

ESCUELA POLITÉCNICA DEL EJÉRCITO

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

DESARROLLO DE UNA APLICACIÓN DE BUSINESS  
INTELLIGENCE (BI) PARA LA EMPRESA EMPAQPLAST.

Previa a la obtención del título de:

INGENIEROS EN SISTEMAS E INFORMÁTICA

POR: SR. BOADA BYRON  
SR. TITUAÑA ALVARO

Sangolquí - Ecuador

2012

## **CERTIFICACIÓN**

Certifico que el presente trabajo fue desarrollado en su totalidad por los señores, Boada Byron y Tituaña Alvaro como requerimiento parcial a la obtención del título de INGENIEROS DE SISTEMAS E INFORMÁTICA

Sangolquí, 24 de Julio del 2012

**DIRECTOR DEL PROYECTO  
ING. LORENA DUQUE**

## **AGRADECIMIENTOS**

Agradezco a Dios por esta oportunidad, a mis padres que son mi principal apoyo, a mi universidad y profesores que me dieron sus enseñanzas y mis compañeros por su amistad.

**Byron Boada**

Dedico este proyecto de tesis a Dios y mis Padres. A Dios porque es la parte principal en mi vida, a mis padres que supieron apoyarme en todo momento.

**Alvaro Tituaña**

## **GLOSARIO DE ABREVIATURAS**

BI	Business Intelligence.
ETL	Extract Transformation Load.
ERP	Enterprise Resource Planning.
MOLAP	Multidimensional Online Analytical Processing.
ROLAP	Processing Analytical On Line Relacional.
HOLAP	Hybrid Online Analytical Process.
OLAP	On-Line Analytical Processing.
ORDBMS	Object Relational Data Base Management System.
CRM	Customer Relationship Management.
GUI	Graphical User Interface.
PDI	Pentaho Data Integration.
MDX	Multidimensional Expression.
JDBC	Java Database Connectivity.
JDNI	Java Naming and Directory Interface.

## Contenido

CAPÍTULO 1 .....	- 1 -
1.1 Tema.....	- 1 -
1.2 Introducción.....	- 1 -
1.3 Planteamiento del Problema .....	- 2 -
1.4 Justificación e Importancia .....	- 3 -
1.5 Alcance.....	- 3 -
1.6 Objetivos.....	- 4 -
1.6.1 Objetivo General .....	- 4 -
1.6.2 Objetivos Específicos .....	- 4 -
CAPÍTULO 2 .....	- 5 -
MARCO TEÓRICO .....	- 5 -
2.1 Data Warehouse .....	- 5 -
2.1.1 Base de datos relacional Oracle.....	- 5 -
2.1.2 Arquitectura de la base de datos.....	- 5 -
2.1.2.1 Instancia .....	- 5 -
2.1.2.2 Base de datos. ....	- 6 -
2.1.3 Almacén de datos .....	- 7 -
2.1.3.1 Descripción de un Data Warehouse.....	- 7 -
2.1.3.2 Función de un Data Warehouse.....	- 8 -
2.1.4 Data Mart .....	- 8 -
2.1.5 Cubos OLAP.....	- 8 -
2.1.5.1 Formas de almacenamiento de los cubos .....	- 9 -
2.1.6 Estructura del cubo del almacén de datos .....	- 10 -
2.1.7 Esquemas para el modelado de datos.....	- 10 -
2.2 Business Intelligence .....	- 11 -
2.2.1 Introducción .....	- 11 -
2.2.2 Conceptos de BI .....	- 12 -
2.2.3 Definición de BI.....	- 12 -
2.2.4 Importancia de BI.....	- 13 -
2.2.5 Beneficios de BI.....	- 13 -
2.2.6 Arquitectura .....	- 14 -
2.3 Metodologías de desarrollo de un proyecto de BI.....	- 15 -
2.3.1 Elección de la Metodología para el desarrollo del proyecto .....	- 16 -

2.3.2 Metodología de Ralph Kimball .....	- 17 -
2.3.2.1. Planificación y Administración del Proyecto .....	- 18 -
2.3.2.2. Definición de los Requerimientos del Negocio .....	- 19 -
2.3.2.3 Diseño Técnico de la arquitectura.....	- 20 -
2.3.2.4 Selección de Productos e Instalación .....	- 22 -
2.3.2.5 Modelado Dimensional .....	- 23 -
2.3.2.6 Diseño Físico .....	- 26 -
2.3.2.7 Diseño y Desarrollo de la Presentación de Datos .....	- 26 -
2.3.2.8 Especificación de Aplicaciones para Usuarios Finales .....	- 27 -
2.3.2.9 Desarrollo de Aplicaciones para Usuarios Finales .....	- 27 -
2.3.2.10. Implementación .....	- 28 -
2.3.2.11. Mantenimiento y crecimiento .....	- 28 -
2.3.2.12. Gestión del proyecto .....	- 28 -
2.3.2.13. Formato para toma de requerimientos.....	- 28 -
2.4 Herramientas .....	- 29 -
2.4. 1 Plataforma Pentaho Open Source Business Intelligence .....	- 30 -
2.4.2 Módulos de la plataforma Pentaho BI .....	- 30 -
2.4.2.1 Reporting.....	- 30 -
2.4.2.2 Análisis.....	- 31 -
2.4.2.3 Mondrian.....	- 32 -
2.4.2.4 Dashboards .....	- 34 -
2.4.2.5 Integración de Datos .....	- 35 -
2.4.2.6 Diseño de cubos.....	- 37 -
2.4.2.7 Base de datos .....	- 37 -
2.4.2.8 Modelado de datos.....	- 38 -
2.4.2.9 Servidor de Base de datos. ....	- 39 -
CAPÍTULO 3 .....	- 40 -
DESARROLLO DEL PROYECTO .....	- 40 -
3.1 Desarrollo del BI para la empresa Empaqplast .....	- 40 -
3.2 Planeación y administración del proyecto .....	- 40 -
3.2.1 Definición del Proyecto .....	- 40 -
3.2.2 Preparación para un proyecto de Data Warehouse. ....	- 40 -
3.2.3 Alcance .....	- 40 -
3.2.4 Justificación .....	- 41 -
3.2.5 Planificación del Proyecto .....	- 42 -

3.2.6	Administración del proyecto .....	- 42 -
3.3	Definición de los Requerimientos del Negocio .....	- 42 -
3.3.1	Requerimientos área de ventas .....	- 43 -
3.3.2	Requerimientos área de compras.....	- 46 -
3.3.3	Requerimientos área de inventarios.....	- 50 -
3.3.4	Análisis y requerimientos.....	- 51 -
3.3.5	Esquema de la base de datos original .....	- 52 -
3.4	Diseño técnico de la Arquitectura .....	- 57 -
3.4.1	Estándares .....	- 57 -
3.4.1.1	Estándar para el modelado .....	- 57 -
3.4.1.2	Estándar de las secuencias.....	- 60 -
3.4.1.3	Estándar para el Proceso ETL .....	- 60 -
3.4.2	Entorno Back Room .....	- 63 -
3.4.3	Entorno Front Room.....	- 64 -
3.5	Selección de Productos e Instalación .....	- 64 -
3.6	Modelado Dimensional .....	- 66 -
3.6.1	Bus Matrix Data Warehouse .....	- 66 -
3.6.2	Modelado Dimensional.....	- 67 -
3.6.2.1	Diagrama general origen de datos Data Mart ventas por factura.....	- 67 -
3.6.2.2	Mapeo de datos ventas por factura.....	- 68 -
3.6.2.3	Modelo dimensional Data Mart ventas por factura.....	- 69 -
3.6.2.4	Mapeo y detalle datos de dimensiones y hechos Data Mart ventas por factura..	- 70 -
3.6.2.5	Modelo de base de datos Data Mart ventas por factura. ....	- 74 -
3.6.2.6	Diagrama general origen de datos Data Mart ventas por remisión.....	- 75 -
3.6.2.7	Mapeo de datos ventas por remisión.....	- 75 -
3.6.2.8	Modelo dimensional Data Mart ventas por remisión.....	76
3.6.2.9	Mapeo de datos de dimensiones y hechos Data Mart ventas por remisión .....	- 78 -
3.6.2.10	Modelo de base de datos Data Mart ventas por remisión.....	- 82 -
3.6.2.11	Diagrama general origen de datos Data Mart compras por factura.....	- 83 -
3.6.2.12	Mapeo de datos compras por factura .....	- 83 -
3.6.2.13	Modelo dimensional Data Mart compras por factura.....	- 85 -
3.6.2.14	Mapeo de datos de dimensiones y hechos Data Mart compras por factura .....	- 86 -
3.6.2.15	Modelo de base de datos Data Mart compras por factura .....	- 90 -
3.6.2.16	Diagrama general origen de datos Data Mart inventario por bodega.....	- 91 -
3.6.2.17	Mapeo de datos compras por factura .....	- 91 -

3.6.2.18 Modelo dimensional Data Mart inventario por bodega.....	- 93 -
3.6.2.19 Mapeo de datos de dimensiones y hechos Data Mart inventario por bodega. ....	- 94 -
3.6.2.20 Modelo de base de datos Data Mart inventario por bodega .....	- 97 -
3.6.3 Diseño Físico.....	- 98 -
3.6.4 Diseño y desarrollo de la presentación de datos.....	- 99 -
3.6.4.1 Proceso de Extracción Transformación y Carga.....	- 99 -
3.6.5 Especificación de Aplicaciones para Usuarios Finales.....	- 137 -
3.6.6 Desarrollo de Aplicaciones para Usuarios Finales .....	- 143 -
3.6.7 Implementación.....	- 148 -
3.6.8 Mantenimiento y crecimiento.....	- 148 -
3.6.9 Gestión del Proyecto.....	- 148 -
<b>CAPÍTULO 4 .....</b>	<b>- 149 -</b>
<b>CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>- 149 -</b>
4.1 Conclusiones.....	- 149 -
4.2 Recomendaciones.....	- 150 -
<b>Bibliografía.....</b>	<b>- 151 -</b>
<b>Webgrafía.....</b>	<b>- 152 -</b>



## LISTADO DE TABLAS

Tabla 2.1 Formato para la toma de requerimientos.....	29
Tabla 2.2 Listado de herramientas para el desarrollo del proyecto.....	29
Tabla 2.3 Servidor p bladesystem C3000.....	39
Tabla 3.1 R001 área de ventas.....	43
Tabla 3.2 R002 área de ventas.....	43
Tabla 3.3 R003 área de ventas.....	43
Tabla 3.4 R004 área de ventas.....	44
Tabla 3.5 R005 área de ventas.....	44
Tabla 3.6 R006 área de ventas.....	44
Tabla 3.7 R007 área de ventas.....	45
Tabla 3.8 R008 área de ventas.....	45
Tabla 3.9 R009 área de ventas.....	45
Tabla 3.10 R0010 área de ventas.....	46
Tabla 3.11 R0011 área de ventas.....	46
Tabla 3.12 R0012 área de compras .....	46
Tabla 3.13 R0013 área de compras .....	47
Tabla 3.14 R0014 área de compras .....	47
Tabla 3.15 R0015 área de compras .....	47
Tabla 3.16 R0016 área de compras .....	48
Tabla 3.17 R0017 área de compras .....	48
Tabla 3.18 R0018 área de compras .....	48
Tabla 3.19 R0019 área de compras .....	49
Tabla 3.20 R0020 área de compras .....	49
Tabla 3.21 R0021 área de compras .....	49
Tabla 3.22 R0022 área de inventarios .....	50
Tabla 3.23 R0023 área de inventarios .....	50
Tabla 3.24 Características del servidor de base de datos.....	51
Tabla 3.25 Nombres para tablas según el estándar.....	58
Tabla 3.26 Estándar para campos en las dimensiones.....	59
Tabla 3.27 Estándar de nombres para campos en las tablas de hechos.....	59
Tabla 3.28 Estándar para nombre para secuencias .....	60
Tabla 3.29 Estándar para el proceso etl de una dimensión.....	61
Tabla 3.30 Estándar para el proceso etl de una tabla de hechos.....	62
Tabla 3.31 Productos de instalación.....	65

## LISTADO DE FIGURAS.

Figura 2.1 Arquitectura del BI .....	14
Figura 2.2 Ciclo de vida de la metodología de Ralph Kimball.....	17
Figura 2.3 Arquitectura back room.....	21
Figura 2.4 Arquitectura front room.....	22
Figura 2.5 Herramientas de la suite de pentaho.....	30
Figura 2.6 Carpeta contenedora report designer.....	31
Figura 2.7 Arquitectura servidor mondrian.....	32
Figura 2.8 Cubo mondrian.....	33
Figura 2.9 Ejemplo dashboard .....	35
Figura 2.10 Ambiente gráfico spoon.....	36
Figura 2.11 Schema Workbench.....	37
Figura 2.12 Oracle 10g.....	38
Figura 2.13 Sybase-PowerDesigner.....	39
Figura 3.1 Topología de red BI.....	51
Figura 3.2 Diagrama de datos origen del cubo de ventas por factura.....	53
Figura 3.3 Diagrama de datos origen del cubo de ventas por remisión.....	54
Figura 3.4 Diagrama de datos origen del cubo de compras por factura.....	55
Figura 3.5 Diagrama de datos origen del cubo de inventarios por bodega.....	56
Figura 3.6 Entorno back room.....	63
Figura 3.7 Entorno front room.....	64
Figura 3.48 Editar la conexión a bases de datos Sec_Auditoria.....	114
Figura 3.49 Selección de un Table Output.....	114
Figura 3.50 Editar la conexión Table Output.....	115
Figura 3.51 Probar la conexión correcta Table Output.....	115
Figura 3.52 Conexión correcta Table Output.....	116
Figura 3.53 Seleccionar el esquema de la base de datos.....	116
Figura 3.54 Selección de tabla de Auditoria.....	117
Figura 3.55 Carga ETL para Dim_Cliente.....	117
Figura 3.56 Selección de un Table Input.....	118
Figura 3.57 Editar la conexión a la base de datos Table Input.....	118
Figura 3.58 Probar la conexión Table Input.....	119
Figura 3.59 Conexión correcta Table Input .....	119
Figura 3.60 Preview para ejecutar el select hacia la tabla c_bpartner.....	120
Figura 3.61 Presentación de Datos de la tabla c_bpartner.....	121
Figura 3.62 Insertar un Database Lookup.....	122
Figura 3.63 Editar la conexión en el database lookup.....	122
Figura 3.64 Join entre las tablas dim_cliente y c_bpartner.....	122

Figura 3.65 Asignar sk al cliente.....	123
Figura 3.66 Cálculos y medidas para la tabla de hechos.....	123
Figura 3.67 Asignación de las medidas y cálculos.....	123
Figura 3.68 Parámetros del sistema para la tabla de auditoría.....	124
Figura 3.69 Datos del sistema para la carga de una tabla de hechos.....	124
Figura 3.70 Datos del sistema para la tabla de auditoría.....	122
Figura 3.71 Crear un Insert/Update.....	125
Figura 3.72 Editar la conexión a la base Insert/Update.....	125
Figura 3.73 Verificar la conexión Insert/Update.....	126
Figura3.74 conexión correcta Insert/Update .....	126
Figura 3.75 Esquema DW.....	124
Figura 3.76 Seleccionar tabla Fact_venta_fac.....	127
Figura 3.77 Unión de las claves primarias.....	128
Figura 3.78 Añadir una secuencia.....	128
Figura 3.79 Secuencia de error para la carga de la Fact_Venta_Fac.....	129
Figura 3.80 Campos de error.....	129
Figura 3.81 Editar la conexión secuencia.....	130
Figura 3.82 Selección de un Table Output para la tabla de auditoría.....	130
Figura 3.83 Editar la conexión Table Output.....	131
Figura 3.84 Probar la conexión correcta Table Output.....	131
Figura 3.85 Conexión correcta Table Output.....	132
Figura 3.86 Seleccionar el esquema de la base de datos.....	132
Figura 3.87 Selección de tabla de Auditoria.....	133
Figura 3.88 Carga ETL para Fact_Venta_Fac.....	133
Figura 3.89 Tiempos de ejecución ETL.....	134
Figura 3.90 Transformación por cada Dimensión y Fact.....	134
Figura 3.91 Selección de la Dimensión o Fact.....	135
Figura 3.92 Éxito de carga.....	135
Figura 3.93 Interfaz Schema Workbench.....	136
Figura 3.94 Menú principal consola de administración.....	137
Figura 3.95 Menú para roles y usuarios.....	138
Figura 3.96 Pantalla de login pentaho bi server.....	138
Figura 3.97 Autenticación de usuario y contraseña.....	139
Figura 3.98 Menú principal pentaho bi server.....	139
Figura 3.99 Menú de herramientas BI.....	140
Figura 3.100 Nueva vista de análisis.....	140
Figura 3.101 Vista de un cubo.....	141
Figura 3.102 Opción para exportar a PDF.....	141
Figura 3.103 Vista de un reporte.....	142
Figura 3.104 Vista de un dashboard.....	142

Figura 3.105 Logo Report Designer.....	143
Figura 3.106 Menu pentaho report designer.....	144
Figura 3.107 Creación nuevo dashboard.....	145
Figura3.108 Perspectivas para el diseño de tableros de control.....	146
Figura 3.109 Vista previa del tablero de control.....	146
Figura 3.110 Vista previa diseño tablero de control.....	147
Figura 3.111 Dashboard completo.....	147

## LISTADO DE ANEXOS

<b>Anexo A</b>	Entorno de pruebas .....	152
<b>Anexo B</b>	Consultas ETL .....	152
<b>Anexo C</b>	Script base de datos .....	152
<b>Anexo D</b>	Constancia de los requerimientos .....	152
<b>Anexo E</b>	Constancia de las reuniones.....	152
<b>Anexo F</b>	Manual instalación productos pentaho .....	152
<b>Anexo G</b>	Manual técnico de Schema workbench.....	152
<b>Anexo H</b>	Manual técnico de Report Designer .....	152
<b>Anexo I</b>	Manual técnico CDE.....	152
<b>Anexo J</b>	Manual técnico servidor Mondrian.....	152

## RESUMEN

El proyecto de tesis “Desarrollo de una aplicación de Business Intelligence para la empresa Empaqplast” fue desarrollado con la finalidad de dar soporte en la toma de decisiones, para las gerencias de las áreas de negocio de compras, ventas e inventarios.

Para el desarrollo del proyecto se utilizó la metodología de Ralph Kimball, ya que es una de las más usadas, seguras y probadas al momento de implementar un proyecto de Business Intelligence, cubriendo todas las fases de ciclo de vida de un proyecto de BI. Empezando por la planificación hasta el mantenimiento y administración de la aplicación.

Todas las etapas del proyecto se realizaron con las herramientas de la suite de Pentaho. Pentaho Data Integration para el proceso de ETL, Pentaho Schema Workbench para el diseño y creación de los cubos multidimensionales, Pentaho Report Designer para el diseño y construcción de reportes, Pentaho Dashboard Editor para el diseño y creación de tableros de control y por último Pentaho BI Server para la publicación y visualización de los resultados.

El Datawarehouse está almacenado en una base de datos Oracle 10g perteneciente a la empresa Empaqplast.

Actualmente la empresa hace uso de la aplicación para generar reportes de las áreas de negocio involucradas, vistas de análisis, y monitoreo del negocio mediante los tableros de control, logrando así un acceso a información organizada, depurada en tiempo real para la toma de decisiones acertadas.

# CAPÍTULO 1

## 1.1 Tema

Desarrollo de una aplicación de Business Intelligence para la empresa EMPAQPLAST.

## 1.2 Introducción

La empresa EMPAQPLAST S.A.[1] se creó en Quito, Ecuador en el año de 1992 para atender las necesidades de los fabricantes de aceites comestibles y productos de limpieza localizados en la sierra. Esta necesidad surgió debido a la falta de abastecimiento en la región sierra y para facilitar la logística, requerimientos de entrega y cumplimiento de los productores locales.

La empresa se inició con cuatro máquinas de la más avanzada tecnología que incluía sopladoras para envases de PVC y sus respectivas tapas. Con el transcurrir del tiempo el mercado de envases fue creciendo, llevando a la empresa a un crecimiento a nivel nacional.

Entre sus características está el buen servicio, calidad y variedad de sus productos. Incursionando en las áreas de soplado[2], inyección[3], extrusión[4], coextrusión[5], e impresión[6].

El volumen de ventas que tiene la empresa le ha permitido en una década ser la empresa fabricante de plásticos más grande del país, apoyándose en su infraestructura y tecnología. Entregando productos certificados y aptos para el uso humano.

[1] Nombre Propio de la Empresa de plásticos.

[2] proceso utilizado para fabricar piezas de plástico huecas gracias a la expansión del material. Esto se consigue por medio de la presión que ejerce el aire en las paredes de la preforma.

[3] proceso semicontinuo que consiste en inyectar un polímero, cerámico o un metal en estado fundido (o ahulado) en un molde cerrado a presión y frío, a través de un orificio pequeño llamado compuerta.

[4] proceso industrial, en donde se realiza una acción de prensado, moldeado del plástico, que por flujo continuo con presión y empuje, se lo hace pasar por un molde encargado de darle la forma deseada.

[5] Proceso que se hace con el polietileno para la creación de láminas plásticas.

[6] Proceso de Impresión de marcas o nombres en las fajillas plásticas.

Trabaja con una gran variedad de materiales. Entre éstos están el Tereftalato de Polietileno (PET)[7], Policarbonato (PC)[8], Polipropileno (PP)[9], Polietileno de Alta Densidad (PEAD)[10], Polietileno de Baja Densidad (PEBD)[11], Estireno Butadieno Acrilonitrilo (ABS)[12], y Cloruro de Polivinilo (PVC)[13].

### **1.3 Planteamiento del Problema**

La empresa EMPAQPLAST actualmente no cuenta con el suficiente flujo de información para las gerencias de los departamentos de ventas, compras e inventarios. Entendiendo que la información no se encuentra estructurada y procesada.

Los datos están almacenados en bases de datos operacionales y no se tiene la facilidad para el análisis de una forma específica y personalizada por cada departamento. Los gerentes requieren tener el acceso a la información de una manera más personalizada, debido a que en algunas ocasiones se ha perdido tiempo en tomar acciones en eventualidades por la falta inmediata de información estructurada, de forma que se pueda analizar y tener un soporte en la toma de decisiones.

Uno de los problemas principales se da en cuanto a la generación de reportes, estos son realizados de una forma manual, lo que requiere tiempo para el área de sistemas en la generación de los mismos además causando un gran tráfico en la base de datos de producción. Reflejándose en el tiempo de espera de cada consulta realizada a la base de datos.

[7] es un tipo de plástico muy usado en envases de bebidas.

[8] grupo de termoplásticos fácil de trabajar, moldear y termoformar, y son utilizados ampliamente en la manufactura moderna.

[9] Pertenecen al grupo de las poliolefinas y es utilizado en una amplia variedad de aplicaciones que incluyen empaques para alimentos, tejidos, equipo de laboratorio.

[10] Este material se encuentran en envases plásticos desechables.

[11] Es un polímerotermoplástico conformado por unidades repetitivas de etileno.

[12]plástico muy resistente al impacto (golpes) muy utilizado en automoción y otros usos tanto industriales como domésticos. Es un termoplástico amorfo.

[13] polímero por adición y además una resina que resulta de la polimerización del cloruro de vinilo o cloroeteno.



## **1.4 Justificación e Importancia**

El proyecto propuesto brindará el soporte para la toma de decisiones gerenciales. De las áreas de ventas, inventarios y compras.

La solución que se plantea para el análisis de la información está basada en la elaboración de una aplicación de Business Intelligence, que estará conformado por los Data Mart<sup>[1]</sup> de las áreas ya mencionadas. Se busca relacionar los datos con el negocio para obtener información relevante sobre la situación de la empresa. Tomando en cuenta que la empresa trabaja con un sistema ERP <sup>[2]</sup> (OpenSource) llamado Open Bravo. Se considerará el uso de la herramienta Pentaho (OpenSource) esta herramienta facilitará el camino para conseguir una solución completa de Business Intelligence y una rápida integración con la infraestructura que existe actualmente en la empresa.

Aplicando este sistema de Business Intelligence, se pretende reducir los costos y optimizar los tiempos en lo que se refiere al tratamiento de la información se facilitará la administración de la misma personalizándola y adaptándola a las necesidades lo que permitirá la Integración y depuración de los datos.

## **1.5 Alcance**

Se realizará el desarrollo de una aplicación de Business Intelligence, para el soporte en la toma de decisiones en la empresa EMPAQPLAST.

Debido a la dimensión de la empresa, su flujo de datos y cantidad de transacciones el aplicativo abarcará las siguientes áreas:

- Ventas.
- Inventarios.
- Compras.

[1] Son subconjuntos de datos con el propósito de ayudar a que un área específica dentro del negocio pueda tomar mejores decisiones

[2] sistemas de información gerenciales que integran y manejan muchos de los negocios asociados con las operaciones de producción y de los aspectos de distribución de una compañía en la producción de bienes o servicios.

Para cada una de estas áreas se desarrollará, los cubos de información, siendo los gerentes de cada departamento los responsables del análisis de los cubos.

El proyecto se desarrollará con el uso de las Herramientas Open Source Pentaho en todas sus etapas.

Se desarrollará los Data Mart de las áreas de ventas, inventarios y compras que conformarán el Data Warehouse.

Se generarán reportes que se adaptarán a las necesidades de información concerniente a las áreas ya mencionadas.

El análisis y despliegue de la información para los usuarios finales se la realizará a través de la consola de usuario de Pentaho.

El proyecto incluye capacitación básica los usuarios.

## **1.6 Objetivos**

### **1.6.1 Objetivo General**

Desarrollar una aplicación de Business Intelligence utilizando la metodología de Ralph Kimball con el uso de herramientas Open Source de la suite Pentaho para la empresa EMPAQPLAST.

### **1.6.2 Objetivos Específicos**

- Realizar el levantamiento y análisis de los requerimientos para la construcción de la aplicación Business Intelligence.
- Utilizar la metodología de Ralph Kimball para el desarrollo del proyecto.
- Diseñar el Data Warehouse para las áreas de Ventas, Inventarios y Compras.
- Realizar un entorno de pruebas con la participación de los representantes de las áreas de ventas, inventarios y compras.

## **CAPÍTULO 2**

### **MARCO TEÓRICO**

#### **2.1 Data Warehouse**

##### **2.1.1 Base de datos relacional Oracle**

La base de datos Oracle es un sistema de administración de base de datos relacionales (ORDBMS acrónimo en inglés de Object Relational Data Base Management System). Este modelo consiste en utilizar tablas bidimensionales para almacenar información. Oracle es uno de los sistemas de bases de datos más completos, destacando:

- Soporte de transacciones.
- Estabilidad.
- Escalabilidad.
- Soporte multiplataforma.

##### **2.1.2 Arquitectura de la base de datos**

Un servidor Oracle está conformado por la instancia y la base de datos. Durante el proceso de creación de una base de datos, se crea primero la instancia, por último se crea la base de datos.

###### **2.1.2.1 Instancia**

La instancia de Oracle consta de un área de memoria compartida conocida como System Global Area (SGA) esta área de memoria consta de tres estructuras básicas:

###### **Shared Pool**

Incluye el Library Cache (área de memoria que almacena código recientemente ejecutado) y el Data Dictionary Cache (área que almacena las definiciones de objetos usados recientemente).

#### **2.1.2.2 Base de datos.**

Esta es la estructura física y se divide en dos tipos de archivos, requeridos y externos.

##### **Archivos requeridos:**

###### **Control File**

Almacena el status de las estructuras físicas de la base de datos.

###### **Online Redo Log File**

Almacenan un registro de los cambios realizados a la base de datos.

###### **Datafiles**

Son el repositorio de la información.

##### **Archivos externos:**

###### **Parameter File**

Define la instancia y los parámetros de inicialización. Hay de dos tipos dinámico (binario, que no se puede ejecutar y se actualiza constantemente) y estático (solamente es leído una sola vez cuando la instancia se inicia).

###### **Password File**

Archivo de sistema que almacena los nombres de usuario y contraseña.

###### **Archive Log Files**

Copias de los Online Red o Log Files llenos.

## 2.1.3 Almacén de datos

### 2.1.3.1 Descripción de un Data Warehouse

“En el contexto de la informática, un almacén de datos (del inglés *Data Warehouse*) es una colección de datos orientada a un determinado ámbito (empresa, organización, etc.), integrado, no volátil y variable en el tiempo, que ayuda a la toma de decisiones en la entidad en la que se utiliza. Se trata, sobre todo, de un expediente completo de una organización, más allá de la información transaccional y operacional, almacenado en una base de datos diseñada para favorecer el análisis y la divulgación eficiente de datos (especialmente OLAP, *procesamiento analítico en línea*). El almacenamiento de los datos no debe usarse con datos de uso actual. Los almacenes de datos contienen a menudo grandes cantidades de información que se subdividen a veces en unidades lógicas más pequeñas dependiendo del subsistema de la entidad del que procedan o para el que sean necesario”. [1]

Data Warehouse (almacén de datos) puede verse como un repositorio de datos.

**Orientado a temas:** Los datos en la base de datos están organizados de manera que todos los elementos de datos relativos al mismo evento u objeto del mundo real queden unidos entre sí.

**Variante en el tiempo:** Registra los cambios que se producen a lo largo del tiempo, reflejando los datos originales en el informe final.

**No volátil:** esta es información solo de lectura, no se puede eliminar ni modificar una vez almacenada.

**Integrado.-** La base de datos contiene los datos de los sistemas operacionales que existen en la organización, estos datos deben ser consistentes.

[1] Wikipedia

En forma general un Data Warehouse es un almacén de datos, pero se debe tomar muy en cuenta cuales son los medios para obtener y analizar esos datos.

Para lo cual se utilizan herramientas para extraer, transformar y cargar los datos en el almacén de datos (Data Warehouse) y herramientas para recuperar los metadatos.

### **2.1.3.2 Función de un Data Warehouse**

En la creación de un Data Warehouse lo que se busca es almacenar los datos que son necesarios o útiles para la organización. De esta manera se trata de crear un repositorio de datos estructurados y depurados y convertir estos datos en información útil para el usuario final. El usuario puede realizar consultas e informes de manera óptima y segura.

### **2.1.4 Data Mart**

Los Data Marts son bases de datos departamentales es decir subconjuntos de datos de áreas específicas de la organización, se caracteriza por tener una estructura óptima de los datos que pueden ser alimentados desde una base de datos transaccional y son los formarán parte de un Data Warehouse.

#### **Características de un Data Mart**

- Mayor rapidez de consulta.
- Área específica.
- Tiene un propósito específico.
- Consultas SQL sencillas.
- Permite llevar un historial de la información.

### **2.1.5 Cubos OLAP**

Los cubos OLAP (*OnLine Analytical Processing*) se crean en función a bases de datos multidimensionales, que permiten procesar grandes volúmenes de información, en

campos bien definidos, dando un acceso inmediato a los datos para su consulta y posterior análisis.

Esta estructura multidimensional de los cubos OLAP es gracias a que son formados por vectores.

#### **2.1.5.1 Formas de almacenamiento de los cubos**

##### **MOLAP**

Los datos fuente del cubo son almacenados junto con sus agregaciones, en una estructura multidimensional de alto rendimiento. El almacenaje de MOLAP, provee excelente rendimiento y compresión de datos.

Tiene el mejor tiempo de respuesta, dependiendo solo en el porcentaje y diseño de las agregaciones del cubo. En general este método, es muy apropiado para cubos con uso frecuente por su rápida respuesta.

##### **ROLAP**

Toda la información del cubo, sus datos, su agregación, sumas son almacenados en una base de datos relacional. ROLAP no almacena copia de la base de datos, tiene acceso a las tablas originales cuando necesita responder a preguntas, es generalmente, mucho más lenta que las otras dos estrategias de almacenaje. Típicamente ROLAP se usa, para largos conjuntos de datos que no son frecuentemente buscados, tales como datos históricos de de los años más recientes.

##### **HOLAP**

HOLAP combina atributos de MOLAP y ROLAP, la agregación de datos es almacenada en una estructura multidimensional usada por MOLAP, y la base de datos fuentes, en una base de datos relacional. Para procedimientos de búsqueda que accesan datos sumariados, HOLAP es equivalente a MOLAP, por el contrario

estos procesos accederán datos fuentes como los drilldown, estos deben de buscar los datos en la base de datos relacional y esto no es tan rápido comparado a si los datos estuvieran almacenados en una estructura MOLAP.

Los cubos almacenados como HOLAP, son más pequeños que los MOLAP y responden más rápidos que los ROLAP. HOLAP es generalmente usado para cubos que requieren rápida respuesta, para sumalizaciones basadas en una gran cantidad de datos.

### **2.1.6 Estructura del cubo del almacén de datos**

**Tablas de hechos:** Contiene todas las claves primarias de cada una de las dimensiones y las medidas que nos ayudarán en el análisis de los cubos.

#### **Tablas de dimensiones:**

En estas tablas encontraremos todos los valores que se utilizarán en la tabla de hechos.

#### **Relaciones de la tabla de hechos**

Las tablas de hechos del almacén de datos se relacionan entre sí a través de las dimensiones que comparten.

### **2.1.7 Esquemas para el modelado de datos**

#### **Esquema en estrella**

Este esquema se caracteriza por tener una tabla de hechos en el centro de su estructura, a la cual llegan las tablas de dimensiones mediante las claves primarias simples de cada dimensión. Concentrando de esta manera la información en una tabla denominada tabla de hechos.



## **Esquema en copo de nieve**

En este esquema encontramos la particularidad que una tabla de dimensiones puede estar formada por más de una tabla de datos lo que le hace diferente a el esquema de estrella permitiéndonos tener varios caminos para llegar a los datos influyendo directamente sobre el rendimiento de las consultas.

## **2.2 Business Intelligence**

### **2.2.1 Introducción**

En el mundo de los negocios actual, las empresas que quieren mantenerse en un buen sitio y ser competitivas no solo deben caracterizarse por la calidad de sus productos sino también por el grado de información que se maneja con sus clientes, empleados, gerentes y socios. En el caso de los directivos de la empresas, se tienen que enfrentar ante ciertos escenarios como disponer de más información pero menos tiempo para analizarla, sistemas de información que no ayuda a la toma de decisiones ágiles y además responsables de generar información urgente en muchos de los casos están saturados por las peticiones de información y no pueden cumplir con todas las peticiones.

Es a partir de estos problemas que nace el concepto de Inteligencia de Negocios o sus siglas en inglés (Business Intelligence BI) el cual engloba los sistemas de información de una empresa para obtener algo más que información, se lo usa para obtener conocimiento.

Las empresas en los últimos años han hecho grandes inversiones en sistemas ERP (Enterprise Resource Planning) y CRM (Customer Relationship Management) los cuales proveen una gran cantidad de datos para las empresas, las cuales ahora desean poder usar esta gran cantidad de información para la toma de decisiones y acciones para un mejor desempeño de sus negocios. Por dichas razones se están adoptando en las empresas en uso de sistemas BI.

### **2.2.2 Conceptos de BI**

“Las aplicaciones de Business Intelligence (BI) son herramientas de soporte de decisiones que permiten en tiempo real, acceso interactivo, análisis y manipulación de información crítica para la empresa. Estas aplicaciones proporcionan a los usuarios un mayor entendimiento que les permite identificar las oportunidades y los problemas de los negocios. Los usuarios son capaces de acceder y apalancar una vasta cantidad de información y analizar sus relaciones y entender las tendencias que últimamente están apoyando las decisiones de los negocios. Estas herramientas previenen una potencial pérdida de conocimiento dentro de la empresa que resulta de una acumulación masiva reinformación que no es fácil de leer o de usar” [1]

“Business Intelligence es la habilidad de consolidar información y analizarla con la suficiente velocidad y precisión para descubrir ventajas y tomar mejores decisiones de negocios. Definición compatible con la necesidad actual de los negocios que ante la presión de ser cada día más competitivos, para mantenerse tienen la doble tarea no sólo de permanecer sino de ser lucrativos” [2]

"Business Intelligence es simplemente la habilidad de los usuarios finales para acceder y analizar tipos cuantitativos de información y ser capaz de actuar en consecuencia" [3]

### **2.2.3 Definición de BI**

Se puede definir de una manera simplificada que BI es el procesamiento de grandes cantidades de datos, utilizando metodologías, herramientas y tecnología que servirá para transformarlos y convertirlos en conocimiento para la toma de decisiones y acciones en tiempo real para beneficio de la empresa.

[1] (CherryTree& Co., 2000)

[2] (Cano, 1999)

[3] (Grupo Gartner, Howard Dresner, citado en Hilson, 2001)

#### **2.2.4 Importancia de BI**

Generalmente, en las organizaciones se genera una gran cantidad de datos e información que en muchos de los casos el análisis de la misma se convierte en un verdadero problema para los directivos.

Las tecnologías y los sistemas de BI permiten realizar un análisis mucho más ágil y comprensible para la toma de decisiones empresariales, las aplicaciones BI buscan incrementar la eficiencia en la organización. Podemos decir que la información, correctamente analizada e interpretada, es la mayor fuente de poder de las empresas, ya que da pistas muy claras acerca del camino a seguir en futuras acciones.

#### **2.2.5 Beneficios de BI**

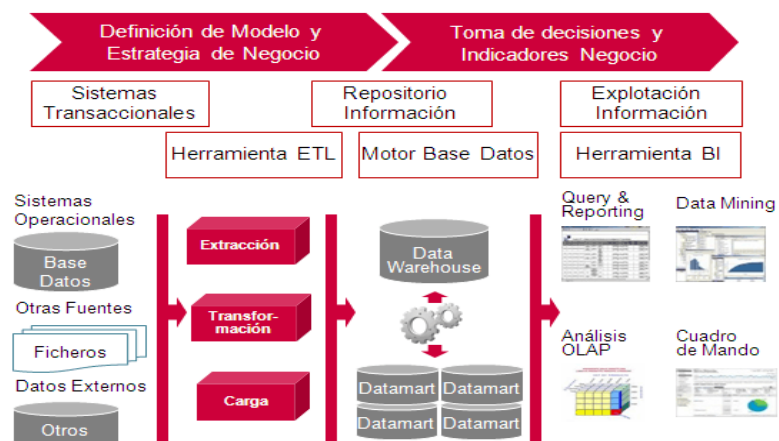
Entre los beneficios más importantes que brinda una aplicación BI a las organizaciones, se puede mencionar los siguientes:

- Minimiza el tiempo de carga de datos, debido a que todos los datos se encuentran en un mismo repositorio o fuente de información.
- Los procesos de extracción y carga de la información son automáticos debido al uso de procesos definidos y metodologías.
- Las herramientas BI permiten realizar análisis, y establecer comparaciones para la toma de decisiones.
- Permite a los usuarios no depender de reportes o informes programados, porque los mismos serán generados de manera dinámica.
- Posibilita la formulación preguntas y respuestas que son claves para el desempeño de la organización.
- Permite acceder y analizar directamente los indicadores de éxito.

- Identifica cuáles son los factores que inciden en el buen o mal funcionamiento de la organización.
- Permite detectar situaciones fuera de lo normal.
- Predice el comportamiento futuro con un alto porcentaje de certeza, basado en el entendimiento del pasado.
- Permite consultar y analizar los datos de manera sencilla e intuitiva.
- Elimina los gastos innecesarios en la organización.
- Aumenta ingresos y producción de bienes.
- Mejora el conocimiento de la empresa sobre clientes, empleados, costos etc. Lo que facilita la gestión de los directivos.

## 2.2.6 Arquitectura

Para crear una aplicación BI se muestra la siguiente figura de su arquitectura.



**Figura 2.1: Arquitectura del BI [1]**

[1] © Atos Spain

Una solución BI empieza, desde los sistemas de origen o los sistemas operacionales de la organización es decir las bases de datos, archivos planos, hojas de cálculo, sistemas ERP que son los que generan datos de la organización. Sobre los datos obtenidos se realiza un proceso de extracción de los datos de sus diferentes fuentes, transformación que consiste en una estandarización de los datos y carga de los datos en un nuevo repositorio como un Data Warehouse o en varios Data Marts para de esta manera ser estructurados y presentados a los usuarios finales en forma de Reportes, Tableros de mando, etc.

### **2.3 Metodologías de desarrollo de un proyecto de BI.**

Para la selección de una metodología de desarrollo de un proyecto de BI se debe tomar en cuenta varios factores como, las necesidades de los usuarios en lo que se refiere a la presentación de informes y análisis, también se debe entender cómo están estructurados los datos de los diferentes sistemas dentro de la organización y como se dan las relaciones entre los mismos.

Existen varias metodologías para el desarrollo de un proyecto estas se clasifican en dos grupos: metodologías Top-down y metodologías Bottom-up en cada una de estas metodologías existen autores que son los más representativos Bill Inmon “Top-down” y Ralph Kimball “Bottom up” que son considerados precursores de las metodologías y del desarrollo de Data Warehouse.

El enfoque de Bill Inmon principalmente está basado en que el Data Warehouse debe ayudar a las necesidades de todos los usuarios en la organización y no solo de un grupo particular. Se trata de un método que minimiza los problemas de integración pero es costoso debido a que se trabaja con una gran cantidad de datos. Este método realiza un resumen del sistema sin especificar detalles y después cada parte se va refinando con detalle. Esta metodología no cumple el ciclo de vida normal de las aplicaciones sino que los requisitos acompañan al proyecto según este vaya comprobándose su necesidad muchas veces esta perspectiva puede atraer riesgos a organización ya que se invierte grandes

esfuerzos para el desarrollo de un Data Warehouse y solo hasta la aparición de los Data Marts es cuando se empieza a tener los beneficios y el retorno de la inversión.

Por la contraparte el enfoque de la metodología de Ralph Kimball está basada en la elaboración de experimentos y prototipos, no requiere grandes inversiones por que la idea consiste en construir Data Marts independientes que se diseñan con detalle y después se relacionen con otros Data Marts para formar un sistema completo.

Ya descritas las metodologías más utilizadas para el desarrollo un Data Warehouse, se podrá seleccionar cuál de ellas es la que se va ajustar mejor al planteamiento de proyecto y brindará mayores ventajas.

Debido a la existencia de mayores fuentes documentación y casos de éxito en diferentes sectores de negocio se seleccionará de entre la metodología de Ralph Kimball y la metodología de Bill Inmon, la que mejor se adapte a las necesidades del proyecto.

