# Introducción

## Definición de Inteligencia de Negocios

### Business Intelligence

Se puede describir a La inteligencia de negocios (Business Intelligence – BI) , como un concepto que integra por un lado el almacenamiento y por el otro el procesamiento de grandes cantidades de datos, con el principal objetivo de transformarlos en conocimiento y en decisiones en tiempo real, a través de un sencillo análisis y exploración. La definición antes expuesta puede representarse a través de la siguiente fórmula:

Datos+Analisis=Conocimiento

#### Beneficios

Entre los beneficios más importantes que BI proporciona a las organizaciones, vale la pena destacar los siguientes:

* Reduce el tiempo mínimo que se requiere para recoger toda la información relevante del negocio, ya que la misma se encontrará integrada en una fuente única de fácil acceso.
* Automatiza la asimilación de la información, debido a que la extracción y carga de los datos necesarios se realizará a través de procesos predefinidos.
* Proporciona herramientas de análisis para establecer comparaciones y tomar decisiones.
* Cierra el círculo que hace pasar de la decisión a la acción.
* Permite a los usuarios no depender de reportes o informes programados, porque los mismos serán generados de manera dinámica.
* Posibilita la formulación y respuesta de preguntas que son claves para el desempeño de la empresa.
* Permite acceder y analizar directamente los indicadores de éxito.
* Se pueden identificar cuáles son los factores que inciden en el buen o mal funcionamiento de la empresa.
* Se podrán detectar situaciones fuera de lo normal.
* Se encontrarán y/o descubrirán cuáles son los factores que maximizarán el beneficio.
* Permitirá predecir el comportamiento futuro con un alto porcentaje de certeza, basado en el entendimiento del pasado.
* El usuario podrá consultar y analizar los datos de manera sencilla.

## Data Warehouse

Debido a que para llevar a cabo BI, es necesario gestionar datos guardados en diversos formatos, fuentes y tipos, para luego depurarlos e integrarlos, además de almacenarlos en un solo destino, depósito o base de datos que permita su posterior análisis y exploración, es imperativo y de vital importancia contar con una herramienta que satisfaga todas estas necesidades. Esta herramienta es el Data Warehouse (DW), que básicamente se encarga de consolidar, integrar y centralizar los datos que la empresa genera en todos los ámbitos de una actividad de negocios (Compras, Ventas, Producción, etc), para luego ser almacenados mediante una estructura que permite el acceso y exploración de la información requerida con buena performance, facilitando posteriormente, una amplia gama de posibilidad de análisis multivariables, que permitirá la toma de decisiones estratégicas y tácticas.

El DW posibilita la extracción de datos de sistemas operacionales y fuentes externas, permite la integración y homogeneización de los datos de toda la empresa, provee información que ha sido transformada, para que ayude en el proceso de toma de decisiones estratégicas y tácticas. El DW, convertirá entonces los datos operacionales de la empresa en una herramienta competitiva, debido a que pondrá a disposición de los usuarios indicados la información pertinente, correcta e integrada, en el momento que se necesita. Una de las definiciones más famosas sobre DW, es la de W. H. Inmon, quien define: “Un Data Warehouse es una colección de datos orientada al negocio, integrada, variante en el tiempo y no volátil para el soporte del proceso de toma de decisiones de la gerencia”.

### ARQUITECTURA DE UN DATA WAREHOUSE

Los Datos son extraídos desde las aplicaciones, bases de datos transaccionales, archivos, etc. los cuales son transformados e integrados para ser cargados en la estructura del DW. La información en el DW se estructura en cubos multidimensionales, los cuales preparan la información para poder responder a las consultas dinámicas optimizando la performance. Las cuales pueden ser realizadas tanto por usuarios no expertos como por usuarios avanzados, accediendo al DW a través de diversas herramientas de consultas, reportes, análisis.

