



UNIDAD ACADÉMICA:

DEPARTAMENTO DE INVESTIGACIÓN Y POSTGRADOS

TEMA:

IMPLEMENTACIÓN DE UN DATAWAREHOUSE CORPORATIVO PARA LA INTEGRACIÓN DE DATOS DE LAS ÁREAS DE LA ORGANIZACIÓN PARA LA TOMA DE DECISIONES EN LA EMPRESA TUBASEC C.A.

Proyecto de investigación y Desarrollo previo a la obtención del título de

Magister en Gerencia Informática

Línea de Investigación, Innovación y Desarrollo principal:

Sistemas de Información y/o nuevas tecnologías de la Información y Comunicación y sus Aplicaciones

Caracterización técnica del trabajo:

Aplicación

Autor:

José Fernando Esparza Parra

Director:

José Marcelo Balseca Manzano, Mg.

Ambato – Ecuador

Septiembre 2016

Implementación de un Datawarehouse Corporativo para la Integración de Datos de las Áreas de la Organización para la Toma de Decisiones en la Empresa Tubasec C.A.

Informe de Trabajo de Titulación
presentado ante la
Pontificia Universidad Católica del
Ecuador Sede Ambato

por

José Fernando Esparza Parra

En cumplimiento parcial
de los requisitos para el Grado
de Magister en Gerencia
Informática



Departamento de Investigación y Postgrados
Septiembre 2016

Implementación de un Datawarehouse Corporativo para la Integración de Datos de las Áreas de la Organización para la Toma de Decisiones en la Empresa Tubasec C.A.

Aprobado por:

Varna Hernández, PhD
Presidente del Comité Calificador
Director DIP

Dennis Vinicio Chicaiza Castillo Mg.
Miembro Calificador

José Marcelo Balseca Manzano, Mg.
Miembro Calificador
Director de Proyecto

Dr. Hugo Altamirano Villaroel
Secretario General

Ricardo Patricio Medina Chicaiza, Mg.
Miembro Calificador

Fecha de aprobación:
Septiembre 2016

Ficha Técnica

Programa: Magister en Gerencia Informática.

Tema: Implementación de un Datawarehouse Corporativo para la Integración de Datos de las Áreas de la Organización para la Toma de Decisiones en la Empresa Tubasec C.A.

Tipo de trabajo: Proyecto de Investigación y Desarrollo.

Clasificación técnica del trabajo: Aplicación.

Autor: José Fernando Esparza Parra.

Director: José Marcelo Balseca Manzano, Mg

Líneas de Investigación, Innovación y Desarrollo

Principal: Sistemas de Información y/o Nuevas Tecnologías de la Información y Comunicación y sus aplicaciones.

Resumen Ejecutivo

El presente proyecto tiene como objetivo implementar un *Data Warehouse* en la empresa Tubasec C.A. cuya actividad económica es la fabricación de prefabricados para la construcción de viviendas y terminado de viviendas, durante la investigación inicial se logró identificar que los procesos productivos y de ventas generan gran cantidad de información, misma que es ingresada, almacenada y procesada en varios sistemas que trabajan de manera independiente, lo que ocasionaba que la información esté fraccionada y la toma de decisiones se torne en una tarea complicada.

La implementación del *Data Warehouse* se realizó en base a la metodología de Hefesto 2.0, dicha metodología permitió delimitar el proyecto gracias a las fases denominadas Análisis de Requerimientos y Análisis de las fuentes. En las etapas posteriores se diseñó un modelo lógico de la Base de Datos, capaz de soportar y almacenar toda la información necesaria para generar indicadores, cuadros de control y reportes.

Con el modelo lógico del *Data Warehouse* preparado, se procedió con la extracción, transformación y carga de datos, para ello se usó Pentaho Data Integration en la limpieza, depuración y carga de datos, esto evitó redundancia de datos innecesaria y facilitó su visualización en los reportes BI.