### **2.3.1 Elección de la Metodología para el desarrollo del proyecto**

Tomando en cuenta que la metodología de Ralph Kimball conduce a una solución completa de BI en un corto periodo de tiempo, acceso a documentación que se puede encontrar y a los numerosos ejemplos aportados en diferentes entornos de negocio, esta metodología permite encontrar una respuesta a casi todas las preguntas que puedan surgir, sobre todo cuando no se dispone de la experiencia previa necesaria.

Por otro lado, este tipo de metodología **bottom-up** permite que, partiendo de cero, se pueda empezar a obtener información útil en cuestión de días y después de los prototipos iniciales, comenzar el ciclo de vida normal que nos ofrezca una solución completa de BI.

Los Data Marts resultantes son fácilmente consultables tanto para los desarrolladores como para los usuarios finales. La relación directa entre los hechos y dimensiones conceden a cualquier usuario la posibilidad de construir consultas sencillas.

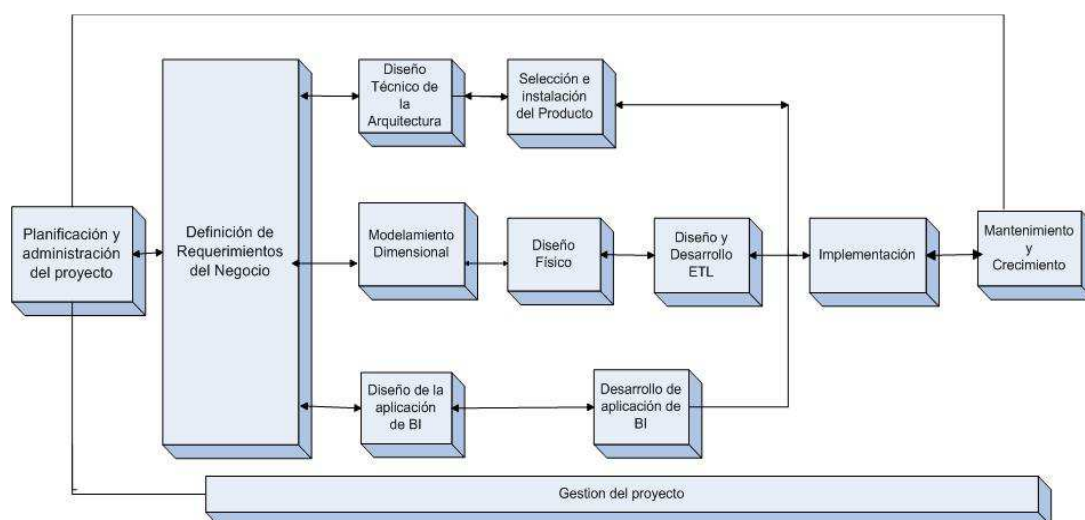
La metodología de Kimball es ideal para los primeros pasos de la implantación de BI a un cliente. Por lo que el desarrollo del proyecto “Desarrollo de una aplicación de Business Intelligence para la empresa EMPAQPLAST” se lo realizará con dicha metodología.

### 2.3.2 Metodología de Ralph Kimball

Ralph Kimball es considerado uno de los representantes más importantes del Data Warehouse y Business Intelligence. Su metodología ha sido probada en muchos escenarios de negocio y se podría decir que se ha llegado a convertir en un estándar de proyectos BI.

En el año 1998 se publica la primera edición del libro “The Data Warehouse Lifecycle Toolkit” donde se expone dicha metodología.

En la siguiente figura se muestra el ciclo de vida propuesta por Ralph Kimball y sus fases.



**Figura 2.2: Ciclo de vida de la metodología de Ralph Kimball**

### **2.3.2.1. Planificación y Administración del Proyecto**

La planificación busca identificar la definición y el alcance del proyecto, incluyendo las justificaciones del negocio y las evaluaciones de factibilidad.

La metodología en esta etapa propone identificar el alcance basado en requerimientos de negocios y no en fechas establecidas. Así el cumplimiento del proyecto será directamente relacionado con el negocio de la empresa.

#### **Definición del proyecto**

Como paso inicial para el desarrollo de un proyecto de BI se debe identificar de donde proviene la necesidad de información la cual puede darse de un sector específico como los directivos o gerentes de la organización o a su vez puede provenir de varios sectores de la empresa.

#### **Preparación para un proyecto de Data Warehouse**

Ralph Kimball indica que se deben tomar en cuenta factores como el apoyo y patrocinio de la gerencia, interés de la empresa en un proyecto de BI, apoyo del departamento de sistemas y tecnología de la empresa, la empresa debe presentar un entorno factible para la realización del proyecto.

#### **Alcance**

Se debe definir los límites del proyecto para poder desarrollarlo satisfactoriamente en base a los requerimientos del negocio.



## **Justificación**

Se debe tomar en cuenta cual será la inversión para el proyecto y cuál será el beneficio para la empresa.

En esta fase de debe definir justificaciones en términos de negocio que sustentan el proyecto y hacer las respectivas evaluaciones de factibilidad.

## **Planificación del proyecto**

A nivel de planificación del proyecto se establece la identidad del mismo es decir se lo maneja con un nombre, el personal (los usuarios, gerentes del proyecto, equipo del proyecto, desarrolladores), desarrollo del plan del proyecto, el seguimiento y la monitorización.

## **Administración del proyecto**

En esta etapa se verifica mediante reuniones con los involucrados en el proyecto el avance del mismo y el cumplimiento de los requerimientos.

### **2.3.2.2. Definición de los Requerimientos del Negocio**

El punto clave para el proceso de desarrollo un Data Warehouse es la manera como se interpretan y se analizan los requerimientos la interpretación correcta de los diferentes niveles de requerimientos proporcionara la visión de cómo se realizara el diseño del Data Warehouse.

Para iniciar con el proceso de obtención de requerimientos se debe hablar con los principales encargados del negocio, se debe enfocar el proceso de toma de requerimientos en cosas como: en base a qué información se toman decisiones, en qué consisten sus trabajos, que información es la que se maneja con más frecuencia, en que actividades invierte más tiempo.

Los diseñadores del Data Warehouse deben comprender cuales son los factores y las reglas que dirigen el negocio, cual es el funcionamiento y los procesos que se llevan a cabo para la actividad de negocio, determinar y analizar los requerimientos para traducirlos en consideraciones de diseño apropiadas.

### **Entrevistas**

Para iniciar el proceso de entrevistas se debe formar un equipo entrevistador, el cual este conformado del entrevistador y una persona que tome nota, es importante iniciar el levantamiento de requerimientos observando los reportes que ya se maneja en la empresa, cuales son las fuentes de datos y conocer las maneras como se lleva a cabo el análisis la información dentro de la empresa, saber en base a que parámetros se toman las decisiones del negocio esto contribuye al análisis de requerimientos. Esto proporciona una guía para la elaboración del modelo dimensional de datos se puede tener una visión general de las dimensiones que se necesitarán para el diseño del Data Warehouse.

Es de vital importancia dirigir las entrevistas a los directivos debido a que son los que tienen un mejor conocimiento de las reglas de negocio, las entrevistas también deben ser realizadas al departamento de tecnología para conocer con que recursos se cuenta para el desarrollo del proyecto.

#### **2.3.2.3 Diseño Técnico de la arquitectura**

El diseño de un Data Warehouse requiere la integración de varias tecnologías por lo que se debe tener en cuenta elementos como: requerimientos del negocio, el entorno técnico y las estrategias de diseño.

### Entorno Back Room:

Es el sitio donde toman lugar los procesos de Data Staging (Adquisición de los datos) consiste en el proceso de Extracción, Transformación y carga (ETL) de los datos desde las fuentes de origen y la carga de los mismo en el Data Warehouse.

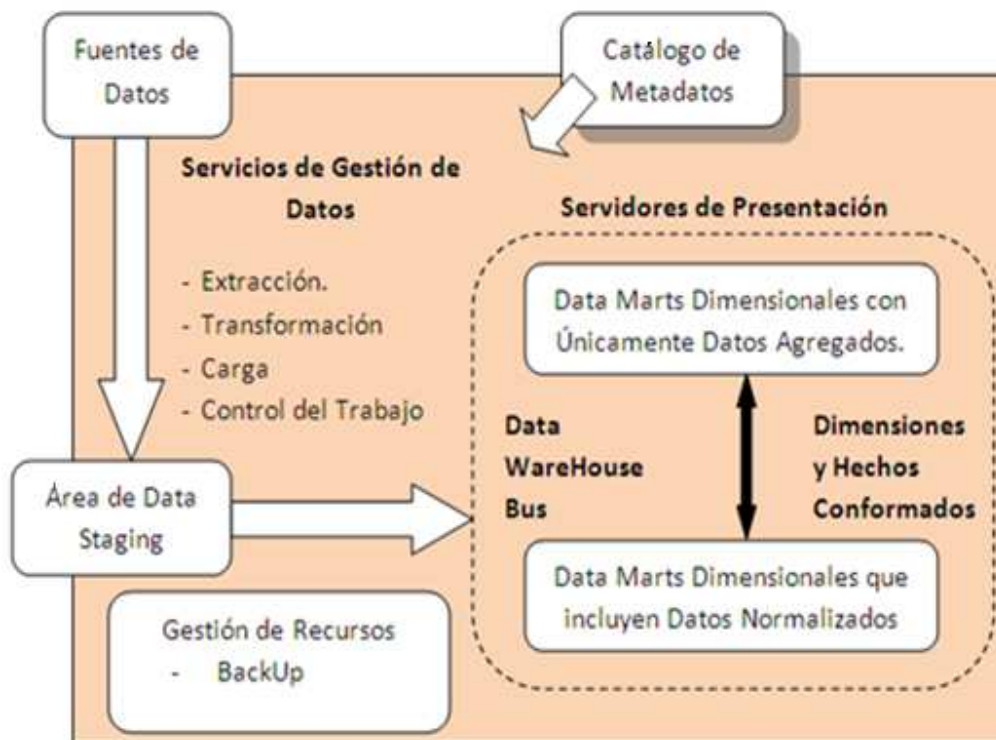


Figura 2.3: Arquitectura back room [1]

### Entorno Front Room

Es la carta de presentación de todo Data Warehouse por tanto debe ser capaz de explotar al máximo todas las funcionalidades y características que puede ofrecer el sistema en general.

[1] Wikispaces

Entre los servicios del Front Room se encuentra navegación, seguridad, monitoreo, generación de reportes y por supuesto manejo de consultas y otros servicios de escritorio. El Front Room corresponde a la capa que representa los datos al usuario, escondiendo toda la complejidad y el origen de los datos.

### Arquitectura técnica del Front Room

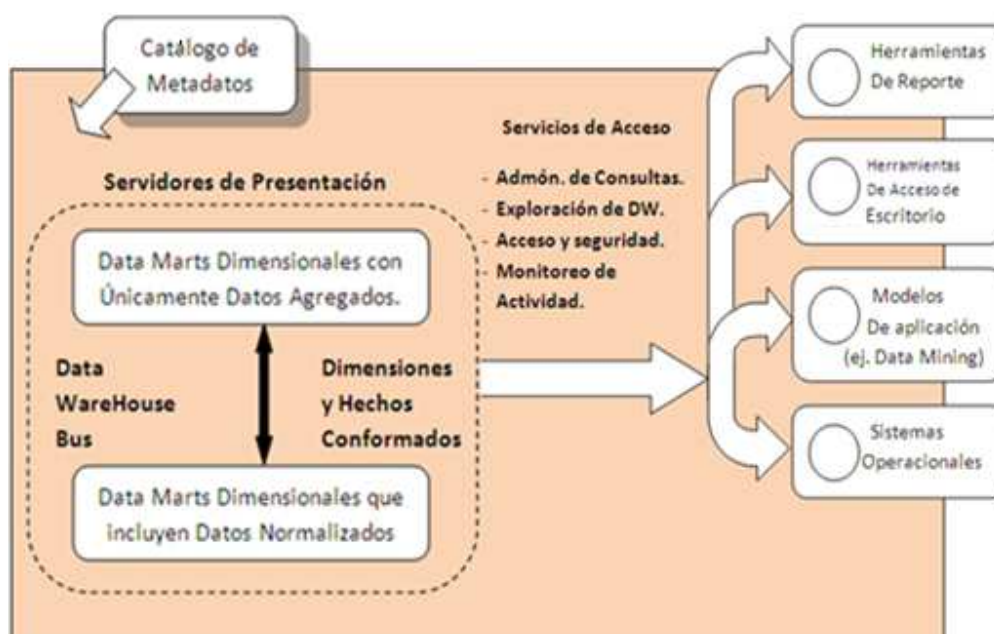


Figura 2.4: Arquitectura front room [1]

#### 2.3.2.4 Selección de Productos e Instalación

Utilizando el diseño de arquitectura técnica como marco es necesario evaluar en base a costos y factores de desempeño los componentes específicos de la arquitectura, como la plataforma de hardware, el motor de base de datos, la herramienta de ETL, las herramientas de acceso.

Una vez evaluados y seleccionados los componentes se procede con la instalación y prueba de los mismos en un ambiente integrado de Data Warehouse.

[1] Wikispaces

### 2.3.2.5 Modelado Dimensional

Ralph Kimball menciona que el modelado dimensional proporciona varias ventajas en el desempeño del sistema BI en cuanto a las consultas.

El Modelado Dimensional, según su creador Ralph Kimball, “es el diseño físico y lógico que transformará las antiguas fuentes de datos en las estructuras finales del Data Warehouse, a través de una técnica que busca la presentación de los datos en un marco de trabajo estándar que es intuitivo y permite un acceso de alto desempeño. Cada modelo dimensional está compuesto de una tabla que tiene una llave compuesta llamada tabla de hechos y un conjunto de tablas más pequeñas llamadas dimensiones. Cada tabla dimensión tiene una llave primaria simple, que corresponde exactamente a una de las partes de la llave compuesta en la tabla de hechos. Esta estructura característica es usualmente llamada esquema estrella” [1]

Los pasos necesarios para convertir un Diagrama Entidad-Relación (ERD) a un conjunto de diagramas de modelado dimensional son:

**Separar el ERD de la organización:** Se debe separar los procesos de negocios y modelar cada uno separadamente.

**Seleccionar las relaciones:** Las relaciones de varios a varios en el modelo entidad-relación que contengan cantidades numéricas y aditivas se las designa a las tablas de hechos.

**Desnormalizar tablas:** las tablas restantes con llaves simples que conectan directamente a las tablas de hechos son las tablas se convierten en las dimensiones. En los casos que una tabla dimensión se conecte a más de una tabla de hechos, se representa esta misma tabla dimensión en ambos esquemas estas tablas adquieren el nombre de dimensiones conformadas.

[1] Kimball98, p144.

El objetivo de aplicar el modelado dimensional a una estructura de Data Warehouse proporciona un marco de trabajo predecible, que se obtiene al utilizar dimensiones conformadas, esto mejora la presentación y también el desempeño, ya que las uniones entre tablas de dimensiones tablas de hechos son sencillas.

Soporta cambios inesperados en el comportamiento del usuario y es fácilmente extensible para acomodar nuevos elementos de datos inesperados y nuevas decisiones de diseño.

### **La Arquitectura del Data Warehouse**

“Un Data Warehouse está hecho de las unión de muchos de sus Data Marts” [1]

“Cada Data Mart debe ser representado por un modelo dimensional y dentro de un único Data Warehouse, todos estos Data Marts deben ser construidos a partir de dimensiones conformadas y hechos. Esta es la base de la arquitectura de bus del Data Warehouse.

La planeación del Data Warehouse debe iniciarse con un fase pequeña de arquitectura de datos que cubra todo y que a la vez tenga metas muy finitas y específicas, luego se debe seguir esta arquitectura con una implementación paso a paso de Data Marts separados, donde cada implementación se adhiere a la arquitectura predefinida” [2]

El autor Ralph Kimball sugiere las siguientes recomendaciones para la construcción de un Data Warehouse.

Crear una arquitectura que defina el alcance e implementación del Data Warehouse completo. Supervisar la construcción de cada pieza del Data Warehouse.

[1] Kimball98, p19

[2]Kimball98, p153

Una tabla dimensión puede ser usada contra múltiples tablas de hechos en el mismo espacio de base de datos.

### **Técnicas de Modelado Dimensional**

“La idea fundamental del modelado dimensional es que casi cada tipo de datos de negocio puede ser representado como un tipo de cubo de datos, donde cada celda del cubo contiene un valor medido y las aristas del cubo definen las dimensiones naturales de los datos” [1]

### **Seleccionar el proceso de Negocio**

En base a los requerimientos que ya se han tomado se debe determinar cuál es el proceso de negocio que se va a modelar el cual puede tener una o más tablas de hechos.

### **Declaración de la granularidad**

El nivel de granularidad se refiere a la presentación de las medidas en la tabla de hechos e es decir una granularidad fina o menor presenta mayor detalle de los datos y una granularidad gruesa o mayor presenta menor detalle en los datos.

### **Identificación de las dimensiones**

Cuando se agregan las diferentes dimensiones éstas deben ser de un objeto del negocio por ejemplo un cliente, producto etc. y deben cumplir con la granularidad establecida previamente.

[1] Kimball98, p165

## **Identificación de los hechos**

Los hechos vienen a ser las medidas, son generalmente cantidades y están orientados a los procesos del negocio. Ejemplos de medidas son totales, precios, cantidades etc.

### **2.3.2.6 Diseño Físico**

En lo que se refiere a la estructura física la etapa incluye tareas como la configuración de la base de datos debe incluir los nombres de columna físicos, los tipos de datos y constraints.

La creación de tablas para datos y metadatos, donde se alojarán dimensiones y tablas de hechos.

Creación de secuencias para los procesos ETL.

Creación de tablas temporales, tablas históricas, tablas para errores.

Generación de Scripts y diagramas físicos de base de datos.

### **2.3.2.7 Diseño y Desarrollo de la Presentación de Datos**

Esta etapa está conformada primeramente por las actividades de extracción, transformación y carga de datos (ETL). Se realiza un análisis de los datos como se van a integrar y como se pueden resolver problemas de inconsistencias.

Se definen como proceso de extracción obtención de los datos de diferentes fuentes que permitirán efectuar la carga del Modelo Físico diseñado.



El proceso de transformación consiste en estandarizar y estructurar los datos extraídos de las diferentes fuentes para convertir los datos y finalmente cargarlos en el Data Warehouse.

### **Actividades para el proceso ETL**

- Realizar un plan cada proceso de ETL.
- Seleccionar una herramienta ETL.
- Estrategias para proceso de carga.
- Carga inicial de Datos(Dimensiones y Hechos)
- Carga incremental de Datos(Dimensiones y Hechos)
- Automatización del proceso ETL

Las tareas de ETL son altamente críticas pues tienen que ver con parte medular del negocio ya que manejan directamente los datos.

La calidad de los datos en un Data Warehouse es un factor determinante en el éxito de un proyecto de BI. En la etapa de ETL se debe resolver todos los inconvenientes relacionados con la calidad de los datos fuente. Para que los datos del Data Warehouse sean confiables.

### **2.3.2.8 Especificación de Aplicaciones para Usuarios Finales**

Cada usuario del Sistema de Business Intelligence necesita un role o perfil de usuario para los diferentes tipos de aplicaciones, acceso a carpetas, acceso a cierto tipo de reportes, entre otras por lo que en esta etapa se definen los diferentes tipos de permisos para los usuarios.

### **2.3.2.9 Desarrollo de Aplicaciones para Usuarios Finales**

Los usuarios acceden al Data Warehouse por medio del BI Server herramienta grafica, que contiene la información de cada área de negocio, se pueden desplegar

reportes, vistas de análisis, tableros de control. Los cubos, reportes y tableros de control son implementados conforme a los requerimientos.

#### **2.3.2.10. Implementación**

La implementación representa la unión de la herramienta de BI, los datos y las aplicaciones de usuarios finales. Existen factores extras que aseguran el correcto funcionamiento de todos los elementos, entre ellos se encuentran la capacitación, el soporte técnico y la comunicación. Todas estas tareas deben tenerse en cuenta antes de que cualquier usuario pueda tener acceso al sistema de BI.

#### **2.3.2.11. Mantenimiento y crecimiento**

Como en todo proyecto se necesita tener actualizaciones de forma constante para poder dar un ciclo de vida adecuado al producto. Es importante establecer las prioridades para poder manejar los nuevos requerimientos de los usuarios y de esa forma poder evolucionar y crecer.

#### **2.3.2.12. Gestión del proyecto**

La gestión del proyecto asegura que las actividades del ciclo de vida se lleven a cabo de manera sincronizada. La gestión del proyecto acompaña todo el ciclo de vida. Entre sus actividades principales se encuentra la monitorización del estado del proyecto y el acoplamiento entre los requerimientos del negocio y las restricciones de los sistemas de información para poder manejar correctamente las expectativas en ambos sentidos.

#### **2.3.2.13. Formato para toma de requerimientos**

Después de tener el planteamiento del problema y los responsables de cada área de negocio se debe diseñar un formato para la toma de requerimientos.

Un factor determinante en el éxito de un BI, es la interpretación correcta de los diferentes niveles de requerimientos expresados por los distintos grupos de usuarios, el formato tiene la siguiente estructura.

**Tabla 2.1: Formato para toma de requerimientos**

<b>IDENTIFICADOR</b>	<b>R001</b>	<b>NOMBRE</b>	<b>Nombre del requerimiento.</b>
<b>FECHA</b>	Fecha del día que se realizó el requerimiento.		
<b>AREA</b>	Nombre del área de negocio.		
<b>DESCRIPCION</b>	Explicar la función del requerimiento lo más detallado posible		

Este formato se utilizara para la toma de requerimientos en las aéreas de ventas, compras e inventarios. El responsable de cada área al finalizar el proyecto, aceptara el cumplimiento firmando el documento de requerimientos.

## 2.4 Herramientas

Las herramientas a utilizar en el desarrollo del proyecto se muestran en la siguiente tabla.

**Tabla 2.2: Listado de herramientas para el desarrollo del proyecto**

<b>Productos Software</b>	<b>Característica</b>
<b>Pentaho BI Server 3.8</b>	Servidor OLAP.
<b>Pentaho Report Designer</b>	Reporteador.
<b>Pentaho Data Integration</b>	Integración de Datos (ETL).
<b>Pentaho Schema Work Bench</b>	Diseñador de Cubos Dimensionales.
<b>Base de Datos Oracle 10g</b>	Repositorio del Data Warehouse.
<b>Powerdesigner 15</b>	Diseño del esquema de base de datos.

## 2.4. 1 Plataforma Pentaho Open Source Business Intelligence

La herramienta Open Source Pentaho Business Intelligence permite realizar análisis de los datos y de informes empresariales. Estas soluciones están escritas en lenguaje java y su ambiente de implementación de igual manera en java.



Figura 2.5: Herramientas de la suite de Pentaho [1]

## 2.4.2 Módulos de la plataforma Pentaho BI

### 2.4.2.1 Reporting

Pentaho Reporting permite realizar informes de manera ágil y de gran capacidad para usuarios, es una solución basada en el proyecto JFreeReport. Pentaho Reporting permite la distribución de los resultados del análisis en múltiples formatos, todos los informes incluyen la opción de imprimir o exportar a formato PDF, XLS, HTML y texto. Los reportes Pentaho permiten también programación de tareas y ejecución automática de informes con una determinada periodicidad.

**Diseñador gráfico:** Basado en “arrastre y soltar” (drag & drop) que provee completo.

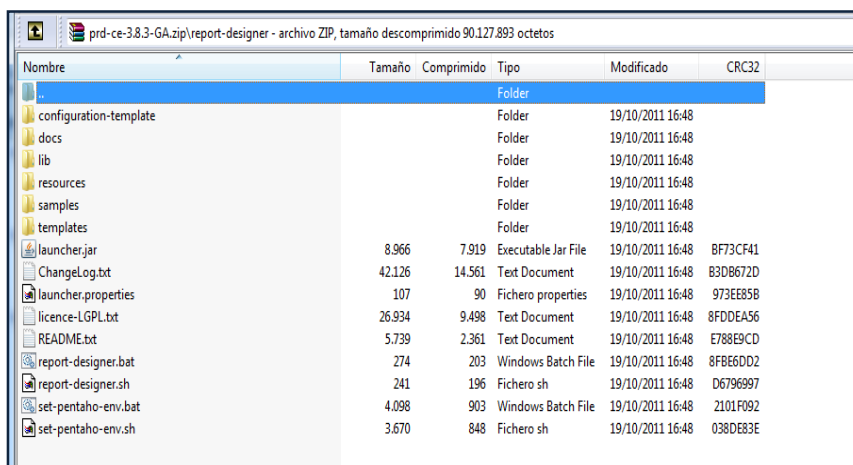
**Plantillas de reportes:** Mejoran la presentación de reportes y la organización de los datos.

[1] [opensourceclub.net](http://opensourceclub.net)

Pentaho Repor Designer, puede ser descargado desde la página web de Pentaho. (<http://www.pentaho.com/download>), dentro del archivo comprimido podemos encontrar diferentes archivos para las plataformas (Linux, Windows).

Report-designer.sh > Para Linux

Report-designer.bat > Para Windows



Nombre	Tamaño	Comprimido	Tipo	Modificado	CRC32
.			Folder		
configuration-template			Folder	19/10/2011 16:48	
docs			Folder	19/10/2011 16:48	
lib			Folder	19/10/2011 16:48	
resources			Folder	19/10/2011 16:48	
samples			Folder	19/10/2011 16:48	
templates			Folder	19/10/2011 16:48	
launcher.jar	8.966	7.919	Executable Jar File	19/10/2011 16:48	BF73CF41
ChangeLog.txt	42.126	14.561	Text Document	19/10/2011 16:48	B3D8672D
launcher.properties	107	90	Fichero properties	19/10/2011 16:48	973EE858
licence-LGPL.txt	26.934	9.498	Text Document	19/10/2011 16:48	8FDDEA56
README.txt	5.739	2.361	Text Document	19/10/2011 16:48	E788E9CD
report-designer.bat	274	203	Windows Batch File	19/10/2011 16:48	8FB6DD2
report-designer.sh	241	196	Fichero sh	19/10/2011 16:48	D6796997
set-pentaho-env.bat	4.098	903	Windows Batch File	19/10/2011 16:48	2101F092
set-pentaho-env.sh	3.670	848	Fichero sh	19/10/2011 16:48	038DE83E

**Figura 2.6: Carpeta contenedora report designer**

#### 2.4.2.2 Análisis

“Pentaho Análisis” da a los usuarios un sistema avanzado de análisis de información. Mediante el uso de las tablas dinámicas (pivot tables, crosstabs), generadas por el servidor Mondrian y JPivot. Facilita el análisis de los datos en un Data Warehouse a través de una interfaz de tabla cruzada donde podemos navegar por las diferentes dimensiones definidas en el modelo dimensional para desarrollar este esquema dimensional se puede utilizando Pentaho Schema Workbench. Teniendo definidos nuestros esquemas dimensionales podemos hacer el análisis utilizando Jpivot a nivel de interfaz de usuario y Mondria como servidor.

### 2.4.2.3 Mondrian

Es la parte que recibe las solicitudes de información de Jpivot, realiza las consultas contra la base de datos y devuelve la información en formato multidimensional. El núcleo de Mondrian es un JAR que actúa como "JDBC para OLAP": proporcionando conexiones y ejecutando consultas SQL contra la base de datos.

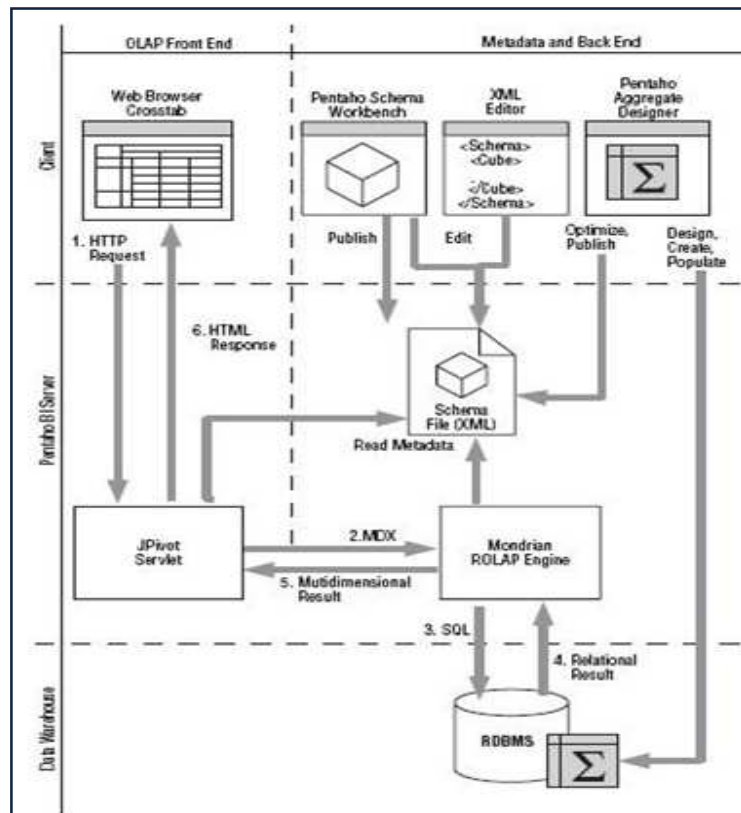


Figura 2.7: Arquitectura servidor mondrian [1]

### Capas de Mondrian

#### Capa de presentación

El usuario puede interactuar de manera grafica con sus reportes filtrando los datos

[1] Rincon del BI

multidimensionales, incluyendo tablas pivote, graficas de pastel, diagramas de barra y líneas, esas herramientas están escritas en lenguaje JSP. Los gráficos pueden ser JPG, GIF. Modrian ha desarrollado una interfaz llamada Jpivot Imagen.

Region	Department	Positions	Medidas			
			Actual	Budget	Variance	Variance Percent
All Regions	All Departments	All Positions	143.639.982,00	143.199.389,00	-440.593,00	-31%
		Account Executive	1.279.975,00	1.323.525,00	-43.550,00	-3,29%
		Administration	2.049.625,00	2.046.500,00	-3.125,00	-0,15%
		Administrative Assistant	3.583.061,00	3.457.094,00	-125.967,00	-3,54%
		Analyst Relations	1.133.375,00	1.212.625,00	79.250,00	6,94%
		CEO	2.049.625,00	1.988.500,00	-61.125,00	-3,00%
		CFO	3.265.672,00	3.167.202,00	-98.470,00	-3,01%
		CMO	3.583.061,00	3.450.094,00	-132.967,00	-3,73%
		CTO	1.257.985,00	1.326.440,00	68.455,00	5,44%
		Controller	2.163.973,00	2.160.942,00	-3.031,00	-0,14%
		District Manager	2.782.625,00	2.696.000,00	-86.625,00	-3,11%
		EOE	1.942.607,00	1.948.753,00	6.146,00	0,32%
		Engineer	2.855.925,00	2.785.950,00	-69.975,00	-2,45%
		Graphics	3.332.375,00	3.221.125,00	-111.250,00	-3,34%
		HR Generalists	3.739.190,00	3.589.699,00	-149.491,00	-4,00%
		HR Training	1.211.073,00	1.280.592,00	69.519,00	5,74%
		IS	2.166.099,00	2.162.883,00	-3.216,00	-0,15%
		Payroll	1.045.415,00	1.139.285,00	93.870,00	8,97%
		Pre-Sales	2.599.375,00	2.528.625,00	-70.750,00	-2,72%
		Press Relations	1.761.226,00	1.786.084,00	24.858,00	1,39%
		Product Marketing Mgr	2.842.731,00	2.773.899,00	-68.832,00	-2,42%
		QA Engineer	1.045.415,00	1.132.285,00	86.870,00	8,27%

Figura 2.8: Cubo mondrian

### Capa de cálculo

En esta capa se valida la ejecución de las consultas escritas al servidor Mondria el lenguaje utilizado por este servidor se denomina MDX("Multi-Dimensional, Expressions") , cuya sintaxis es similar a la del lenguaje SQL.

### Capa de agregación

“Una agregación es un conjunto de valores de medida ("células") en la memoria, calificado por un conjunto de valores de las columnas de dimensión. La capa de dimensiones envía las solicitudes de grupos de células. Si las células solicitada no están en la caché, o que pueda deducirse por enrollar una agregación en la memoria

caché, el administrador de la agregación envía una solicitud a la capa de almacenamiento” [1]

### **Capa de almacenamiento**

Esta capa se conformada por un sistema de bases de datos relacionales (SGBDR). Lo que le permite a Mondrian ser un servidor ROLAP. Es responsable de proveer agregaciones de datos y atributos de tablas de dimensión.

Estas capas pueden existir en una misma máquina, o pueden estar distribuidas en varias maquinas. La capa 2 y 3 que comprenden el servidor Mondrian deberán estar en una misma capa. La capa de presentación podrá encontrarse separada teniendo el acceso a esta mediante una conexión JDBC.

### **Api**

Modrian provee un API para que las aplicaciones clientes ejecuten los queries. El lenguaje que se usa Mondrian para ejecutar queries es MDX(Multidimensional Expression). El jdbc ejecuta el SQL normal.

### **Mdx**

Mdx nos permite realizar las consultas a la base de datos relacional, este es un estándar en los sistemas OLAP. Sintaxis es similar a la de SQL

#### **2.4.2.4 Dashboards**

Todos los componentes del modulo Pentaho Reporting y Pentaho Análisis pueden formar parte de un Dashboard. En Pentaho Dashboards es muy fácil incorporar una

[1] Tallersoft2011



variedad en tipos de gráficos, tablas y velocímetros (dashboard widgets) e integrarlos con los Portlets JSP, en donde podrá visualizar informes, gráficos y análisis OLAP.



**Figura 2.9: Ejemplo dashboard [1]**

#### **2.4.2.5 Integración de Datos**

Pentaho Data Integration (PDI o también llamado Kettle) en el componente de pentaho responsable de la extracción, transformación, y carga (ETL). La herramienta de ETL frecuentemente es utilizada en un ambiente para desarrollo de Data Warehouse. Kettle puede también ser usado para otros propósitos como:

- Migración de datos entre las aplicaciones o bases de datos.
- Exportación de datos desde base de datos a archivos planos.
- Carga de datos en forma masiva hacia la base de datos.
- Integración de aplicaciones.

#### **Kettle incluye las herramientas:**

##### **Spoon**

Es el diseñador gráfico de transformaciones y trabajos del sistema de ETL de Pentaho Data Integration (PDI), también conocido como Kettle.

[1]MicroStrategy

Está diseñado para ayudar en los procesos ETL, que incluyen la Extracción, Transformación, Transporte y Carga de datos.

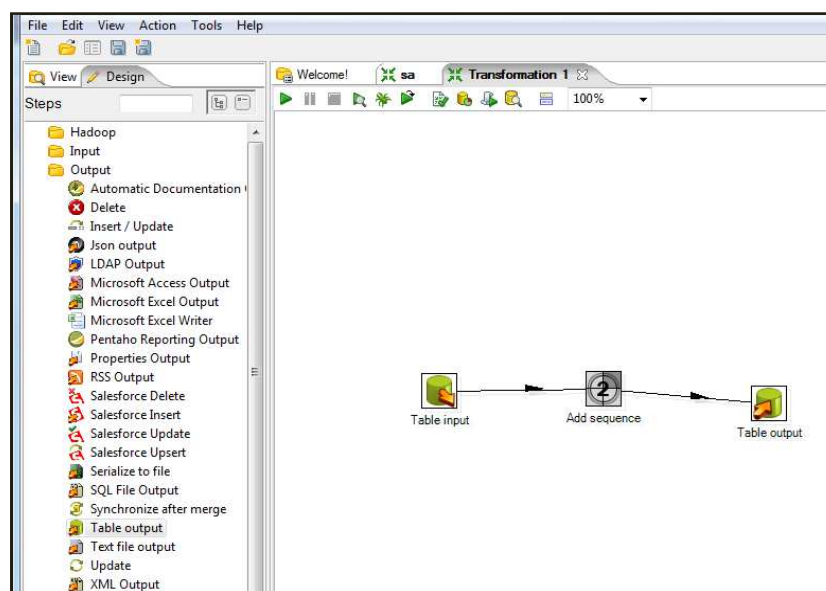
Spoon es una Interfaz Gráfica de Usuario, que permite diseñar transformaciones y trabajos que se pueden ejecutar con las herramientas de Kettle (Pan y Kitchen).

## Pan

Es un motor de transformación de datos que realiza muchas funciones tales como lectura, manipulación, y escritura de datos hacia y desde varias fuentes de datos.

## Kitchen

Es un programa que ejecuta las transformaciones realizadas por Spoon. En XML o en un catálogo de base de datos.



**Figura 2.10: Ambiente gráfico spoon**

#### 2.4.2.6 Diseño de cubos.

Para la construcción de los cubos, después del análisis de los requerimientos y el diseño del Data Warehouse. Se utilizará la herramienta Schema Workbench para el diseño de los cubos OLAP.

Schema Workbench es una interfaz de diseño que permite crear y probar esquemas de Cubos OLAP, para posteriormente ser visualizados en el servidor Mondrian. Modrian procesa las solicitudes de MDX con los Rolap (Relational OLAP). Los archivos que se generán son de tipo XML. Tienen una estructura especifica y son usados por El servidor Mondrian, estos modelos XML son considerados como estructuras de cubos que se utilizan cuando existen tablas de Hechos y dimensiones.

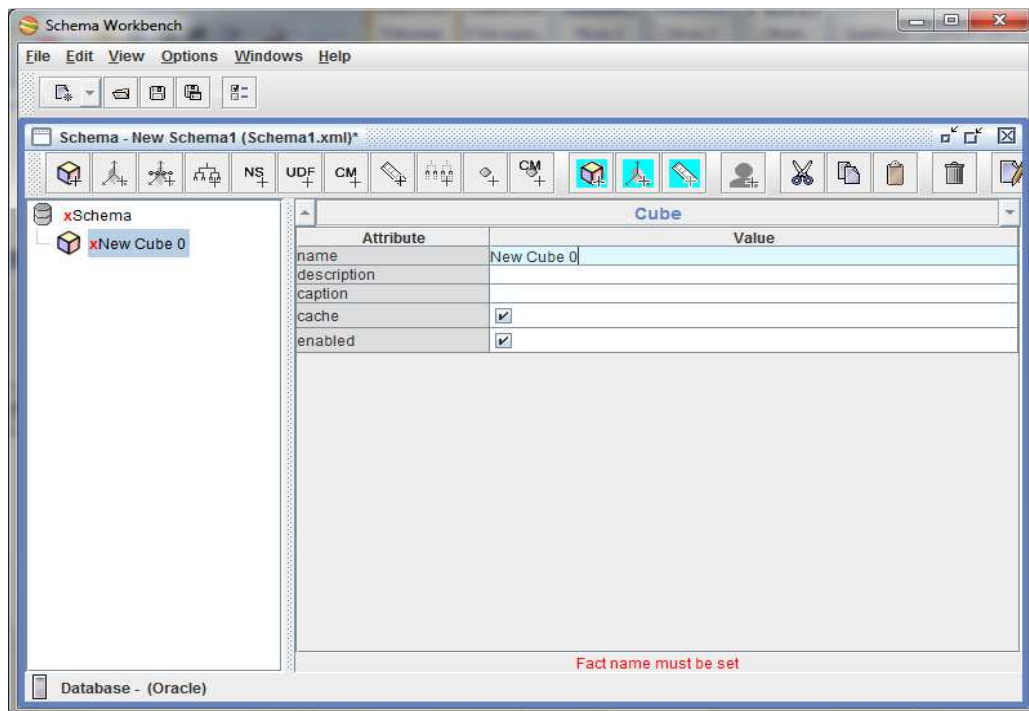


Figura 2.11: Schema Workbench.

#### 2.4.2.7 Base de datos

Para el almacenamiento del Data Warehouse se utilizará una base de datos Oracle 10g montada en un servidor local de bases de datos de la empresa.

La base de datos Oracle es un sistema de administración de base de datos relacionales

(ORDBMS acrónimo en inglés de Object Relational Data Base Management System).

Este modelo consiste en utilizar tablas bidimensionales para almacenar información. Oracle es uno de los sistemas de bases de datos más completos, destacando:

- Soporte de transacciones.
- Estabilidad.
- Escalabilidad.
- Soporte multiplataforma.



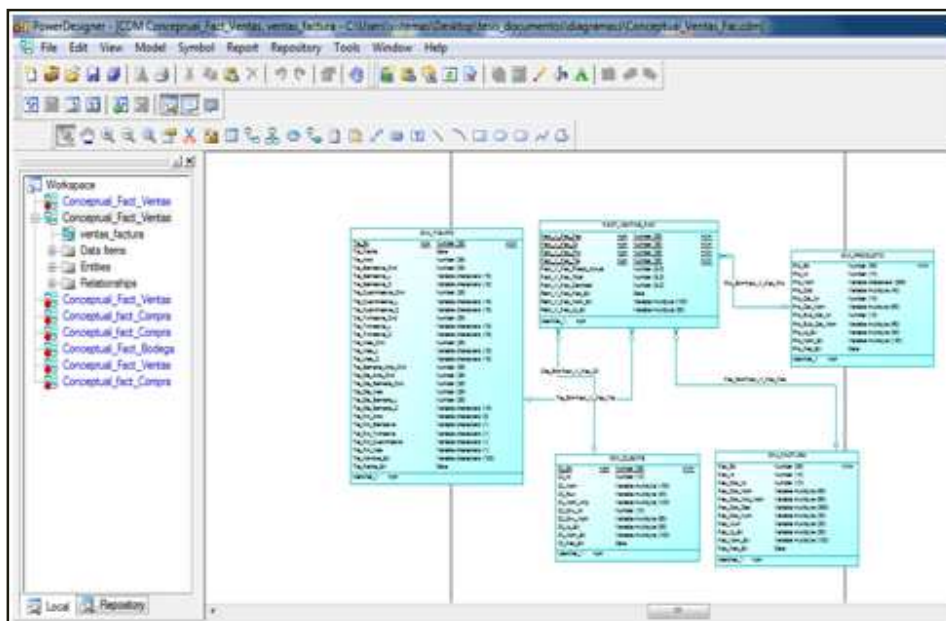
**Figura 2.12: Oracle 10g**

#### **2.4.2.8 Modelado de datos**

Para modelar el esquema de base de datos del Data Warehouse, se utilizará la herramienta de modelado Sybase-PowerDesigner.