La arquitectura del Data Warehouse está formada por diversos componentes que interactúan entre si y en los cuales cada uno provee una función específica dentro del DW:

#### OLTP (On Line Transaction Processing)

Este componente está formado por toda la información transaccional que se genera en la organización, como también por aquellas fuentes externas de datos que puede llegar a disponer.

Entre los OLTP más habituales se encuentran: Archivos de textos, Hojas de cálculos, informes, Bases de Datos Transaccionales.

#### Load Manager (ETL)

Los ETL son los encargados manipular los datos anteriores para integrarlos, transformarlos y cárgalos en el DW.

En síntesis, las funciones específicas de los ETL son tres:

* Extracción.
* Transformación.
* Carga.

#### Data Warehouse Manager

El data Warehouse Manager se encarga de gestionar el depósito de datos y lo organiza en torno a una base de datos multidimensional y gestionar y mantener metadatos.

##### Base de datos multidimensional

Las Bases de datos multidimensionales proveen una estructura que permite tener acceso flexible a los datos. Estas se pueden visualizar como un cubo multidimensional, en donde las variables asociadas existen a lo largo de varias dimensiones y la intersección de las mismas representa la medida, indicador o el hecho que se está evaluando

Estas bases implican tres variantes posibles de modelado:

* Esquema Estrella
* Esquema Copo de Nieve
* Esquema Constelación o copo de estrellas

Estos pueden ser implementados de diversas maneras que según sea requieren que toda la estructura de datos este desnormalizada o semi desnormalizada.

Los diferentes tipos de implementación son los siguientes:

* Relacional – ROLAP
* Multidimensional – MOLAP
* Híbrido – HOLAP

##### ¿Qué es un esquema estrella?

El esquema estrella se denomina de dicha forma porque el diagrama entidad-relación recrea la figura de una estrella. El centro de la estrella consiste de una larga tabla de hechos y los extremos de la estrella lo conforman las tablas de dimensiones.

##### Tabla de hechos y tabla de dimensiones

Las tablas de hechos contienen información sobre los eventos de negocio (ventas, órdenes de compra) los hechos, medidas o indicadores que serán utilizados por los analistas de negocio para apoyar el proceso de toma de decisiones. Los hechos son datos instantáneos en el tiempo, que son filtrados, agrupados y explorados a través de condiciones definidas en las tablas de dimensiones.

Las tablas de dimensiones contienen información sobre las entidades del negocio ( clientes, productos, stock), definen como están los datos organizados lógicamente y proveen el medio para analizar el contexto del negocio. Representan los ejes del cubo, y los aspectos de interés, mediante los cuales el usuario podrá filtrar y manipular la información almacenada en la tabla de hechos.

#### Query Manager

El Query Manager es el encargado de realizar las operaciones necesarias para soportar los procesos de gestión ejecución de consultas.

Este recibe las consultas del usuario, las aplica a las talas correspondientes y devuelve los resultados obtenidos.

Las operaciones que se pueden realizar sobre modelos multidimensionales y que permitirán al usuario explorar y desmenuzar los datos en busca de respuestas son:

* Drill-down: bajar por la jerarquía de una dimensión viendo los datos con mayor detalle.
* Drill-up: subir por la jerarquía de una dimensión viendo los datos con menor detalle.
* Drill-across: agrega como nuevo criterio de análisis una dimensión.
* Roll-across: quita una dimensión como criterio de análisis.
* Pivot: selecciona el orden de visualización, con el fin de analizar la información desde diferentes perspectivas
* Page-Slice: presente el cubo dividido en secciones como si se tratase de páginas de un libro.

#### Herramientas de Consulta y Análisis

Las Herramientas de consulta y Analisis son las que permiten al usuario obtener la información y explorar los datos, son el nexo entre el deposito de datos y los usuarios.

