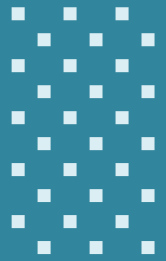
ETL - PARTE 2

Eric Gustavo Coronel Castillo
youtube.com/DesarrollaSoftware
gcoronelc@gmail.com

Logro Esperado

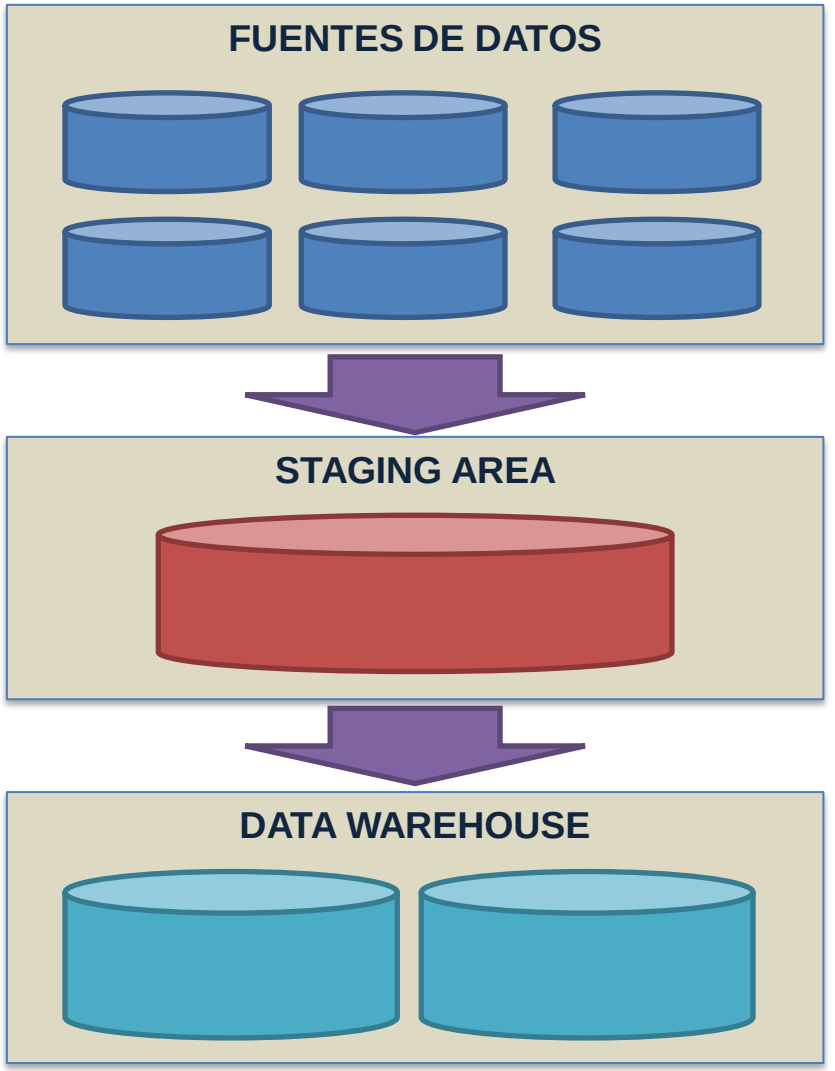
Al finalizar esta presentación, se espera que el participante aplique SSIS para crear procesos ETL Avanzados.