Considerando los datos obtenidos en la investigación preliminar y verificando los resultados obtenidos al implementar el Data Warehouse, se considera que una implementación de este tipo de tecnologías es de gran ayuda para tener control de la información y mejorar la toma de decisiones gerenciales.

Declaración de Originalidad y Responsabilidad

Yo, José Fernando Esparza Parra, portador de la cédula de ciudadanía y/o pasaporte No. 1711911923, declaro que los resultados obtenidos en el proyecto de titulación y presentados en el informe final, previo a la obtención del título de Magister en Gerencia Informática, son absolutamente originales y personales. En tal virtud, declaro que el contenido, las conclusiones y los efectos legales y académicos que se desprenden del trabajo propuesto, y luego de la redacción de este documento, son y serán de mi sola y exclusiva responsabilidad legal y académica.

José Fernando Esparza Parra
1711911923

A DIOS quien siempre ha sabido guiarme por el camino correcto, a mis padres quienes han sabido formarme hasta llegar a ser una buena persona, a mi amada esposa Carmita, quien ha sido el complemento ideal en mi vida y quien me motivo a retomar mis estudios, a mis adorados hijos Giselle y Mateo, por quienes hoy tengo la fortaleza y motivación de cada día tratar de ser mejor.

A mis maestros, compañeros y amigos, quienes han compartido sus conocimientos y vivencias en este importante camino.

Reconocimientos

Al Mg. Marcelo Balseca por su apoyo incondicional durante toda la maestría, no solo ha sido un excelente maestro, sino también un compañero más, quien me brindó la confianza necesaria durante el proceso educativo y también en la realización del presente trabajo.

A los maestros que colaboraron directamente con la formación académica y al personal administrativo quienes cumplen una labor bastante eficiente.

Resumen

El presente proyecto tiene como objetivo implementar un Data Warehouse en la empresa Tubasec C.A. cuya actividad económica es la fabricación de prefabricados para la construcción y terminado de viviendas. Durante la investigación se logró identificar que el problema se presenta cuando los procesos productivos y de ventas generan gran cantidad de información, misma que es ingresada, almacenada y procesada en varios sistemas independientes, lo que ocasionaba que la información esté fraccionada y la generación de reportes gerenciales se torne en una tarea muy complicada. La implementación del Data Warehouse se realizó en base a la metodología de Hefesto 2.0, dicha metodología permitió delimitar el proyecto gracias a las fases denominadas Análisis de Requerimientos y Análisis de las fuentes para determinar el modelo lógico, con este modelo se procedió con la extracción, transformación y carga de datos, para ello se usó Pentaho Data Integration. Finalmente como resultado de estos procesos, se diseñaron cuadros de mando gerenciales y reportes en Qlick Sense que son de ayuda en la toma de decisiones gerenciales de la empresa. Con los resultados obtenidos en Tubasec C.A., se puede concluir que la implementación de Data Warehouse en empresas que fabrican productos para la construcción es factible y beneficiosa, puesto que ayudan a integrar procesos y departamentos que normalmente están separados, pero que son parte de los costos indirectos de fabricación.

Palabras clave: almacén de datos, datos, reportes

Abstract

This paper aims to implement a data warehouse in Tubasec C.A. that is engaged to manufacture prefabricated and finished construction products. During the investigation the problem was identified, which occurs during manufacturing and sales processes, generating a huge amount of information. Such information is entered, stored and processed in several independent systems; causing incomplete information and developing management reports becomes a complicated task. The Data Warehouse implementation was performed based on the HEFESTO 2.0 methodology, which allowed defining the scope of the project through two stages: Analysis Requirements and Sources Analysis; which are used to determine the logical model. Data extraction, transformation and loading were performed using this model by means of the Pentaho data Integration. Finally as a result of these processes, a management dashboard and reports using "Sense Qlick" were developed, which are very helpful to make any management decisions of the company. It is concluded that the implementation of Data Warehouse in companies that manufacture construction products is feasible and beneficial based on the information gathered from Tubasec C.A., since they help to integrate processes and departments that are normally separated and are part of indirect costs of manufacturing.