Existen diferentes tipos de herramientas de consulta y análisis, estas son :

* Reportes y Consultas
* OLAP
* Data Mining
* EIS

#### OLAP

OLAP es el acrónimo de Procesamiento Analitico en línea es la componente más poderosa de los DW, ya que es el motor de consultas especializado del mismo. La idea de una base de datos OLAP es la de utilizar un repositorio de datos optimizado para el análisis de datos multidimensionales ofreciendo flexibilidad al usuario y un respuesta rápida a las consultas realizadas.

Existen tres tipos de base de datos OLAP de acuerdo al modo en que los datos son almacenados:

* MOLAP (OLAP Multidimensional) Los datos son guardados en un formato multidimensional. Toda la información detallada y resumida se encuentre almacenada en los archivos del cubo. Un Ejemplo de BD MOLAP es PALO
* ROLAP (OLAP relacional): Los datos y los resúmenes se encuentran almacenados en una base de datos transaccional. El motor ROLAP transforma las consultas multidimensionales en consultas SQL optimizadas agregando además capacidades de cacheo. Pentaho Mondrian es un ejemplo de motor ROLAP.
* HOLAP (OLAP Hibrido) los resúmenes y datos de navegación se encuentran almacenados en una estructura MOLAP y la información detallada es almacenada en una base de datos transaccional.

## Pentaho BI Suite

Pentaho es una suite de herramientas de BI que trabajan en forma conjunta para brindar soluciones de BI.

Las soluciones de Pentaho están escritas en Java y tienen un ambiente de implementación también basado en Java.

### Capa de presentación

Pentaho contiene una interfaz web llamada “user console”. Esta permite a un usuario interactuar con el servidor. La capa de presentación permite al usuario navegar y abrir el contenido existente (dashboards, reportes, análisis) y también crear nuevos contenidos de BI.

### Capa de Componentes de BI

Esta capa esta formada por un conjunto de componentes que ofrecen las herramientas básicas de BI :

* Capa de Metadatos
* Servicios de Reportes Ad Hoc
* Motor ETL
* Motor de Reportes
* Motor OLAP
* Motor de Data Mining

###### Capa de Metadatos

La Función de la capa de Metadatos ( PML) es separar al usuario de la complejidad de SQL y las base de datos, es el componente de penthao BI que brinda las soluciones para crear modelos de metadatos.

###### Ad hoc Reporting Service

The Web Ad Hoc Query and Reporting service, o WAQR, ofrece al usuario una forma facil de crear reports utilizando la capa de metadatos.

###### The ETL Engine

El Motor ETL de pentaho denominado Kettle es el encargado de ejecutar las tareas generadas por Pentaho Data Integration tools.

Ambos components brindan soluciones para capa ETL de la arquitectura del DW.

###### Reporting Engines

La plataforma de pentaho permite alojar varios motores de reportes.

###### The OLAP Engine

Mondrian es el motor OLAP de pentaho que ademas trasfroma las consultas MDX en consultas SQL basado en el modelo multidimensional.

###### The Data Mining Engine

Weka es el motor de data mining usado por Pentahoque permite manejar tares de data mining.