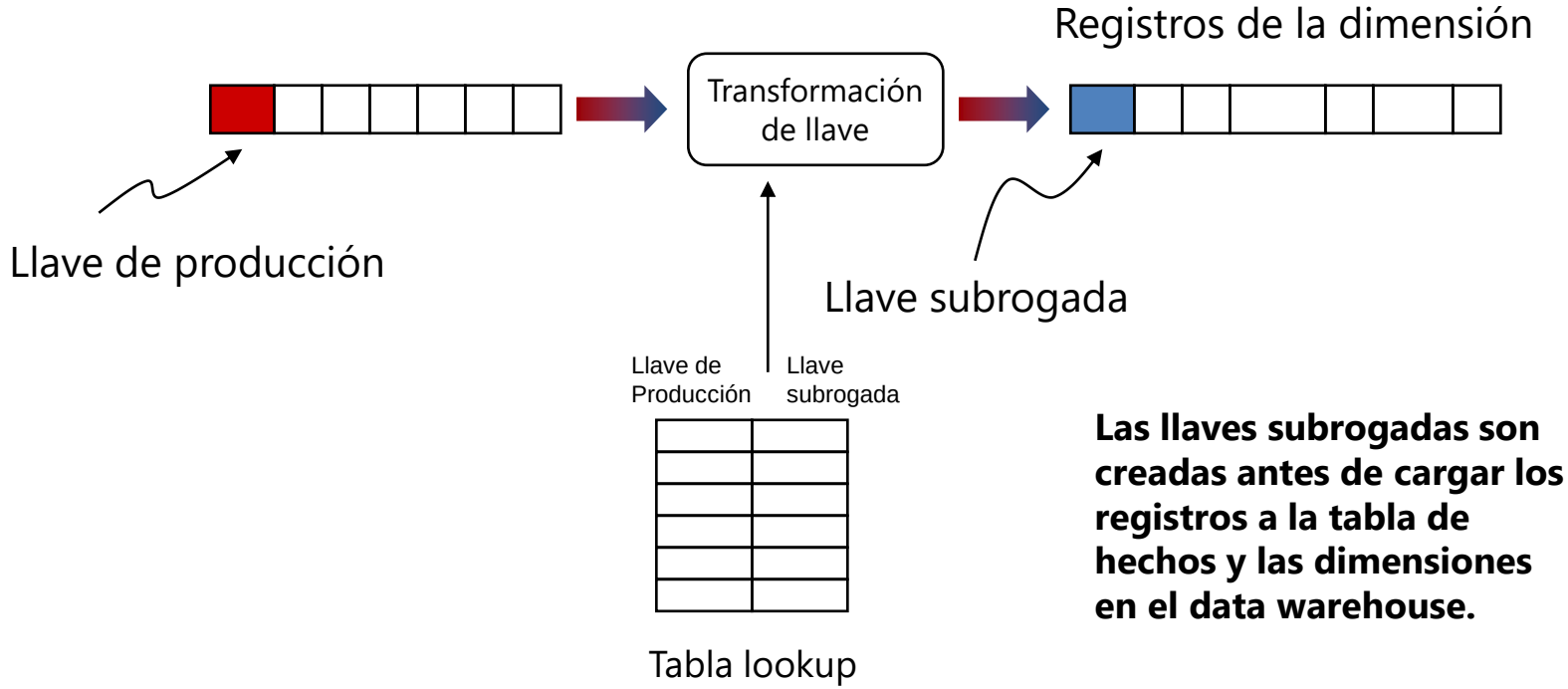
STAGING AREA

STAGING AREA: Fundamentos

- Área de almacenamiento intermedio utilizada para el procesamiento de calidad de datos durante las operaciones de extracción, transformación y carga.
- Contiene tablas normalizadas y archivos usados para:
  - Staging e integridad de datos del Sistema OLTP.
  - Validación.
  - Referencias cruzadas y búsquedas.
- La data OLTP fluye a través del staging area y es eventualmente almacenada en un esquema estrella (dimensional).
- El acceso a la data por parte de los usuarios de negocio está siempre direccionado a la base de datos dimensional.



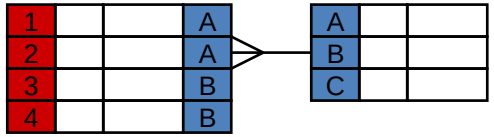
STAGING AREA: Asignación de llaves



Las llaves subrogadas son creadas antes de cargar los registros a la tabla de hechos y las dimensiones en el data warehouse.

Clave subrogada: identificador único en una tabla, y que no tiene ningún significado para el negocio. Su objetivo es diferenciar a un registro de los otros.

STAGING AREA: Integridad Referencial



Esquema E-R

Validar las relaciones uno-a-uno y uno-a-muchos entre atributos almacenados en una tabla de dimensión.