Key words: data warehouse, data, reports.

Tabla de Contenidos

Ficha Técnica	iii
Declaración de Originalidad y Responsabilidad.....	iv
Dedicatoria.....	v
Reconocimientos	vi
Resumen	vii
Abstract	viii
Tabla de Contenidos	ix
Lista de Tablas	xii
Lista de Figuras	xiii
CAPITULOS	
1. Introducción	1
1.1. Presentación del trabajo.....	2
1.3. Descripción del documento.....	3
2. Planteamiento de la Propuesta de Trabajo.....	4
2.1. Información técnica básica	4
2.3. Preguntas básicas	5
2.4. Formulación de meta.....	6
2.5. Objetivos.....	6
2.6. Delimitación funcional.....	6
3. Marco Teórico	8
3.1. Definiciones y conceptos	8
3.1.1. <i>Data Warehouse</i>	8
3.1.1.2. Metodología	10
3.1.1.4. Elementos del <i>Data Warehouse</i>	11
3.1.1.4. Esquemas.....	12
3.1.2. <i>Data Warehousing</i>	15

3.1.2.1. OLTP.....	16
3.1.2.2. Load Manager.....	16
3.1.2.3. <i>Data Warehouse Manager</i>	18
3.1.2.4. <i>Query Manager</i>	19
3.1.2.5. Herramientas de consulta y análisis	19
3.1.3. <i>Data Mart</i>	20
3.1.4. ROLAP y MOLAP.....	21
3.2. Estado del Arte	22
4. Metodología.....	24
4.1. Diagnóstico	24
4.2. Método(s) aplicado(s).....	24
4.2.1. Investigación.....	25
4.2.2. Desarrollo	27
4.2.2.1 Análisis de Requerimientos	27
4.2.2.2 Análisis de los OLTP (Online Transaction Processing).....	28
4.2.2.3 Modelo Lógico del Datawarehouse	29
4.2.2.4 Integración de Datos.....	30
5 .Resultados.....	31
5.1. ANÁLISIS DE RESULTADOS	31
5.1.1. Comparación entre herramientas de Extracción, Carga y Depuración de Datos.	31
5.1.2. Determinación herramienta SGBD (Sistema Gestor de Base de Datos).....	35
5.1.3. Comparación entre herramientas BI	37
5.1.2 Desarrollo de la Aplicación	42
5.1.2.1. Análisis de requerimientos	42
5.1.2.2 Análisis de los Orígenes de Datos	45
5.1.2.3 Modelo Lógico de Data Warehouse	49
5.1.2.4 Integración de Datos.....	57
5.2. Evaluación preliminar	90
6 .Conclusiones y Recomendaciones	93

6.1. Conclusiones.....	93
6.2. Recomendaciones.....	93
APÉNDICES	
Apéndice A: Instalacion Postgresql	95
Apéndice B: Instalación Pentaho Suit.....	110
Apéndice C: Instalación de Qlik Sense	114
Apéndice D: Scripts para la extraccion de informacion de las bases de datos.....	117
Referencias	124

Lista de Tablas

1. Matriz de Medición y Análisis de software.....	27
2. Matriz de conformación de Indicadores.....	29
3. Matriz de relación de Datos.....	29
4. Evaluación software de Integración de Datos.....	33
5. Puntuación software de Integración de Datos.....	34
6. Evaluación Motor de Base de Datos.....	36
7. Puntuación Motor de Base de Datos.....	36
8. Evaluación herramientas <i>Business Intelligence</i>	40
9. Puntuación herramientas Business Intelligence.....	41
10. Indicadores de Ventas.....	46
11. Indicadores de Producción	46
12. Indicadores de Proveedores	47
13. Indicadores de Talento Humano.....	48
14. Seguridad y Salud Ocupacional.....	49
15. Matriz de Relación de datos de inventarios	50
16. Matriz de Relación de datos de inventarios.....	51
17. Matriz de Relación de datos de Personal.....	52
18. Matriz de Relación de ventas.....	54
19. Matriz de evaluación entre Data Warehouse vs. Proceso Manual	91
20. Resumen evaluación entre Data Warehouse vs. Proceso Manual.....	91