Sybase-PowerDesigner. Es una herramienta para el análisis, diseño inteligente y construcción sólida de una base de datos y un desarrollo orientado a modelos de datos a nivel físico y conceptual, que da a los desarrolladores Cliente/Servidor la más firme base para aplicaciones de alto rendimiento.

[1] ModderClocker



**Figura 2.13: Sybase-PowerDesigner**

#### 2.4.2.9 Servidor de Base de datos.

La base de datos origen se encuentra almacenada en el servidor p bladesystem c3000, en la bahía 2 especificada en la siguiente tabla.

**Tabla 2.3 Servidor p bladesystem c3000**

BAHÍAS	MODELO	FIRMWARE	SISTEMA OPERATIVO	MEMORIA	DISCO DURO
Bahía 1	BL460c	Versión 9.0	Redhat 5.4	10 GB	150 GB
Bahía 2	BL460c	Versión 9.0	Redhat 5.4	10 GB	150 GB
Bahía 5	BL460c	Versión 9.0	Windows Server 2008 R2 Standard (64 bits)	10 GB	150 GB

## **CAPÍTULO 3**

### **DESARROLLO DEL PROYECTO**

#### **3.1 Desarrollo del BI para la empresa Empaqplast**

Para el desarrollo del Data Warehouse se utilizará la metodología de Ralph Kimball lo que permitirá dar un seguimiento adecuado al desarrollo del proyecto.

#### **3.2 Planeación y administración del proyecto**

##### **3.2.1 Definición del Proyecto**

Partiendo de la necesidad de la empresa EMPAQPLAST en el desarrollo de un sistema BI para dar soporte al análisis de la información y la toma de decisiones por los gerentes de las áreas de ventas, compras e inventarios. La empresa a decidido implementar este proyecto.

##### **3.2.2 Preparación para un proyecto de Data Warehouse.**

La empresa EMPAQPLAST muestra interés en el desarrollo del proyecto, por lo que se tiene el apoyo de la gerencia general y las gerencias involucradas en el desarrollo, el departamento de tecnología de la empresa fue el encargado de presentar este proyecto como soporte para el análisis de información y toma de decisiones, para los gerentes de las áreas previamente mencionadas.

##### **3.2.3 Alcance**

El proyecto estará centrado en el desarrollo de una aplicación de Business Intelligence (BI) para el soporte en la toma de decisiones en la empresa EMPAQPLAST.

El aplicativo abarcará las siguientes áreas:

- Ventas.
- Inventarios.
- Compras.

Para cada una de estas áreas se desarrollará, los cubos de información, siendo los gerentes de cada departamento los responsables del análisis de los cubos.

Se desarrollará los Data Mart de las áreas de ventas, inventarios y compras que conformarán el Data Warehouse.

Se generarán reportes que se adaptarán a las necesidades de información concerniente a las áreas ya mencionadas.

### **3.2.4 Justificación**

La solución que se plantea para el análisis de la información está basada en la elaboración de una aplicación de Business Intelligence (BI) que estará conformado por los Data Mart de las áreas ya mencionadas.

Se busca relacionar los datos con el negocio para obtener información relevante sobre la situación de la empresa. Tomando en cuenta que la empresa trabaja con un sistema ERP (OpenSource) llamado Open Bravo. Se considerará el uso de la herramienta Pentaho BI (OpenSource) esta herramienta facilitará el camino para conseguir una completa solución de Business Intelligence (BI) y una rápida integración con la infraestructura que existe actualmente en la empresa. Aplicando este sistema de Business Intelligence (BI), se pretende reducir los costos y optimizar los tiempos en lo que se refiere al tratamiento de la información se facilitará la administración de la misma personalizándola y adaptándola a las necesidades lo que permitirá la Integración y depuración de los datos.

### 3.2.5 Planificación del Proyecto

Asignarán roles a las personas involucradas en el ciclo de vida del desarrollo del proyecto de BI. Con el objetivo de tener responsables de cada fase del ciclo de vida del proyecto. Tomando en cuenta la necesidad e interés de la empresa EMPAQPLAST en el desarrollo del proyecto se asignó los siguientes responsables.

- **Patrocinador del Proyecto:** Empresa EMPAQPLAST.
- **Administrador de Bases de Datos:** Gerente de departamento de Tecnología (Ing. Pablo Beltrán)
- **Director del Proyecto:** Gerente de departamento de Tecnología (Ing. Pablo Beltrán)
- **Desarrollador:** Estudiantes Egresados (Boada Byron, Alvaro Tituaña)
- **Arquitecto de Datos:** Estudiantes Egresados (Boada Byron, Alvaro Tituaña)
- **Diseñador:** Estudiantes Egresados (Boada Byron, Alvaro Tituaña)
- **Personal involucrado en el proyecto:** gerentes y empleados de las áreas de Ventas, Compras, Inventarios, Financiera, Contable y Sistemas.

### 3.2.6 Administración del proyecto

Semanalmente se realiza un reunión para monitoreo del avance del proyecto por parte del gerente del área de tecnología y los estudiantes encargados del desarrollo de proyecto.

La constancia de las reuniones y el cronograma de actividades, se encuentran en el **Anexo E**.

## 3.3 Definición de los Requerimientos del Negocio

La toma de requerimientos se lo realizo mediante reuniones con los gerentes de las áreas de negocio en bases a las necesidades de información, cada requerimiento fue descrito en el formato previamente establecido en la metodología.

A continuación se describen los requerimientos divididos por áreas de negocio.



### 3.3.1 Requerimientos área de ventas

**Tabla 3.1: R001 área de ventas**

<b>IDENTIFICADOR</b>	<b>R001</b>	<b>NOMBRE</b>	<b>VENTAS POR PRODUCTO</b>
<b>FECHA</b>	31 Enero del 2012		
<b>ÁREA</b>	Ventas		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer los movimientos por producto valorados en precio de venta al público por periodos de tiempo.		

**Tabla 3.2: R002 área de ventas**

<b>IDENTIFICADOR</b>	<b>R002</b>	<b>NOMBRE</b>	<b>VENTAS TOTALES POR PRODUCTO</b>
<b>FECHA</b>	31 Enero del 2012		
<b>ÁREA</b>	Ventas		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las ventas totales realizadas por producto.		

**Tabla 3.3: R003 área de ventas**

<b>IDENTIFICADOR</b>	<b>R003</b>	<b>NOMBRE</b>	<b>VENTAS POR CATEGORIA DE PRODUCTO</b>
<b>FECHA</b>	31 Enero del 2012		
<b>ÁREA</b>	Ventas		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer los movimientos de productos por categoría de producto y por periodos de tiempo.		

**Tabla 3.4: R004 área de ventas**

<b>IDENTIFICADOR</b>	<b>R004</b>	<b>NOMBRE</b>	<b>VENTAS POR SUBCATEGORIA DE PRODUCTO</b>
<b>FECHA</b>	31 Enero del 2012		
<b>ÁREA</b>	Ventas		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer los movimientos de productos por subcategoría de producto por periodos de tiempo.		

**Tabla 3.5: R005 área de ventas**

<b>IDENTIFICADOR</b>	<b>R005</b>	<b>NOMBRE</b>	<b>VENTAS POR CLIENTE</b>
<b>FECHA</b>	31 Enero del 2012		
<b>ÁREA</b>	Ventas		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las ventas totales realizadas por cliente por periodos de tiempo.		

**Tabla 3.6: R006 área de ventas**

<b>IDENTIFICADOR</b>	<b>R006</b>	<b>NOMBRE</b>	<b>VENTAS POR CATEGORIA DE CLIENTE</b>
<b>FECHA</b>	31 Enero del 2012		
<b>ÁREA</b>	Ventas		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las ventas totales realizadas por cada categoría de cliente por periodos.		

**Tabla 3.7: R007 área de ventas**

<b>IDENTIFICADOR</b>	<b>R007</b>	<b>NOMBRE</b>	<b>FACTURAS DE VENTA POR PRODUCTO</b>
<b>FECHA</b>	31 Enero del 2012		
<b>ÁREA</b>	Ventas		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las facturas por producto		

**Tabla 3.8: R008 área de ventas**

<b>IDENTIFICADOR</b>	<b>R008</b>	<b>NOMBRE</b>	<b>FACTURAS DE VENTA POR CLIENTE</b>
<b>FECHA</b>	31 Enero del 2012		
<b>ÁREA</b>	Ventas		
<b>DESCRIPCIÓN</b>	El sistema debe permitir visualizar las facturas por cliente.		

**Tabla 3.9: R009 área de ventas**

<b>IDENTIFICADOR</b>	<b>R009</b>	<b>NOMBRE</b>	<b>FACTURAS DE VENTA ANULADAS POR CLIENTE</b>
<b>FECHA</b>	31 Enero del 2012		
<b>ÁREA</b>	Ventas		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las facturas anuladas por cliente.		

**Tabla 3.10: R010 área de ventas**

<b>IDENTIFICADOR</b>	<b>R010</b>	<b>NOMBRE</b>	<b>FACTURAS DE VENTA ANULADAS POR PRODUCTO</b>
<b>FECHA</b>	31 Enero del 2012		
<b>ÁREA</b>	Ventas		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las facturas anuladas por producto.		

**Tabla 3.11: R011 área de ventas**

<b>IDENTIFICADOR</b>	<b>R011</b>	<b>NOMBRE</b>	<b>REMISIONES POR CLIENTE</b>
<b>FECHA</b>	31 Enero del 2012		
<b>ÁREA</b>	Ventas		
<b>DESCRIPCIÓN</b>	El sistema debe permitir visualizar la cantidad de producto anulado, enviado, devuelto por cliente.		

### 3.3.2 Requerimientos área de compras

**Tabla 3.12: R012 área de compras**

<b>IDENTIFICADOR</b>	<b>R012</b>	<b>NOMBRE</b>	<b>COMPRAS POR PRODUCTO</b>
<b>FECHA</b>	1 Febrero del 2012		
<b>ÁREA</b>	Compras		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las compras por producto valorados en precio unitario, por periodos de tiempo.		

**Tabla 3.13: R013 área de compras**

<b>IDENTIFICADOR</b>	<b>R013</b>	<b>NOMBRE</b>	<b>COMPRAS TOTALES POR PRODUCTO</b>
<b>FECHA</b>	1 Febrero del 2012		
<b>ÁREA</b>	Compras		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las compras totales realizadas por producto, por periodos de tiempo.		

**Tabla 3.14: R014 área de compras**

<b>IDENTIFICADOR</b>	<b>R014</b>	<b>NOMBRE</b>	<b>COMPRAS POR CATEGORÍA DE PRODUCTO</b>
<b>FECHA</b>	1 Febrero del 2012		
<b>ÁREA</b>	Compras		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer los movimientos de compra de productos por categoría, por periodos de tiempo.		

**Tabla 3.15: R015 área de compras**

<b>IDENTIFICADOR</b>	<b>R015</b>	<b>NOMBRE</b>	<b>COMPRAS POR SUBCATEGORÍA DE PRODUCTO</b>
<b>FECHA</b>	1 Febrero del 2012		
<b>ÁREA</b>	Compras		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer los movimientos de compra de productos por subcategoría, por periodos de tiempo.		

**Tabla 3.16: R016 área de compras**

<b>IDENTIFICADOR</b>	<b>R016</b>	<b>NOMBRE</b>	<b>COMPRAS POR PROVEEDOR</b>
<b>FECHA</b>	1 Febrero del 2012		
<b>ÁREA</b>	Compras		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las compras totales realizadas por proveedor, por periodos de tiempo.		

**Tabla 3.17: R017 área de compras**

<b>IDENTIFICADOR</b>	<b>R017</b>	<b>NOMBRE</b>	<b>COMPRAS POR CATEGORÍA DE PROVEEDOR</b>
<b>FECHA</b>	1 Febrero del 2012		
<b>ÁREA</b>	Compras		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las compras totales realizadas por cada categoría de proveedor, por periodos de tiempo.		

**Tabla 3.18: R018 área de compras**

<b>IDENTIFICADOR</b>	<b>R018</b>	<b>NOMBRE</b>	<b>FACTURAS POR PROVEEDOR</b>
<b>FECHA</b>	1 Febrero del 2012		
<b>ÁREA</b>	Compras		
<b>DESCRIPCIÓN</b>	El sistema debe permitir visualizar las facturas de compra por proveedor.		

**Tabla 3.19: R019 área de compras**

<b>IDENTIFICADOR</b>	<b>R019</b>	<b>NOMBRE</b>	<b>FACTURAS DE COMPRA POR PRODUCTO</b>
<b>FECHA</b>	1 Febrero del 2012		
<b>ÁREA</b>	Compras		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las facturas de compra por producto		

**Tabla 3.20: R020 área de compras**

<b>IDENTIFICADOR</b>	<b>R020</b>	<b>NOMBRE</b>	<b>FACTURAS DE COMPRA ANULADAS POR PROVEEDOR</b>
<b>FECHA</b>	1 Febrero del 2012		
<b>ÁREA</b>	Compras		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las facturas de compra anuladas por proveedor.		

**Tabla 3.21: R021 área de compras**

<b>IDENTIFICADOR</b>	<b>R021</b>	<b>NOMBRE</b>	<b>FACTURAS DE COMPRA ANULADAS PRODUCTO</b>
<b>FECHA</b>	1 Febrero del 2012		
<b>ÁREA</b>	Compras		
<b>DESCRIPCIÓN</b>	El sistema debe permitir conocer las facturas de compra anuladas por producto.		

### 3.3.3 Requerimientos área de inventarios.

**Tabla 3.22: R022 área de inventarios**

IDENTIFICADOR	R022	NOMBRE	INVENTARIO DE PRODUCTOS POR CATEGORIA
FECHA	2 febrero del 2012		
ÁREA	Inventarios		
DESCRIPCIÓN	El sistema debe permitir conocer el stock de producto por categoría.		

**Tabla 3.23: R023 área de inventarios**

IDENTIFICADOR	R023	NOMBRE	INVENTARIO DE PRODUCTOS POR BODEGA
FECHA	2 febrero del 2012		
ÁREA	Inventarios		
DESCRIPCIÓN	El sistema debe permitir conocer el stock de producto por bodega.		

Todos los requerimientos fueron aceptados por la gerencia de sistemas de la empresa y cumplidos por los desarrolladores del proyecto, la constancia se encuentra en el **Anexo D**.



### 3.3.4 Análisis y requerimientos

Partimos del análisis de la base de datos original instalada en un servidor con las siguientes características.

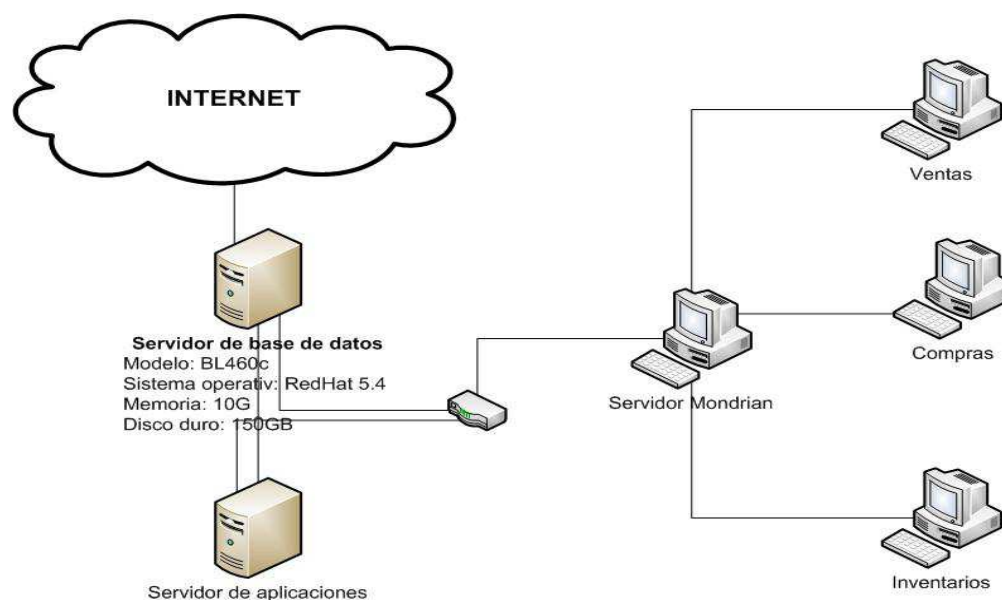
**Tabla 3.24: Características del servidor de base de datos.**

BAHÍAS	MODELO	FIRMWARE	SISTEMA OPERATIVO	MEMORIA	DISCODURO
Bahía 2	BL460c	Versión 9.0	Redhat 5.4	10 GB	150 GB

Esta base de datos es consumida por un sistema ERP (Open Bravo) donde la empresa maneja todas las transacciones y procesos de negocio.

El objetivo de usar un sistema de BI, para las áreas de ventas, compras e inventarios, es poder obtener reportes e información detallada en base a los requerimientos establecidos en cada área estratégica de negocio, de forma independiente al ERP de la empresa.

La siguiente figura muestra la topología de red en la que funcionará la aplicación de BI.



**Figura 3.1 Topología de red BI**

### **3.3.5 Esquema de la base de datos original**

Después del análisis de los requerimientos se diseño un esquema para cada área de negocio de la base de datos original de donde se extraerán los datos para las aéreas de ventas, compras e inventarios.

A continuación se muestra los diagramas de origen de datos para cada área de negocio. Se especificará las tablas de la base de datos de la empresa de donde se tomarán los datos para alimentar a los diferentes Data Marts.

**Espacio en blanco intencional**

**Diagrama base de datos de origen para el cubo de ventas por factura.**

**Diagrama base de datos de origen para el cubo de ventas por remisión.**

**Diagrama base de datos de origen para el cubo de compras por factura.**

**Diagrama base de datos de origen para el cubo de inventario por bodega.**

## **3.4 Diseño técnico de la Arquitectura**

### **3.4.1. Estándares**

Se definió estándares para los nombres de dimensiones, tablas de hechos, secuencias y en el proceso ETL. Con el objetivo de que el sistema BI pueda crecer y ser entendido por los técnicos sin complicación.

#### **3.4.1.1 Estándar para el modelado**

Palabras claves para los nombres de los campos y tablas de dimensión, hechos y auditoría.

##### **DIM**

Prefijo para las dimensiones.

##### **FACT**

Prefijo para las tablas de hechos.

##### **HIS**

Prefijo de las tablas de auditoría.

##### **ÁREA DE NEGOCIO**

Asignar un nombre en singular relacionado a la entidad.

##### **IDENTIFICADOR**

Asignar un nombre, máximo de tres caracteres en base al documento o entidad relacionada con los datos.

##### **SK**

Usará una clave subrogada (SK), de tipo numérica, la cual puede ser generada por una secuencia de la base de datos o por las herramientas ETL.

## ÁREA DE NEGOCIO 1

Serán los tres primeros caracteres del **área de negocio**.

## MEDIDA

Estará relacionada con operación numérica que se realice o el campo numérico que represente.

## ÁREA DE NEGOCIO 2

Será la primera letra del **área de negocio**.

## Estándar de nombres para las tablas.

Se diseñó el estándar que se muestra en la siguiente tabla, donde se puede ver como asignar los nombres a las tablas de dimensión, auditoría y hechos. Estas tablas se crearon para el diseño de cada Data Mart, correspondiente a las áreas de compras, ventas e inventarios con el mismo formato.

**Tabla 3.25: Nombres para tablas según el estándar.**

Alcance	Prefijo	Numero de caracteres	Ejemplo	Descripción
Para las tablas de dimensiones	DIM_	Máximo 25	DIM_CLIENTE	DIM_ÁREA DE NEGOCIO
Para tablas de hechos	FACT_	MAXIMO 40	FACT_VENTA_FAC	FACT_ÁREA DE NEGOCIO_IDENTIFICADOR
Para la tabla de auditoría	HIS_	MAXIMO 15	HIS_AUDITORIA	HIS_ÁREA DE NEGOCIO

## Estándar de nombres de los campos para las dimensiones

Las claves primarias de las dimensiones tendrán una clave subrogada denominada SK. Esta se generó mediante una secuencia en la base de datos.



Tomando con ejemplo la secuencia de la dimensión cliente llamada SEC\_DIM\_CLIENTE este formato se aplica para todas las dimensiones de los Data Marts de cada área de negocio.

**Tabla 3.26: Estándar de nombres para campos en tablas de dimensiones**

Alcance	Prefijo	Numero de caracteres	Ejemplo	Descripción
Claves primarias en las dimensiones (SK)	Cli_	Máximo 15	Cli_sk	ÁREA DE NEGOCIO1_SK
Campos	Cli_	MAXIMO 15	Cli_Doc_Nom	ÁREA DE NEGOCIO 1_ IDENTIFICADOR _IDENTIFICADOR

#### **Estándar de nombres de los campos para las tablas de hechos**

Para los campos de las tablas de hechos se aplicara el estándar especificado en la tabla en la siguiente tabla.

La columna de la descripción esta previamente explicada en la sección de estándares para el modelado. Se debe tomar en cuenta que la tabla de hechos tiene una clave primaria es compuesta y está conformada por las claves primarias de cada dimensión.

**Tabla 3.27: Estándar de nombres para campos en tablas de hechos**

Alcance	Prefijo	Numero de caracteres	Ejemplo	Descripción
Claves primarias en las tablas de hechos	Fact_x_	Máximo 15	Fact_C_Fac_Cli	FACT_ ÁREA DE NEGOCIO2_ IDENTIFICADOR_ ÁREA DE NEGOCIO 1
Campos de las tablas de hechos	Fact_x_	MAXIMO 15	Fact_C_Fac_Total	FACT_ ÁREA DE NEGOCIO2_ IDENTIFICADOR_ MEDIDA

### 3.4.1.2 Estándar de las secuencias

Las secuencias se crearon para generar las claves subrogadas SK para las dimensiones y la tabla de auditoría. Estas pasaran a ser las claves primarias de cada dimensión

**SEC:** Prefijo para las secuencias.

**DIM\_ÁREA DE NEGOCIO:** Representa el nombre de la dimensión.

**Tabla 3.28: Estándar de nombres para secuencias**

Alcance	Prefijo	Ejemplo	Descripción
Secuencia para la dimensión cliente	SEC_	SEC_DIM_CLIENETE	SEC_DIM_AREA DE NEGOCIO
Secuencia para la dimensión bodega	SEC_	SEC_DIM_BODEGA	SEC_DIM_AREA DE NEGOCIO
Secuencia para la dimensión factura	SEC_	SEC_DIM_FACTURA	SEC_DIM_AREA DE NEGOCIO
Secuencia para la dimensión producto	SEC_	SEC_DIM_PRODUCTO	SEC_DIM_AREA DE NEGOCIO
Secuencia para la dimensión remisión	SEC_	SEC_DIM_REMISIÓN	SEC_DIM_AREA DE NEGOCIO
Secuencia para la dimensión tiempo	SEC_	SEC_DIM_TIEMPO	SEC_DIM_AREA DE NEGOCIO
Secuencia para la dimensión auditoria	SEC_	SEC_HIS_AUDIORIA	SEC_HIS_AREA DE NEGOCIO

### 3.4.1.3 Estándar para el Proceso ETL

#### Estándar para proceso ETL de una dimensión.

En este caso explicaremos como nombrar la dimensión cliente (DIM\_CLIENTE).

El mismo proceso se utilizara para las demás dimensiones.

Para la carga de las dimensiones se utilizan las siguientes herramientas del Spoon.

**Table input:** Se asignara el nombre original de la tabla en la base de datos de la empresa.

**Add sequence:** Las secuencias tendrán el mismo nombre con que fueron creadas en la base de datos del Data Warehouse.

**Get System Info:** Esta opción tendrá un prefijo INFO\_ seguido del nombre de la dimensión.

**Insert/Update:** Se pondrá el nombre de la dimensión.

**Table Output:** Tendrá el nombre de la tabla de auditoría.

**Tabla 3.29: Estándar para proceso ETL de una dimensión.**

Herramientas Spoon	Nombre ETL	Descripción
Table input	C_BPARTNER	Tabla que contiene los datos extraídos mediante un SQL.
Add sequence	SEC_DIM_CLIENTE SEC_HIS_AUDITORIA	Secuencia para las dimensiones y la tabla de auditoría.
Get System Info	INFO_DIM_CLIENTE	Extraer información como la IP, nombre ETL, fecha ETL para en casos de auditorías.
Insert/Update	DIM_CLIENTE	Controla lo que se inserta o actualiza en la base de datos del Data Warehouse.
Table Output	SEC_HIS_AUDITORIA	Contiene los datos para la auditoria extraidos con el Get System Info

### **Estándar para proceso ETL de una tabla de hechos.**

En este caso explicaremos como nombrar la Fact (FACT\_INVENTARIO\_BOD).

El mismo proceso se utilizara para las demás dimensiones.

Para la carga de las Facts se utilizan las siguientes herramientas del Spoon.

**Table input:** Se asignara el nombre original de la tabla en la base de datos de la empresa.

**Database Lookup:** tendrá el mismo nombre de la dimensión si el prefijo DIM.

**Group by:** Tendrá el nombre VALORES.

**Add sequence:** Las secuencias tendrán el mismo nombre con que fueron creadas en la base de datos del Data Warehouse.

**Get System Info:** Tendrá un prefijo INFO\_ seguido del nombre de la Fact.

**Insert/Update:** Se pondrá el nombre de la Fact.

**Table Output:** Tendrá el nombre de la tabla de auditoría.

**Tabla 3.30: Estándar para proceso ETL para una tabla de hechos.**

Herramientas Spoon	Nombre ETL	Descripción
Table input	DBEMPAQPLAST	Tabla que contiene los datos extraídos mediante un SQL.
Database Lookup	BODEGA PRODUCTO	Se realiza la conexión entre la dimensión y la tabla de la base de datos de la empresa
Group by	VALORES	Aquí se realizan las operaciones o cálculos, sumas, promedios.
Add sequence	SEC_DIM_CLIENTE SEC_HIS_AUDITORIA	Secuencia para las dimensiones y la tabla de auditoría.
Get System Info	INFO_DIM_CLIENTE	Extraer información como la IP, nombre ETL, fecha ETL para en casos de auditorías.
Insert/Update	DIM_CLIENTE	Controla lo que se inserta o actualiza en la base de datos del Data Warehouse.
Table Output	SEC_HIS_AUDITORIA	Contiene los datos para la auditoría extraídos con el Get System Info

### 3.4.2 Entorno Back Room

Los datos serán extraídos de la base de datos de la empresa que se encuentra en Oracle 10g, el proceso ETL se lo realizara con la herramienta Pentaho Data Integration (PDI) de la suite de Pentaho, el alojamiento de los datos en el Data Warehouse se lo hará en una base de datos Oracle 10g subida en un servidor Centos.



**Figura 3.6 Entorno Back Room**

### 3.4.3 Entorno Front Room

Una vez poblado el Data Warehouse con los datos respectivos se podrá visualizar los resultados a través del servidor Mondrian de Pentaho.

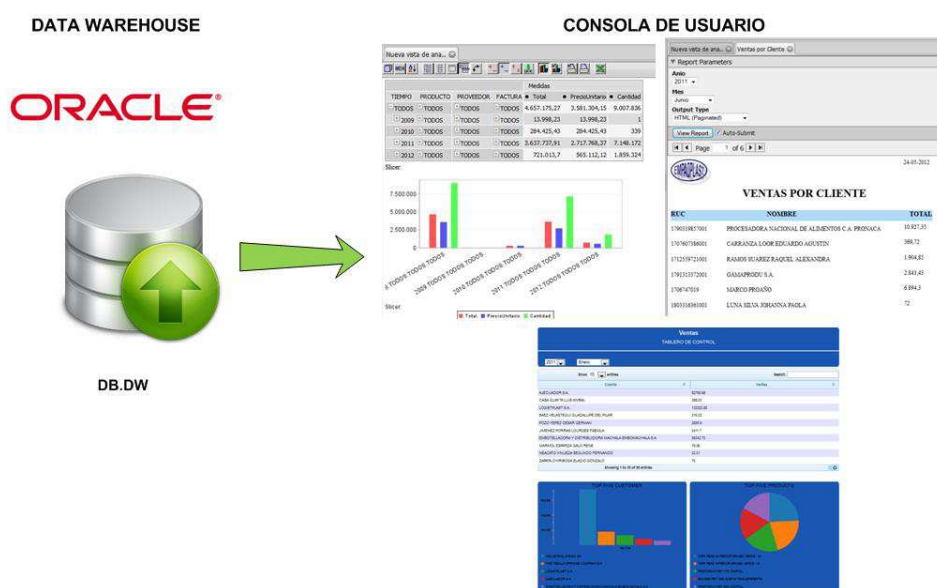



Figura 3.7 Front Room

### 3.5 Selección de Productos e Instalación

Para el desarrollo del proyecto de BI se trabajo con las herramientas de Pentaho Community. Detalladas en el siguiente cuadro.

La instalación de todos los productos se la realiza de forma detallada en el **Anexo F**.

**Tabla 3.31 productos de instalación**

<b>Producto</b> 	<b>Característica</b>	<b>Uso</b>	<b>Hardware</b>
<b>Pentaho BI Server 3.8</b>	Servidor OLAP	Administración de usuarios  Publicación de reportes, cubos  Diseño de tableros de mando (dashboard).	Centos 6 4 GB Ram Disco duro de 500 GB Intel Core i7
<b>Pentaho Report Designer</b>	Reporteador	Diseño y construcción de reportes	Centos 6 4 GB Ram Disco duro de 500 GB Intel Core i7
<b>Pentaho Data Integration</b>	Integración de Datos (ETL)	Diseño y desarrollo del proceso de extracción transformación y carga (ETL)	Centos 6 4 GB Ram Disco duro de 500 GB Intel Core i7
<b>Pentaho Schema Work Bench</b>	Diseñador de Cubos Dimensionales.	Diseño y construcción de cubos multidimensionales.	Centos 6 4 GB Ram Disco duro de 500 GB Procesador Intel Core i7
<b>CDE</b>	Comunity Dashboard Editor.	Diseño y construcción de tableros de control.	Centos 6 4 GB Ram Disco duro de 500 GB Procesador Intel Core i7

## 3.6 Modelado Dimensional

### 3.6.1 Bus Matrix Data Warehouse

La tabla de bus matrix define cuales son las dimensiones que se van a compartir en los diferentes Data Marts, la granularidad describe el detalle de los datos que se van a mostrar, a mayor granularidad mayor es el detalle de las consultas. Para el caso del presente se utilizo granularidad fina.

**Tabla 3.32: Bus Matrix Data Warehouse**

Proceso de Negocio	Tablas de Hecho	Medidas _Fact	Granularidad	Dimensiones					
				Dim_factura	Dim_remision	Dim_cliente	Dim_producto	Dim_tiempo	Dim_bodega
Ventas	Fact_venta_fac	Cantidad Precio actual Valor total	Detalle de ventas por fechas, categorías y subcategorías de clientes, categorías y subcategorías de productos, mostrando a que factura pertenece	*		*	*	*	
	Fact_venta_Rem	Cantidad Precio actual Valor total	Detalle de ventas por periodos de tiempo, fechas, categorías y subcategorías de clientes, productos mostrando a que remisión pertenecen		*	*	*	*	
Compras	Fact_compra_Fac	Cantidad Precio actual Valor total	Detalle de compras por periodos de tiempo, fechas, clientes, productos, categorías de productos, categorías de clientes, mostrando a que factura pertenecen	*		*	*	*	
Inventarios	Fact_inventario_Bod	Cantidad	Detalle de los movimientos de los productos en las bodegas y su stock actual				*		*



### 3.6.2 Modelado Dimensional

El modelo lógico para todas las áreas del negocio se lo realizó siguiendo el esquema de estrella que optimiza el tiempo de respuesta en consultas complejas, siguiendo los pasos del modelo dimensional de Ralph Kimball, se creó cuatro Data Marts, dos para el área de ventas, uno para el área de compras y uno para el área de inventarios. Que conforman el Data Warehouse. Se creó dimensiones que contienen las características de las entidades de cada proceso del negocio, y tablas de hechos que contienen las medidas o cantidades numéricas para el análisis del negocio. A continuación se detalla el proceso de modelado para cada una de las áreas de negocio.

#### 3.6.2.1 Diagrama general origen de datos Data Mart ventas por factura.

El diagrama muestra las dimensiones y la relación que tienen con las tablas de la base de datos origen. Para comprender de donde provienen los datos que se utilizan para poblar el Data Mart de ventas por factura.

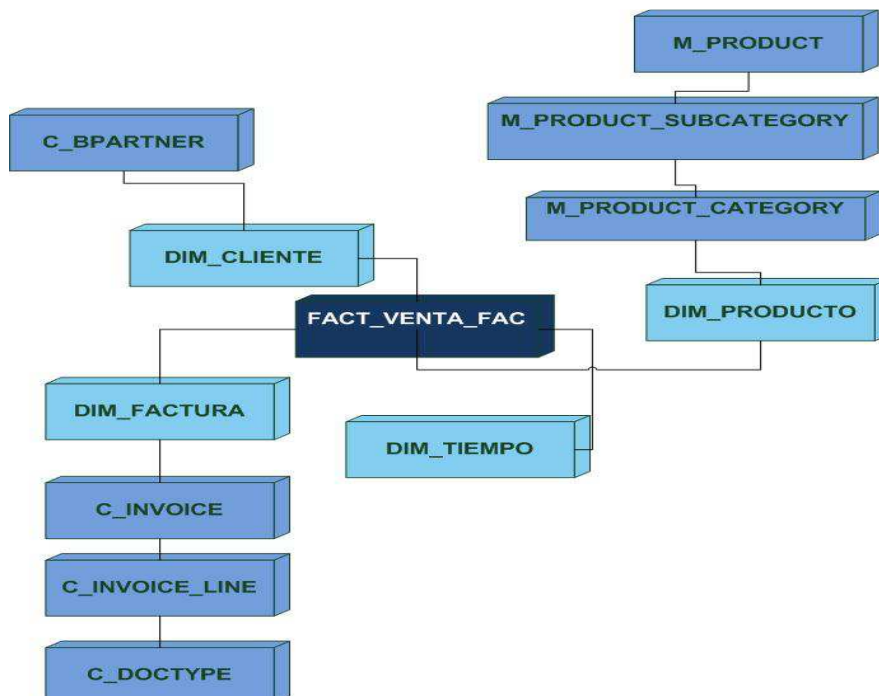


Figura 3.8: Origen de datos general data mart ventas por factura

### 3.6.2.2 Mapeo de datos ventas por factura

En la siguiente tabla se describe el origen de datos de la base de datos de la empresa hacia Las dimensiones y tablas de hechos del Data Warehouse.

**Tabla 3.33: Mapeo de datos ventas por factura**

Fuente de datos original (BD)	Descripción	Cargados	
		Dimensión (DIM)	Tabla de hechos(FACT)
<b>M_PRODUCT</b>	Tabla que contiene los datos de productos.	DIM_PRODUCTO	FACT_VENTA_FAC
<b>M_PRODUCT_SUBCATEGORY</b>	Tabla que contiene las subcategorías del producto.	DIM_PRODUCTO	FACT_VENTA_FAC
<b>M_PRODUCT_CATEGORY</b>	Tabla que contiene las categorías de los productos.	DIM_PRODUCTO	FACT_VENTA_FAC
<b>C_BPARTNER</b>	Tabla que contiene Los clientes.	DIM_CLIENTE	FACT_VENTA_FAC
<b>C_INVOICE</b>	Tabla que contiene las facturas.	DIM_FACTURA	FACT_VENTA_FAC
<b>C_INVOICE_LINE</b>	Tabla que contiene las líneas de las facturas.	DIM_FACTURA	FACT_VENTA_FAC
<b>C_DOCTYPE</b>	Tabla que contiene los tipos de documentos.	DIM_FACTURA	FACT_VENTA_FAC

### 3.6.2.3 Modelo dimensional Data Mart ventas por factura.

Con el previo análisis de los requerimientos se diseñó el modelo lógico para el Data Mart de ventas por factura, la figura muestra los detalles en cuanto a relaciones, nombres de campos y tipos de datos.

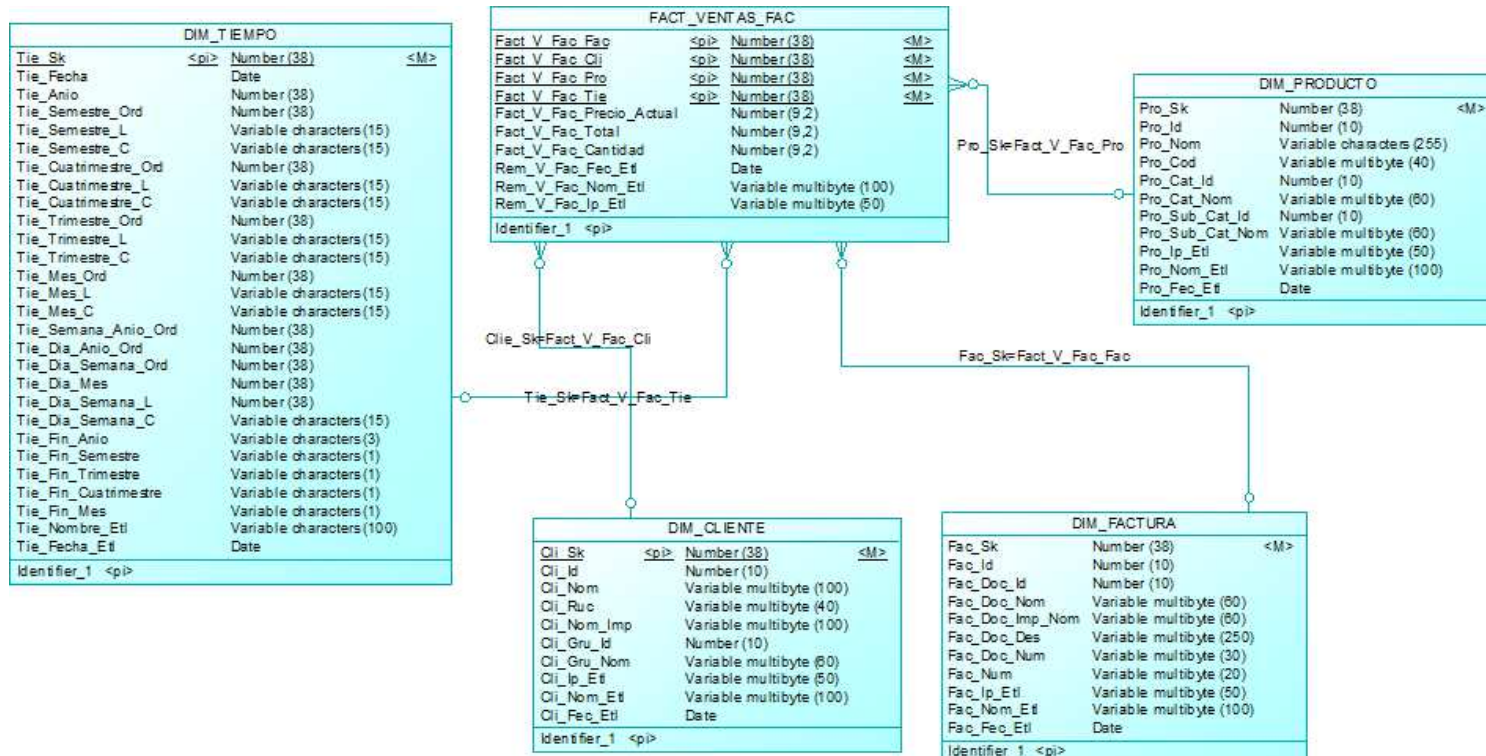


Figura 3.9 Modelo dimensional Data Mart ventas por factura

### 3.6.2.4 Mapeo y detalle datos de dimensiones y hechos Data Mart ventas por factura.

Las siguientes tablas muestran el destino y origen de los datos para dimensiones y tablas de hechos del Data Mart de ventas por factura, se detalla los nombres de los campos de destino y origen los tipos de datos y sus esquemas respectivos.