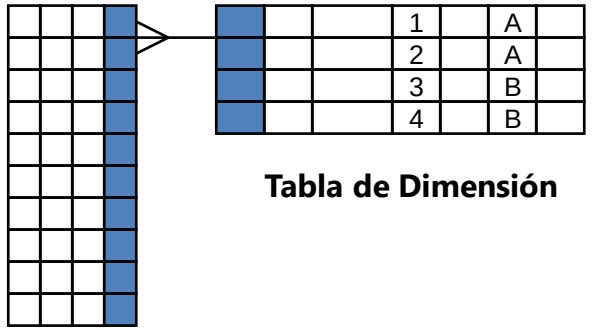
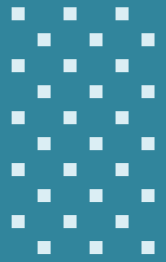


Tabla de Dimensión

Tabla de Hechos (Fact Table)

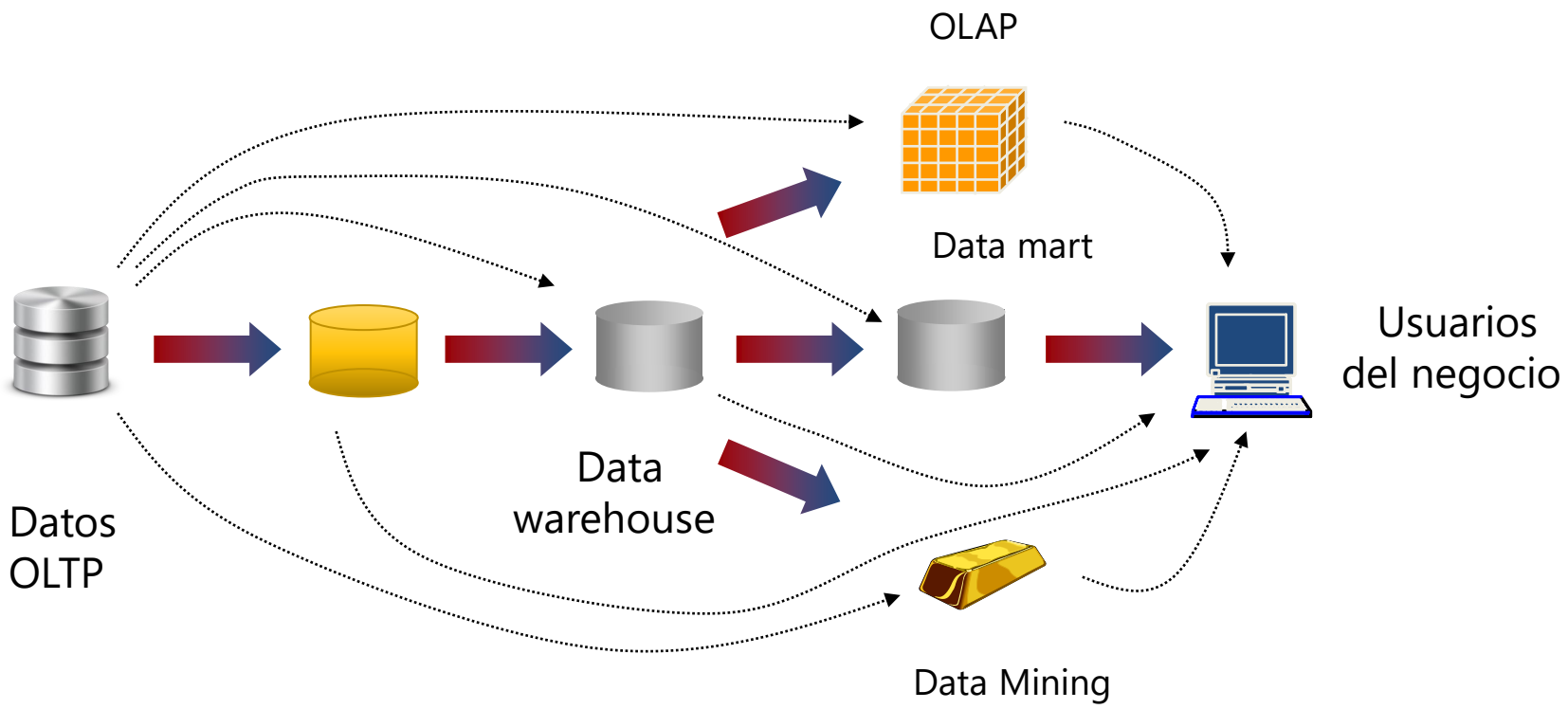
- Use un esquema E-R para el Staging Area.
- Procesamiento secuencial sobre las tablas de dimensión.
  - Uno-a-uno: Ordenar sobre una columna y verificar que existe exactamente un solo valor en la otra columna.
  - Uno-a-muchos: Ordenar sobre los atributos y verificar que cada valor tenga solo un valor en la otra tabla de atributos.