Lista de Figuras

1. Arquitectura del <i>Data Warehouse</i>	8
2. DW Variante en el tiempo	9
3. DW no Volátil.....	9
4. Metodología Hefesto.....	10
5. Tablas de Dimensiones.....	11
6. Tabla de Hechos.....	12
7. Esquema Estrella.....	13
8. Esquema Copo de Nieve.....	14
9. Esquema constelación	15
10. Cubo tridimensional	15
11. Arquitectura del <i>Data Warehousing</i>	16
12. Procesos de transformación de datos - Codificación	17
13. Procesos de transformación de datos – Medida de Atributos	17
14. Procesos de transformación de datos – Etiquetado	17
15. Procesos de transformación de datos – Fuentes Múltiples.....	18
16. <i>DW Manager</i>	18
17. Query Manager.....	19
18. Herramienta de consulta y Análisis	19
19. Data Mart Top-Down.....	20
20. <i>Data Mart Top-Down</i>	20
21. MOLAP vs ROLAP.....	21
22. Flujo de decisión para.....	25
23. Ejemplo de ilustración de modelo conceptual.....	28
24. Cuadrante Mágico de Gartner Data Integration.....	32
25. Análisis software de Integración de Datos	34
26. Análisis Motor de Base de Datos.....	37
27. Cuadrante Mágico de Gartner BI.....	38
28. Comparación entre Tableau vs Qlikiew	39
29. Análisis herramientas de Business Intelligence.....	41
30. Modelo Lógico Ventas.....	44

31. Modelo Lógico Producción	44
32. Modelo Lógico Talento Humano	44
33. Modelo Compras.....	45
34. Seguridad y Salud Ocupacional.....	45
35. Modelo lógico fact_inventario	51
36. Modelo lógico fact_proveedor	52
37. Modelo lógico fact_empleados	53
38. Modelo lógico fact_ventas	55
39. Modelos Lógico Data Warehouse Tubasec C.A.	56
40. Creación de una nueva transformación	57
41. Entrada de una fuente de datos.....	58
42. Carga de Datos.....	59
43. Conexión a la Base de Datos.....	59
44. Error de conexión	60
45. Pre visualización de resultados	61
46. Filtrado de datos	61
47. Transformación de tipo de datos.....	62
48. Carga de Datos.....	62
49. Creación de la Tabla desde Spoon	63
50. Ejecución Script creación tabla.....	64
51. Mensaje de ejecución de Script.....	64
52. Tabla de resultados de la extracción de datos	65
53. Creación de un Trabajo en Kettle.....	66
54. Objetos para crear un trabajo automático.....	66
55. Programación de las transformaciones.	67
56. Selección de la transformación a programar	68
57. Ejecución del trabajo de forma exitosa	68
58. Errores de tarea inconclusa de estación.....	69
59. Conexión ODBC.....	70
60. Conexión de ODBC exitosa.....	70
61. Aplicación Qlik Sense	71
62. Aplicación creada en Qlik Sense.....	71
63. Añadir datos Qlik Sense.	72