**Tabla 3.34 Mapeo y detalle dimensión cliente**

Tabla	DIM_CLIENTE									
Tipo de Tabla	Dimensional									
Descripción	Tabla que contiene detalles del cliente									
Esquema	DW									
Destino							Origen			
Columna	Descripción	Tipo de Dato	Tamaño	Clave	FK para	Null	Esquema	Tabla	Campo	Tipo de Dato
Cli_Sk	Clave Subrogada de la dimensión cliente	Integer		Pk	cli_sk	N	DW	dim_cliente	cli_sk	Integer
Cli_Id	Código del cliente	Number	10			Y	obtest	c_Bpartner	c_bpartner_id	Number(10)
Cli_Nom	Nombre del cliente	Nvarchar2	100			Y	obtest	c_Bpartner	name	NVarchar2(100)
Cli_Ruc	Ruc del cliente	Nvarchar2	40			Y	obtest	c_Bpartner	value	NVarchar2(40)
Cli_Nom_Imp	Nombre a imprimir del cliente	Nvarchar2	100			Y	obtest	c_Bpartner	Name2	NVarchar2(100)
Cli_Gru_Id	Código del grupo de cliente	Number	10			Y	obtest	c_bp_group	c_bp_group_id	Number(10)
Cli_Gru_Nom	Nombre del grupo de cliente	Nvarchar2	60			Y	obtest	c_bp_group	name	NVarchar2(60)
Cli_Ip_Etl	Ip de la maquina donde se realizo el etl	Nvarchar2	40			Y	Sistema	Sistema	Sistema	NVarchar2(50)
Cli_Nom_Etl	Nombre del ETL	Nvarcahr2	100			Y	Sistema	Sistema	Sistema	NVarchar2(100)
Cli_Fec_Etl	Fecha de la ejecución o carga del ETL	Date				Y	Sistema	Sistema	Sistema	Date

**Tabla 3.35 Mapeo y detalle dimensión producto**

<b>Tabla</b>				DIM_PRODUCTO		
<b>Tipo de Tabla</b>				Dimensional		
<b>Descripción</b>				Tabla que contiene los datos de los productos		
<b>Esquema</b>				DW		
<b>Origen</b>						
Columna	Descripción	Tipo de Dato	Tamaño	Tabla	Campo	Tipo de Dato
Pro_Sk	Clave Subrogada de la dimensión producto	Integer		dim_producto	pro_sk	Integer
Pro_Id	Id del producto en el sistema	Number	10	m_product	m_product_id	Number(10)
Pro_Nom	Nombre del producto	Nvarchar2	225	m_product	name	NVarchar2(225)
Pro_Cod	Código del producto para la identificación	Nvarchar2	40	m_product	value	NVarchar2(40)
Pro_Cat_Nom	Nombre de la categoría del producto	Nvarchar2	60	m_product_category	name	NVarchar2(60)
Pro_Cat_Id	Id de la categoría del producto	Number	10	m_product_category	m_product_category_id	Number(10)
Pro_Sub_Cat_Nom	Nombre de la subcategoría del producto	Nvarchar2	60	um_product_subcategory	name	NVarchar2(60)

**Tabla 3.36 Mapeo y detalle dimensión de factura**

<b>Tabla</b>	DIM_FACTURA									
<b>Tipo de Tabla</b>										
<b>Descripción</b>	Dimensional Tabla que contiene las facturas y sus detalles									
<b>Esquema</b>	DW									
<b>Destino</b>							<b>Origen</b>			
Columna	Descripción	Tipo de Dato	Tamaño	Clave	FK para	Null	Esquema	Tabla	Campo	Tipo de Dato
Fac_sk	Clave Subrogada de la dimensión factura	Integer		Pk	fac_sk	N	DW	dim_factura	Fac_sk	Integer
Fac_Doc_Id	Id del tipo de documento	Number	10			Y	obtest	c_doctype	c_doctype_id	Number(10)
Fac_Doc_Nom	Nombre del tipo de documento	Nvarchar2	60			Y	obtest	c_doctype	name	NVarchar2(60)
Fac_Doc_Imp_Nom	Nombre simplificado del tipo de documento	Nvarchar2	60			Y	obtest	c_doctype	printname	NVarchar2(60)
Fac_Doc_Des	Descripción del tipo de documento	Nvarchar2	250			Y	obtest	c_doctype	description	NVarchar2(250)
Fac_Doc_Num	Código del tipo de documento	Nvarchar2	30			Y	obtest	c_invoice	documentno	NVarchar2(30)
Fac_Id	Código de la factura	Number	10			Y	obtest	c_invoice	c_invoice_id	Number (10)
Fac_Num	Número de factura	Nvarchar2	20			Y	obtest	c_invoice	poreference	NVarchar2(20)
Fac_Ip_Etl	Ip de la maquina donde se realizo el etl	Nvarchar2	50			Y	Sistema	Sistema	Sistema	NVarchar2(50)
Fac_Nom_Etl	Nombre del ETL	Nvarchar2	100			Y	Sistema	Sistema	Sistema	NVarchar2(100)
Fac_Fec_Etl	Fecha de la ejecución o carga del ETL	date				Y	Sistema	Sistema	Sistema	Date

**Tabla 3.37 Mapeo y detalle fact de ventas por factura**

Tabla										
Tipo de Tabla		FACT_VENTA_C								
Descripción		Hechos								
Esquema		Tabla de hechos para el Datamart de ventas con relación a la entidad factura DW								
Destino							Origen			
Columna	Descripción	Tipo de Dato	Tamaño	Clave	FK para	Nul	Esquema	Tabla	Campo	Tipo de Dato
Fact_V_Fac_Tie	Clave Subrogada de la dimensión tiempo	Integer		Pk	Tie_sk	N	DW	Dim_Tiempo	Tie_sk	Number(38)
Fact_V_Fac_Pro	Clave Subrogada de la dimensión producto	Integer		Pk	Pro_sk	N	DW	Dim_Producto	Pro_sk	Number(10)
Fact_V_Fac_Cli	Clave Subrogada de la dimensión cliente	Integer		Pk	Cli_sk	N	DW	Dim_Cliente	Cli_sk	NVarchar2(225)
Fact_V_Fac_Fac	Clave Subrogada de la dimensión factura	Integer		Pk	Fac_sk	N	DW	Dim_Factura	Fac_sk	NVarchar2(40)
Fact_V_Fac_Cantidad	Medida que representa la cantidad de producto.	Number	9			Y	Obtest	c_invoiceline	qtyinvoice	Number
Fact_V_Fac_Precio_Actual	Medida que representa el precio del producto.	Number	9,2			Y	Obtest	c_invoiceline	priceactual	Number
Fact_V_Fac_Total	Medida que representa el total del producto.	Number	20,2			Y	DW	Fact_ventas	Fact_v_total	Number(20,2)
Fact_V_Fac_Ip_Et	Ip de la maquina donde se realizó el ETL	Nvarchar2	40			Y	Sistema	Sistema	Sistema	NVarchar2(50)
Fact_V_Fac_Nombre_Etl	Nombre del ETL	Nvarcahr2	100			Y	Sistema	Sistema	Sistema	NVarchar2(100)
Fact_V_Fac_Fecha_Etl	Fecha de la ejecución o carga del ETL	Date				Y	Sistema	Sistema	Sistema	Date

### 3.6.2.5 Modelo de base de datos Data Mart ventas por factura.

Finalmente el diagrama de base de datos muestra estructura del Data Mart de ventas por factura. El script de generación esta en el **Anexo C**.

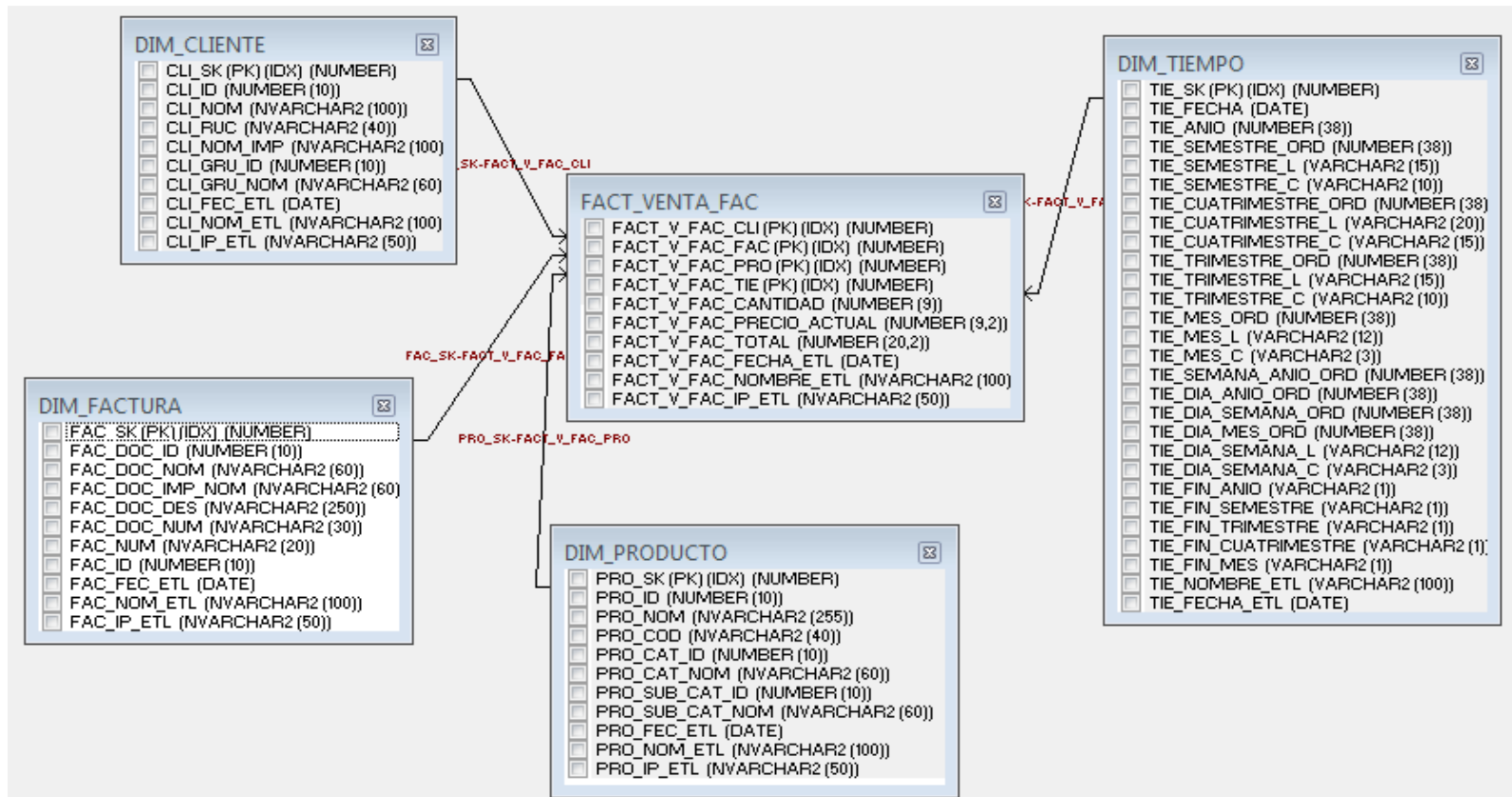


Figura 3.10 Modelo de base de datos Data Mart ventas por factura



### 3.6.2.6 Diagrama general origen de datos Data Mart ventas por remisión

El diagrama muestra las dimensiones y la relación que tienen con las tablas de la base de datos origen. Para comprender de donde provienen los datos que se utilizan para poblar el Data Mart de ventas por remisión.

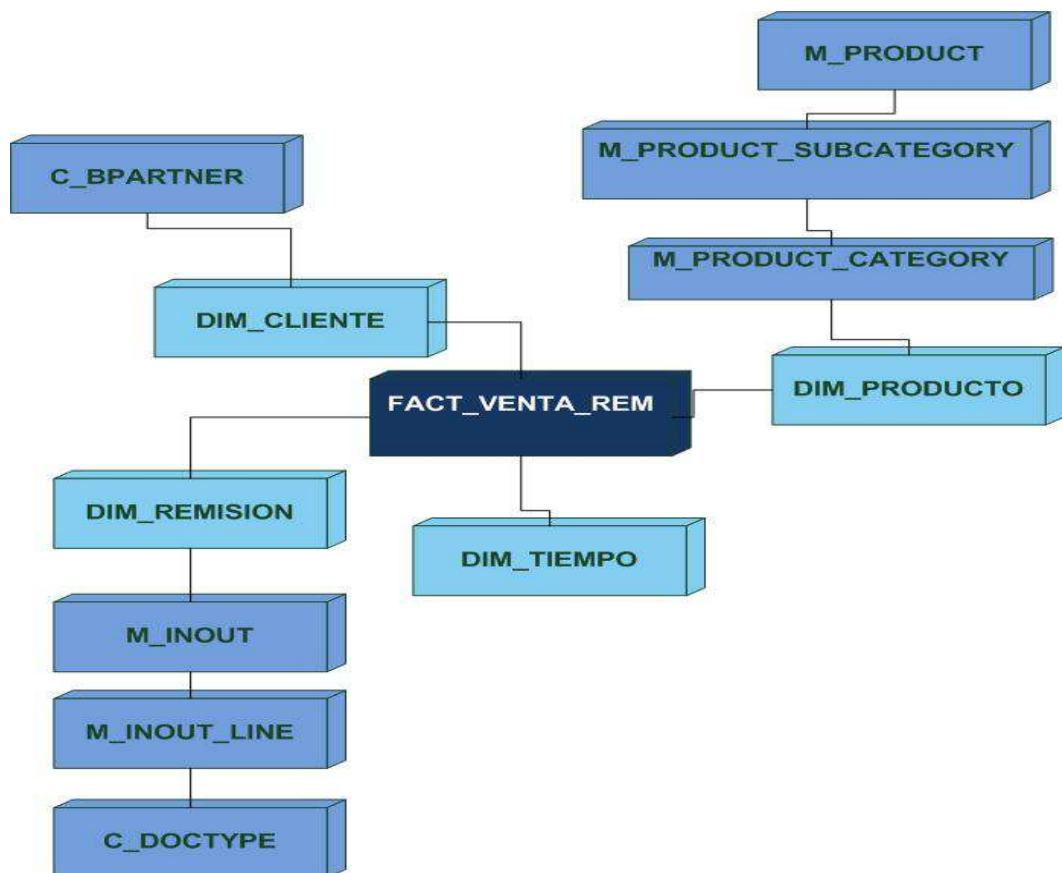


Figura 3.11 Origen de datos general Data Mart ventas por remisión

### 3.6.2.7 Mapeo de datos ventas por remisión

En la siguiente tabla se describe el origen de datos de la base de datos de la empresa hacia las dimensiones y tablas de hechos del Data Warehouse.

**Tabla 3.38: Mapeo de datos ventas por remisión**

Fuente de datos original (BD)	Descripción	Cargados	
		Dimensión (DIM)	Tabla de hechos(FACT)
<b>M_PRODUCT</b>	Tabla que contiene los datos de productos.	DIM_PRODUCTO	FACT_VENTA_REM
<b>M_PRODUCT_SUBCATEGORY</b>	Tabla que contiene las subcategorías del producto.	DIM_PRODUCTO	FACT_VENTA_REM
<b>M_PRODUCT_CATEGORY</b>	Tabla que contiene las categorías de los productos.	DIM_PRODUCTO	FACT_VENTA_REM
<b>C_BPARTNER</b>	Tabla que contiene Los clientes.	DIM_CLIENTE	FACT_VENTA_REM
<b>C_INOUT</b>	Tabla que contiene las remisiones.	DIM_REMISION	FACT_VENTA_REM
<b>C_INOUT_LINE</b>	Tabla que contiene las líneas de las remisiones.	DIM_REMISION	FACT_VENTA_REM
<b>C_DOCTYPE</b>	Tabla que contiene los tipos de documentos.	DIM_REMISION	FACT_VENTA_REM

### 3.6.2.8 Modelo dimensional Data Mart ventas por remisión

Con el previo análisis de los requerimientos se diseñó el modelo lógico para el Data Mart de ventas por remisión, la figura muestra los detalles en cuanto a relaciones, nombres de campos y tipos de datos.

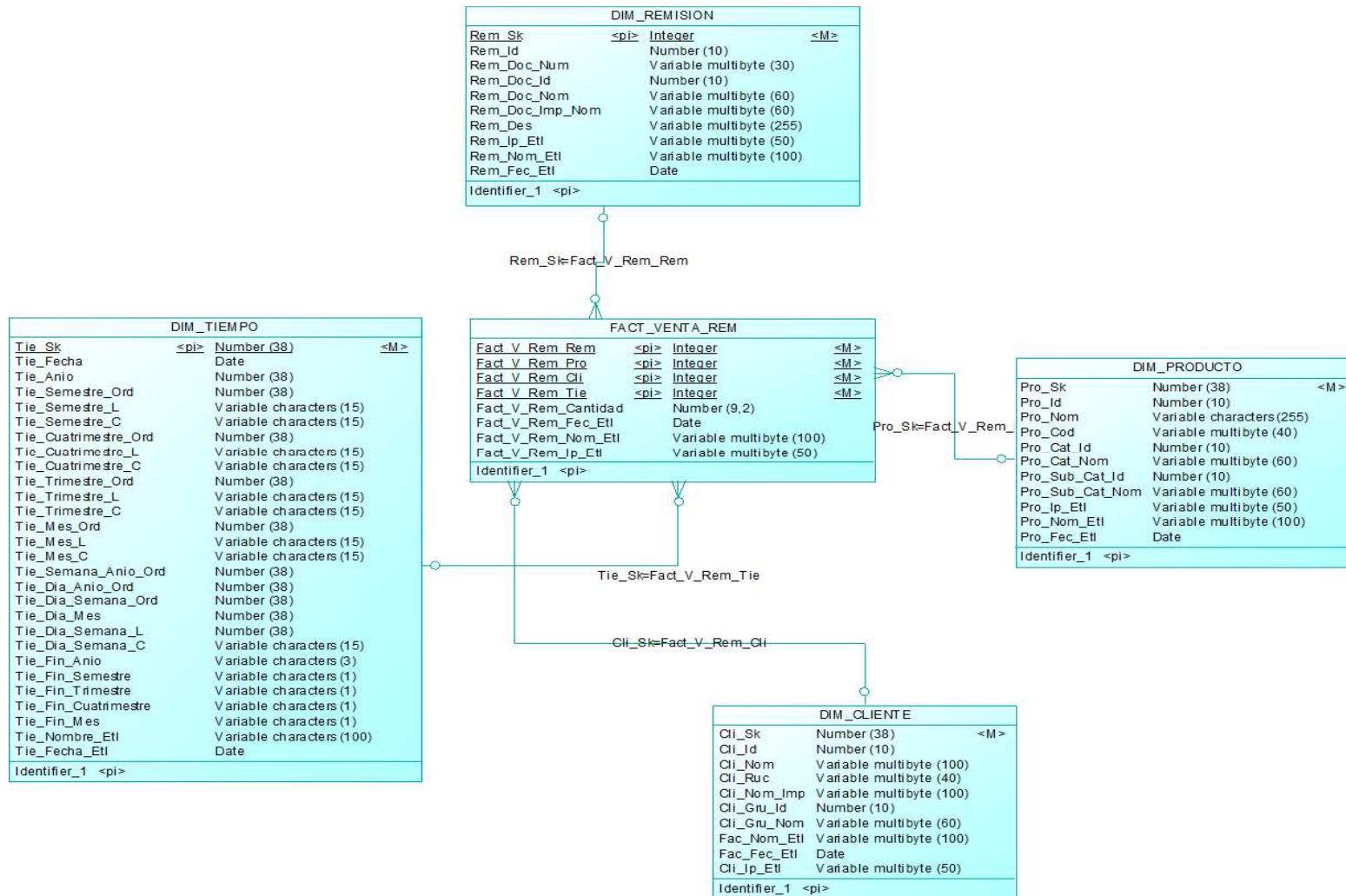


Figura 3.12 Modelo dimensional Data Mart ventas por remisión

### 3.6.2.9 Mapeo y detalle de datos de dimensiones y hechos Data Mart ventas por remisión

Las siguientes tablas muestran el destino y origen de los datos de dimensiones y tablas de hechos para el Data Mart de ventas por remisión, se detalla los nombres de los campos de destino y origen, los tipos de datos y sus esquemas respectivos.

**Tabla 3.39 Mapeo y detalle dimensión cliente**

Tabla	DIM_CLIENTE							
Tipo de Tabla	Dimensional							
Descripción								
TablaEsquema	DW							
Origen								
Columna	Descripción	Tipo de Dato	Tamaño	FK para	Esquema	Tabla	Campo	Tipo de Dato
Cli_Sk	Clave Subrogada de la dimensión cliente	Integer		cli_sk	DW	dim_cliente	cli_sk	Integer
Cli_Id	Código del cliente	Number	10		obtest	c_Bpartner	c_bpartner_id	Number(10)
Cli_Nom	Nombre del cliente	Nvarchar2	100		obtest	c_Bpartner	name	NVarchar2(100)
Cli_Ruc	Ruc del cliente	Nvarchar2	40		obtest	c_Bpartner	value	NVarchar2(40)
Cli_Nom_Im p	Nombre a imprimir del cliente	Nvarchar2	100		obtest	c_Bpartner	Name2	NVarchar2(100)
Cli_Gru_Id	Código del grupo de cliente	Number	10		obtest	c_bp_group	c_bp_group_id	Number(10)
Cli_Gru_Nom	Nombre del grupo de cliente	Nvarchar2	60		obtest	c_bp_group	name	NVarchar2(60)
Cli_Ip_Etl	Ip de la maquina donde se realizo el etl	Nvarchar2	40		Sistema	Sistema	Sistema	NVarchar2(50)
Cli_Nom_Etl	Nombre del ETL	Nvarcahr2	100		Sistema	Sistema	Sistema	NVarchar2(100)
Cli_Fec_Etl	Fecha de la ejecución o carga del ETL	Date			Sistema	Sistema	Sistema	Date

**Tabla 3.40 Mapeo y detalle dimensión producto**

Tabla	DIM_PRODUCTO						
Tipo de Tabla	Dimensional						
Descripción							
fEsquema	DW						
Origen							
Columna	Descripción	Tipo de Dato	Tamaño	Esquema	Tabla	Campo	Tipo de Dato
Pro_Sk	Clave Subrogada de la dimensión producto	Integer		DW	dim_producto	pro_sk	Integer
Pro_Id	Id del producto en el sistema	Number	10	obtest	m_product	m_product_id	Number(10)
Pro_Nom	Nombre del producto	Nvarchar2	225	obtest	m_product	name	NVarchar2(225)
Pro_Cod	Código del producto para la identificación	Nvarchar2	40	obtest	m_product	value	NVarchar2(40)
Pro_Cat_Nom	Nombre de la categoría del producto	Nvarchar2	60	obtest	m_product_category	name	NVarchar2(60)
Pro_Cat_Id	Id de la categoría del producto	Number	10	obtest	m_product_category	m_product_category_id	Number(10)
Pro_Sub_Cat_Nom	Nombre de la subcategoría del producto	Nvarchar2	60	obtest	um_product_subcategory	name	NVarchar2(60)
Pro_Sub_Cat_Id	Id de la subcategoría del producto	Number	10	obtest	um_product_subcategory	um_product_subcategory_id	Number(10)
Pro_Ip_Etl	Ip de la maquina donde se realizó el etl	Nvarchar2	40	Sistema	Sistema	Sistema	NVarchar2(50)
Pro_Nom_Etl	Nombre del ETL	Nvarcahr2	100	Sistema	Sistema	Sistema	NVarchar2(100)
Pro_Fec_Etl	Fecha de la ejecución o carga del ETL	Date		Sistema	Sistema	Sistema	Date

**Tabla 3.41 Mapeo y detalle dimensión remisión**

Origen							
Tabla	DIM_REMISION						
Tipo de Tabla	Dimensional						
Descripción							
Esquema	DW						
Columna	Descripción	Tipo de Dato	Tamaño	Esquema	Tabla	Campo	Tipo de Dato
Rem_sk	Clave Subrogada de la dimensión remisión	Integer	10	Dw	dim_remision	cli_sk	Number(38)
Rem_id	Id del la remisión	Number	10	obtest	m_inout	m_inout_id	Number(10)
Rem_Doc_Num	Numero del tipo de documento	Nvarchar2	30	obtest	m_inout	documentno	NVarchar2(30)
Rem_Doc_Id	Id del tipo de documento	Number	10	obtest	c_doctype	c_doctype_id	Number(10)
Rem_Doc_nom	Nombre del tipo de documento	Nvarchar2	60	obtest	c_doctype	name	NVarchar2(60)
Rem_Imp_Nom	Nombre 2 de la remisión	Nvarchar2	60	obtest	c_doctype	printname	NVarchar2(60)
Rem_Des	Descripción de la remisión	Nvarchar2	255	obtest	c_doctype	description	NVarchar2(255)
Rem_Ip_Etl	Ip de la maquina donde se realizo el etl	Nvarchar2	40	Sistema	Sistema	Sistema	NVarchar2(50)
Rem_Nom_Etl	Nombre del ETL	Nvarcahr2	100	Sistema	Sistema	Sistema	NVarchar2(100)
Rem_Fec_Etl	Fecha de la ejecución o carga del ETL	Date		Sistema	Sistema	Sistema	Date

**Tabla 3.42 Mapeo y detalle fact de ventas por remisión**

Origen							
Tabla							
Tipo de Tabla	FACT_VENTA_REM						
Descripción	Hechos						
Esquema	DW						
Columna	Descripción	Tipo de Dato	Tamaño	Esquema	Tabla	Campo	Tipo de Dato
Fact_V_Rem_Pro	Clave Subrogada de la dimensión producto	Integer		obtest	Dim_Producto	Pro_Sk	Integer
Fact_V_Rem_Cli	Clave Subrogada de la dimensión cliente	Integer		obtest	Dim_Cliente	Clie_Sk	Integer
Fact_V_Rem_Rem	Clave Subrogada de la dimensión remisión	Integer		obtest	Dim_Remision	Rem_Sk	Integer
Fact_V_Rem_Tie	Clave Subrogada de la dimensión tiempo	integer		obtest	Dim_tiempo	Tie_Sk	Integer
Fact_V_Rem_Cantidad	Cantidad de Producto por remisión	Number		obtest	m_inoutline	movementqty	Number
Fact_V_Rem_Ip_Etl	Ip de la maquina donde se realizo el etl	Nvarchar2	40	Sistema	Sistema	Sistema	NVarchar2(50)
Fact_V_Rem_Nom_Etl	Nombre del ETL	Nvarcahr2	100	Sistema	Sistema	Sistema	NVarchar2(100)
Fact_V_Rem_Fec_Etl	Fecha de la ejecución o carga del ETL	Date		Sistema	Sistema	Sistema	Date

### 3.6.2.10 Modelo de base de datos Data Mart ventas por remisión

Finalmente el diagrama de base de datos muestra estructura del Data Mart de ventas por factura. El script de generación esta en el **Anexo C**.

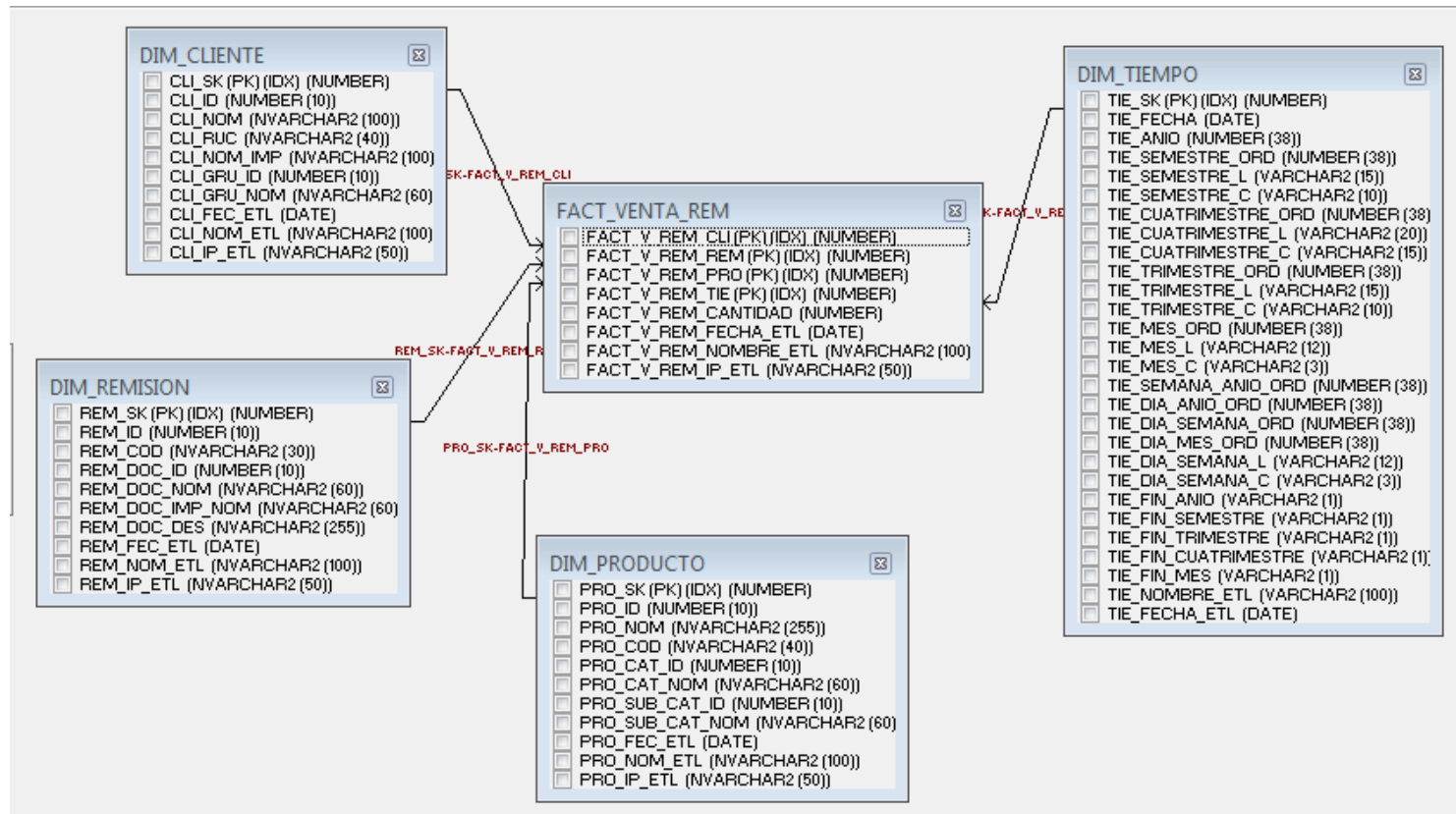


Figura 3.13 Modelo de base de datos Data Mart ventas por remisión.



### 3.6.2.11 Diagrama general origen de datos Data Mart compras por factura

El diagrama muestra las dimensiones y la relación que tienen con las tablas de la base de datos origen. Para comprender de donde provienen los datos que se utilizan para poblar el Data Mart de compras por factura.

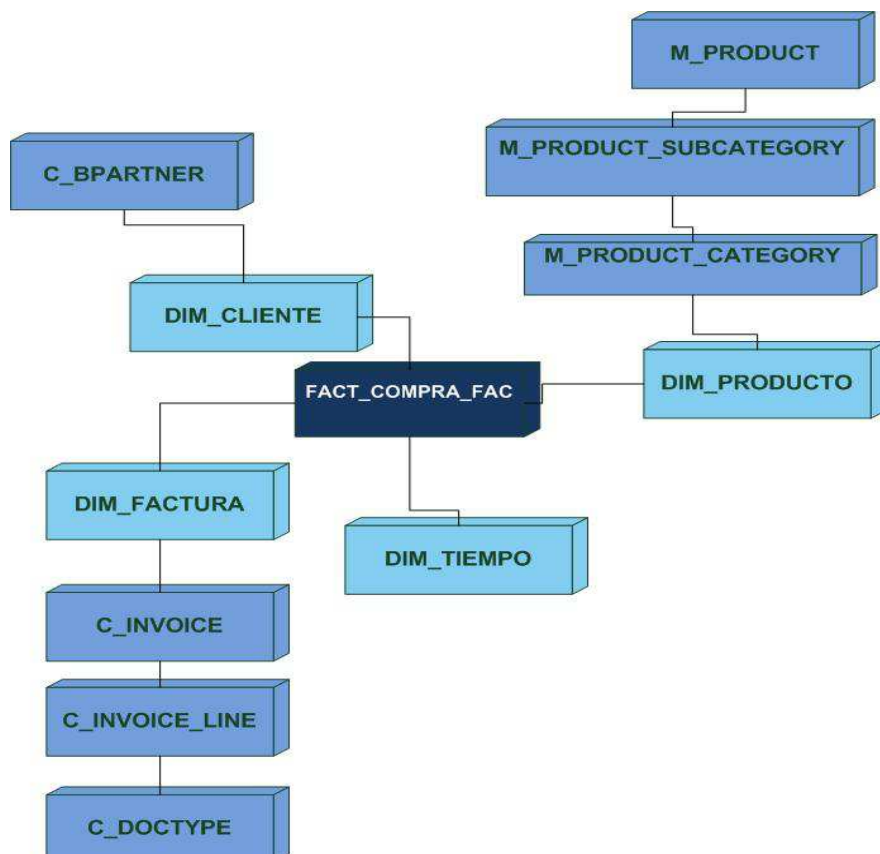


Figura 3.14 Origen de datos general Data Mart compras por factura

### 3.6.2.12 Mapeo de datos compras por factura

En la siguiente tabla se describe el origen de datos de la base de datos de la empresa hacia Las dimensiones y tablas de hechos del Data Warehouse.

**Tabla 3.43: Mapeo de datos compras por factura**

		<b>Cargados</b>	
<b>Fuente de datos original (BD)</b>	<b>Descripción</b>	<b>Dimensión (DIM)</b>	<b>Tabla de hechos(FACT)</b>
<b>M_PRODUCT</b>	Tabla que contiene los datos de productos.	DIM_PRODUCTO	FACT_COMPRAS_FAC
<b>M_PRODUCT_SUBCATEGORY</b>	Tabla que contiene las sub-categorías del producto.	DIM_PRODUCTO	FACT_COMPRAS_FAC
<b>M_PRODUCT_CATEGORY</b>	Tabla que contiene las categorías de los productos.	DIM_PRODUCTO	FACT_COMPRAS_FAC
<b>C_BPARTNER</b>	Tabla que contiene Los clientes.	DIM_CLIENTE	FACT_COMPRAS_FAC
<b>C_INVOICE</b>	Tabla que contiene las facturas.	DIM_FACTURA	FACT_COMPRAS_FAC
<b>C_INVOICE_LINE</b>	Tabla que contiene las líneas de las facturas.	DIM_FACTURA	FACT_COMPRAS_FAC
<b>C_DOCTYPE</b>	Tabla que contiene los tipos de documentos.	DIM_FACTURA	FACT_COMPRAS_FAC

### 3.6.2.13 Modelo dimensional Data Mart compras por factura

Con el previo análisis de los requerimientos se diseño el modelo lógico para el Data Mart de compras por factura, la figura muestra los detalles en cuanto a relaciones, nombres de campos y tipos de datos.

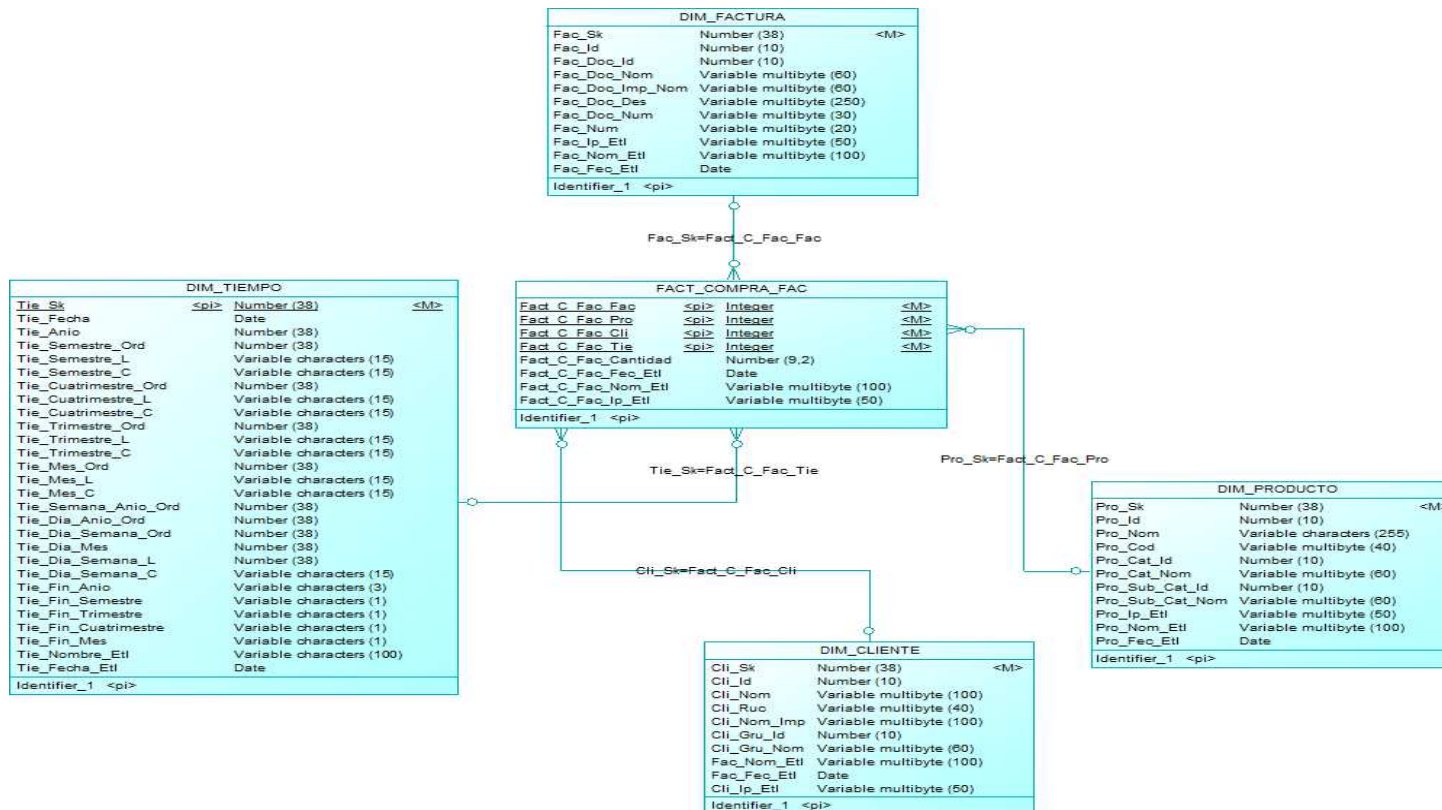


Figura 3.15 Modelo dimensional Data Mart compras por factura

### 3.6.2.14 Mapeo y detalle de datos de dimensiones y hechos Data Mart compras por factura

Las siguientes tablas muestran el destino y origen de los datos de dimensiones y tablas de hechos para el Data Mart de compras por factura, se detalla los nombres de los campos de destino y origen los tipos de datos y sus esquemas respectivos.