EXTRACCIÓN DE DATOS

GUSTAVO CORONEL gcoronelc@gmail.com

EXTRACCIÓN DE DATOS: Fuentes y destino



EXTRACCIÓN DE DATOS: Especificación

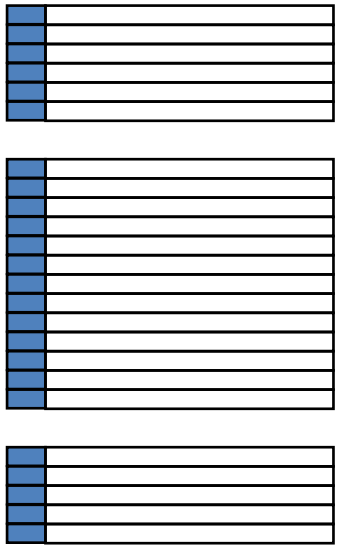
Fuentes de Datos DBMS/Archivos Planos		Transformaciones	Atributos del data warehouse
DBMS/ Arch Planos	Atributos		

PARTICIONAMIENTO DEL DATA WAREHOUSE

- Es el proceso de dividir una tabla en unidades más pequeñas.
- Ventajas:
  - Las consultas son respondidas más rápidamente.
  - Acelera el proceso de backup y restore de la data incremental.
  - Disminuye el tiempo requerido para cargar las tablas indexadas.
- El particionamiento no es libre.
  - La mayor cantidad de consultas son requeridas para determinar cual tabla contiene la data más requerida por el usuario.
  - Adicionalmente la metadata es requerida para definir el particionamiento en el warehouse.

PARTICIONAMIENTO DEL DATA WAREHOUSE: Horizontal y vertical

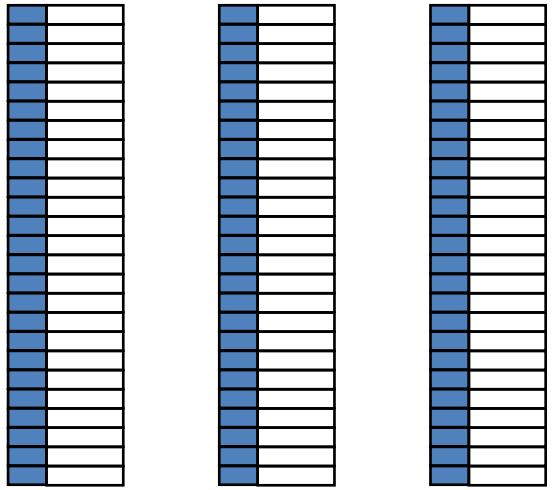
Tabla particionada horizontal (HTP)



Criterio para particionar:
Tiempo:
Año, Mes.
Geografía:
Norte, Sur.
Organización:
Marketing,
Manufactura.

- Una práctica común es particionar por tiempo, dado que es frecuente encontrar este tipo de preguntas en las consultas a los Sistemas de Soporte a la Toma de Decisiones.
- Elegir el criterio de HTP de una tabla puede ser un desafío, dado que la estrategia de una partición favorece a un conjunto de queries sobre otro.

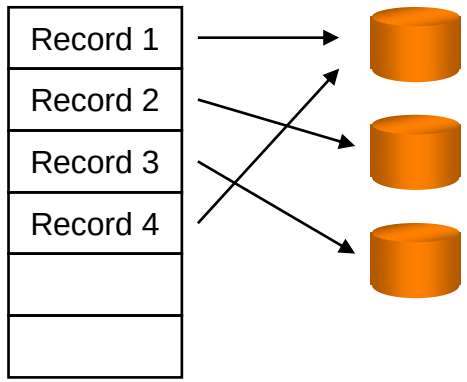
Tabla particionada vertical (VTP)



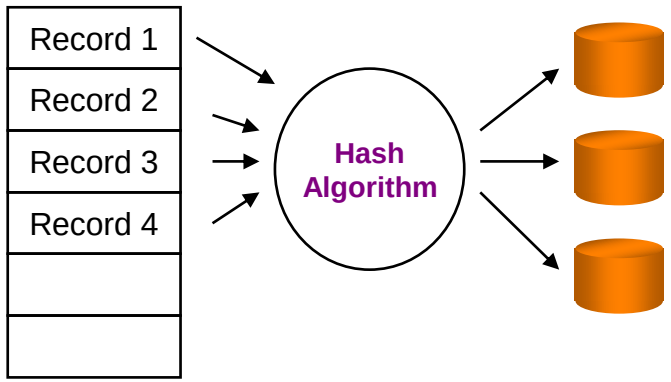
- Una razón común para aplicar VTP a una tabla es acelerar el acceso a los queries emitidos por un conjunto muy diverso de usuarios.
- Transporta la sobrecarga de espacio asociada con la duplicación de claves en todas las particiones.