64. Conci3n ODBC en Qlick Sense.....	72
65. Selecci3n Base de Datos Qlik Sense.	73
66. Abrir Gestor de datos Qlik Sense	73
67. Carga de datos Qlik Sense.	74
68. Editar Hoja en Clik Sense.....	74
69. A3adir dimensi3n Qlik Sense.....	75
70. A3adir agregaci3n Qlik Sense.....	75
71. A3adir filtros Qlik Sense.	76
72. Aspecto gr3ficas Qlik Sense.....	77
73. Personal por ciudad y por sexo	77
74. Indicador de Ventas – Valor Facturado.....	78
75. Indicador de Ventas – Expediciones por peso.	79
76. Indicador de Ventas – Porcentaje de Ventas por peso.	80
77. Indicador de Ventas – Porcentaje de Ventas por peso	81
78. Indicador de Ventas – Porcentaje devoluciones	82
79. Indicador de Producci3n – Producci3n e Inventario	84
80. Indicador de Producci3n – Porcentaje de producci3n	85
81. Indicador de Producci3n – Porcentaje de Inventario	86
82. Indicador de Producci3n – Bajas.....	87
83. Indicador de Personal – Horas Extras	88
84. Indicador de Personal – Remuneraci3n.....	89
85. Indicador de Proveedores – Cantidad de compras.....	90
86. An3lisis Data Warehouse vs. Proceso Manual.....	91
87. Descarga PostgreSQL	95
88. Seleccion opci3n download para redireccionamiento de p3gina	96
89. Selecciona del software de acuerdo al sistema operativo.....	96
90. Bienvenida al proceso de Instalaci3n postgresQL.	97
91. Direccionamiento del lugar de alojamiento de Postgresql.....	97
92. Directorio de localizaci3n de los datos de postgres.	98
93. Inicio de la instalacion.....	98
94. Extracci3n de archivos de instalacion de PostgreSQL.....	99
95. Extracci3n de archivos de instalacion de PostgreSQL.....	99
96. Diagrama de la estructura de <i>Business Intelligence</i> y sus herramientas.	100

97. Descarga Java.....	101
98. Pantalla de inicio de instalación del Jdk.....	101

Capítulo 1

Introducción

En éstos días, sobrevivir en el mundo de negocios es una guerra de información y toma de decisiones acertadas, éstas decisiones deben estar basadas en el análisis de los datos más relevantes que la empresa posee, sin embargo una empresa normalmente no siempre tiene organizada la información en un solo sistema y consecuentemente eso implica tener varias bases de datos y plataformas en las que almacenan sus datos. Los *Data Warehouses* fueron propuestos con el fin de responder mejor a las crecientes demandas de los usuarios de toma de decisiones (Elzbieta Malinowski, 2009).

El *Data Warehouse* requiere una arquitectura que comienza al ver el conjunto, para posteriormente ver los detalles. Ciertamente, los detalles son importantes en todo el *Data Warehouse*, pero solamente cuando se ven en un contexto más amplio. (Inmon, 2002).

Para crear una *Data Warehouse* o Almacén de Datos es preciso realizar un estudio de la información que la empresa posea y cómo estos datos están almacenados, sin dejar de lado el análisis respectivo para obtener dicha información. El *Data Warehousing* (DWH), es el encargado de extraer, transformar, consolidar, integrar y centralizar los datos que una organización genera en todos los ámbitos de su actividad diaria (compras, ventas, producción, etc.) y/o información externa relacionada. De esta manera se permite el acceso y exploración de la información requerida, a través de una amplia gama de posibilidades de análisis multivariantes, con el objetivo final de dar soporte al proceso de toma de decisiones estratégico y táctico. (Hefesto, 2010)

La metodología usada para el diseño y construcción del *Data Warehouse* es la propuesta por HEFESTO en su versión 2.0, esta metodología es orientada exclusivamente para la construcción de Almacenes de Datos y se acopló perfectamente al entorno que maneja Tubasec C.A. en todas sus fuentes de datos actuales. HEFESTO está compuesta de cuatro etapas bien definidas que van desde el análisis de requerimientos hasta la integración de datos, lo que permitió que el *Data Warehouse* almacene la información necesaria para generar los cuadros de mando en herramientas BI, útiles para la alta gerencia al momento de toma de decisiones empresariales.