**Tabla 3.44 Mapeo y detalle dimensión cliente**

Tabla	DIM_CLIENTE							
Tipo de Tabla	Dimensional							
Descripción								
Esquema	DW							
Origen								
Columna	Descripción	Tipo de Dato	Tamaño	FK para	Esquema	Tabla	Campo	Tipo de Dato
Cli_Sk	Clave Subrogada de la dimensión cliente	Integer		cli_sk	DW	dim_cliente	cli_sk	Integer
Cli_Id	Código del cliente	Number	10		obtest	c_Bpartner	c_bpartner_id	Number(10)
Cli_Nom	Nombre del cliente	Nvarchar2	100		obtest	c_Bpartner	name	NVarchar2(100)
Cli_Ruc	Ruc del cliente	Nvarchar2	40		obtest	c_Bpartner	value	NVarchar2(40)
Cli_Nom_Imp	Nombre a imprimir del cliente	Nvarchar2	100		obtest	c_Bpartner	Name2	NVarchar2(100)
Cli_Gru_Id	Código del grupo de cliente	Number	10		obtest	c_bp_group	c_bp_group_id	Number(10)
Cli_Gru_Nom	Nombre del grupo de cliente	Nvarchar2	60		obtest	c_bp_group	name	NVarchar2(60)
Cli_Ip_Etl	Ip de la maquina donde se realizo el etl	Nvarchar2	40		Sistema	Sistema	Sistema	NVarchar2(50)
Cli_Nom_Etl	Nombre del ETL	Nvarcahr2	100		Sistema	Sistema	Sistema	NVarchar2(100)
Cli_Fec_Etl	Fecha de la ejecución o carga del ETL	Date			Sistema	Sistema	Sistema	Date

**Tabla 3.45 Mapeo y detalle dimensión producto**

<b>Tabla</b>	DIM_PRODUCTO						
<b>Tipo de Tabla</b>	Dimensional						
<b>Descripción</b>							
<b>fEsquema</b>	DW						
<b>Origen</b>							
Columna	Descripción	Tipo de Dato	Tamaño	Esquema	Tabla	Campo	Tipo de Dato
Pro_Sk	Clave Subrogada de la dimensión producto	Integer		DW	dim_producto	pro_sk	Integer
Pro_Id	Id del producto en el sistema	Number	10	obtest	m_product	m_product_id	Number(10)
Pro_Nom	Nombre del producto	Nvarchar2	225	obtest	m_product	name	NVarchar2(225)
Pro_Cod	Código del producto para la identificación	Nvarchar2	40	obtest	m_product	value	NVarchar2(40)
Pro_Cat_Nom	Nombre de la categoría del producto	Nvarchar2	60	obtest	m_product_category	name	NVarchar2(60)
Pro_Cat_Id	Id de la categoría del producto	Number	10	obtest	m_product_category	m_product_category_id	Number(10)
Pro_Sub_Cat_Nom	Nombre de la subcategoría del producto	Nvarchar2	60	obtest	um_product_subcategory	name	NVarchar2(60)
Pro_Sub_Cat_Id	Id de la subcategoría del producto	Number	10	obtest	um_product_subcategory	um_product_subcategory_id	Number(10)
Pro_Ip_Etl	Ip de la maquina	Nvarchar2	40	Sistema	Sistema	Sistema	NVarchar2(50)
Pro_Nom_Etl	Nombre del ETL	Nvarcahr2	100	Sistema	Sistema	Sistema	NVarchar2(100)
Pro_Fec_Etl	Fecha de la ejecución o carga del ETL	Date		Sistema	Sistema	Sistema	Date

**Tabla 3.46 Mapeo y detalle dimensión de factura**

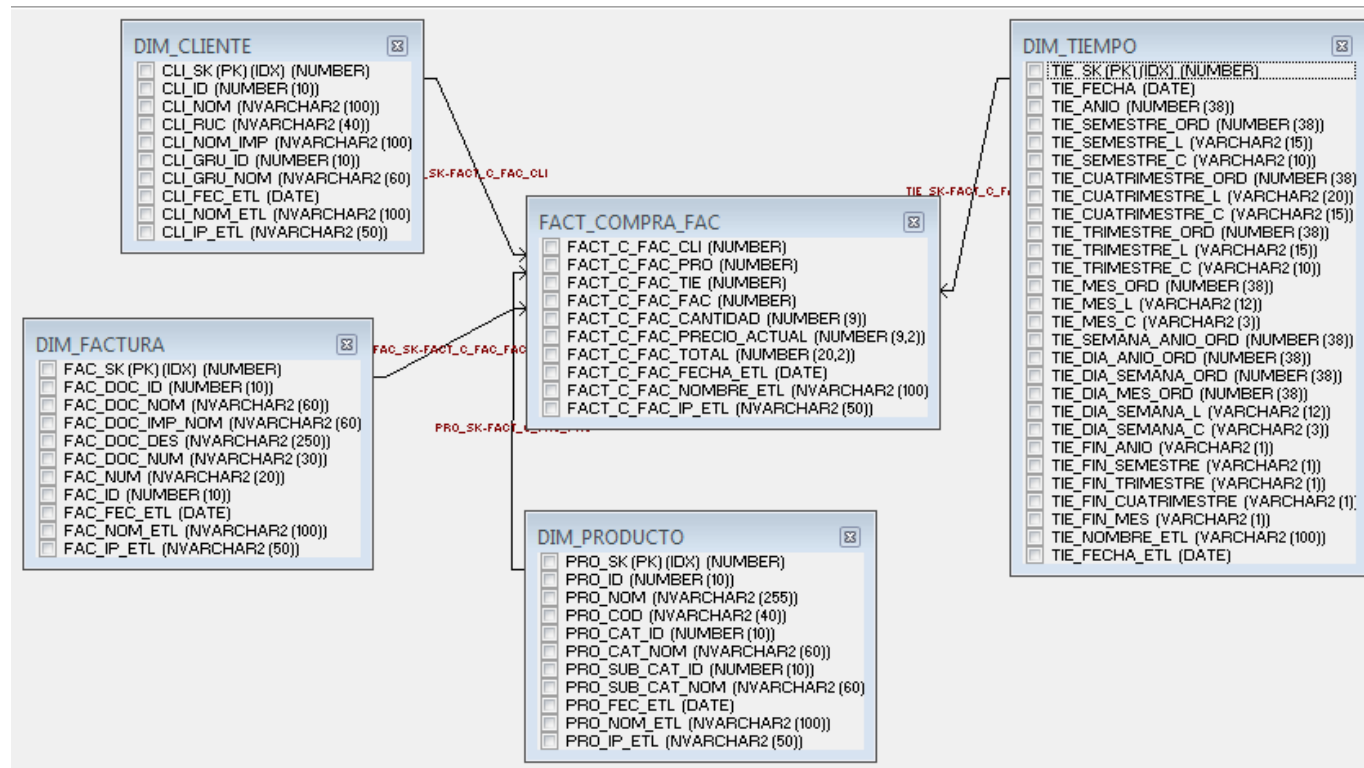
<b>Tabla</b>	DIM_FACTURA									
<b>Tipo de Tabla</b>										
	Dimensional									
<b>Descripción</b>	Tabla que contiene las facturas y sus detalles									
<b>Esquema</b>	DW									
<b>Destino</b>							<b>Origen</b>			
<b>Columna</b>	<b>Descripción</b>	<b>Tipo de Dato</b>	<b>Tamaño</b>	<b>Clave</b>	<b>FK para</b>	<b>Null</b>	<b>Esquema</b>	<b>Tabla</b>	<b>Campo</b>	<b>Tipo de Dato</b>
Fac_sk	Clave Subrogada de la dimensión factura	Integer		Pk	fac_sk	N	DW	dim_factura	Fac_sk	Integer
Fac_Doc_Id	Id del tipo de documento	Number	10			Y	obtest	c_doctype	c_doctype_id	Number(10)
Fac_Doc_Nom	Nombre del tipo de documento	Nvarchar2	60			Y	obtest	c_doctype	name	NVarchar2(60)
Fac_Doc_Imp_Nom	Nombre simplificado del tipo de documento	Nvarchar2	60			Y	obtest	c_doctype	printname	NVarchar2(60)
Fac_Doc_Des	Descripción del tipo de documento	Nvarchar2	250			Y	obtest	c_doctype	description	NVarchar2(250)
Fac_Doc_Num	Código del tipo de documento	Nvarchar2	30			Y	obtest	c_invoice	documentno	NVarchar2(30)
Fac_Id	Código de la factura	Number	10			Y	obtest	c_invoice	c_invoice_id	Number (10)
Fac_Num	Número de factura	Nvarchar2	20			Y	obtest	c_invoice	poreference	NVarchar2(20)
Fac_Ip_Etl	Ip de la maquina donde se realizo el etl	Nvarchar2	50			Y	Sistema	Sistema	Sistema	NVarchar2(50)
Fac_Nom_Etl	Nombre del ETL	Nvarchar2	100			Y	Sistema	Sistema	Sistema	NVarchar2(100)
Fac_Fec_Etl	Fecha de la carga del ETL	date				Y	Sistema	Sistema	Sistema	Date

**Tabla 3.47 Mapeo y detalle fact de compras por factura**

Tabla								
Tipo de Tabla	FACT_COMPRA_FAC							
Descripción	Hechos							
Esquema	DW							
Origen								
Columna	Descripción	Tipo de Dato	Tamaño	FK para	Esquema	Tabla	Campo	Tipo de Dato
Fact_C_Fac_Tie	Clave Subrogada de la dimensión tiempo	Integer		Tie_sk	DW	Dim_Tiempo	Tie_sk	Number(38)
Fact_C_Fac_Pro	Clave Subrogada de la dimensión producto	Integer		Pro_sk	DW	Dim_Producto	Pro_sk	Number(10)
Fact_C_Fac_Cli	Clave Subrogada de la dimensión cliente	Integer		Cli_sk	DW	Dim_Cliente	Cli_sk	NVarchar2(225)
Fact_C_Fac_Fac	Clave Subrogada de la dimensión factura	Integer		Fac_sk	DW	Dim_Factura	Fac_sk	NVarchar2(40)
Fact_C_Fac_Cantidad	Medida que representa la cantidad de producto comprado	Number	9		Obtest	c_invoiceline	qtyinvoice	Number
Fact_C_Fac_Precio_Actual	Medida que representa el precio del producto comprado.	Number	9,2		Obtest	c_invoiceline	priceactual	Number
Fact_C_Fac_Total	Medida que representa el total del producto comprado	Number	20,2		DW	Fact_ventas	Fact_v_total	Number(20,2)
Fact_C_Fac_Ip_Et	Ip de la maquina donde se realizó el ETL	Nvarchar2	40		Sistema	Sistema	Sistema	NVarchar2(50)
Fact_C_Fac_Nombre_Etl	Nombre del ETL	Nvarcahr2	100		Sistema	Sistema	Sistema	NVarchar2(100)
Fact_C_Fac_Fecha_Etl	Fecha de la ejecución o carga del ETL	Date			Sistema	Sistema	Sistema	Date

### 3.6.2.15 Modelo de base de datos Data Mart compras por factura

Finalmente el diagrama de base de datos muestra estructura del Data Mart de ventas por factura. El script de generación esta en el **Anexo C**.

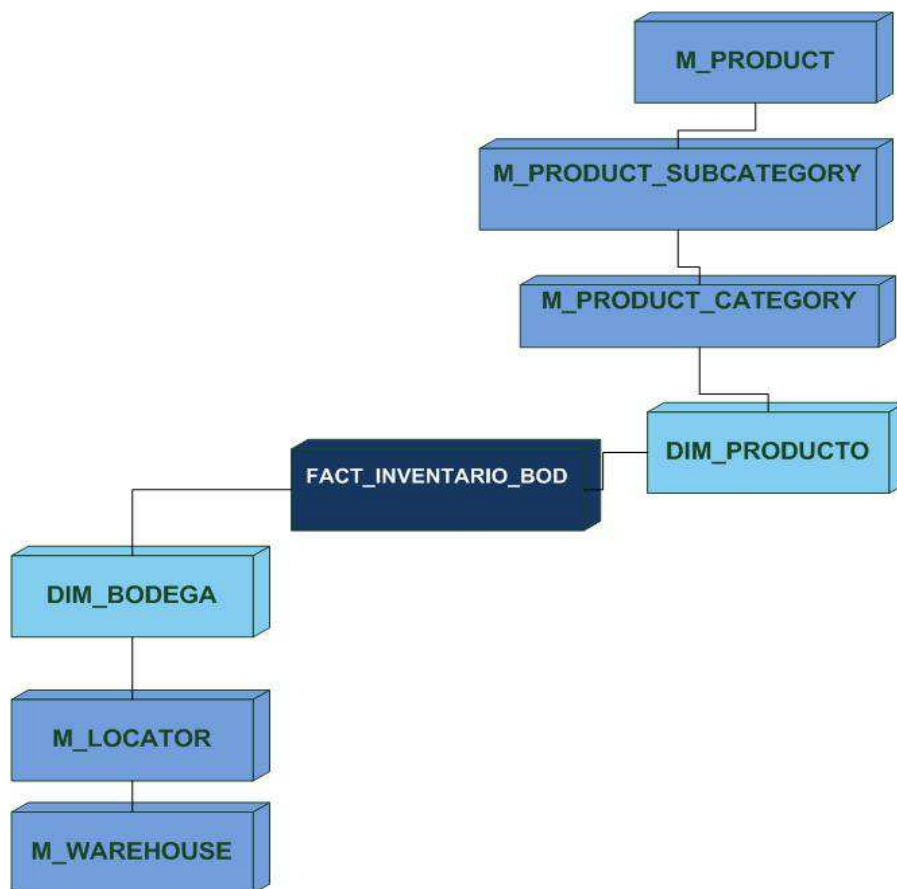


**Figura 3.16** Modelo de base de datos Data Mart compras por factura.



### 3.6.2.16 Diagrama general origen de datos Data Mart inventario por bodega

El diagrama muestra las dimensiones y la relación que tienen con las tablas de la base de datos origen. Para comprender de donde provienen los datos que se utilizan para poblar el Data Mart de inventario por bodega.



**Figura 3.17 Origen de datos general Data Mart inventario por bodega**

### 3.6.2.17 Mapeo de datos compras por factura

En la siguiente tabla se describe el origen de datos de la base de datos de la empresa hacia Las dimensiones y tablas de hechos del Data Warehouse.

**Tabla 3.48: Mapeo de datos compras por factura**

		Cargados	
Fuente de datos original (BD)	Descripción	Dimensión (DIM)	Tabla de hechos(FACT)
<b>M_PRODUCT</b>	Tabla que contiene los datos de productos.	DIM_PRODUCTO	FACT_INVENTARIO_BOD
<b>M_PRODUCT_SUBCATEGORY</b>	Tabla que contiene las subcategorías del producto.	DIM_PRODUCTO	FACT_INVENTARIO_BOD
<b>M_PRODUCT_CATEGORY</b>	Tabla que contiene las categorías de los productos.	DIM_PRODUCTO	FACT_INVENTARIO_BOD
<b>M_LOCATOR</b>	Tabla que contiene la ubicación del producto.	DIM_BODEGA	FACT_INVENTARIO_BOD
<b>M_WAREHOUSE</b>	Tabla que contiene el contenido de las bodegas.	DIM_BODEGA	FACT_INVENTARIO_BOD

### 3.6.2.18 Modelo dimensional Data Mart inventario por bodega

Con el previo análisis de los requerimientos se diseñó el modelo lógico para el Data Mart de inventario por bodega, la figura muestra los detalles en cuanto a relaciones, nombres de campos y tipos de datos.

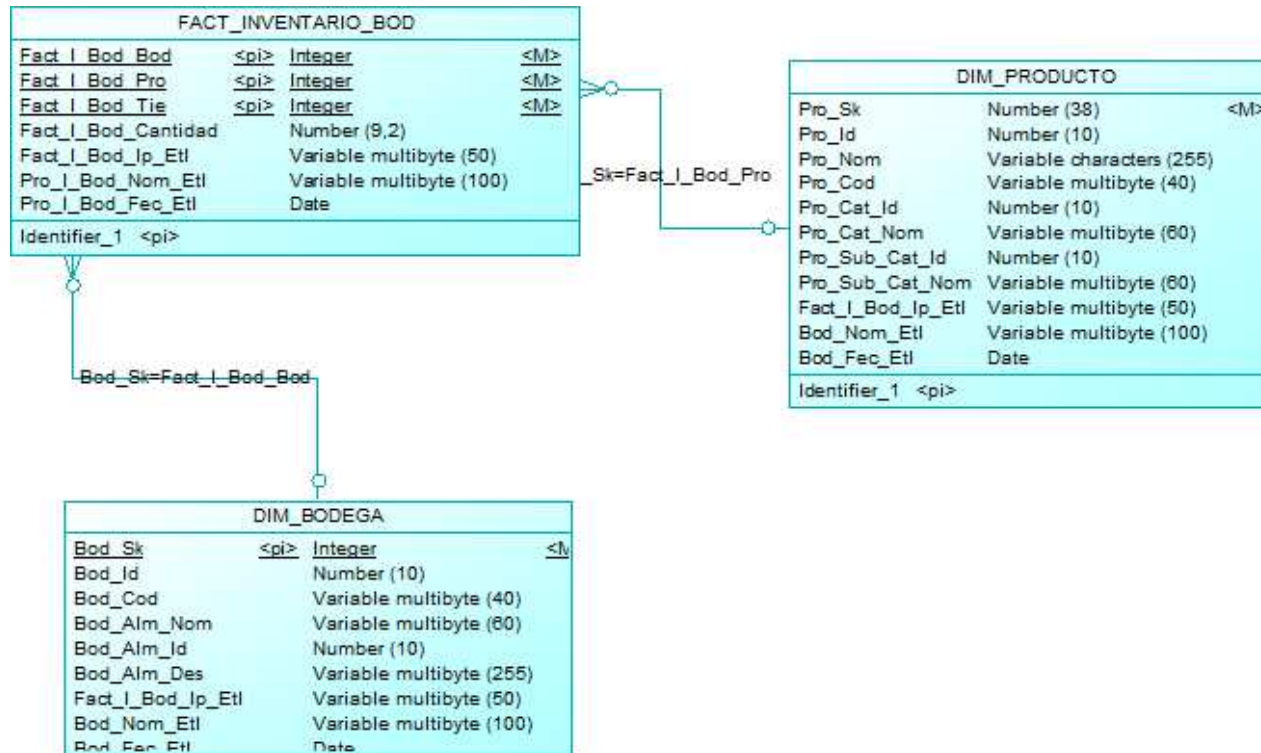


Figura 3.18 Modelo dimensional Data Mart inventario por bodega

### 3.6.2.19 Mapeo y detalle de datos de dimensiones y hechos Data Mart inventario por bodega.

Las siguientes tablas muestran el destino y origen de los datos de dimensiones y tablas de hechos para el Data Mart de inventario por bodega, se detalla los nombres de los campos de destino y origen los tipos de datos y sus esquemas respectivos.

**Tabla 3.49 Mapeo y detalle dimensión cliente**

Tabla	DIM_BODEGA							
Tipo de Tabla	Dimensional							
Descripción	Datamart de Inventario							
Esquema	DW							
Destino					Origen			
Columna	Descripción	Tipo de Dato	Tamaño	Clave	Esquema	Tabla	Campo	Tipo de Dato
Bod_Sk	Clave Subrogada de la dimensión bodega	Integer		Pk	Obtest2	dim_bodega	pro_sk	Integer
Bod_Id	Id de la bodega	Number	10		Obtest2	m_locator	m_locator_id	Number (10)
Bod_Cod	Código de la bodega en el sistema.	Nvarchar2	40		Obtest2	m_warehouse	value	Nvarchar2(40)
Bod_Alm_Id	Id del almacén.	Number	10		Obtest2	m_warehouse	m_warehouse_id	Number (10)
Bod_Alm_Nom	Nombre del almacén.	Nvarchar2	60		Obtest2	m_warehouse	Name	Nvarchar2(60)
Bod_Alm_Des	Descripción del almacén.	Nvarchar2	255		Obtest2	m_warehouse	description	Nvarchar2(255)
Bod_Ip_Etl	Ip de la maquina donde se realizó el etl	Nvarchar2	40		Sistema	Sistema	Sistema	NVarchar2(50)
Bod_Nom_Etl	Nombre del ETL	Nvarcahr2	100		Sistema	Sistema	Sistema	NVarchar2(100)
Bod_Fec_Etl	Fecha de la ejecución o carga del ETL	Date			Sistema	Sistema	Sistema	Date
Pro_Nom_Etl	Nombre del ETL	Nvarcahr2		100	Y	Sistema	Sistema	Sistema

**Tabla 3.50 Mapeo y detalle dimensión producto**

Tabla	DIM_PRODUCTO						
Tipo de Tabla	Dimensional						
Descripción							
fEsquema	DW						
Origen							
Columna	Descripción	Tipo de Dato	Tamaño	Esquema	Tabla	Campo	Tipo de Dato
Pro_Sk	Clave Subrogada de la dimensión producto	Integer		DW	dim_producto	pro_sk	Integer
Pro_Id	Id del producto en el sistema	Number	10	obtest	m_product	m_product_id	Number(10)
Pro_Nom	Nombre del producto	Nvarchar2	225	obtest	m_product	name	NVarchar2(225)
Pro_Cod	Código del producto para la identificación	Nvarchar2	40	obtest	m_product	value	NVarchar2(40)
Pro_Cat_Nom	Nombre de la categoría del producto	Nvarchar2	60	obtest	m_product_category	name	NVarchar2(60)
Pro_Cat_Id	Id de la categoría del producto	Number	10	obtest	m_product_category	m_product_category_id	Number(10)
Pro_Sub_Cat_No m	Nombre de la subcategoría del producto	Nvarchar2	60	obtest	um_product_subcategor y	name	NVarchar2(60)
Pro_Sub_Cat_Id	Id de la subcategoría del producto	Number	10	obtest	um_product_subcategor y	um_product_subcategory_i d	Number(10)
Pro_Ip_Etl	Ip de la maquina	Nvarchar2	40	Sistema	Sistema	Sistema	NVarchar2(50)
Pro_Nom_Etl	Nombre del ETL	Nvarcahr2	100	Sistema	Sistema	Sistema	NVarchar2(100)
Pro_Fec_Etl	Fecha de la ejecución	Date		Sistema	Sistema	Sistema	Date

**Tabla 3.51 Mapeo y detalle fact de inventario por bodega**

Tabla		FACT_INVENTARIO_BOI								
Tipo de Tabla		Dimensional								
Descripción		Datamart de Inventario								
Esquema		DW								
Destino							Origen			
Columna	Descripción	Tipo de Dato	Tamaño	Clave	FK para	Null	Esquema	Tabla	Campo	Tipo de Dato
Fact_I_Bod_Pro	Clave Subrogada de la dimensión producto	Integer		Pk		N	obtest	Dim_Producto	Pro_Sk	Integer
Fact_I_Bod_Bod	Clave Subrogada de la dimensión bodega	Integer		Pk		N	obtest	Dim_Bodega	Bod_Sk	Integer
Fact_I_Bod_Cantidad	Medida de cantidad de producto en bodega	Number					obtest	m_inoutline	movementqty	Number
Fact_I_Bod_Ip_Etl	Ip de la maquina donde se realizo el ETL	Nvarchar2	40				Sistema	Sistema	Sistema	NVarchar2(50)
Fact_I_Bod_Nom_Etl	Nombre del ETL	Nvarcahr2	100				Sistema	Sistema	Sistema	NVarchar2(100)
Fact_I_Bod_fec_Etl	Fecha de la ejecución o carga del ETL	Date					Sistema	Sistema	Sistema	Date

### 3.6.2.20 Modelo de base de datos Data Mart inventario por bodega

Finalmente el diagrama de base de datos muestra estructura del Data Mart de inventario por bodega. El script de generación esta en el **Anexo C**.

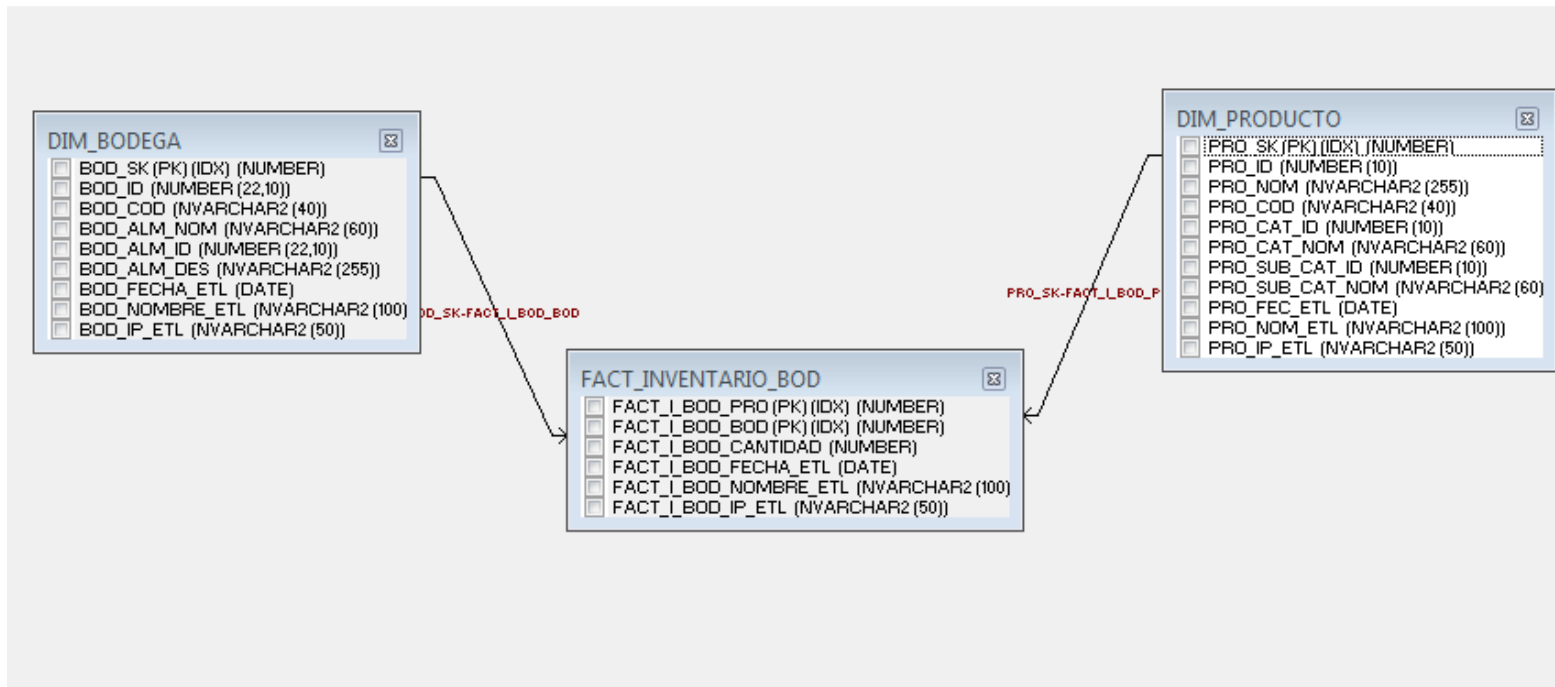


Figura 3.19 Modelo de base de datos Data Mart compras por factura

### 3.6.3 Diseño Físico

Finalmente el diagrama del Data Warehouse tiene la siguiente estructura, conformada por la unión de los Data Mart de las áreas de ventas, compras e inventarios.

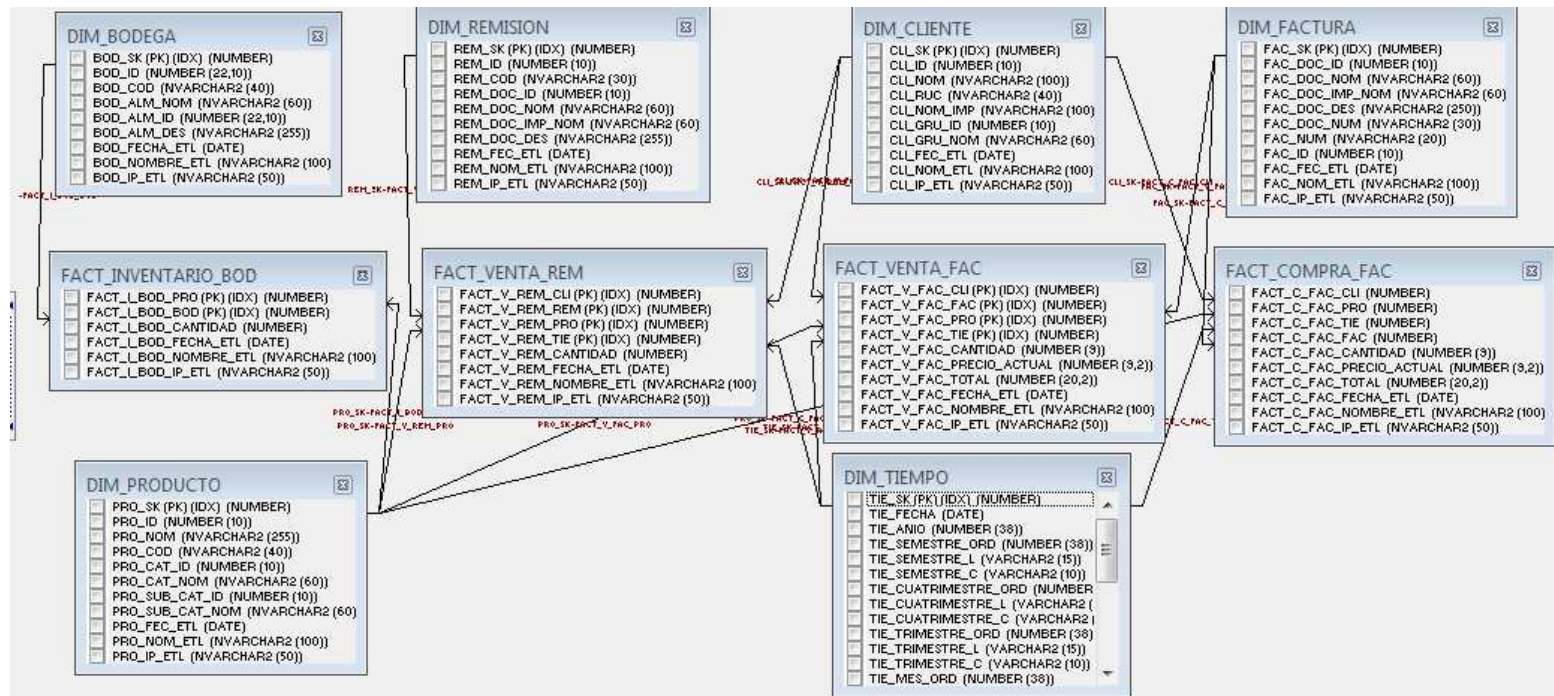


Figura 3.20 Modelo físico general del Data Warehouse



### 3.6.4 Diseño y desarrollo de la presentación de datos

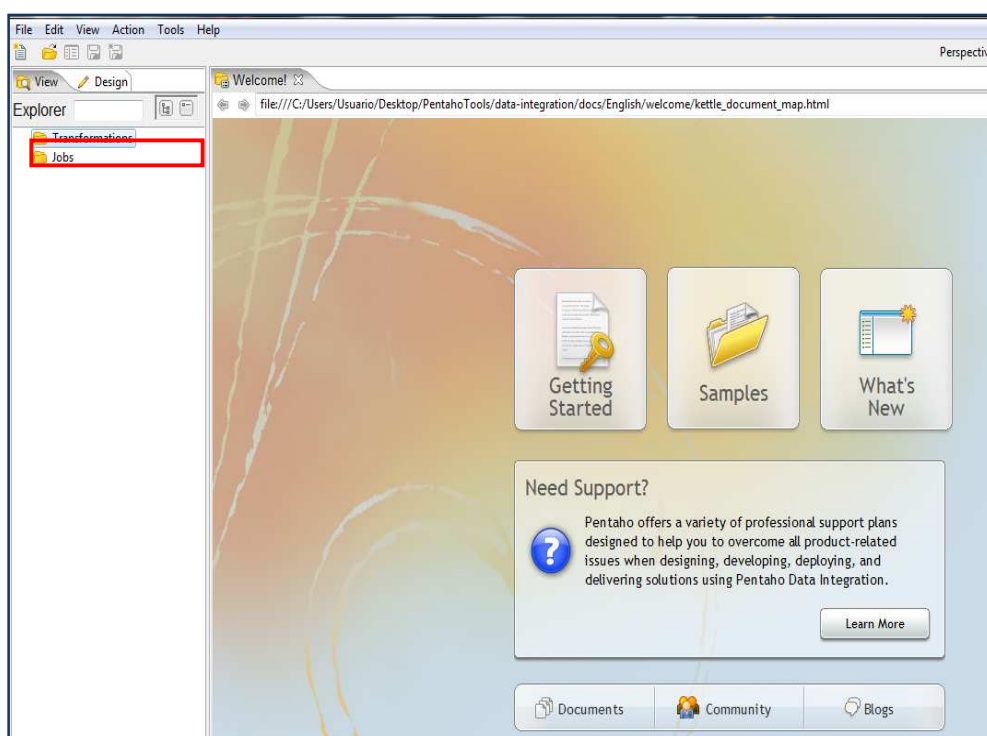
#### 3.6.4.1 Proceso de Extracción Transformación y Carga

El proceso ETL se lo realizó con la herramienta Spoon de Pentaho, en esta sección se describe el procedimiento para el proceso completo de ETL

#### Descripción de la Interfaz de Usuario

Para empezar hacer los procesos ETL se necesita crear una transformación o un trabajo.

Crear una transformación dando doble clic sobre la carpeta transformations.



**Figura 3.21** Pantalla menú principal spoon

El ambiente gráfico permite crear transformaciones con una variedad de opciones y herramientas que se encuentran en la pestaña Design.



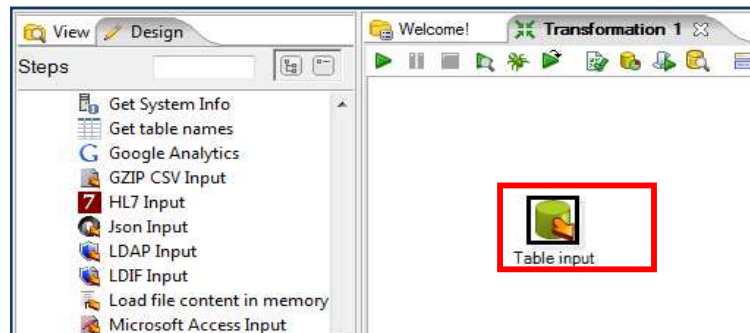
**Figura 3.22 Ambiente gráfico Spoon**

### **Diseño del ETL para nuestro caso practico**

Desarrollo del ETL para la carga de la dimensión cliente. El proceso ETL que se presenta en este ejemplo es igual para todas las dimensiones.

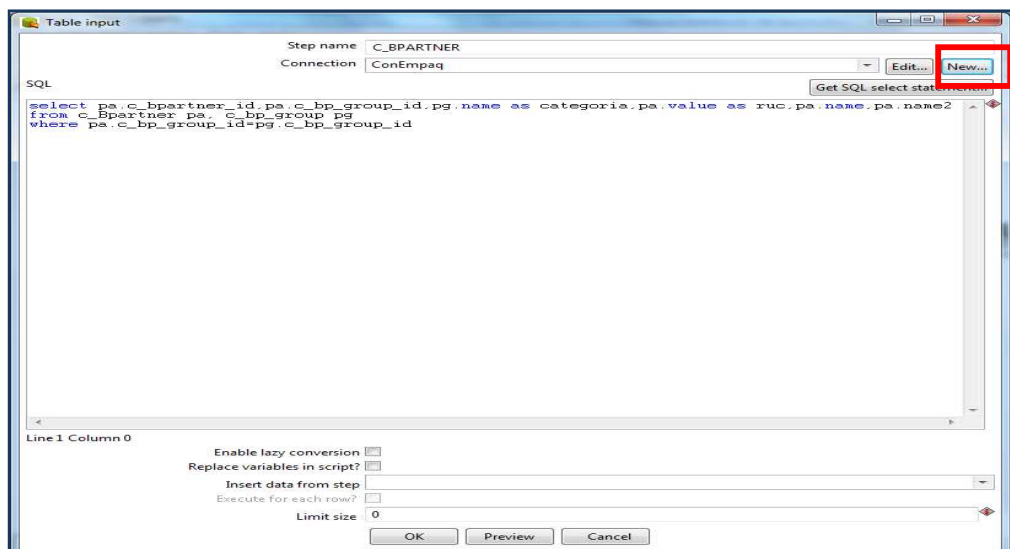
#### **Table Input**

Para empezar a diseñar el ETL se debe insertar un Table Input que se encuentra en las herramientas de Design en la carpeta input. Este ejecuta la sentencia SQL, la cual trae los datos de la base origen los cuales serán usados para la carga del Data Mart.



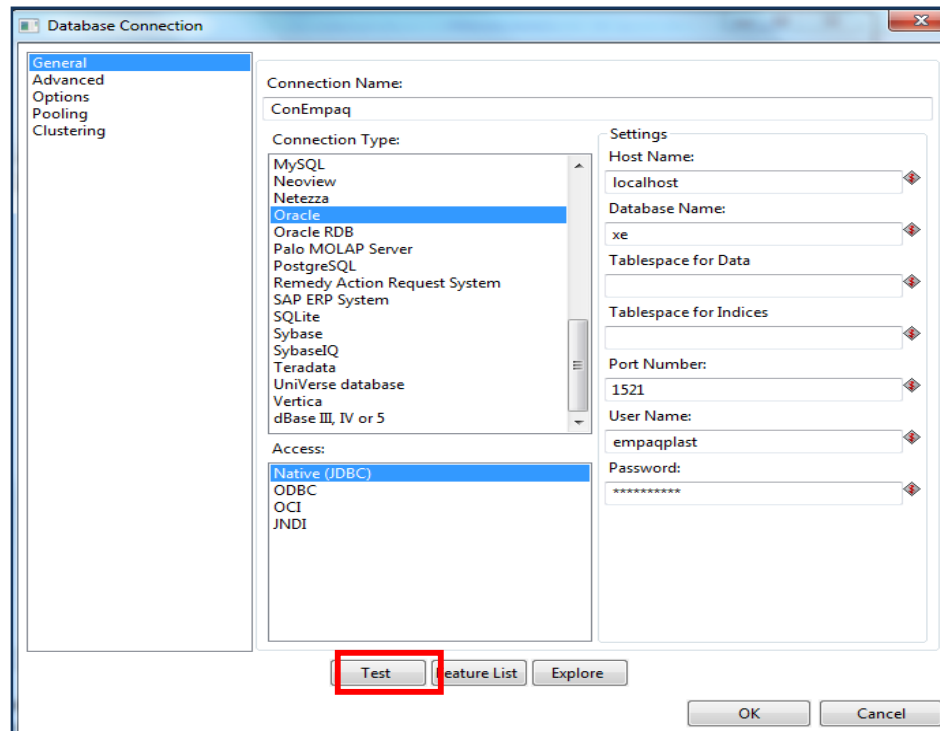
**Figura 3.23 Insertar un table input**

Dar doble clic sobre Table Input para asignar un nombre y la conexión a la base de datos de donde se extrae los datos de origen.



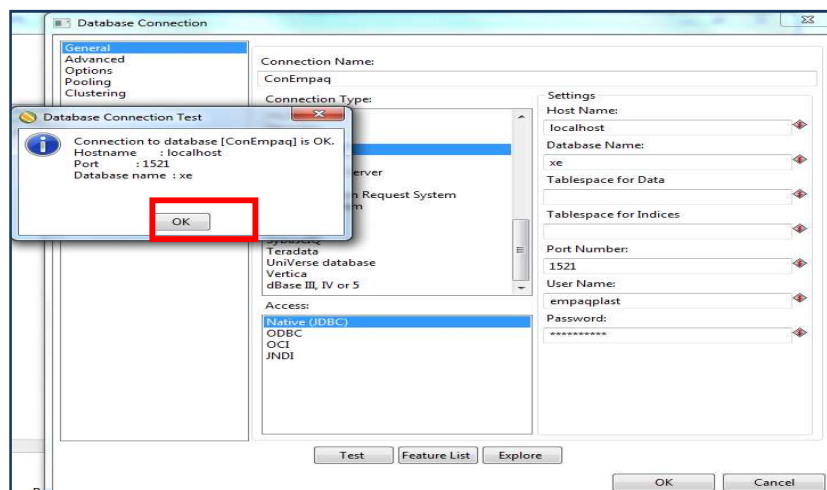
**Figura 3.24 Ventana de table input**

Después de dar clic en nuevo saldrá esta pantalla, escoger la base de datos Oracle. Insertar los parámetros obligatorios. Para terminar damos clic en test.



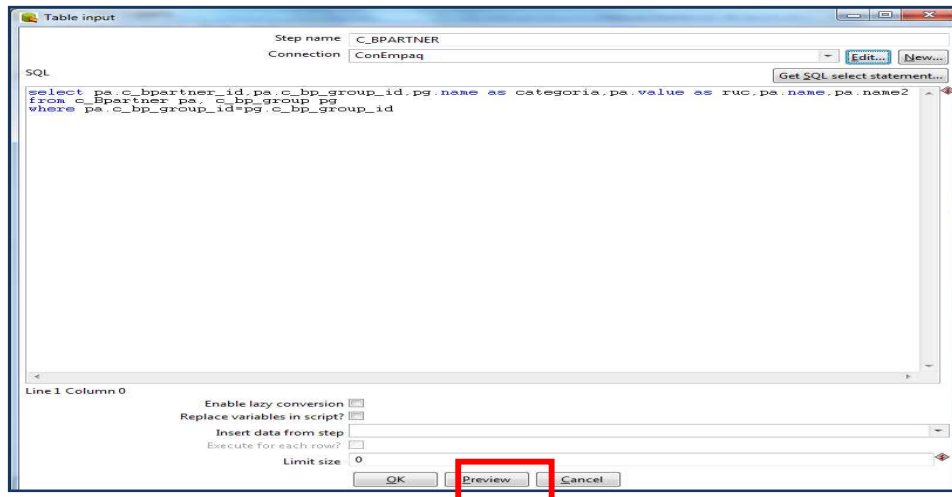
**Figura 3.25 Ventana de conexión a bases de datos**

Después de dar clic en test, verificar si la conexión es correcta, dando clic en el botón OK.



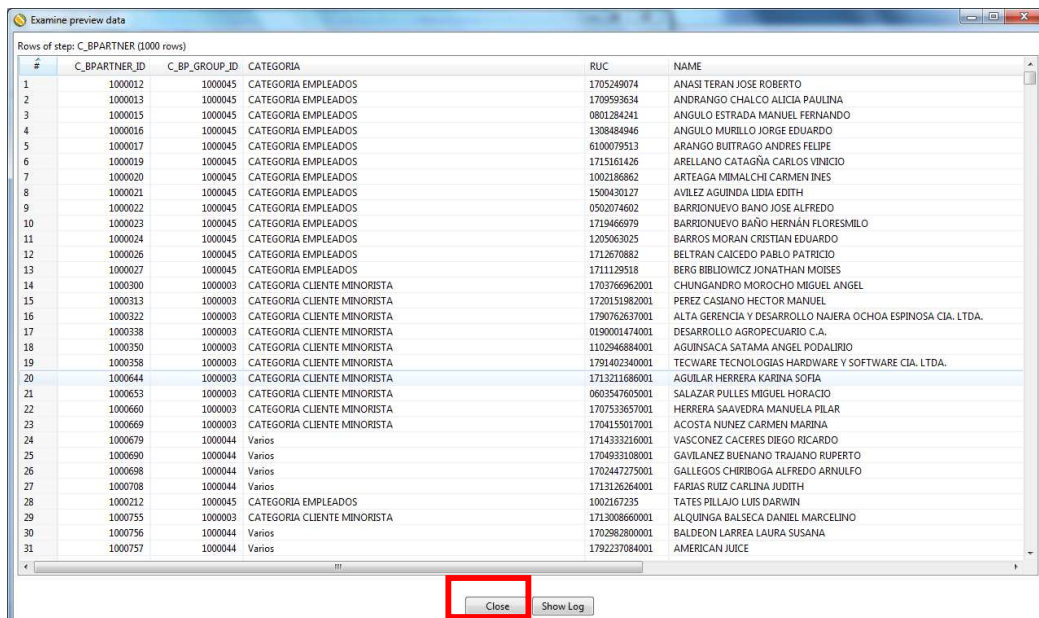
**Figura 3.26 Ventana de conexión exitosa table input**

En la opción de Preview podemos verificar si la sentencia SQL trae los datos correctos. Todas las sentencias SQL, están adjuntas en **Anexo B**.



**Figura 3.27 Ventana para sentencias SQL**

Finalmente si los datos que se van a cargar son correctos dar clic en el botón close.



**Figura 3.28 Ventana de vista previa de datos**

## Add Sequence

Insertar un Add Sequence, esta herramienta permite crear las secuencias en la base de datos de nuestros Data Mart. Estas secuencias se denominan SK.

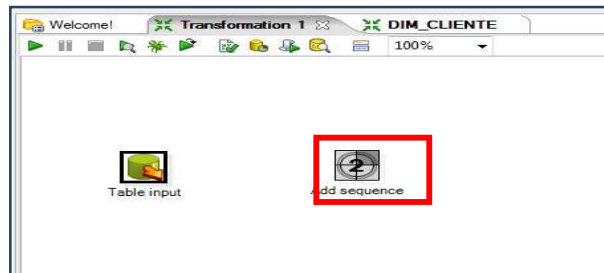


Figura 3.29 Agregar secuencias

Asignar un nombre y valor a la secuencia, el nombre del valor es importante porque al final se unirá con la secuencia creada en la base de datos del Data Warehouse.

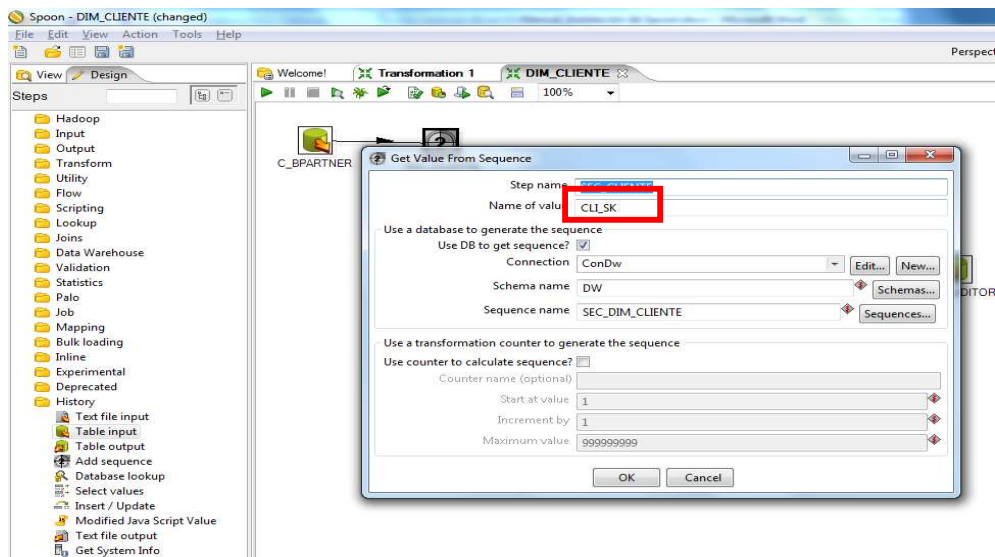


Figura 3.30 Agregar propiedades a una secuencia

Realizar la conexión a la base de datos que almacena el Data Warehouse, en este caso se llama DW. Ingresar los datos obligatorios para que la conexión sea correcta.