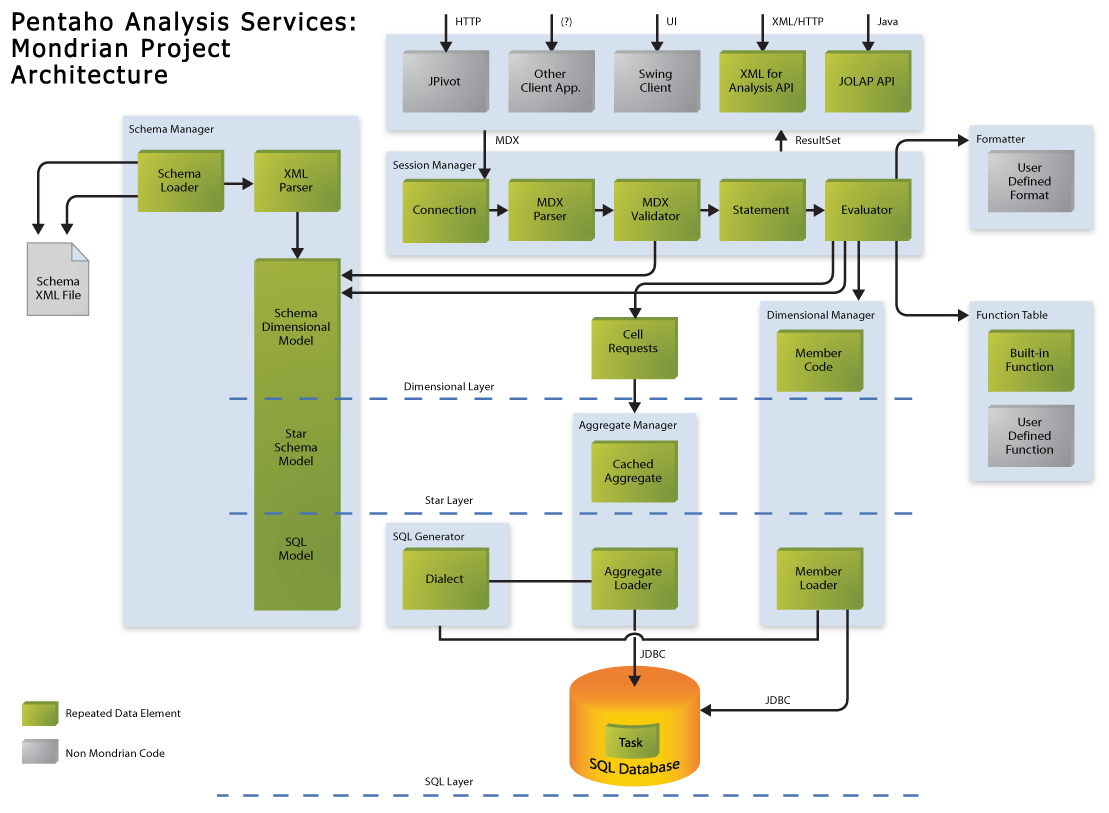
## Pentaho Analysis Services

Pentaho Analysis Services (PAS ) provee las capacidades OLAP a Pentaho , a travez de una arquitectura ROLAP.

PAS esta formado por los siguientes componentes:

* **JPivot** —JPivot es una herramienta de análisis basada en java que sirve de interfaz al usuario para que este pueda trabajar con cubo .
* **Mondrian ROLAP**— Es el Corazón de PAS , el encargado de recibir las consultas MDX y devolver respuestas multidimensionales.
* **Schema Workbench**— Es una herramienta visual para diseñar y testear esquemas de cubos. Mondrian utiliza estos esquemas para interpretar las consultas MDX, trasladarlas a consultas SQL y obtener los datos de un RDBMS.
* **Aggregate Designer**— Es una herramienta visual para generar tablas de agregación y aumentar la performance de las consultas.

### Arquitectura de Mondrian

Un sistema **OLAP Mondrian** consiste de 4 capas: capa de presentación, la capa dimensional, la capa de estrella (Star Layer) y por último la capa de almacenamiento (Storage Layer).

#### Capa de presentación

Determina que es lo que ve el usuario final en el monitor y de que forma interactúa y efectúa preguntas. La forma de representación puede consistir en tablas pivot, diagramas de torta y barras y también herramientas de visualización avanzada como mapas interactivos o gráficos dinámicos. Estos pueden estar escritos en **Swing** o **JSP**, cuadros exhibidos en **JPEG** y **GIF** o transmitidos a una aplicación remota vía **XML**. Por diversas y heterogéneas que parezcan estas formas de presentación, tienen en común la **gramática multidimensional**, medidas y celdas en las cuales la capa de presentación efectúa preguntas y el servidor OLAP retorna la respuesta.

