**Introduction to Data Warehousing and Business Intelligence**

Robert C. Cain

**Course Overview:**

DW (Data Warehousing) /BI (Business Intelligence) es un área ampliamente usada en el mundo de los negocios. Tras la recolección y el almacenamiento de extensas bases de datos, permite obtener estadísticas que facilitan la toma de decisiones dentro del mundo empresarial.

**Introducción:**

A lo largo del curso se verán conceptos como: *Diseño de una Data Warehouse,* Ejemplos de Implementación, *WideWorldImporters, SQL Server 2016.*

**¿Por qué aprender *DW/BI*?**

* Las bases de datos tradicionales difieren mucho de una DW.
* Existen herramientas que son propias de esta área: Ejemplos como *SQL Reporting Service.*
* Evitan la necesidad de crear interfaces especiales para la extracción de información de bases de datos tradicionales.
* También es útil para Administradores de negocios pues podrán ser capaces de extraer de si negocio de forma autosuficiente.

**¿Qué es DataWarehouse?**

* Es una base de datos gigantesca que almacena TODA tu información.
* DW actúa como un repositorio central en donde se agrega información de múltiples sistemas. Esta información combinada permite obtener una vista general del desempeño de una empresa.
* Es *la fuente de toda verdad*, pues almacena información puntual.

**¿Qué es Business Intelligence?**

* Información procesada y de la que dispones para convertir el conocimiento en acciones informadas.
* Provee métodos para mantener la *salud*  de tu negocio.
* Permite examinar tres áreas de interés:
  + Agregaciones: Descomponer volúmenes de información en características más pequeñas que si sean de interés, ejemplo: marca de un producto, beneficios obtenidos en un plazo de tiempo..
  + Tendencias: Al conocer cómo se desenvuelve un producto es posible hacer estimaciones de cómo lo hará más adelante y en base a esto optar por una decisión.
  + Correlación(Data Mining/Machine Learning): Herramientas extras que permiten descubrir relaciones entre productos que a simple vista no se notan. Si un producto vende, puede que otro no lo haga.

**¿Por qué tener una DataWarehouse?**

La primera razón deriva de combinar información de múltiples sistemas. Al estar junta, se facilita obtención de reportes. Permite además resolver inconsistencias entre sistemas. Si por ejemplo en distintas bases de datos se utiliza un código diferente para el mismo producto, es posible asignar un solo código para dicho producto dentro de la DW. Reduce además la carga en sistemas de producción. Muchas de las consultas pasarían a ser ejecutadas en la DW. Provee además un sitio en donde almacenar información de larga estancia y finalmente facilita la consistencia entre sistemas de transición.

**Metodologías de Data Warehousing:** Existen 2

**Bill Inmon – Corporate Information Factory (CIF)**

* Muy similar a las bases de datos tradicionales.
* El usuario puede acceder directamente a los CIF.

**Ralph Kimball – Kimball Method – Start Schema**

* El usuario puede acceder directamente a la DW.
* Es el método más extendido.

**No usar los sistemas transaccionales para tareas de *reporting.* OLTP (*On Line Transaction Processing*)**

* Están diseñadas para acceder a datos individuales.
* Los datos han sido normalizados (Implica remover valores duplicados)
* Debido a la normalización, obtener datos puede involucrar el uso de mucho *joins.*
* No es apropiado para personas del mundo de los negocios.
* Provoca un impacto negativo en las OLTP.

**Una DW difiere en… OLAP(Online Analytical Processing)**

* Posee una mejor arquitectura para realizar tareas de *report.*
* La data no se encuentra normalizada.
* Incrementa el espacio utilizado a diferencia de una base de datos tradicional.
* Número de tablas es reducido.
* Dos esquemas importantes: Start y Snowflake.
  + Start: Todas las dimensiones (información descriptiva) tienen relación con la *Fact Table* (Información objetiva).
  + Snowflake: La dimensiones pueden establecer relaciones con otras dimensiones.
* Se opta por lo general utilizar *Start* sobre *Snowflake.*

Más allá de *Facts* o *Dimensions* destaca el uso de *Surrogate keys* (SK)*.* Por lo general, estas son nuevas llaves primarias establecidas dentro de la DW. Deben de ser pequeñas y fáciles de entender. Algunas de las razones por las que se utilizan se deben a cambios en los sistemas fuentes. Sin embargo no siempre es necesario utilizarlas.

**Fact Tables:**

* Diseñados para resaltar un evento. Describen información concisa y cuantificable

**Dimensions:**

* Utilizados para describir información relacionados a los *fact tables.*
* Por lo general almacenan información como: Puntos geográficos, empleados, productos, clientes, hora, etc…
* Son de carácter SCD (*Slowly Changing Dimension*)

**Dimensiones de tipo 0 (Fijos):** Almacenan información estática (colores y tamaños) que por lo general no cambiará. Sin embargo, eso no evita que se añada nueva información.

**Dimensiones de tipo 1:** Al actualizarse un valor este simplemente se sobrescribe. Su impacto es capaz de alterar información del pasado.

**Dimensiones de tipo 2:** Al actualizarse un valor, el dato antiguo deja de actualizarse y se inserta uno nuevo. Bastante útil al manejar problemas burocráticos y legales (por razones de fecha). La importación de este tipo de datos a partir de un OLTP requiere del uso de columnas adicionales que registren las fechas de cuando se realizaron los cambios. Es importante también ser consistente con los nombres dados y tener en cuenta el no usar palabras reservadas. Se debe además considerar el tipo de dato con el vayamos a trabajar.

**Dimensiones de tipo 3:** Al actualizarse un valor, los datos viejos son cambiados a una columna nueva. Se puede considerar como una dimensión inútil pues favorece la pérdida de información.

**Dimensiones de tipo 4:** Una combinación entre los tipos 1 y 2. Cuando un valor se actualiza, los datos viejos son guardados en una tabla historial y el valor actual se modifica. Las tablas que se utilizan manejan sus propias llaves.

**Existen algunas otras dimensiones…**

* Todas estas dimensiones se aplican a nivel de columna no de fila.
* Debe ser la empresa la que determine que datos requieren de un seguimiento histórico. *Trackear* todo es increíblemente ineficiente.
* Tampoco utilices los términos *Tipo 1, Tipo 2.* Es preferible destacar un atributo.
* Si por ejemplo utilizamos esto en un *Snowflake Schema* encontraremos tablas intermedias que describan algunas de las entradas de nuestra tabla principal.

**Obteniendo Data para nuestro DW:**

* Usar ETL (Extract – Transform – Load)
* Podemos utilizer SSIS – SQL Server Integration Services, Powershell o Aplicaciones Propias.

**Sacar Data de nuestro DW:**

* SSAS – SQL Server Analysis Services
* Azura ML (Machine Learning)

**Reporting:**

* SSRS – SQL Server Reporting Services
* Excel and PowerPoint
* Power BI.