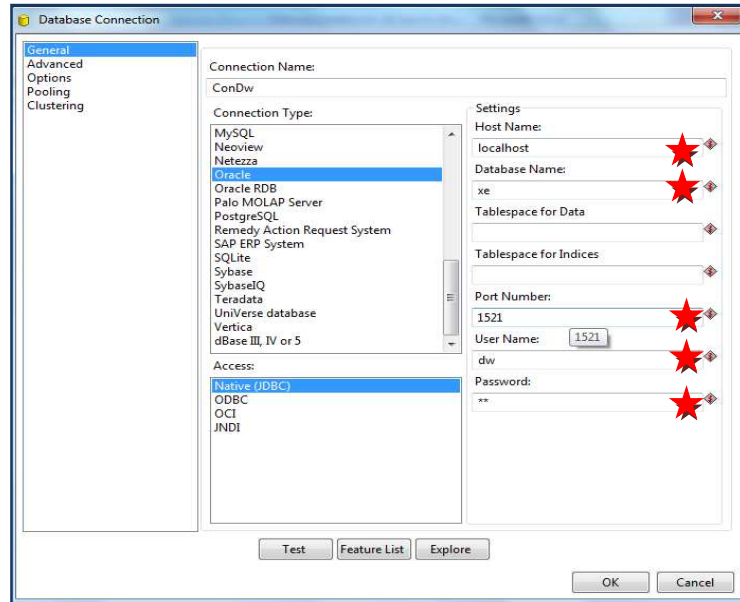


Figura 3.31 Ventana conexión a bases de datos

Dar clic en botón test para verificar si la conexión es correcta, si es correcta dar clic en el botón OK.

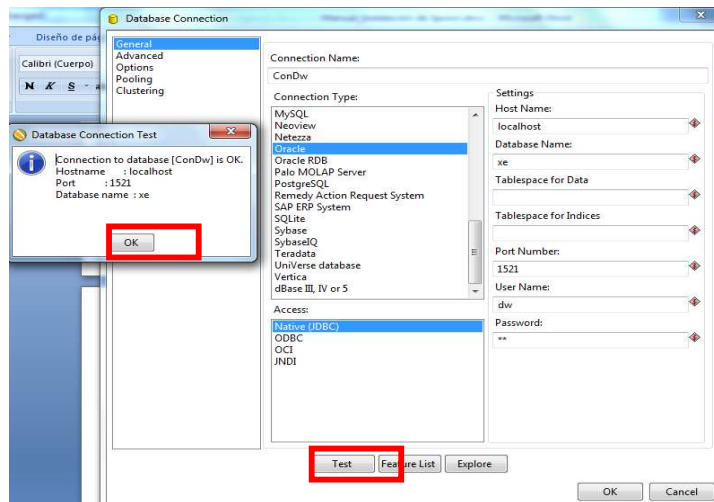
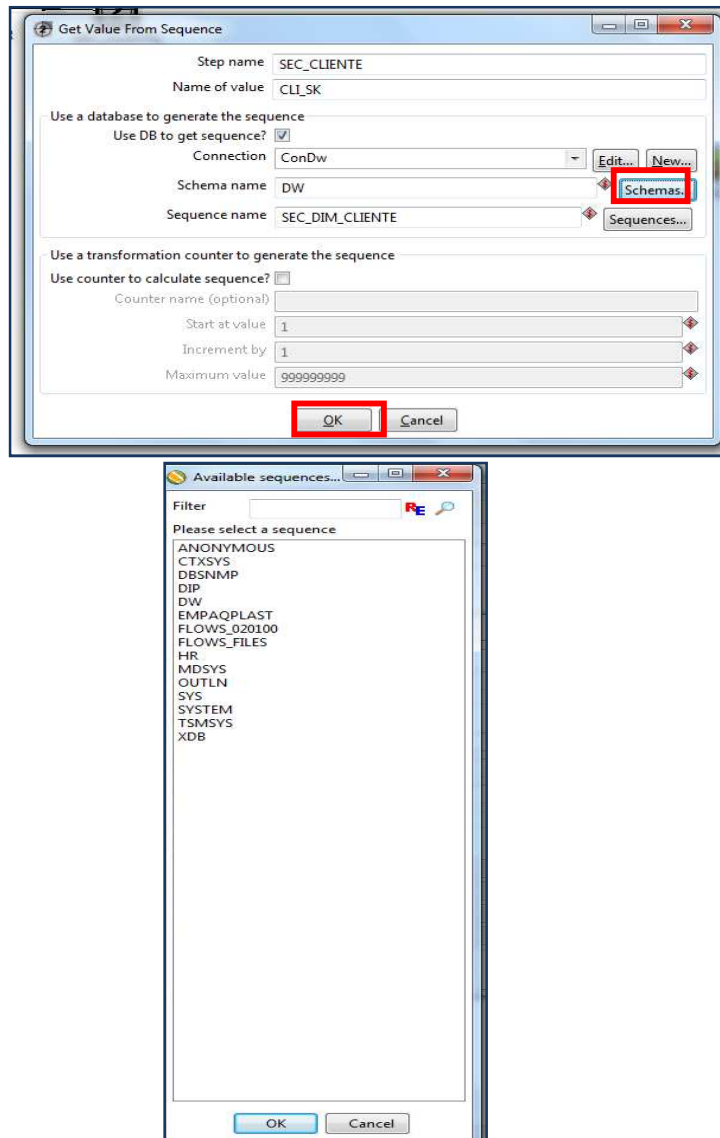


Figura 3.32 Ventana conexión exitosa secuencia

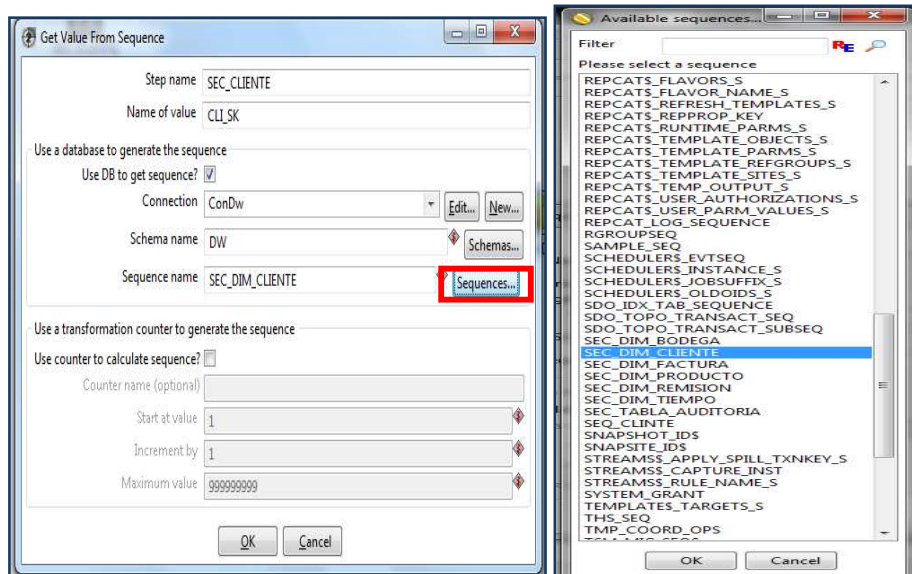
Después se selecciona el esquema o la base de datos donde están nuestras secuencias.



**Figura 3.33 configuración de esquemas para una secuencia**

Dar clic en el botón sequence y añadir la secuencia del “cliente” SEC\_DIM\_CLIENTE.

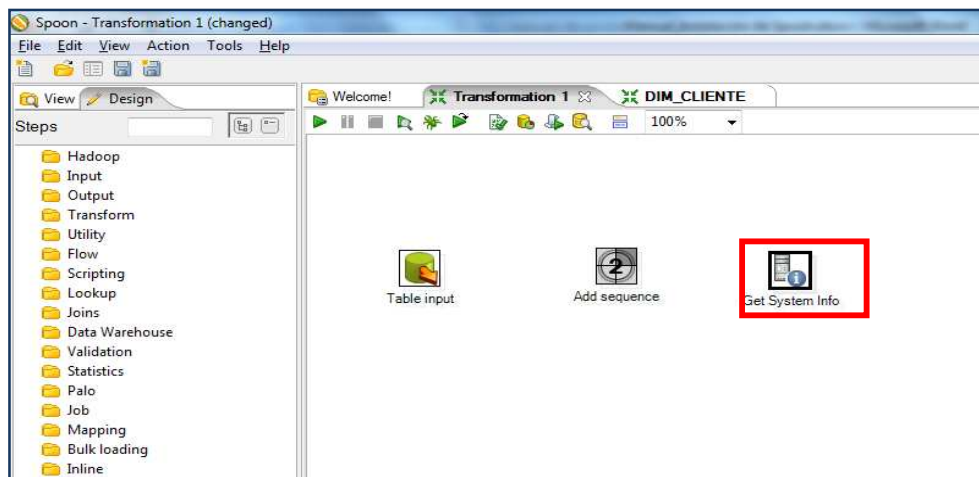




**Figura 3.34 Selección secuencia**

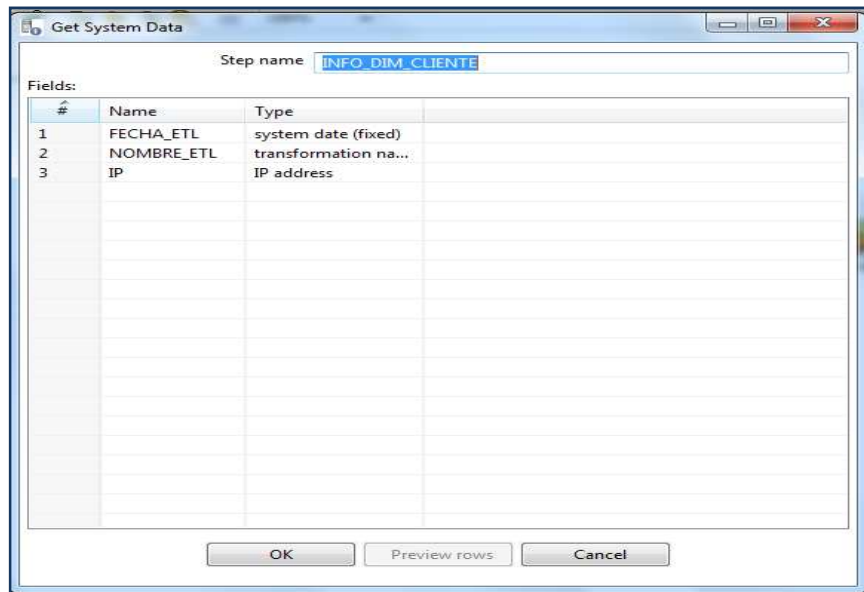
### Get System Info

Con esta herramienta se puede obtener algunos datos del sistema que servirán para la tabla de auditoría.



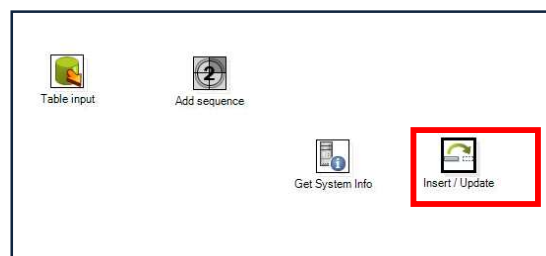
**Figura 3.35 Insertar un Get System Info**

Para este caso práctico se toma del sistema la fecha del ETL, Nombre del ETL, y la IP de la máquina.



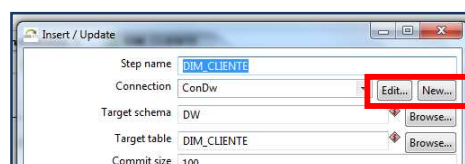
**Figura 3.36 Asignación de nombres en Get System Data**

Crear un Insert/Update, para controlar la inserción y actualización en la base de datos del Data Warehouse DW.



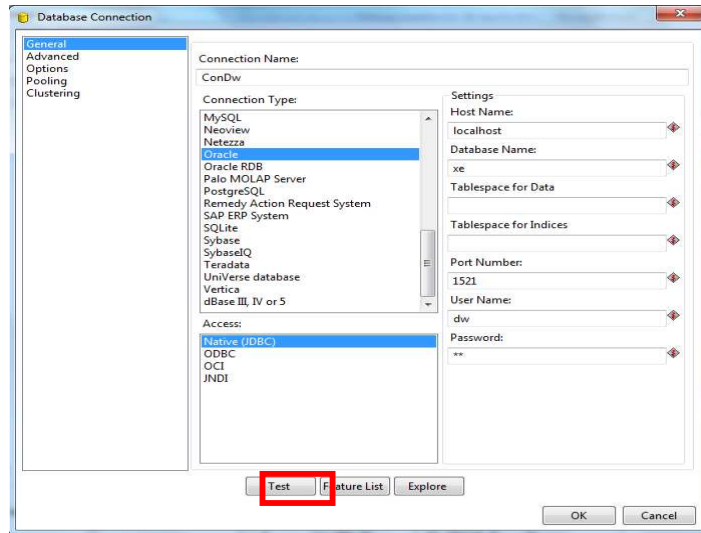
**Figura 3.37 Agregar un Insert / Update**

Asignar un nombre (DIM\_CLIENTE), realizar la conexión a la base de datos del Data Warehouse.



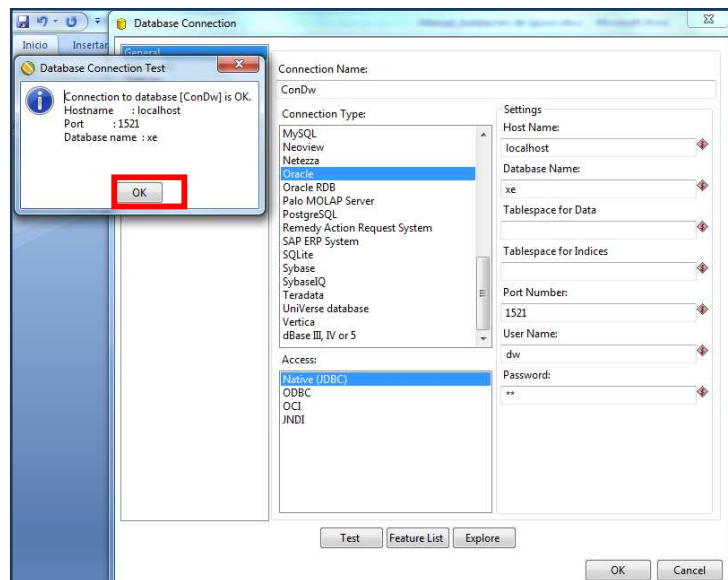
**Figura 3.38 Configuración de la conexión de un Insert/Update**

Configurar la conexión con los parámetros obligatorios.



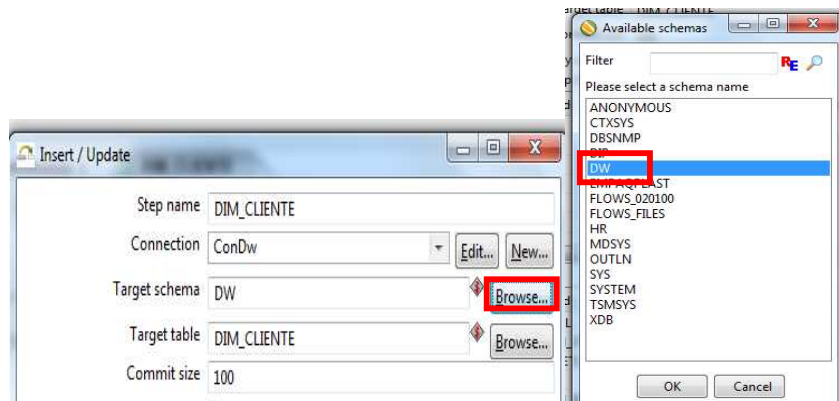
**Figura 3.39 Configuración conexiones a bases de datos Insert/Update**

Después de dar clic en el botón test verificar si la conexión es correcta, dando clic en el botón ok.



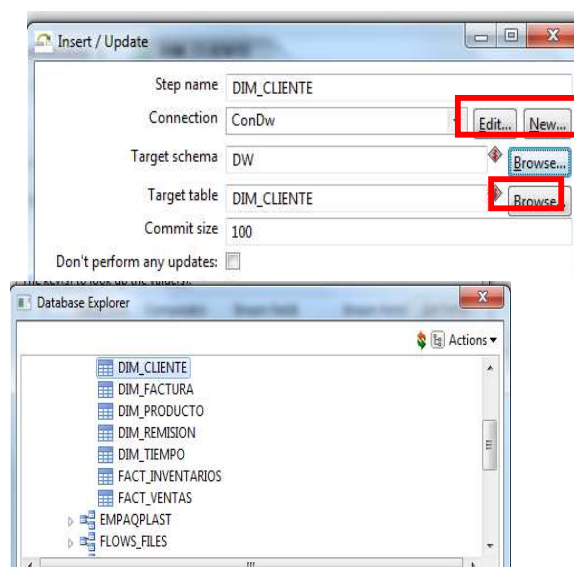
**Figura 3.40 Conexión exitosa Insert/Update**

En este paso seleccionar el esquema de base de datos del Data Warehouse, como se explica en la siguiente figura



**Figura 3.41 Selección de Esquema para un Insert/Update**

Después de seleccionar el esquema del Data Warehouse. Escoger la tabla de dimensión del cliente.



**Figura 3.42 Selección de tabla para un Insert/Update**

Hacer la relación entre el Id de la base de datos origen y el campo que contiene ese Id en la base del Data Warehouse.

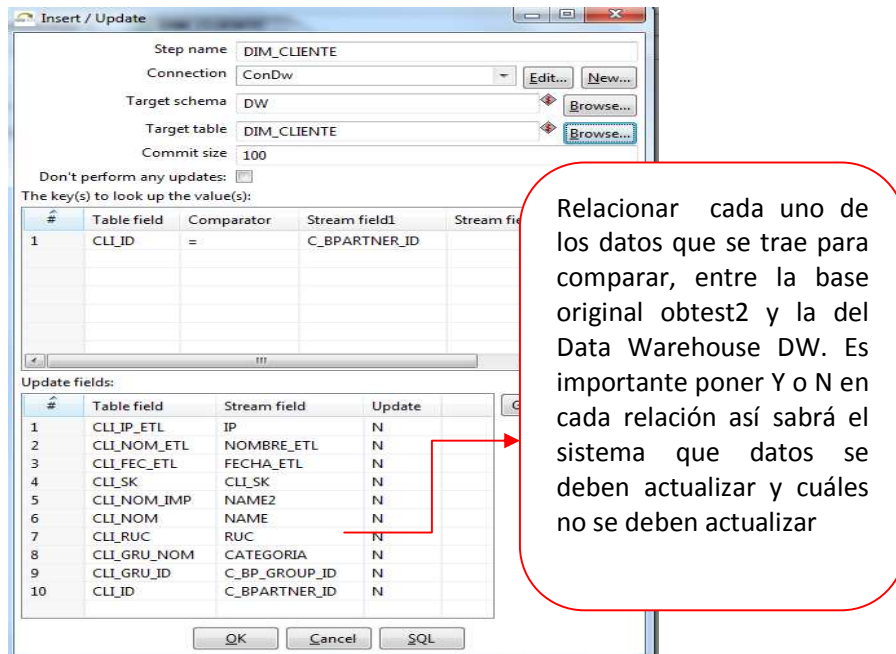
The key(s) to look up the value(s):			
#	Table field	Comparator	Stream field1
1	CLI_ID	=	C_BPARTNER_ID

**Figura 3.43 Join de las claves primarias**

A continuación se presenta la siguiente tabla que muestra un ejemplo de cómo se realiza las relaciones entre claves primarias.

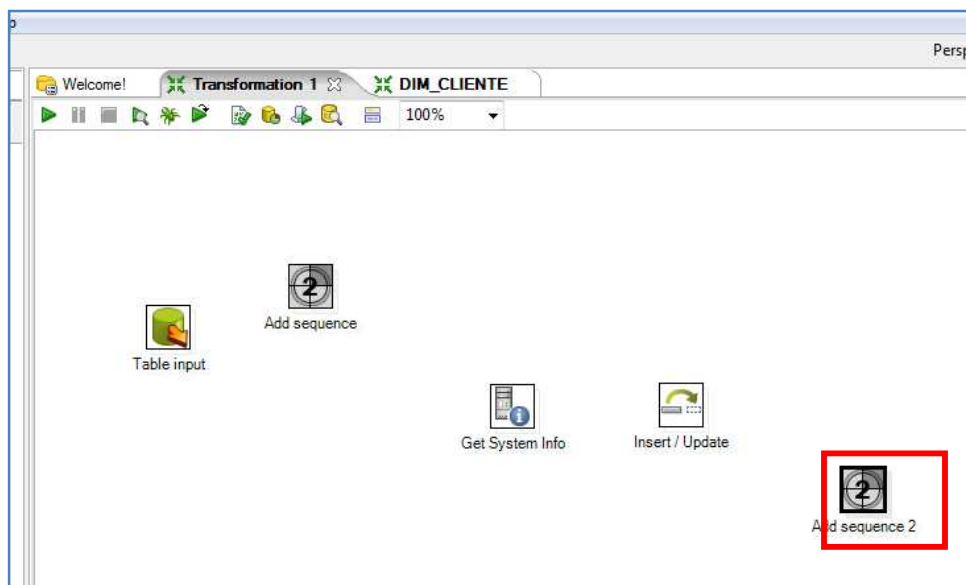
**Tabla 3.52: Join de las claves primarias**

Campo de la “dim_cliente”, que contiene el c_bpartner_id”.	Comparación	Clave primaria de la tabla “c_bpartner”, en la base de datos original obetest2.	Descripción
CLI_ID	=	C_BPARTNER_ID	unir las claves primarias del cliente con la que aremos la comparacion para la carga de los datos.



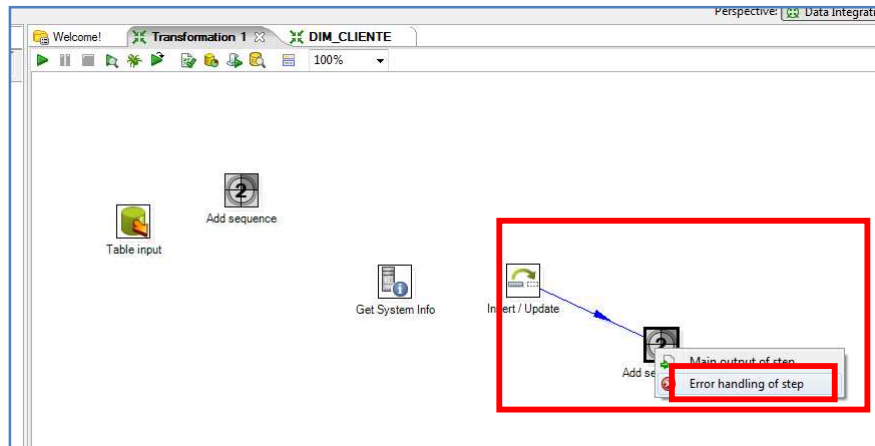
**Figura 3.44 Selección de tabla para un Insert/Update**

Crear una secuencia la cual permite controlar los errores al momento de ejecutar el Insert/Update, estos errores se almacenarán en la tabla de auditoría llamada HIS\_AUDITORIA.



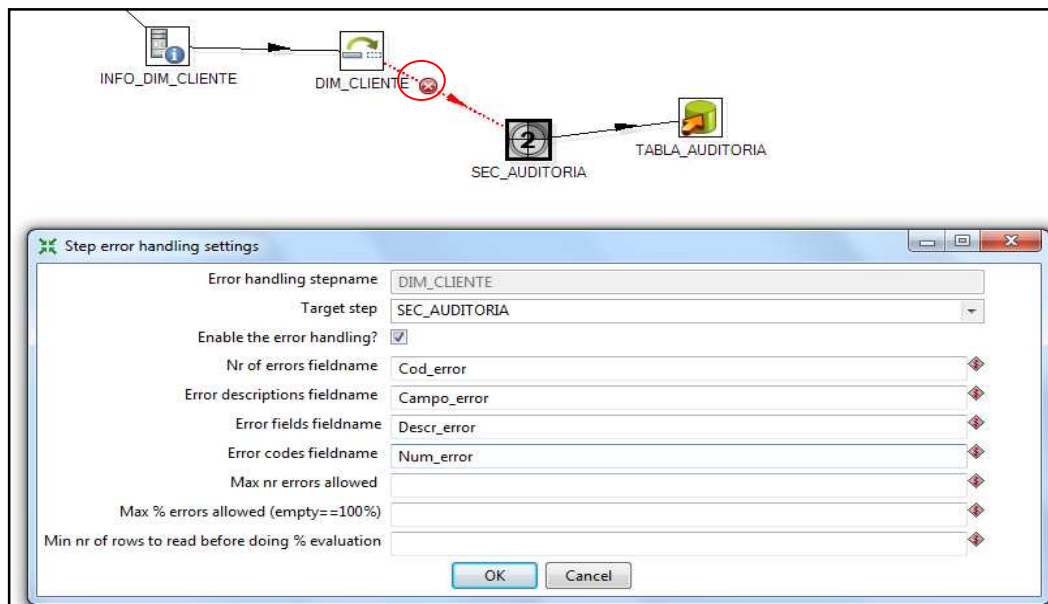
**Figura3.45 Secuencia para Errores**

Arrastrar la flecha desde el Insert/Update hacia la secuencia y se escogemos la opción de Error handling of step como se muestra en la siguiente figura.



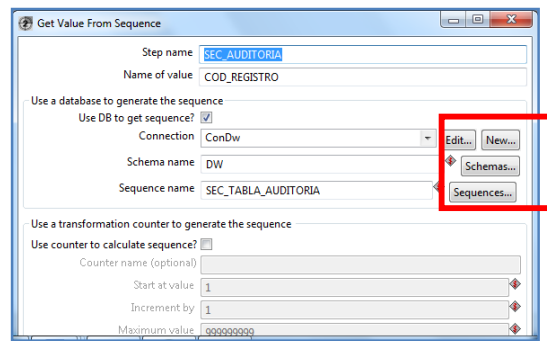
**Figura 3.46 Secuencia Para Errores.**

Ingresar los campos de error que se van a controlar, dando clic en la "X".



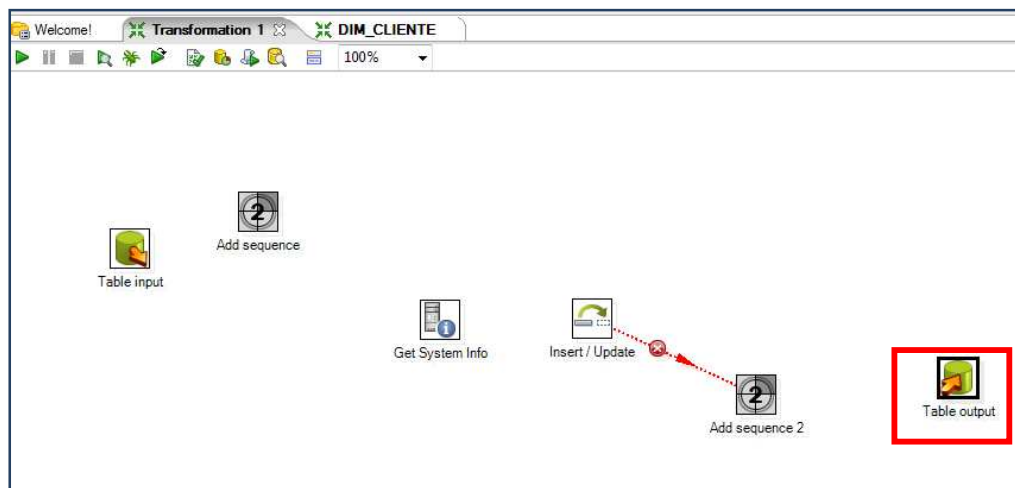
**Figura 3.47 Nombres campos de errores.**

Realizar la conexión de la secuencia SEC\_AUDITORIA, de la misma manera como se hizo la conexión para la anterior secuencia.



**Figura 3.48 Editar la conexión sec\_auditoria**

Para finalizar crear un table output, que se llamara HIS\_AUDITORIA, en esta tabla se almacenarán los errores y los campos de auditoría que se controlaron anteriormente.



**Figura 3.49 Selección de un Table Output.**

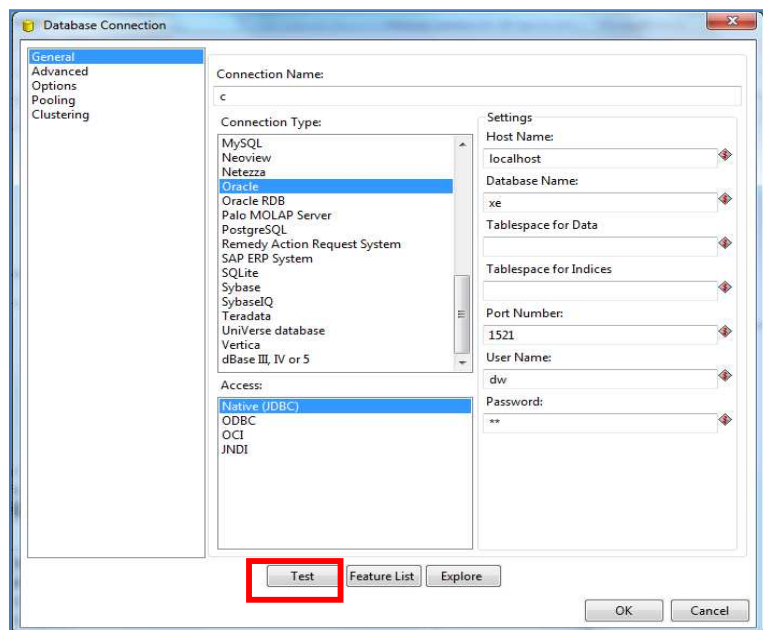


Editar nombre de la conexión para la tabla de auditoría.



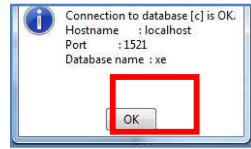
**Figura 3.50 Editar la conexión.**

Después de Configurar los campos obligatorios para la conexión dar clic en el botón test.



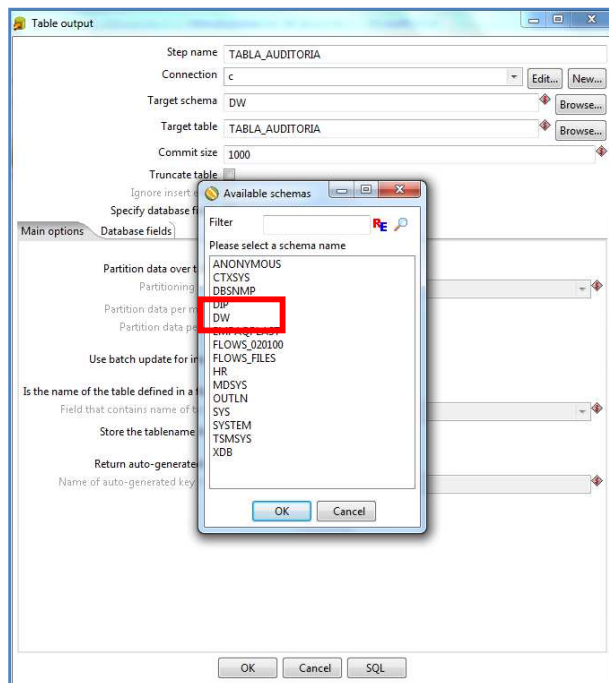
**Figura 3.51 Probar la conexión correcta**

Si la conexión es correcta dar clic en botón ok.



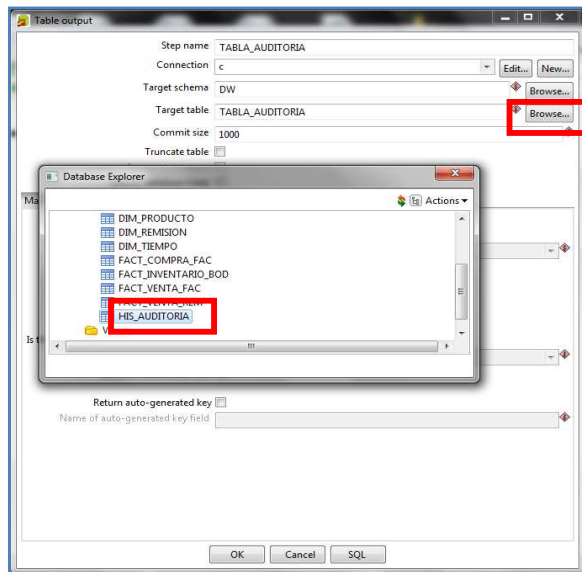
**Figura 3.52 Conexión correcta table output**

Seleccionar el esquema del Data Warehouse DW.



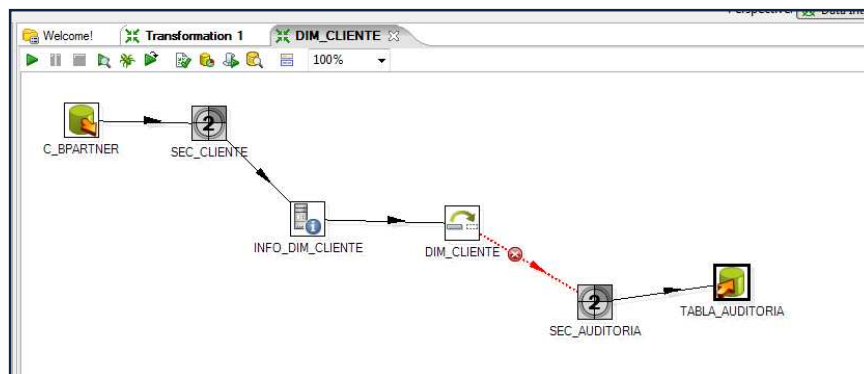
**Figura 3.53 Seleccionar el esquema de la base de datos.**

Dentro del esquema DW se encuentra la tabla HIS\_AUDITORIA. Seleccionar esta tabla.



**Figura 3.54 Selección de tabla de Auditoria.**

Finalmente todas las tablas de dimensiones se cargarán de la misma manera siguiendo los pasos previamente desarrollados obteniendo el siguiente diseño.



**Figura 3.55 Carga ETL para Dim\_Cliente**

### **Proceso ETL para la carga de una tabla de hechos.**

El proceso de ETL para este ejemplo práctico se lo realizó para la tabla de hechos FACT\_VENTAS\_FAC. Este proceso se lo realizará para la carga de todas las tablas de hechos de nuestro Data Warehouse.

## Table Input

Diseño del ETL, para la carga de las tablas de hechos. Insertar un Table Input.

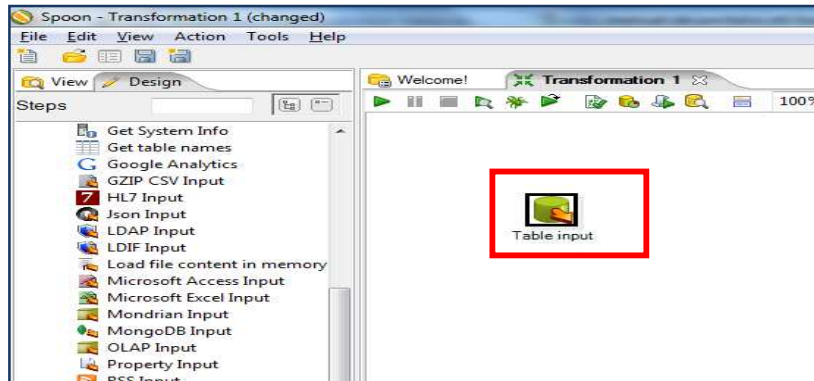


Figura 3.56 Selección de un table input

Ingresar al Table Input dando doble clic, asignar un nombre y la conexión a la base de datos origen.

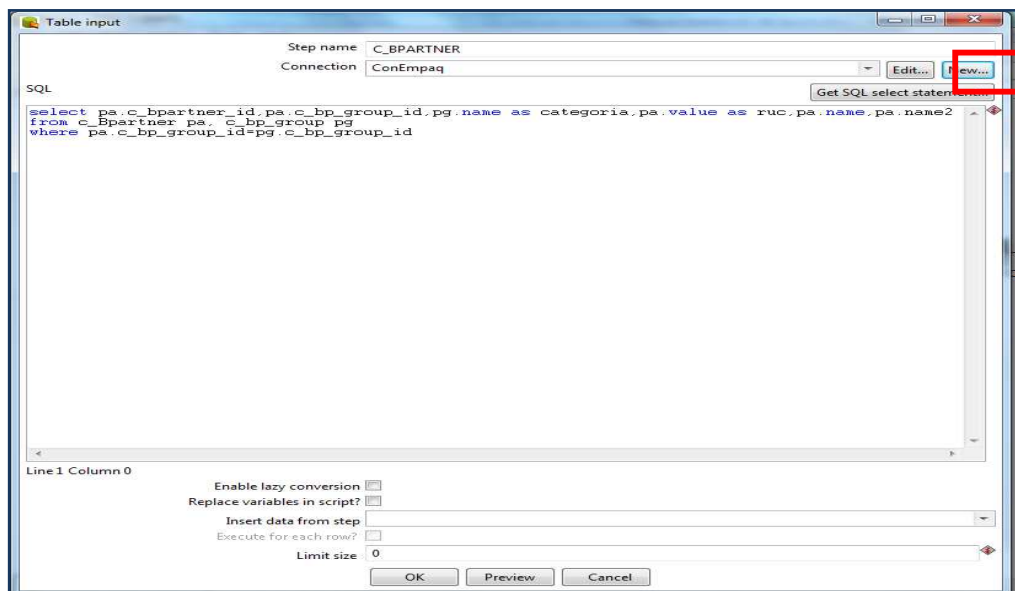
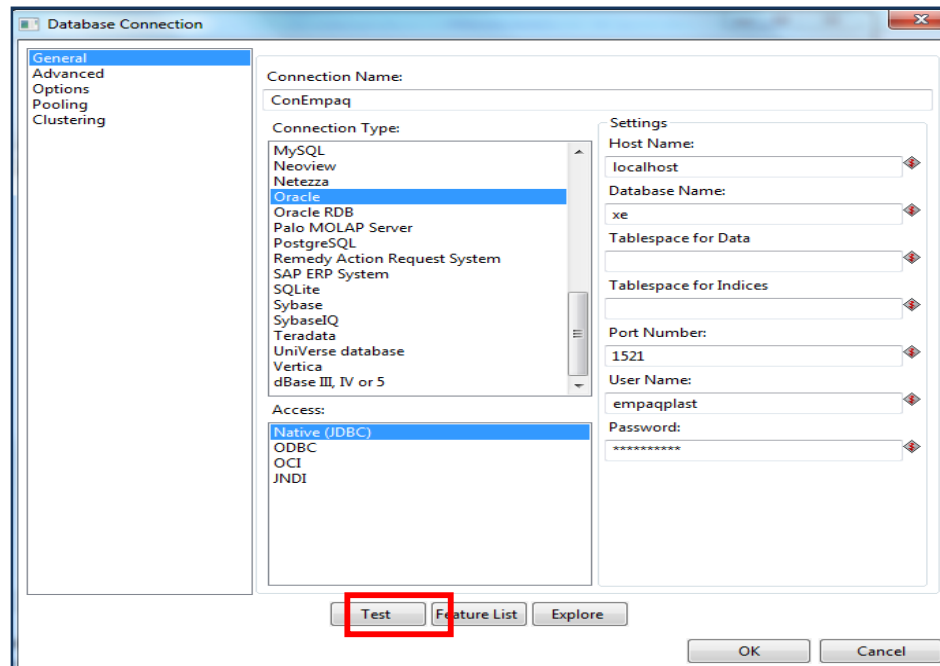
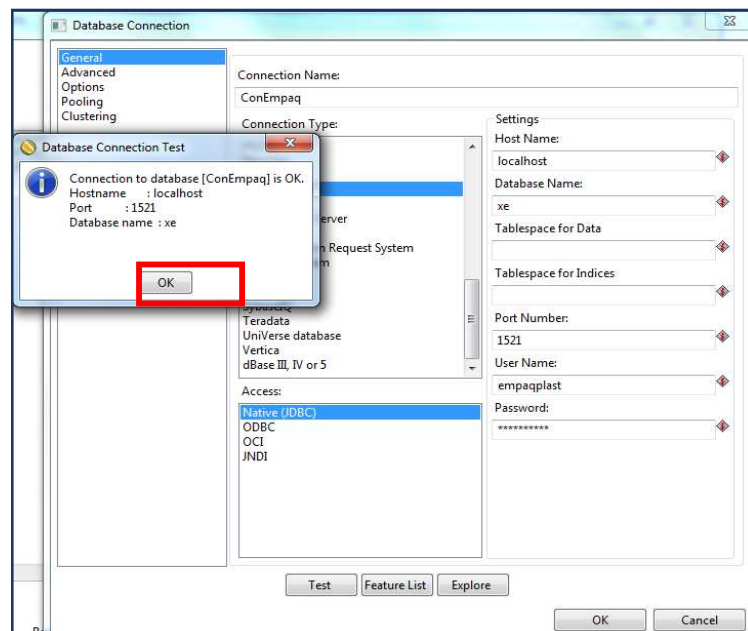


Figura3.57 Editar la conexión a la base de datos



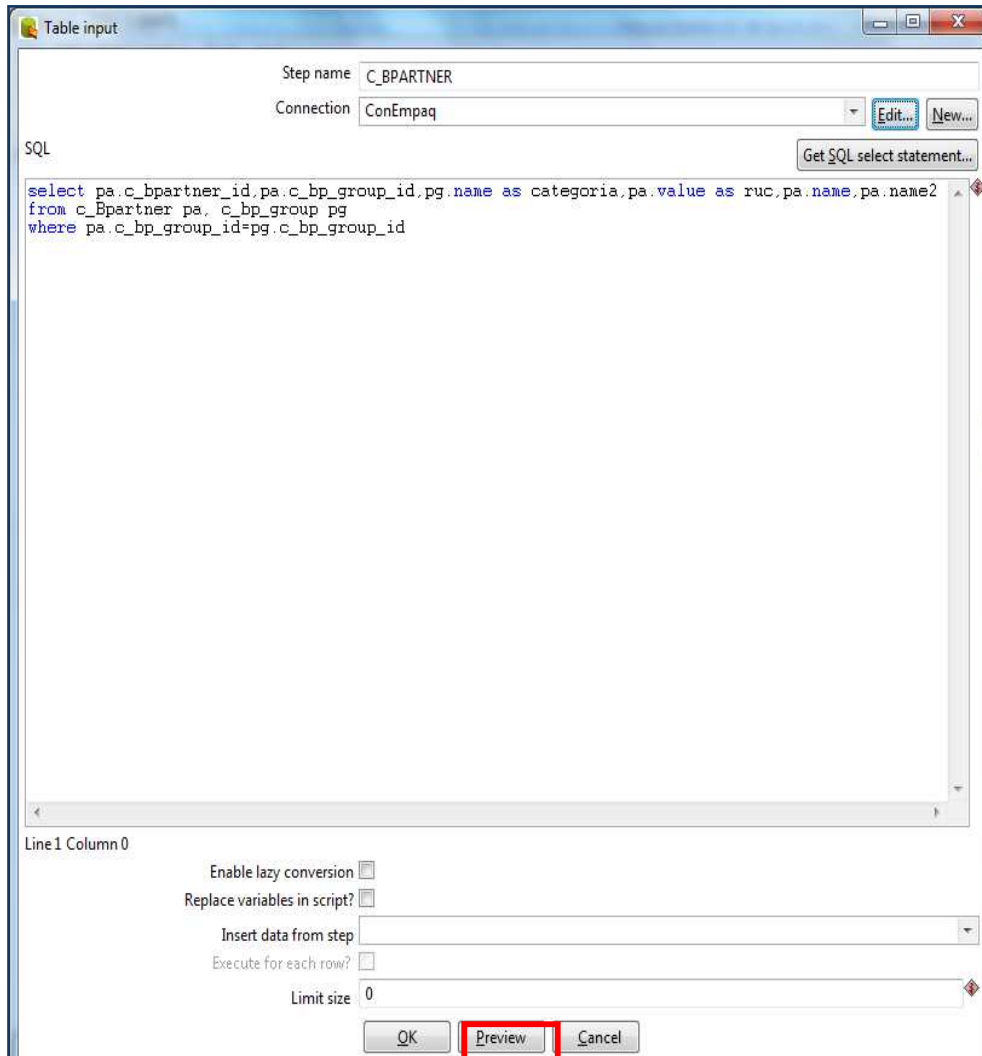
**Figura 3.58 Probar la conexión con un test**

Realizar el Test de conexión si es correcta dar en ok.