#### Capa dimensional

Esta capa analiza gramaticalmente (**parsing**), valida y ejecuta sentencias **MDX**[3]. Una consulta es evaluada en múltiples fases. Los ejes son computados primero y luego los valores de las celdas dentro de los ejes. Por razones de eficiencia, la capa dimensional envía requerimientos de celdas (**cell-requests**) a la capa de gregación en lotes. Un transformador de sentencias (**query transformer**) permite a la aplicación manipular consultas existentes, en vez de construir una sentencia nueva por cada petición. Los metadatos escriben el modelo multidimensional, y de que manera se corresponde o mapea en el modelo relacional.

#### Capa de estrella

Es la responsable de mantener un **caché** de agregaciones. Una **agregación** es un conjunto de valores de medidas (*celdas*) en memoria, descriptas por una serie de valores de columnas dimensionales. Esta capa recibe peticiones por un conjunto de celdas. Si las celdas no se encuentran en caché, o disponibles al realizar ***rollup*** en una agregación en caché, el gestor de agregación envía una petición a la capa de persistencia.

#### Capa de almacenamiento (Nro 4)

Esta capa se encuentra representada por un **SGBDR**, esto es lo que convierte a Mondrian en un servidor **ROLAP**. Es la capa responsable de brindar celdas agregadas de datos y miembros de las tablas dimensionales. La decisión de utilizar un **SGBDR** se basa en que brinda toda la funcionalidad necesaria para efectuar las agregaciones requeridas para *simular* un modelo multidimensional, quitando la complejidad del desarrollo **MOLAP** y manteniendo independiente la decisión del **SGBDR** a ser utilizado, ganando en **flexibilidad**.

## esquema xml CUBOS, Dimensiones y Medidas

El esquema es un documento xml que describe uno o más cubos multidimensionales. Los cubos son estructuras de datos que describen el mapeo entre las dimensiones del cubo y las medidas con las tablas y columnas en una base de datos relacional. Para mondrian el esquema es la clave para trasladar consultas MDX en consultas SQL.

Un cubo esta formado por un conjunto de dimensiones las cuales forman los ejes del cubo y por un conjunto de medidas que son los valores a reportar para cada una de las intersecciones de las dimensiones.

### Jerarquias Niveles y Miembros

Para que un cubo sea útil, se necesita más que valores individuales en las intersecciones de las dimensiones. Es necesario una manera de agrupar los datos a lo largo de las diversas dimensiones. Para este propósito las dimensiones son organizadas en una o más jerarquías.

### Jerarquias

Una jerarquía es una estructura de tipo árbol que es usada para obtener datos del cubo con diferentes niveles de agrupación. ( agregación). El ejemplo más común es la dimensión del tiempo con la jerarquía Año-Trimestre-Mes-Día.

### Niveles y Miembros

Una jerarquía esa compuesta por niveles y cada uno de esos niveles tiene uno o más miembros. En el ejemplo año trimestre mes y dia son niveles y los elementos de estos son los miembros. Es importante que los niveles que los miembros de un nivel no deben solaparse de forma de obtener resultados siempre consistentes

Las dimensiones pueden contener múltiples jerarquías de forma de ofrecer múltiples formas de análisis y caminos de navegación a lo largo de una dimensión, como también servir a diferentes necesidades de negocio.

## JIVE SBS

JIve Sbs o “ Social Business Software” es un herramienta de colaboración y manejo del conocimiento basada en Java EE.

Jive SBS contiene funcionalidades tales como el manejo de comunidades online, microblogging, redes sociales, foros de discuciones, blogs, wikis.

Las aplicaciones SBS son diseñadas para usar en un contexto corporativo, típicamente para sustituir o complementar las intranets de las compañías y suplementar las propiedades de la web que las compañías utilizan para organizar sus comunicaciones exteriores.

SBS fomenta la colaboración informal que normalmente queda afuera de las aplicaciones transaccionales, flujos de trabajos formales, o equipos de ingeniería.