PARTICIONAMIENTO DEL DATA WAREHOUSE: Algoritmos

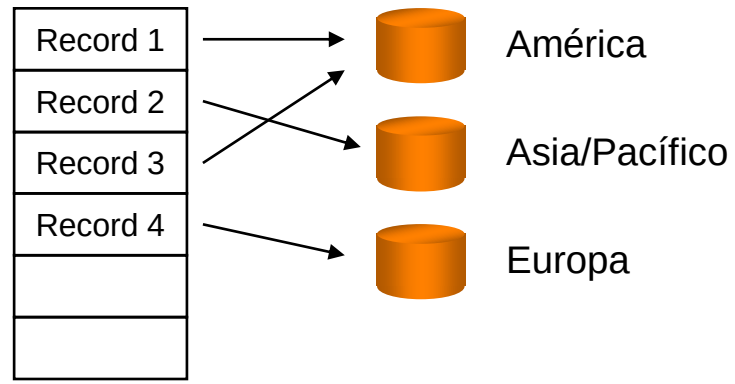
Round-robin



Hash

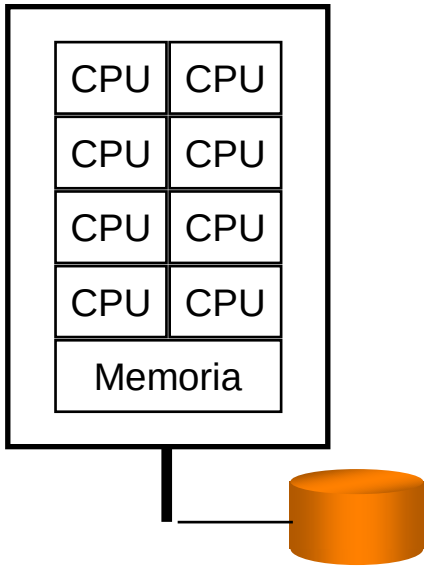


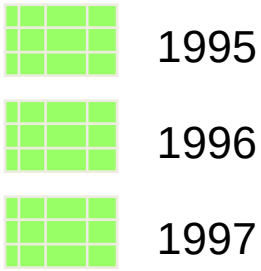
Range



PARTICIONAMIENTO DEL DATA WAREHOUSE: Arquitectura del hardware

SMP
Multiprocesamiento Simétrico



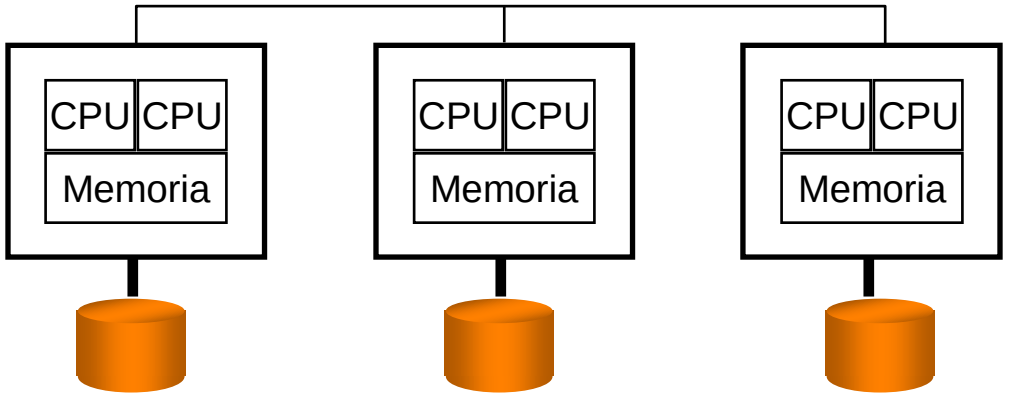


1995

1996

1997

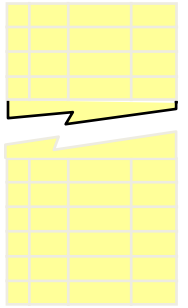
MPP
Procesamiento Masivamente Paralelo



1995

1996

1997

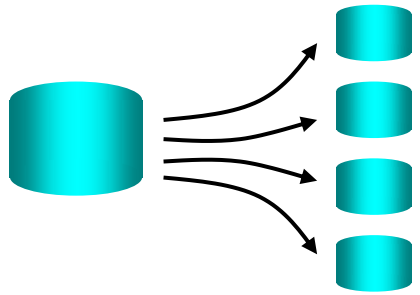


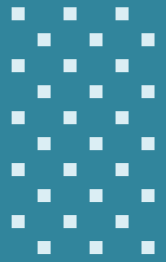
1995-1997

Una estrategia de particionamiento es influenciada no sólo por el tipo de consultas sino también por la arquitectura del hardware.

PARTICIONAMIENTO DEL DATA WAREHOUSE: Beneficios

- Incrementa el paralelismo.
- Reduce los tiempos de backup.
- Incrementa la disponibilidad.
- Mejora la administración.
- Reduce el conjunto de datos para las consultas.
- La eliminación de datos antiguos es más rápida.





PARTICIONAMIENTO EN SQL SERVER

GUSTAVO CORONEL

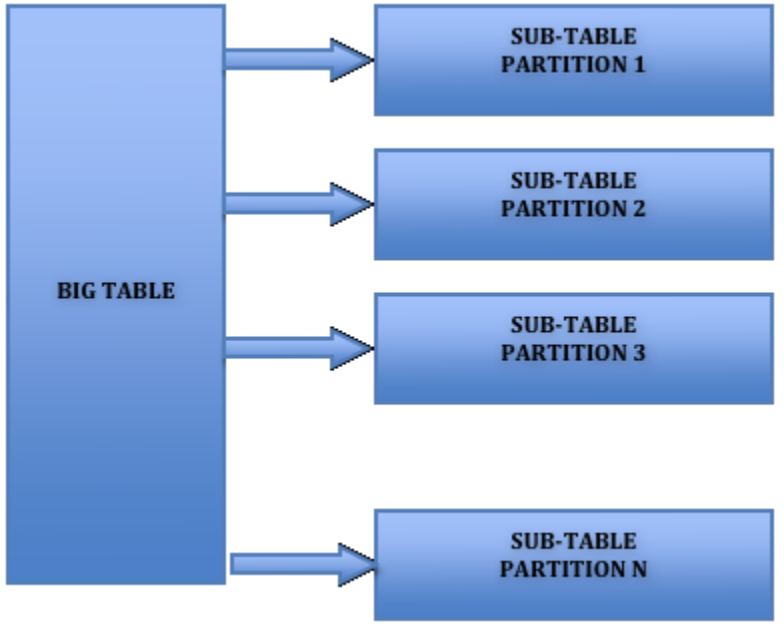
gcoronelc@gmail.com

PARTICIONAMIENTO EN SQL SERVER: Tablas e índices

- ✓ Reparte las filas de la tabla entre varios filegroups, dependiendo del valor que tome una determinada columna.

PARTICIONAMIENTO EN SQL SERVER: Partition Function