**Figura 3.59 Conexión correcta**

Preview para ejecutar el SQL que trae los datos y verificar si son correctos.



**Figura 3.60 Preview para ejecutar el select hacia la tabla c\_bpartner**

Estos son los datos extraídos mediante la sentencia SQL que se encuentra adjunta en el **Anexo B**

Rows of step: DBEEMPAQ (1000 rows)

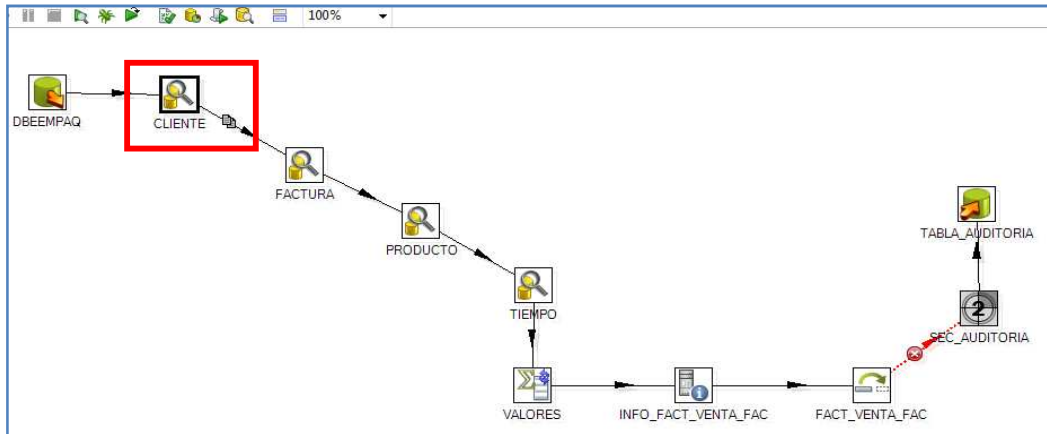
#	C_INVOICE_ID	DATEINVOICED	M_PRODUCT_ID	C_BPARTNER_ID	DECODE(QTYINVOICED,NULL,0,QTYINVOIC...	DECODE(PRICEACTUAL,NULL,0,PRICEACTU...	TOTAL	C_DOCTYPE_ID
1	1004822	2011/01/31 00:00:00.000	1003425	1000487	-31968	0.061	-1950.048	1000067
2	1003907	2011/01/31 00:00:00.000	1003425	1000487	31968	0.061	1950.048	1000001
3	1004823	2011/01/31 00:00:00.000	1003425	1000487	-31968	0.061	-1950.048	1000067
4	1003906	2011/01/31 00:00:00.000	1000085	1000255	16000	0.23039	3686.24	1000001
5	1003906	2011/01/31 00:00:00.000	1000118	1000255	16800	0	0	1000001
6	1003705	2011/01/19 00:00:00.000	1000129	1000282	22000	0.046	1012	1000001
7	1003705	2011/01/19 00:00:00.000	1005794	1000282	3923.3	3.5	13731.55	1000001
8	1003711	2011/01/19 00:00:00.000	1000016	1000442	13676	0.108	1477.008	1000001
9	1003724	2011/01/19 00:00:00.000	1000019	1000507	140000	0.0425	5950	1000001
10	1003724	2011/01/19 00:00:00.000	1000016	1000507	6838	0.095	649.61	1000001
11	1003732	2011/01/20 00:00:00.000	1003393	1001616	1800	0.05823	104.814	1000001
12	1003728	2011/01/20 00:00:00.000	1003393	1000467	41400	0.0685	2835.9	1000001
13	1003732	2011/01/20 00:00:00.000	1003294	1001616	3000	0.00723	21.69	1000001
14	1003737	2011/01/20 00:00:00.000	1000016	1000829	82056	0.083	6810.648	1000001
15	1003742	2011/01/21 00:00:00.000	1003393	1000500	43500	0.0685	2979.75	1000001
16	1003759	2011/01/21 00:00:00.000	1000058	1000255	4840	0.2168	1049.312	1000001
17	1003759	2011/01/21 00:00:00.000	1003409	1000255	160000	0.00846	1353.6	1000001
18	1003759	2011/01/21 00:00:00.000	1005592	1000255	424.4	3.29175	1397.0187	1000001
19	1003759	2011/01/21 00:00:00.000	1003411	1000255	160000	0.00846	1353.6	1000001
20	1003759	2011/01/21 00:00:00.000	1005529	1000255	1	21	21	1000001
21	1003767	2011/01/24 00:00:00.000	1000019	1000829	168000	0.0385	6468	1000001
22	1003767	2011/01/24 00:00:00.000	1003405	1000829	44800	0.106	4748.8	1000001
23	1003789	2011/01/24 00:00:00.000	1005794	1000282	996	3.5	3486	1000001
24	1003808	2011/01/25 00:00:00.000	1000061	1000439	40000	0.09694	3877.6	1000001
25	1003808	2011/01/25 00:00:00.000	1000087	1000439	16786	0.03549	595.73514	1000001
26	1003808	2011/01/25 00:00:00.000	1003287	1000439	20000	0.03549	709.8	1000001
27	1003808	2011/01/25 00:00:00.000	1003318	1000439	1020	0.08586	87.5772	1000001
28	1003808	2011/01/25 00:00:00.000	1000088	1000439	6000	0.03549	212.94	1000001
29	1003771	2011/01/24 00:00:00.000	1003316	1000255	972	2.77	2692.44	1000001
30	1003771	2011/01/24 00:00:00.000	1003334	1000255	1200	0	0	1000001
31	1003771	2011/01/24 00:00:00.000	1000086	1000255	2500	0	0	1000001

Close Show Log

**Figura 3.61 Presentación de datos de la tabla c\_bpartner**

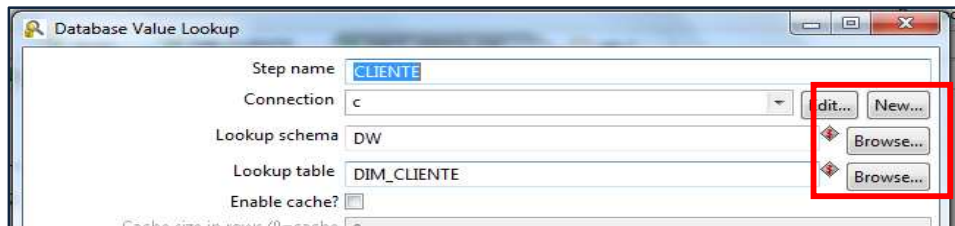
### Database Lookup

Crear un Database Lookup el permitirá realizar la conexión entre las tablas de dimensión, dar doble clic para configurar la conexión y asignar el join que unirá las claves primarias lo que permitirá obtener los datos de la dimensión cliente.



**Figura 3.62 Insertar un Database Lookup.**

Editar una conexión de la misma manera que se realizó para la DIM\_CLIENTE. Seleccionar el esquema de base de datos DW y la tabla DIM\_CLIENTE.



**Figura 3.63 Editar la conexión en el database lookup**

Realizar el join entre la clave primaria de la tabla origen y el campo que contiene la clave primaria en el Data Warehouse.

The key(s) to look up the value(s):			
#	Table field	Comparator	Field1
1	CLI_ID	=	C_BPARTNER_ID

**Figura 3.64 Join entre las tablas dim\_cliente y c\_bpartner**

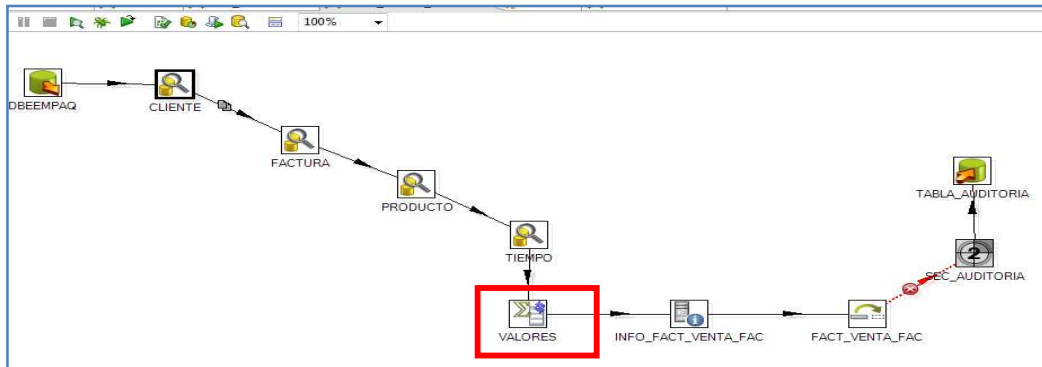


Asignar el SK correspondiente a la tabla DIM\_CLIENTE.

#	Field	New name	Default	Type
1	CLI_SK			Integer

**Figura 3.65** Asignar sk al cliente

Insertar un group by para el cálculos de las medidas.



**Figura 3.66** Cálculos y medidas para la tabla de hechos.

Dar doble clic sobre el group by, para configurar sus opciones.

Step name: VALORES

Always give back a result row:

The fields that make up the group:

#	Group field
1	CLI_SK
2	FAC_SK
3	PRO_SK
4	TIE_SK

Ingresar los SK de cada dimensión.

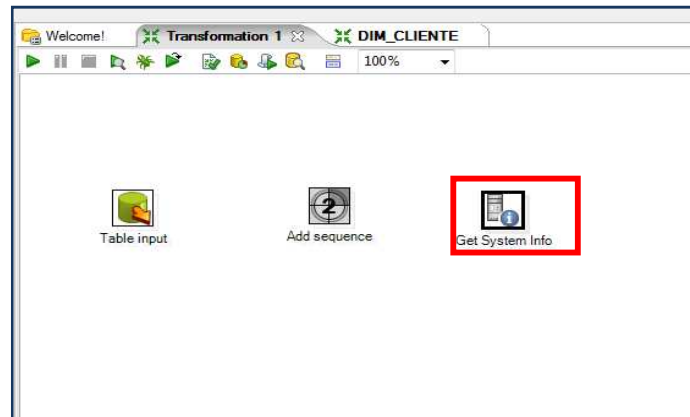
Aggregates:

#	Name	Subject	Type	Value
1	QTYINVOICED	DECODE(QTYINVOICED,NULL,0,QTYINVOICED)	Sum	
2	PRICEACTUAL	DECODE(PRICEACTUAL,NULL,0,PRICEACTUAL)	Sum	
3	TOTAL	TOTAL	Sum	

Ejecutar el cálculo de las medidas.

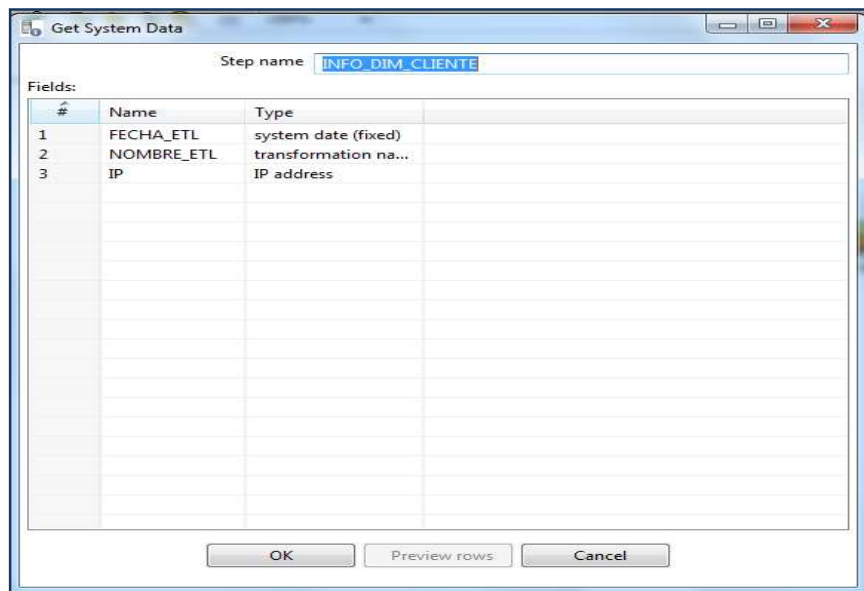
**Figura 3.67** Asignación de medidas y cálculos

Crear un Get System Info, para obtener información acerca del sistema. Este se configura de la misma manera que se lo realizó para la dimensión del cliente.



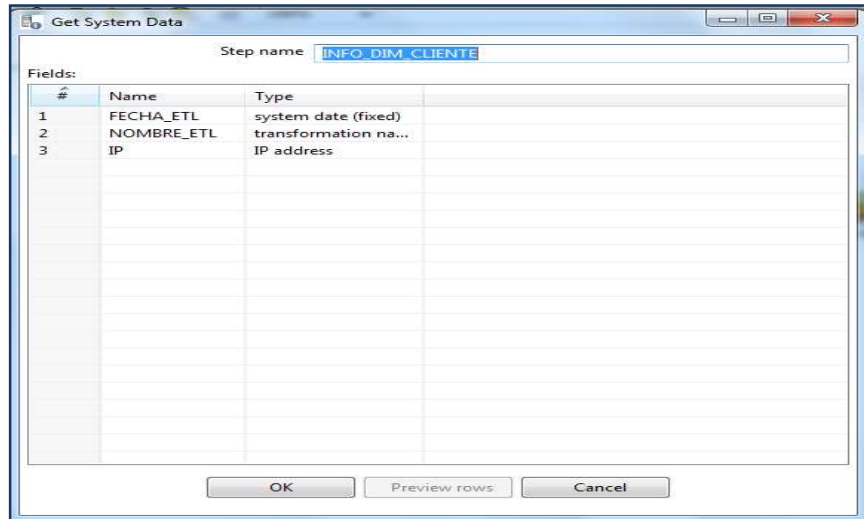
**Figura 3.68** Parámetros del sistema para la tabla de auditoría

Asignar los datos que se desea extraer del sistema como la fecha, nombre y ip correspondiente al ETL.



**Figura 3.69** Datos del sistema para la carga de una tabla de hechos

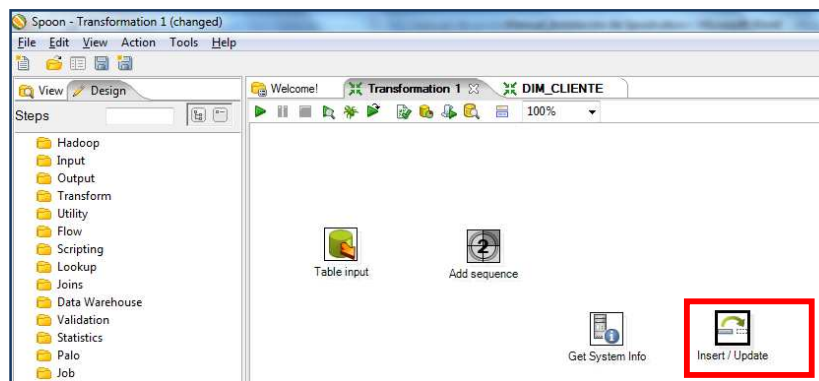
Asignar los datos que se extraerá del sistema.



**Figura 3.70** Datos del sistema para la tabla de auditoría.

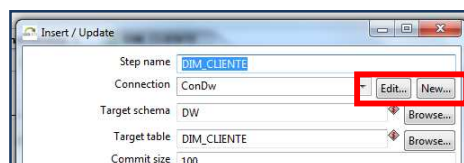
### Insert/Update

Crear un Insert/Update, para controlar la inserción y actualización en la base DW.



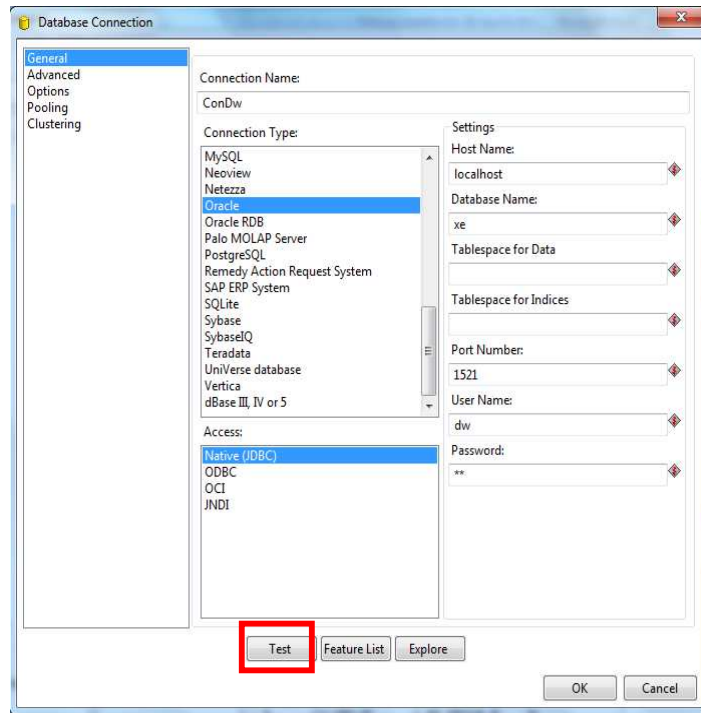
**Figura 3.71** Crear un Insert/Update.

Asignar un nombre y una conexión.



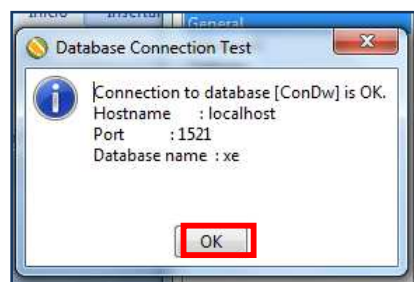
**Figura 3.72** Editar la conexión a la base.

Crear una conexión como ya se explicó anteriormente.



**Figura 3.73 Verificar la conexión Insert/Update**

Verificar que la conexión hacia la base del Data Warehouse es correcta.



**Figura3.74 Conexión correcta Insert/Update**

Escoger el esquema de la base de datos, en este caso la del Data Warehouse (DW).

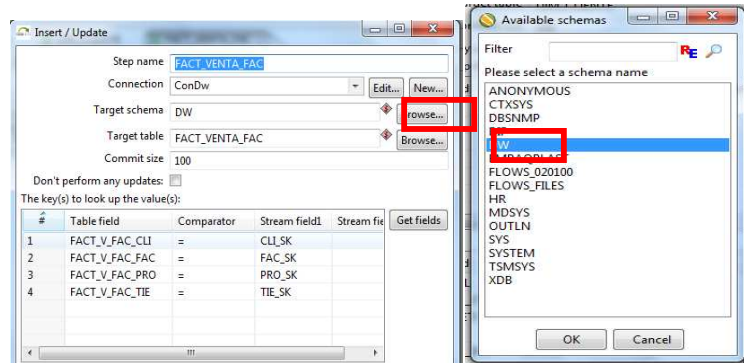


Figura 3.75 Esquema DW

Seleccionar la tabla de hechos donde se cargará los datos, en este ejemplo se realizará la carga de datos a la tabla FACT\_VENTAS.

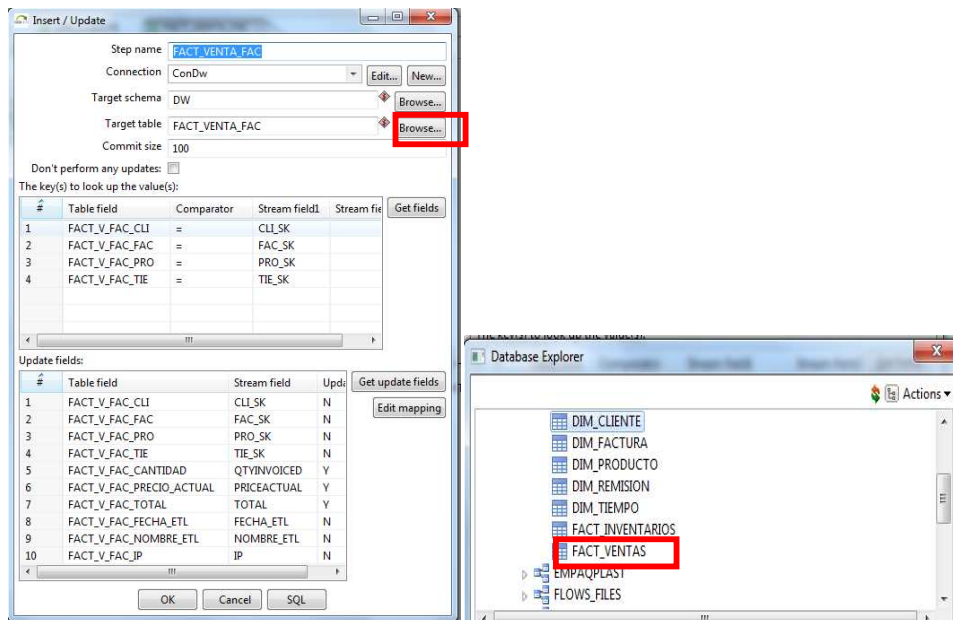
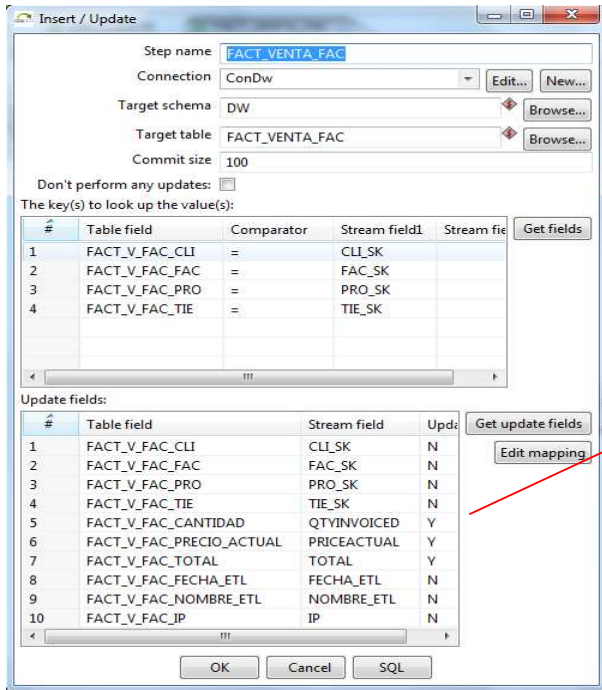


Figura 3.76 Seleccionar tabla Fact\_venta\_fac

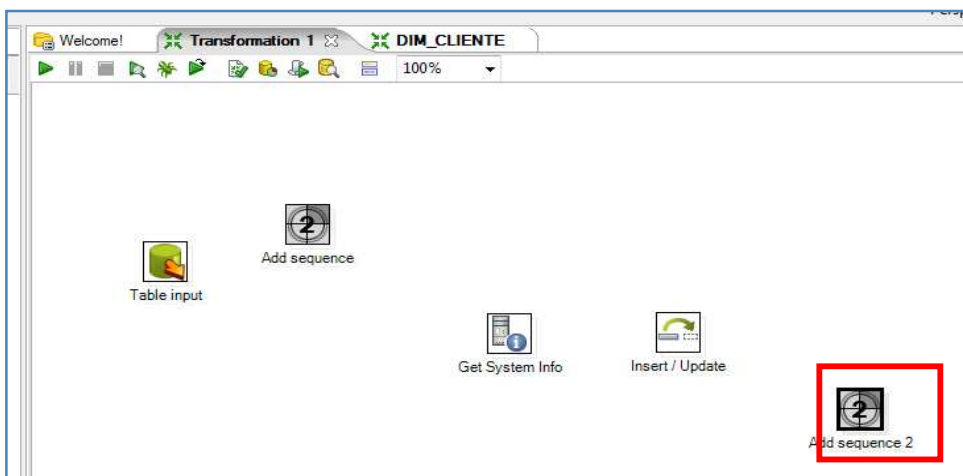
Unir los ID de la base de datos original con los del Data Warehouse (DW).



Relacionar cada uno de los datos que se trae comparar entre la base original obtest2 y la del datawarehouse. Es importante poner Y o N en cada relación así sabrá el sistema qué datos se deben actualizar y cuáles no se deben actualizar

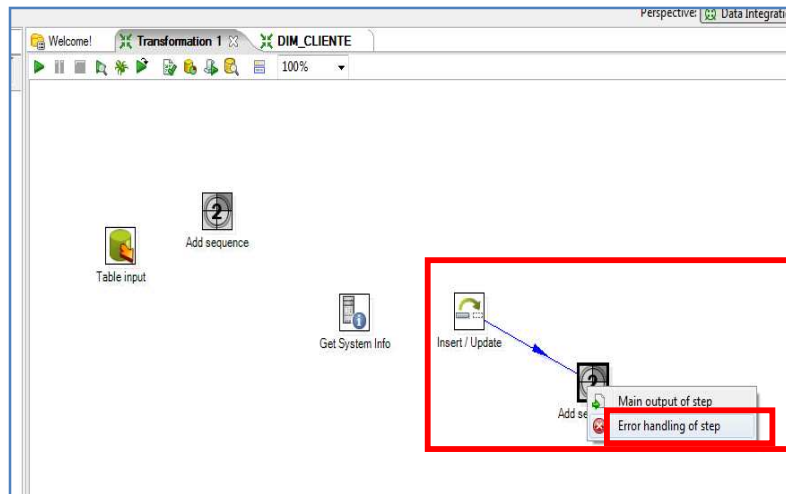
**Figura 3.77 Unión de las claves primarias**

Crear una secuencia para controlar los errores, estos errores se almacenarán en la tabla de auditoría.



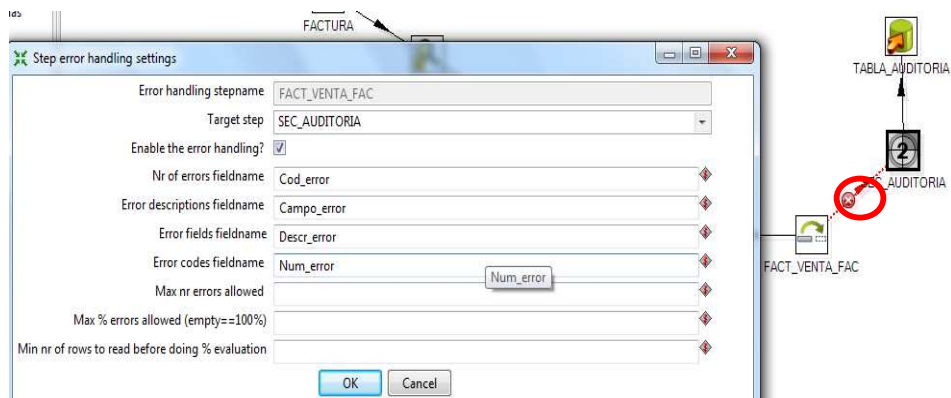
**Figura 3.78 Añadir una secuencia.**

Arrastrar la flecha desde el Insert/Update hacia la secuencia y escoger la opción de Error handling of step como se muestra en la siguiente figura.



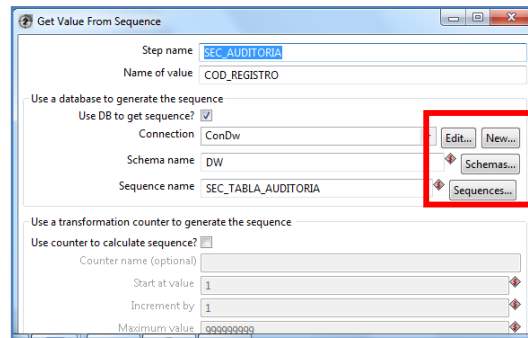
**Figura 3.79** Secuencia de error para la carga de la Fact\_Venta\_Fac

Ingresa los campos de error que van a controlar, dando clic en la "X".



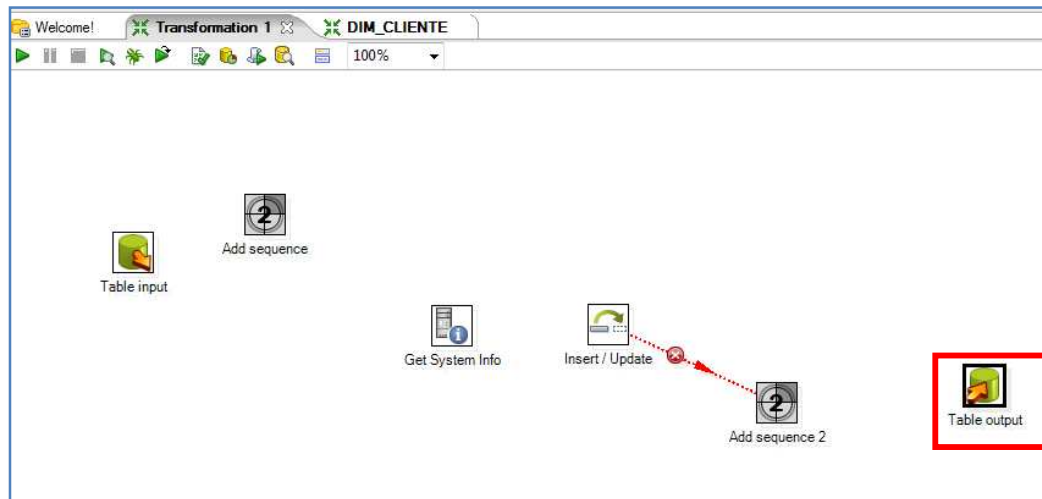
**Figura 3.80** Campos de error

Realizar la conexión de la secuencia SEC\_AUDITORIA, de la misma manera como se hizo la conexión para la anterior secuencia.



**Figura 3.81 Editar la conexión secuencia**

Para finalizar se crea un table output, que se llamará HIS\_AUDITORIA, en esta tabla se almacenarán los errores y los campos de auditoría que se controlaron anteriormente.



**Figura 3.82 Selección de un Table Output para la tabla de auditoría**

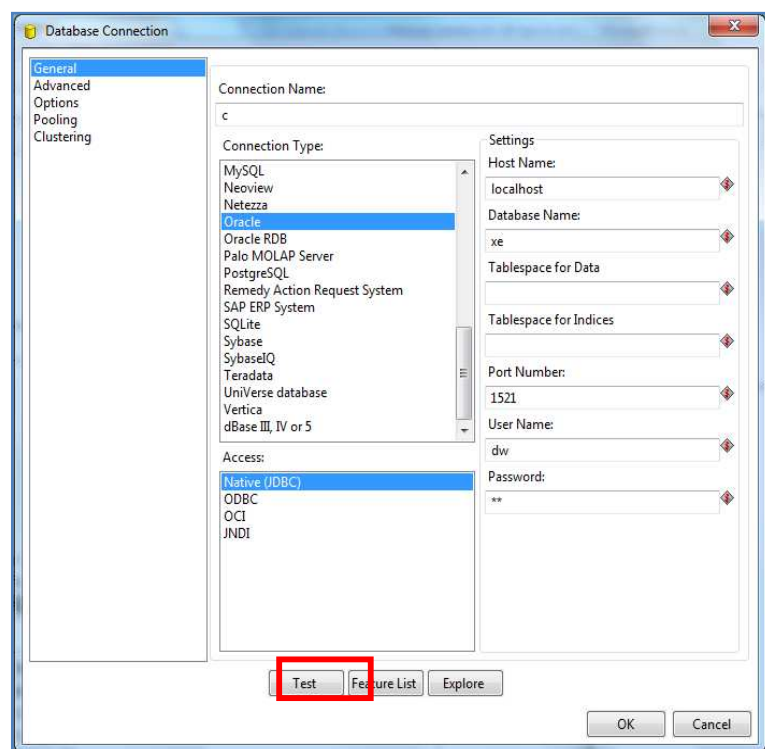


Conexión para la tabla de auditoría.



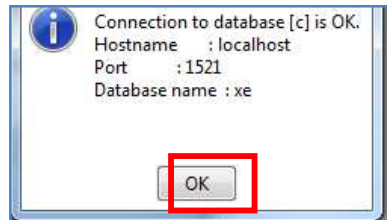
**Figura 3.83 Editar la conexión Table Output**

Configuración de campos obligatorios para la conexión dar clic en test.



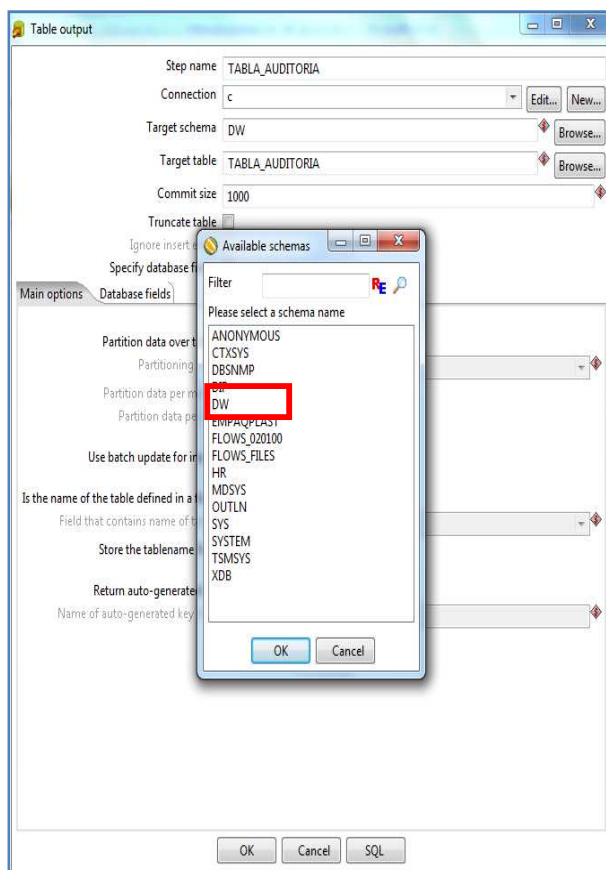
**Figura 3.84 Probar la conexión correcta Table Output**

Si la conexión es correcta dar clic en ok.



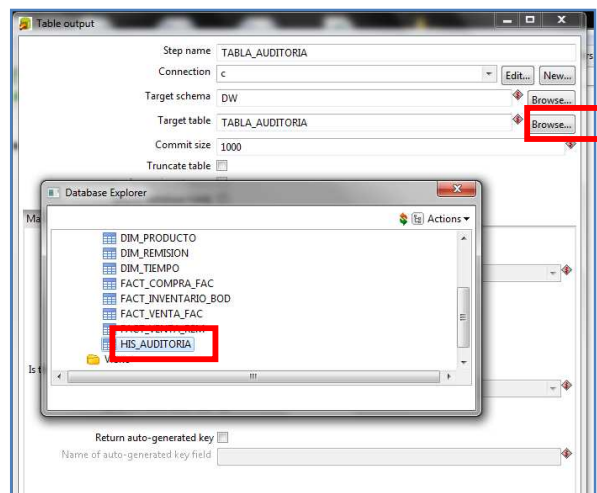
**Figura 3.85 Conexión correcta Table Output**

Seleccionar el esquema del Data Warehouse (DW).



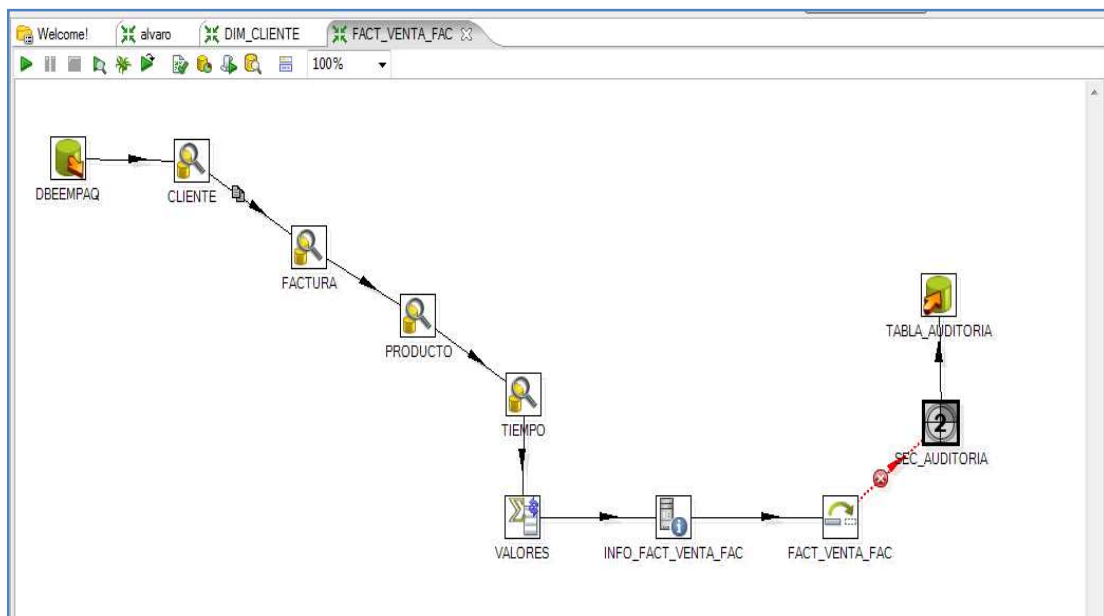
**Figura 3.86 Seleccionar el esquema de la base de datos.**

Dentro del esquema DW se encuentra la tabla HIS\_AUDITORIA. Seleccionar esta tabla.



**Figura 3.87 Selección de tabla de Auditoria.**

Finalmente se tiene el siguiente diseño el cual se aplica para todas las tablas de hechos.



**Figura 3.88 Carga ETL para Fact\_Venta\_Fac**

### Garga de dimensiones y tablas de hechos "Fact", desde un job.

Crear un Start para programar los tiempos de ejecución del ETL.

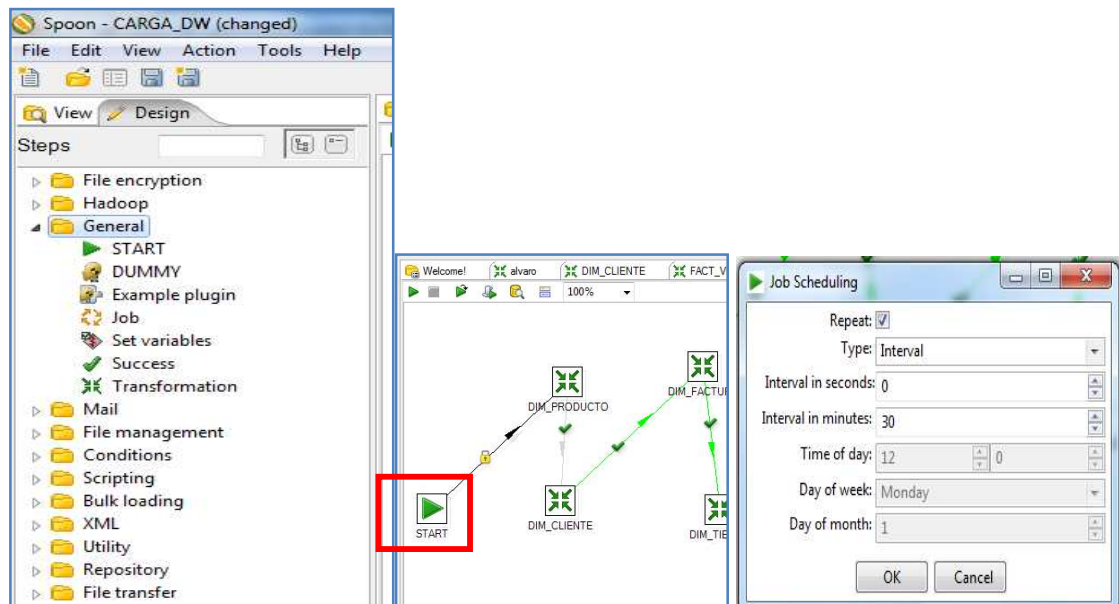


Figura 3.89 Tiempos de ejecución ETL.

