le proporciona las herramientas y constructores de diseño de bases de datos para que el usuario pueda tener todo a su alcance y desarrollar la base de datos que necesite. También cuenta con los métodos de mantenimiento y administración de la base de datos para su uso constante y a futuro en caso de haber algún cambio a su estructura.

Como ingenieros, el grupo que desarrollo de la actividad de aprendizaje está anuente de esta herramienta como una de las posibles que se utilizarán en un ambiente laboral y profesional. Sus elementos son de suma importancia en el desarrollo profesional de cada uno, y será un conocimiento más que enriquecerá ese crecimiento.

IV. Referencias Bibliográficas

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2006). Fundamentos de Bases de Datos. McGraw-Hill.

V. Anexos

Video Grupal

Estudio de Caso DB2 Universal Database de IBM

<https://youtu.be/-x-wgw5iAdg>

Video Joy N. [DB2 IBM Training](#)

DB2 en Cloud, la tendencia más actual.

The screenshot displays the IBM Db2 on Cloud console interface. At the top, there's a navigation bar with 'Service Details' and 'IBM Cloud'. Below it, a 'LOAD' section shows a progress bar with 'Source', 'Target', 'Define', and 'Finalize' stages. A message states: 'You are loading the file sample-spreadsheet-mkcsay-rn: CF234631.MYTABLE'. Below this, there's a table with columns: 'Data page character encoding', 'Date format', 'Time format', 'Header in first row', 'Time & date format', 'Select date type', and 'Data format (data not shown)'. The table contains 8 rows of data, including columns like 'Date', 'Region', 'Item', 'Unit', 'Unit Cost', and 'Total'.

Date	Region	Item	Unit	Unit Cost	Total
04-07-2011	East	Jeans	65	1.85	120.25
03-07-2015	Central	Jeans	62	8.00	504.00
04-07-2015	Central	Jeans	58	4.50	261.00
05-05-2017	Central	Pen	27	10.00	270.00
02-27-2017	West	Boots	55	2.00	110.00
04-25-2017	East	Jeans	65	4.00	262.50
03-27-2017	Central	Jeans	70	1.00	70.00
05-07-2017	Central	Jeans	60	4.00	240.00
05-27-2017	West	Jeans	52	1.00	52.00