- Definen el tipo de dato que se utilizará para particionar la información.
- Todos los tipos de datos, excepto **text**, **ntext**, **image**, **xml**, **timestamp**, **varchar(max)**, **nvarchar(max)**, **varbinary(max)**, son aceptables para particionar.
- Contiene el valor de las fronteras que se utilizarán en la partición.
- El número de particiones es igual al número de las fronteras más 1.



PARTICIONAMIENTO EN SQL SERVER: Sintaxis de Partition Function

- **Sintaxis**

  CREATE PARTITION FUNCTION <Nombre> (tipo de dato)
  AS RANGE [LEFT | RIGHT]
  FOR VALUES (<lista de valores>)

- **Ejemplo**

  CREATE PARTITION FUNCTION pf_FechaDocumento (datetime)
  AS RANGE LEFT
  FOR VALUES ('01/01/2002','01/01/2004','01/01/2006')

- **Fronteras.**
  - Fechas menores a 01/01/2002 incluido el 01/01/2002
  - Fechas entre 01/01/2002 y 01/01/2004 incluido 01/01/2004
  - Fechas entre 01/01/2004 y 01/01/2006 incluido 01/01/2006
  - Fechas Superiores al 01/01/2006

PARTICIONAMIENTO EN SQL SERVER: Partition Scheme

- Mapea las particiones definidas en la Partition Function a los filegroups donde la información será almacenada físicamente.
- En necesario que exista una Partition Function definida.
- Es necesario que existan los filesgroups que se utilizarán en la Partition Scheme.