Insertar una transformación para cada dimension y fact.

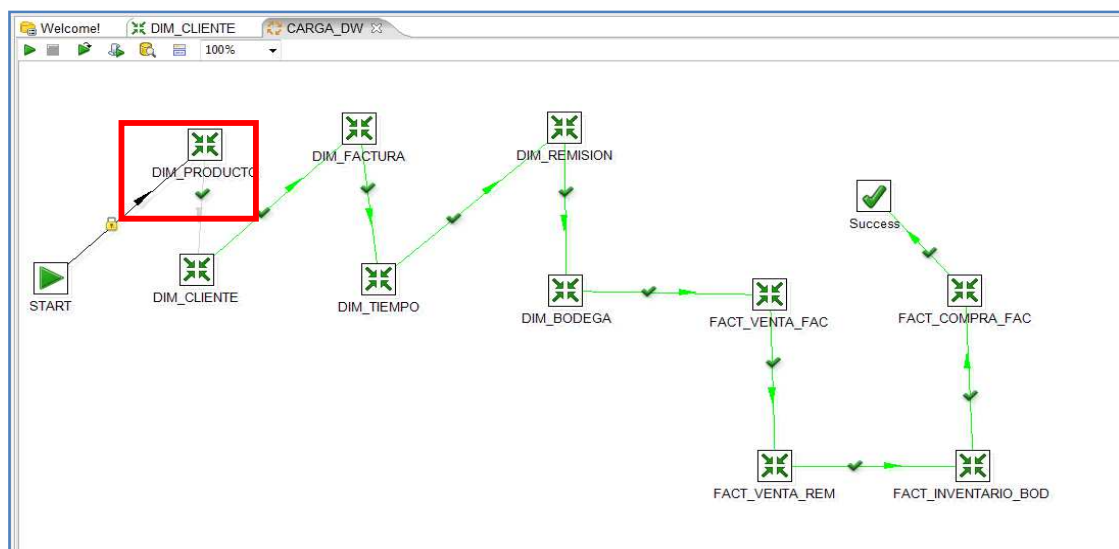
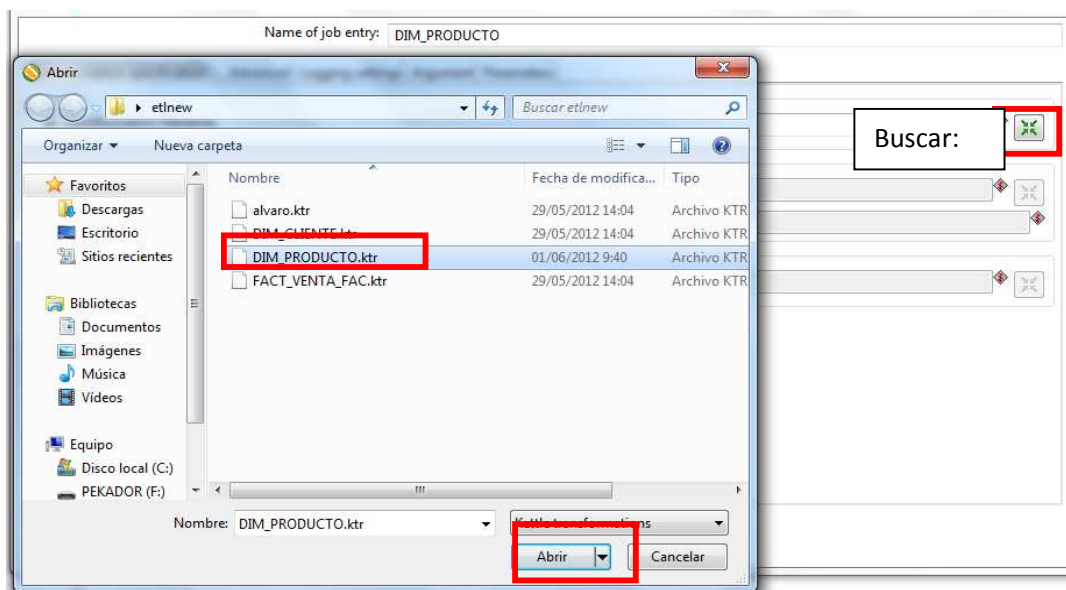


Figura 3.90 Transformación por cada Dimensión y Fact.

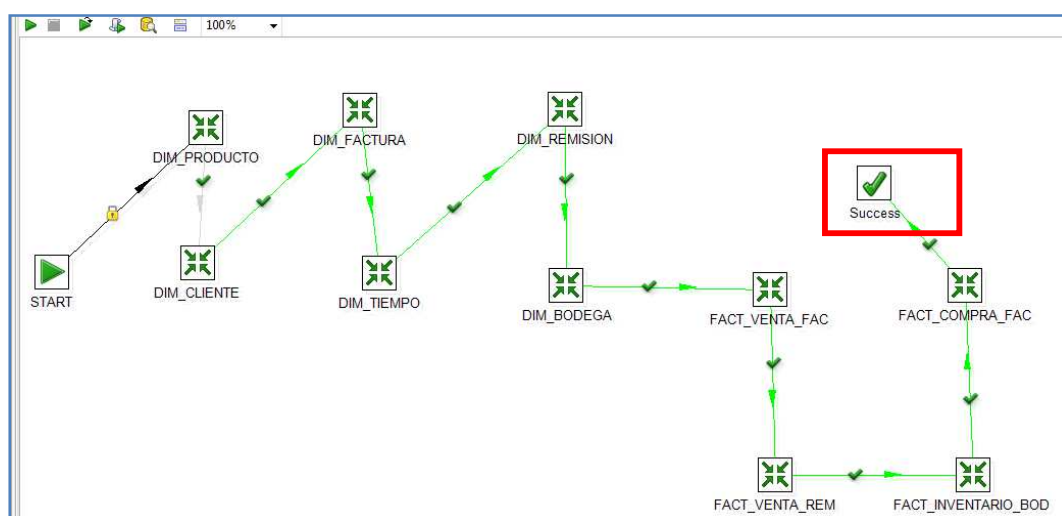
Dar clic en buscar y seleccionar el archivo .ktr donde se encuentran las transformaciones de las dimensiones y las tablas de hechos.

Insertar transformaciones de la misma manera para una dimensión y una tabla de hechos denominadas fact.



**Figura 3.91 Selección de la Dimensión o Fact.**

Insertar un Success, para verificar el éxito de la carga.



**Figura 3.92 Éxito de carga.**

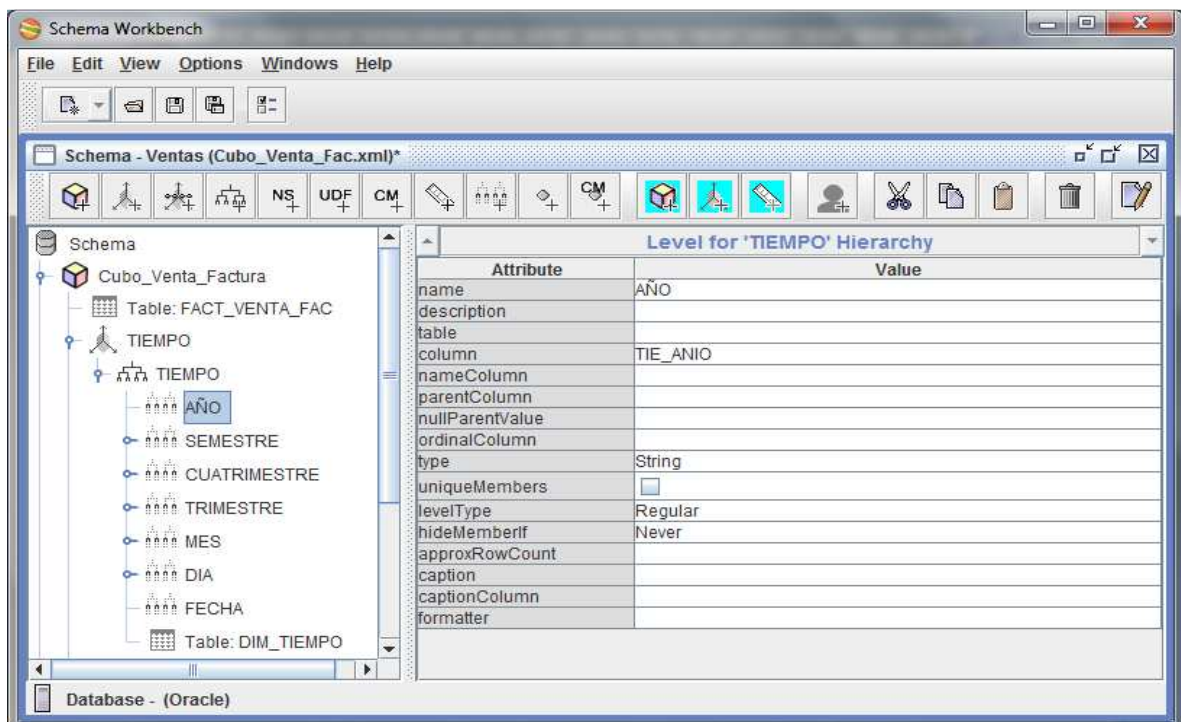
## SCHEMA WORKBENCH

Después de realizar el proceso de ETL y una vez que el Data Mart ya contiene los datos se procede a la construcción de los cubos multidimensionales con la herramienta Schema Workbench.

La instalación de la herramienta esta descrita en el **anexo F**. Para el caso práctico se diseño y creo 2 cubos para el área de ventas, 1 cubo para el área de compras y 1 cubo para el área de inventarios.

Se procederá a la explicación de la construcción de un cubo tomando como ejemplo en cubo de ventas por factura y de la misma manera se procedió en la creación de los demás cubos.

**El desarrollo del ejemplo se encuentra en el anexo G.**



**Figura 3.93** Interfaz Schema Workbench

### **3.6.5 Especificación de Aplicaciones para Usuarios Finales**

En esta etapa se realiza la creación y asignación de los diferentes usuarios y roles configuraciones del servidor y los distintos tipos de acceso, por lo que el procedimiento que se siguió se detallará a continuación.

#### **PENTAHO BI SERVER MONDRIAN**

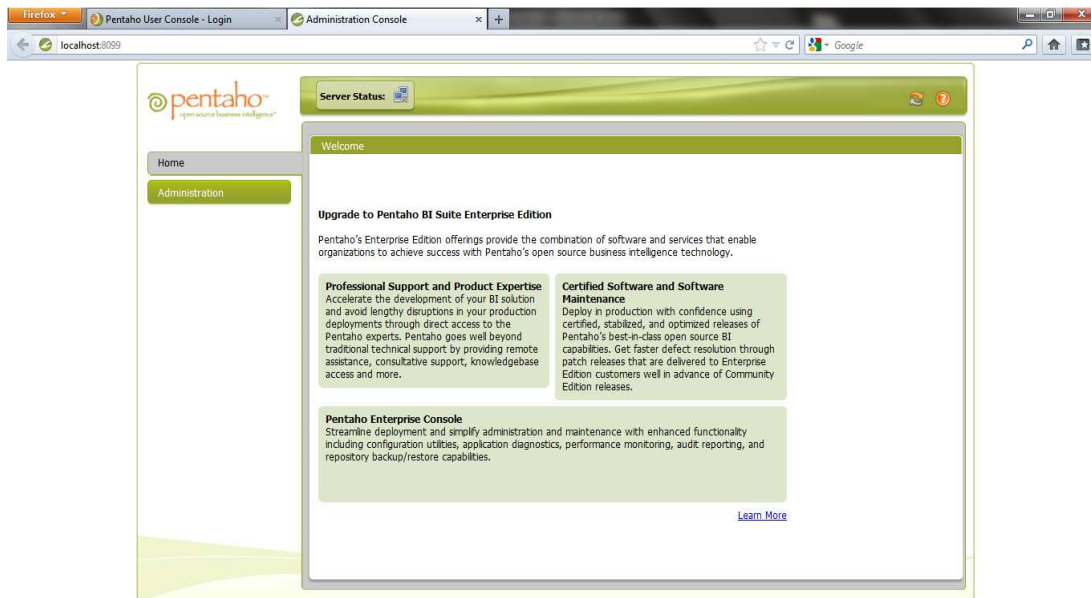
El servidor Mondrian contiene dos carpetas una llamada “biserver-ce” y la otra “administration-console”.

La carpeta “biserver-ce” es la que contiene el framework del servidor de BI y donde se encuentran los archivos para inicializarlo.

En la carpeta “administration-console” se encuentran los archivos para iniciar la consola de administración.

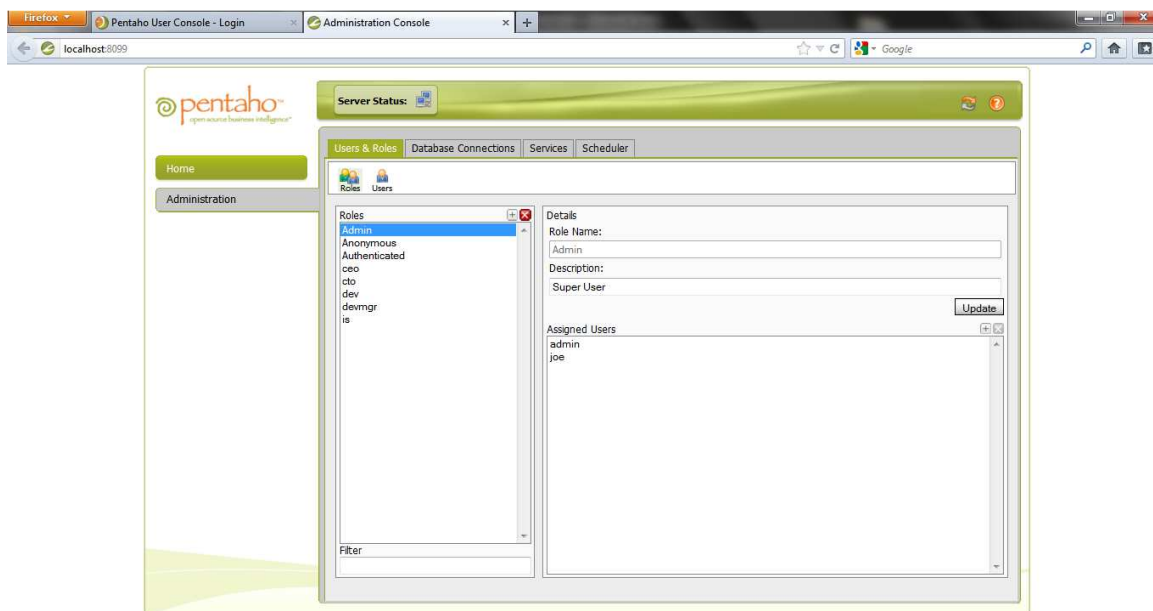
Ejecutando el archivo start-pac.bat para Windows o start-pac.bat para Linux se inicia la consola de administración para ingresar a la misma se ingresa a través de un browser digitando la siguiente dirección [http://ip\\_maquina:8099](http://ip_maquina:8099)

**El manejo del servidor Mondrian y la consola de usuario se encuentran en mayor detalle en el anexo J.**



**Figura 3.94 Menú principal consola de administración**

Una vez dentro de la consola de administración es donde se puede agregar usuarios, roles, crear conexiones a bases de datos entre otras funciones.



**Figura 3.95 Menú para roles y usuarios**



Para ingresar al servidor se debe digitar en un browser la siguiente dirección [http://ip\\_maquina:8080/pentaho](http://ip_maquina:8080/pentaho)



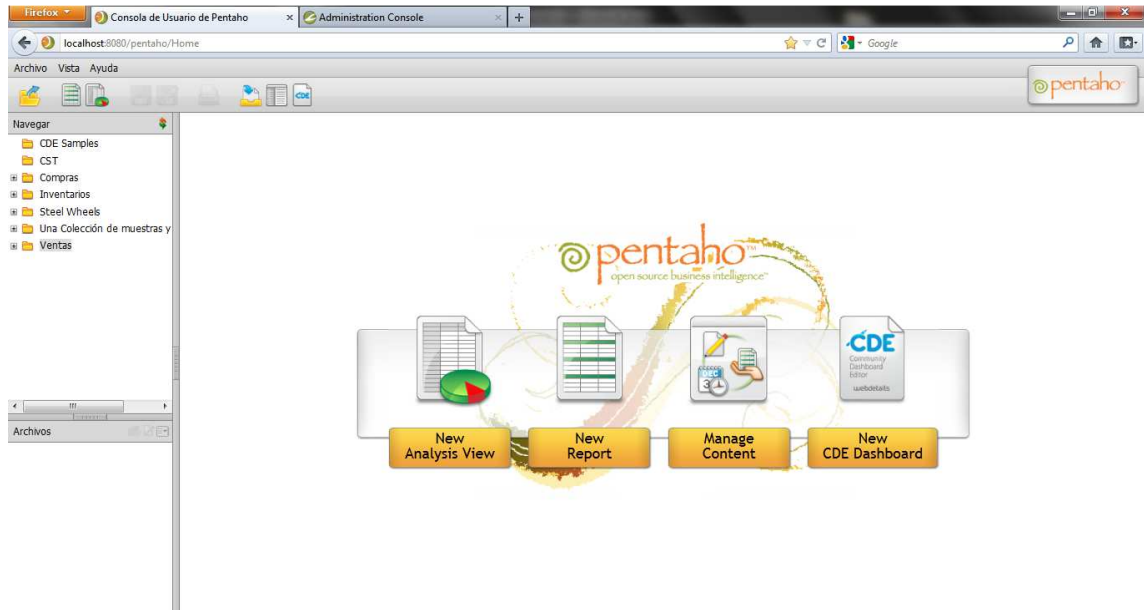
**Figura 3.96** Pantalla de login pentaho bi server

Se ingresa con el usuario y contraseña.

A screenshot of a login form titled 'Ingresar'. It features a dropdown menu for 'Usuario de Ejemplo:' with the text 'Seleccione un usuario'. Below it are input fields for 'Usuario:' containing 'ESPE' and 'Contraseña:' with masked characters. There is a checkbox labeled 'Abrir en ventana nueva' which is unchecked. At the bottom are 'Ingresar' and 'Cancelar' buttons.

**Figura 3.97** Autenticación de usuario y contraseña

Esta pantalla muestra, el menú principal del servidor BI de pentaho, contiene los menús para realizar vistas de análisis, nuevos reportes, Tableros de control y visualizar carpetas de contenidos.

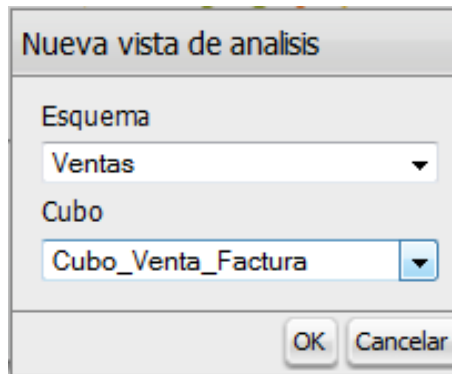


**Figura 3.98 Menú principal pentaho bi server**



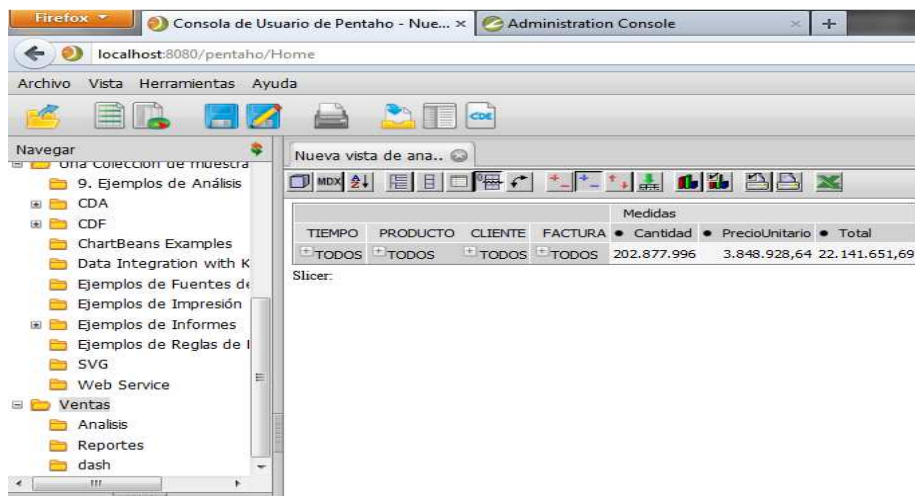
**Figura 3.99 Menú de herramientas BI**

Para analizar un cubo seleccionar New Analysis View. Se selecciona cualquiera de los esquemas y cubos ya publicados por la herramienta schema work bench.



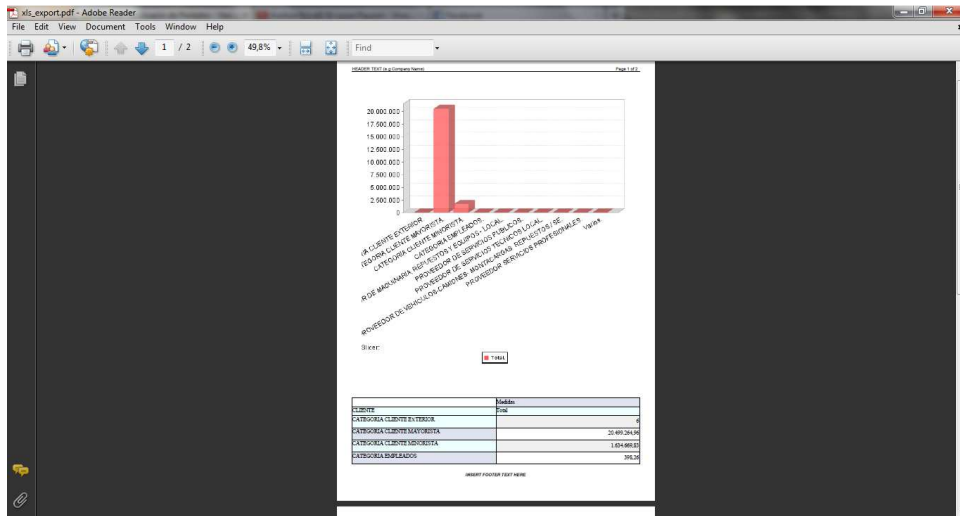
**Figura 3.100 Nueva vista de análisis**

Se obtiene la vista completa del cubo, de acuerdo a las necesidades del usuario se puede fabricar un número indefinido de vistas de análisis y guardarlas en una carpeta.



**Figura 3.101 Vista de un cubo**

Cuando la vista de análisis ya está elaborada de acuerdo a las necesidades del usuario se puede exportar a pdf.



**Figura 3.102 Opción para exportar a PDF**

Desde el servidor BI se visualiza los reportes y tableros de control que se crearon y publicaron previamente

Report Parameters

Anio: 2011

Mes: Febrero

Cliente: ECUAREFRESGOS S.A.

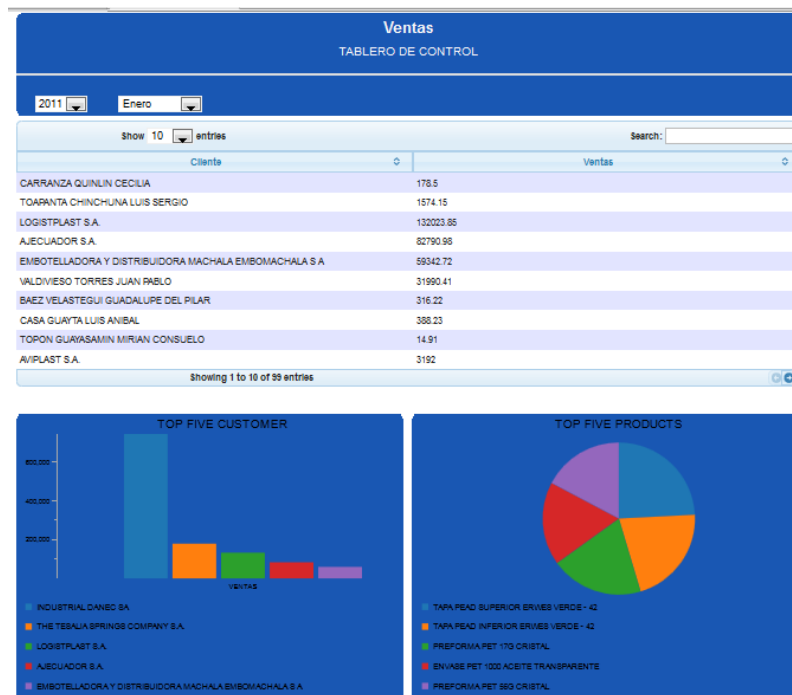
Output Type: HTML (Paginated)

View Report  Auto-Submit

Page 1 of 1

CODIGO	NOMBRE	TOTAL
PTENVA0025	ENVASE PET NATURAL 2000 PETALOIDE	3.300
PTENVA0021	ENVASE PET CRISTAL 4000CC	6.045,2
PTENVA0119	ENVASE TRANSPARENTE PETALOIDE 500 CC	1.922,7
PTENVA0094	ENVASE PET NATURAL 500 PETALOIDE CON GAS	1.233
PTLAMI0047	LAMINA PEBD NATURAL 200CC TROPICAL 32.5CM X 60MICR	12.847,8

**Figura 3.103 Vista de un reporte**



**Figura 3.104 Vista de un dashboard**

### 3.6.6 Desarrollo de Aplicaciones para Usuarios Finales

En esta etapa se define como los usuarios acceden al Data Warehouse y por medio de que herramientas gráficas, para el caso práctico de este proyecto se realizó reportes y Dashboards o tableros de control.

**El proceso de cómo crear reportes se lo realiza con mayor detalle en el anexo H.**

### PENTAHO REPORT DESIGNER



**Figura 3.105 Logo Report Designer**

Para el caso práctico se diseñó y creó 3 reportes para área de ventas, 2 reportes para el área de compras y 1 reporte para el área de inventarios.

Se procederá a la explicación de la construcción de un reporte tomando como ejemplo el reporte de ventas por cliente y de la misma manera se procedió en la creación de los demás reportes.

Iniciar la Herramienta Report Designer, si es en entorno Windows ejecutando el archivo report-designer.bat y si es entorno Linux ejecutando el archivo report-designer.sh

Para iniciar la construcción de un reporte se lo puede realizar de dos formas mediante un Wizard con plantillas predeterminadas de Pentaho Report Designer o un Reporte nuevo en blanco.

En este caso se seleccionará un reporte nuevo para adaptarlo a los formatos de la empresa.

Se tomará como ejemplo la construcción del reporte de ventas por cliente, para el caso práctico se procedió de la misma forma con los demás reportes.



**Figura 3.106** Menu pentaho report designer

## **TABLEROS DE CONTROL**

### **Configuración de CDE.**

El editor de escritorio de la Comunidad (CDE), que está integrado en la consola de usuario de Pentaho (PUC), es una herramienta gráfica para crear, editar y visualizar cuadros de mando de Pentaho.

Para el diseño de un dashboard, CDE ofrece cuatro perspectivas:

- **Layout:** Poder diseñar un dashboard, desde cero o a partir de una plantilla. Se pueden aplicar estilos y agregar elementos HTML como textos o imágenes.
- **Components:** Para agregar y configurar los distintos componentes que conforman el panel de control: Componentes Visuales (cuadros de texto, tablas, gráficos), los parámetros y Scripts.
- **Data Sources:** Para la definición de las fuentes de datos utilizadas por los componentes.
- **Preview:** La opción de vista previa es el método abreviado para probar el comportamiento del dashboard, que está trabajando.

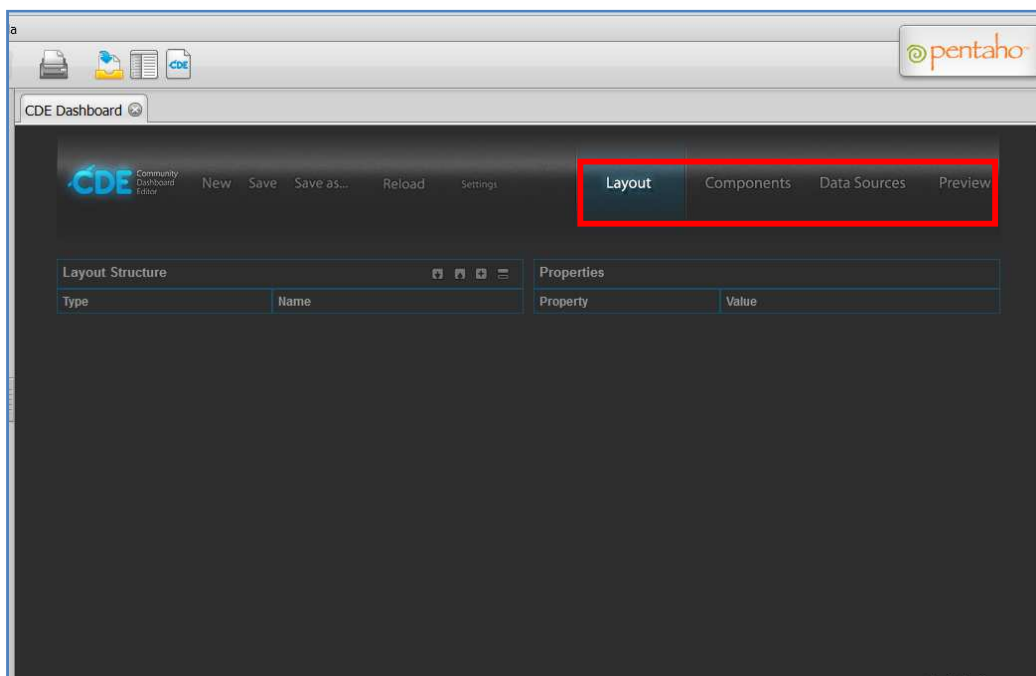
Para empezar a diseñar un tablero de control es necesario iniciar el Bi Server, acceder a CDE, ambiente gráfico para crear un dashboard.

**La creación de tableros de control se encuentra en mayor detalle en el anexo I**



**Figura 3.107 Creación nuevo dashboard**

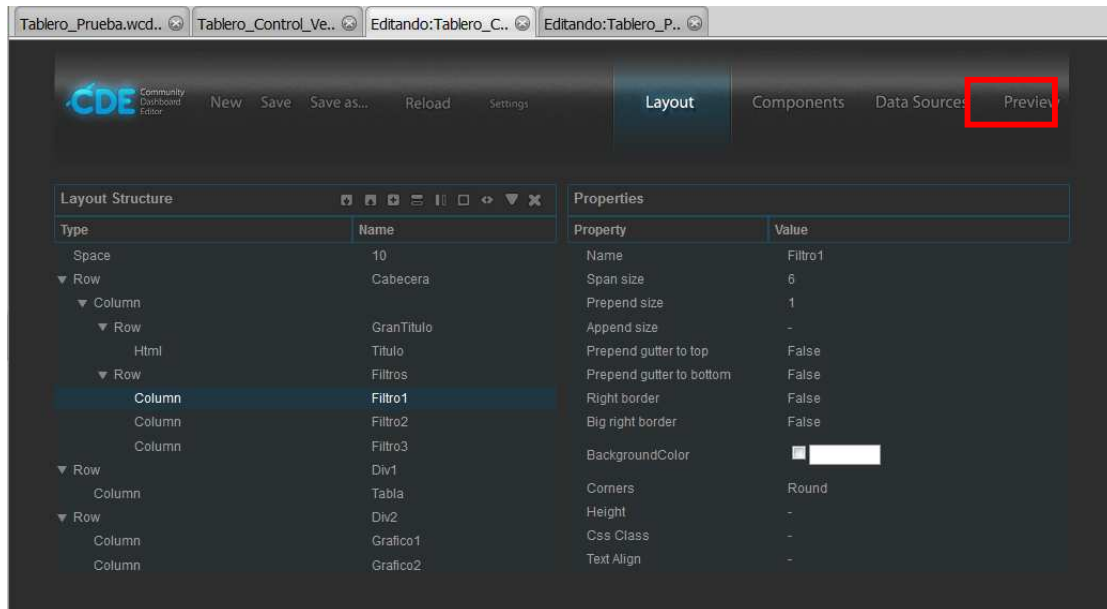
En esta pantalla se crean las cuatro perspectivas para el diseño de tableros de control.



**Figura3.108 Perspectivas para el diseño de tableros de control**

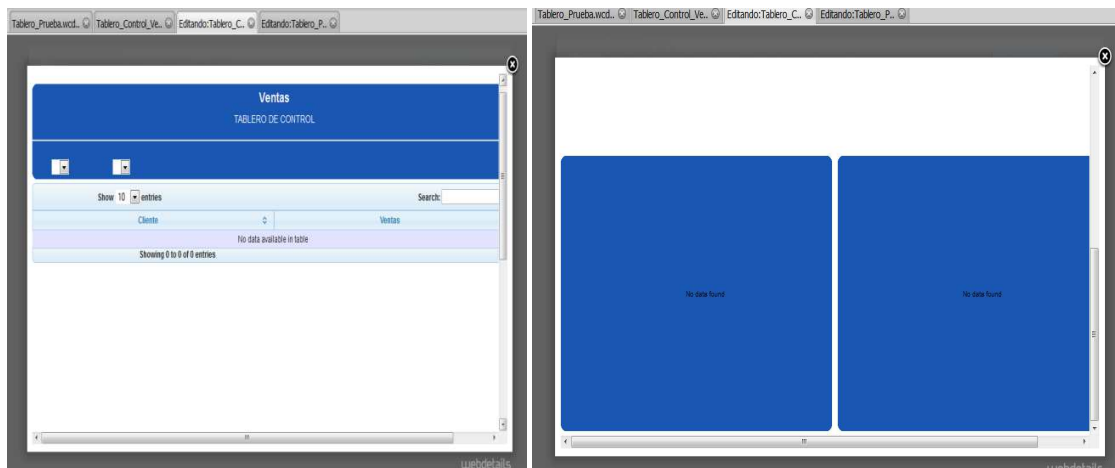


## Diseño de un tablero de control y vista previa



**Figura 3.109 Vista previa del tablero de control**

Vista previa del diseño del tablero de control para ventas.



**Figura 3.110 Vista previa diseño tablero de control.**

Finalmente se obtiene un tablero de control con la información necesaria y el diseño deseado.



**Figura 3.111 Dashboard completo**

### 3.6.7 Implementación

El proyecto ha sido implantado en la empresa EMPAQPLAST tomando en cuenta las siguientes consideraciones.

- El Data Warehouse está alojado en una base de datos oracle 10g.
- El servidor BI correrá en un servidor Centos.
- Los roles y usuarios serán administrados por el gerente de Sistemas.
- Los usuarios del sistema recibirán capacitaciones básicas sobre el uso del sistema.

### 3.6.8 Mantenimiento y crecimiento

El mantenimiento y crecimiento del sistema será responsabilidad del departamento de sistemas de la empresa EMPAQPLAST.

### 3.6.9 Gestión del Proyecto

La gestión del proyecto hasta la etapa de Implementación fue monitoreada por los estudiantes y el departamento de sistemas.

## CAPÍTULO 4

### CONCLUSIONES Y RECOMENDACIONES

#### 4.1 Conclusiones

- El análisis para el desarrollo de un Data Warehouse en las áreas de compras ventas y inventarios, generó información que se fue descartando conforme a los requerimientos de cada área de negocio. Esto permitió que el Data Warehouse, contenga la información necesaria y útil para cada área de negocio. Para la implementación de futuros requerimientos se recomienda trabajar en conjunto con las áreas de negocio y las TIC.
- El objetivo de usar una metodología como Ralph Kimball, facilitó el desarrollo del proyecto, dividiéndolo por etapas al ciclo de vida, donde cada etapa pudo ser evaluada y corregida a tiempo. Empezando desde la planificación hasta el mantenimiento y crecimientos del Data Warehouse.
- El uso de la herramienta Open Source Pentaho previamente investigada y analizada. Permitió la implementación de un Data Warehouse de una manera ágil, fácil de comprender y brindando la estabilidad necesaria al momento de integrar los datos. Se tiene que tomar en cuenta que uno de los inconvenientes comunes es el origen de datos al momento de realizar el proceso de extracción, transformación y carga ETL.
- Un buen diseño del Data Warehouse puede optimizar las consultas reflejadas en tiempos de respuesta obteniendo datos relevantes para el análisis del negocio. En un Data Warehouse a diferencia de una base de datos Transaccional no es posible realizar un CRUD. El proceso de extracción, transformación y carga es ejecutado mediante un proceso ETL.

## 4.2 Recomendaciones

- Es importante identificar las áreas de negocio implicadas, entender sus necesidades, para esto el uso de herramientas como la entrevista, permite recopilar información relevante de los procesos fundamentales de una organización. Esta información pasará a ser analizada por el área de sistemas en conjunto con las áreas de negocio seleccionadas y podrán ser implementadas.
- El uso de una metodología en todo proyecto de desarrollo es importante al momento de darle un estándar a nuestro sistema Business Intelligence, logrando así que el proyecto pueda ser fácil de comprender y de ayuda para nuevos proyectos de Business Intelligence, sobre la herramienta Open Source Pentaho.
- La suite Open Source Pentaho. Es una plataforma compuesta de diferentes programas que satisfacen los requisitos de Business Intelligence. Ofreciendo soluciones para la gestión y análisis de la información, incluyendo el análisis multidimensional OLAP, presentación de informes y creación de cuadros de mando para el usuario por lo que constituye una solución factible para cualquier organización que requiera implementar un sistema Business Intelligence.
- El análisis de la estructura y relaciones entre tablas de la base de datos origen, es importante al momento de desarrollar un Data Warehouse. El uso de un esquema estrella o copo de nieve dependerá directamente del conocimiento de la estructura de la base de datos origen. El cual se verá reflejado en el acceso a los datos, tiempo de una consulta, almacenamiento y la calidad de información que contenga nuestro Data Warehouse.

## **Bibliografía**

- Ing. Bernabeu Ricardo Darío Córdoba. “Data warehousing: investigación y sistematización de conceptos”
- Ing. Bernabeu Ricardo Darío Córdoba “Hefesto: metodología para la construcción de un data warehouse”
- Josep Lluís Cano “Business Intelligence Competir con información ”
- Ralph Kimball “The data warehouse toolkit” second edition.
- Castellan de la Cerda María del Carmen, Rodríguez de León, Estuardo”Bussiness Intelligence y E-commerce”, Proyecto Universidad Francisco Marroquín, Guatemala.
- Erika Graciela Sevilla Berrios”Guía metodológica para la definición y desarrollo de un data warehouse” Proyecto Universidad Americana, Nicaragua.
- Josep Lluís Cano “Business Intelligence Competir con información ” .
- Galuth Irene García Camacho, Carmen Silvana, Murillo Silva “Estudio de herramientas bussiness intelligence para la implementación de un sistema de información gerencial en la unidad de planificación de la esPOCH” Proyecto Escuela Superior Politécnica de Chimborazo.

- Miguel Rodríguez Sanz. “Análisis y diseño de un Data Mart para el seguimiento académico de alumnos en un entorno universitario”, Proyecto Universidad Carlos III Madrid.

## Webgrafía

- Sitio Oficial de Pentaho. Disponible en: (<http://www.pentaho.com>)
- Caso de Estudio - Inteligencia de Negocios. Disponible en: (<http://rimenri.blogspot.com>)
- Oracle. Disponible en: (<http://es.wikipedia.org/wiki/Oracle>)
- Kimball vs Inmon Ampliación de conceptos del Modelado Dimensional. Disponible en: ( <http://churriwifi.wordpress.com/2010/04/19/15-2-ampliacion-conceptos-del-modelado-dimencional>)
- Inteligencia de Negocios. Disponible en: (<http://www.gopac.com.mx/v3/gopacbi/quees.asp>)
- The BI Lifecycle Disponible en: ([http://www.bidsolutions.co.za/bi\\_landscape.html](http://www.bidsolutions.co.za/bi_landscape.html))
- Entendiendo como Oracle Arranca Disponible en: (<http://alejandrotujillo.wordpress.com/2008/10/07/>)
- Conceptos de Business Intelligence Disponible en: ([http://www.sinnexus.com/business\\_intelligence](http://www.sinnexus.com/business_intelligence))
- Arquitectura BI ([http://www.es.atosconsulting.com/es-es/servicios/soluciones/consultoria\\_tecnologica/business\\_intelligence/default.htm](http://www.es.atosconsulting.com/es-es/servicios/soluciones/consultoria_tecnologica/business_intelligence/default.htm))
- Arquitectura BI Disponible en: (<http://casedw.wikispaces.com/BackRoom+Technical+Architecture>)
- <http://www.opensourceclub.net/>

## **ANEXOS**

### **Anexo A - Entorno de pruebas**

Se encuentra en el cd adjunto al proyecto en la carpeta Anexos

### **Anexo B - Consultas ETL**

Se encuentra en el cd adjunto al proyecto en la carpeta Anexos

### **Anexo C - Script base de datos**

Se encuentra en el cd adjunto al proyecto en la carpeta Anexos

### **Anexo D - Constancia de los requerimientos**

Se encuentra en el cd adjunto al proyecto en la carpeta Anexos

### **Anexo E - Constancia de las reuniones**

Se encuentra en el cd adjunto al proyecto en la carpeta Anexos

### **Anexo F - Manual instalación productos pentaho**

Se encuentra en el cd adjunto al proyecto en la carpeta Anexos

### **Anexo G - Manual técnico de Schema workbench**

Se encuentra en el cd adjunto al proyecto en la carpeta Anexos

### **Anexo H - Manual técnico de Report Designer**

Se encuentra en el cd adjunto al proyecto en la carpeta Anexos

### **Anexo I - Manual técnico CDE**

Se encuentra en el cd adjunto al proyecto en la carpeta Anexos

### **Anexo J - Manual técnico servidor Mondrian**

Se encuentra en el cd adjunto al proyecto en la carpeta Anexos

**HOJA DE LEGALIZACION DE FIRMAS  
ELABORADA(O) POR**

Sr. Byron Alejandro Boada Vargas Machuca

Sr. Alvaro Arturo Tituaña Burgos

**DINADOR DE LA CARRERA DE INGENIERIA EN SISTEMAS E  
INFORMATICA**

ING. MAURICIO CAMPAÑA

---

Sr. Ing. Mauricio Campaña

Lugar y fecha: \_\_\_\_\_