La interacción entre los pares toma lugar en línea , a la conveniencia del usuario y en instancias internas eliminando las necesidades de reuniones, llamadas de conferencias y cadenas de mails.

Los usuarios pueden crear espacios de trabajo en sus comunidades dedicadas a un departamento específico donde pueden crear discusiones , realizar preguntas, postear blogs, y crear, modificar y ver documentos.

# Instalación y configuración de Mondrian

El primer paso es tener Java SDK instalado y con la variable de entorno configurado, en caso de no tenerlo descargamos el paquete de Java lo instalamos, y creamos una variable global JAVA\_HOME apuntando al directorio de java SDK.

Descargar Mondrian engine desde mondrian.pentaho.com o sourceforge.net/Project/Mondrian

En este caso trabajamos con la versión 3.2.1.13885

Descomprimimos.

Extraemos el contenido de mondrian.war en el directorio webapps/mondrian de apache (anexo instalar apache)

Si deseamos configurar la base de datos de prueba:

Necesitamos las siguientes librerías :

* eigenbase-xom.jar
* eigenbase-properties.jar
* postgresql-9.0-801.jdbc4.jar
* log4j-1.2.16.jar
* commons-logging-api-1.1.1.jar

Las librerías las colocamos en la carpeta lib en el directorio donde se descomprio mondrian.

Ejecutamos la siguiente sentencia en la línea de comandos.

java -cp "C:\mondrian\commons-logging-api-1.1.1.jar;C:\mondrian\lib\eigenbase-xom.jar;C:\mondrian\lib\eigenbase-resgen.jar;C:\mondrian\lib\eigenbase-properties.jar;C:\mondrian\lib\log4j-1.2.16.jar;C:\mondrian\postgresql-9.0-801.jdbc4.jar" mondrian.test.loader.MondrianFoodMartLoader -tables -data -indexes -jdbcDrivers="org.postgresql.Driver" -inputFile="C:\mondrian\demo\FoodMartCreateData.sql" -outputJdbcURL="jdbc:postgresql://localhost/foodmart?user=postgres&password=33410720"

Configuramos la conexión a la base de datos en los archivos web.xml y mondrian.properties ubicados en apache/mondrian/WEB-INF

En web.xml para el parámetro connectString

Provider=mondrian;Jdbc=jdbc:postgresql://localhost/foodmart?user=postgres&#38;password=33410720;JdbcDrivers=org.postgresql.Driver;Catalog=/WEB-INF/queries/FoodMart.xml

Luego debemos modificar en las consultas ubicados en los archivos jsp en WEB-INF/queries/ los archivos arrows.jsp , colors.jsp , fourhier.jsp , mondrian.jsp

<jp:mondrianQuery id="query01" jdbcDriver="org.postgresql.Driver" jdbcUrl="jdbc:postgresql://localhost/foodmart?user=postgres&password=33410720" catalogUri="/WEB-INF/queries/FoodMart.xml"

connectionPooling="false">

Contenido

[Introducción 1](#_Toc289930427)

[Definición de Inteligencia de Negocios 1](#_Toc289930428)

[Business Intelligence 1](#_Toc289930429)

[Data Warehouse 1](#_Toc289930430)

[ARQUITECTURA DE UN DATA WAREHOUSE 2](#_Toc289930431)

[Pentaho BI Suite 4](#_Toc289930432)

[Capa de presentación 5](#_Toc289930433)

[Capa de Componentes de BI 5](#_Toc289930434)

[Pentaho Analysis Services 7](#_Toc289930435)

[Arquitectura de Mondrian 7](#_Toc289930436)

[esquema xml CUBOS, Dimensiones y Medidas 9](#_Toc289930437)

[Jerarquias Niveles y Miembros 9](#_Toc289930438)

[Jerarquias 9](#_Toc289930439)

[Niveles y Miembros 9](#_Toc289930440)

[JIVE SBS 10](#_Toc289930441)

[Instalación y configuración de Mondrian 11](#_Toc289930442)