PARTICIONAMIENTO EN SQL SERVER: Sintaxis de Partition Scheme

- ✓ **Sintaxis**

  CREATE PARTITION SCHEME <Nombre>

  AS PARTITION <partition function>

  [TO (<lista de filegroups>) | ALL TO (filegroup)

- ✓ **Ejemplo**

  CREATE PARTITION SCHEME ps_FechaDocumento

  AS PARTITION pf_FechaDocumento

  TO (Filegroup1, Filegroup2, Filegroup1, Filegroup2)

PARTICIONAMIENTO EN SQL SERVER: Ejemplos de Partition Scheme

- ✓ **Ejemplo**

```
CREATE PARTITION SCHEME ps_FechaDocumento
AS PARTITION pf_FechaDocumento
TO (Filegroup1, Filegroup2, Filegroup1, Filegroup2, Filegroup1)
```

- ✓ **Ejemplo**

```
CREATE PARTITION SCHEME ps_FechaDocumento
AS PARTITION pf_FechaDocumento
ALL TO (Filegroup1)
```

PARTICIONAMIENTO EN SQL SERVER: Tabla particionada

- Requiere un Partition Scheme.
- Se debe indicar la columna que se utilizará para el particionamiento.
- El tipo de dato de la columna por la que se particionara debe ser el mismo tipo de dato de la Partition Function.

PARTICIONAMIENTO EN SQL SERVER: Sintaxis para crear una tabla particionada

- ✓ **Sintaxis**

  CREATE TABLE <Nombre>

  <Lista de campos>

  ON <partition scheme>(campo de partición)

- ✓ **Ejemplo**

  CREATE TABLE Ventas (

  SalesOrderID int identity,

  OrderDate datetime)

  ON ps_FechaDocumento(OrderDate)

PARTICIONAMIENTO EN SQL SERVER: Consultar particionamiento

- ✓ **Consulta**

```
SELECT
    SalesOrderID, OrderDate,
    $partition.pf_FechaDocumento(OrderDate) NroParicion
FROM Ventas;
GO
```

PARTICIONAMIENTO EN SQL SERVER: Particionamiento de índices

- ✓ **Sintaxis**

  CREATE INDEX <nombre indice>

  ON <tabla>(lista de campos)

  ON partition scheme(campo de particionamiento)

- ✓ **Ejemplo**

  CREATE UNIQUE INDEX idx5_orden

  ON Ventas(SalesOrderId, OrderDate)

  ON ps_FechaDocumento(OrderDate)

CONCLUSIONES

CONCLUSIONES

- EL particionamiento de las bases de datos muy grandes mejora el rendimiento de las consultas a los datos.
- Además del tipo de consultas, la arquitectura del hardware influye en la estrategia del diseño de particionamiento de una base de datos.

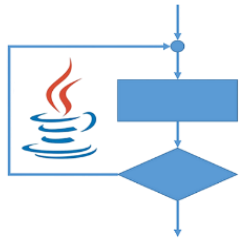
GRACIAS TOTALES

GUSTAVO CORONEL gcoronelc@gmail.com

Gustavo Coronel
gcoronelc.github.io

udemy

gcoronelc.github.io
youtube.com/DesarrollaSoftware
facebook.com/groups/desarrollasoftware



FUNDAMENTOS DE PROGRAMACIÓN CON JAVA

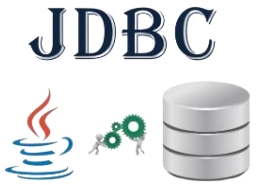
Inicia tu aprendizaje, utilizando las mejores prácticas de programación

CURSO PROFESIONAL DE JAVA ORIENTADA A OBJETOS

Aprende programación en capas, patrones y buenas prácticas

PROGRAMACIÓN DE BASE DE DATOS ORACLE CON PL/SQL

Aprende a obtener el mejor rendimiento de tú base de datos



PROGRAMACIÓN DE BASE DE DATOS ORACLE CON JDBC

Aprende a programar correctamente con JDBC