1.1. Presentación del trabajo

El presente trabajo fue desarrollado en Tubasec C.A., una empresa de la ciudad de Riobamba que elabora productos prefabricados de fibrocemento, hormigón y polipropileno. Esta empresa posee gran cantidad de información que es generada en los procesos productivos, almacenamiento, ventas y distribución, dicha información es almacenada en diferentes repositorios de datos que manejan diferente tecnología de almacenamiento y recuperación, lo que ocasiona retardos y problemas a la hora de conocer la situación empresarial a nivel macro. La investigación surge de esta problemática, planteando generar un almacén de datos que contenga toda la información y que además la relacione para generar cuadros de mando con datos actualizados y cotejados.

Los *Data Warehouse (DW)*, también conocidos como almacenes de datos, son hoy en día la herramienta más importante de la inteligencia de negocios, puesto que en ellos se soportan, almacenan y gestionan grandes cantidades de información para ser plasmadas en herramientas de BI (Inteligencia de Negocios).

El diseño del DW se desarrolló en varias etapas, las fases iniciales fueron necesarias para entender los requerimientos de la alta gerencia con respecto a la manera en cómo los datos deben relacionarse y enfocarse con los objetivos de la empresa, para ello se utilizó la entrevista como herramienta de recolección de datos. Una vez obtenido los datos iniciales se analizaron las fuentes de datos, para lograr determinar la factibilidad del proyecto en base a lo solicitado en el análisis de datos. Determinados los requerimientos y analizadas las fuentes de datos, se procedió a la extracción, relación y almacenamiento de la data, mediante el uso de herramientas ETL(Extracción, transformación y carga), muy importantes a la hora de almacenar información depurada, la herramienta que se utilizó fue Pentaho Data Integration, para luego almacenar la información en una base de datos PostgreSQL.

Finalmente se implementó cuadros de control en Qlik Sense, herramienta de BI gratuita con alta flexibilidad y buen desempeño, éstos cuadros de control son el producto final que la alta gerencia requiere para ayudarse en la toma de decisiones empresariales, mismas que pueden influir en el rumbo que la organización puede tomar, basándose en datos de la situación actual y proyectándose al futuro en corto, mediano o largo plazo.

Éste tipo de soluciones podrán ser implementadas en empresas de similares características, puesto que el flujo de la información es característico y los resultados que éstas esperan obtener

están en la misma línea, por su puesto se debe considerar particularidades que cada organización posee.

1.3. Descripción del documento

El documento está compuesto por seis capítulos, el primer capítulo es la parte introductoria donde se pretende informar al lector del sentido del proyecto, sus orígenes y su razón de ser.

El capítulo 2 se muestra en planteamiento de la propuesta que se implementa en el proyecto, el Capítulo 3 contiene la parte teórica, concretamente identifica los términos y conceptos útiles que den soporte a lo que se plantea hacer, esta descripción está claramente redactada, sin ahondar en términos técnicos que podrían confundir, de hecho se utilizaron términos que cualquier lector no experimentado podrá entender.

En el Capítulo 4 presenta la metodología que se utilizó para desarrollar el proyecto, detallado cómo se logró avanzar en fases para llegar al objetivo final, mientras que el Capítulo 5 muestra los resultados obtenidos y metas cumplidas ya una vez implementado el proyecto, además de los resultados obtenidos con esta implementación.

Finalmente en el capítulo 6, están las conclusiones y recomendaciones, para luego finalizar con los apéndices en los que se describe las instalaciones de herramientas y configuraciones necesarias para la implementación del proyecto.

Capítulo 2

Planteamiento de la Propuesta de Trabajo

2.1. Información técnica básica

Tema: Implementación de un *Data Warehouse* Corporativo para la Integración de Datos de las Áreas de la Organización para la toma de Decisiones en la empresa TUBASEC C.A.

Tipo de Trabajo: Proyecto de Investigación y Desarrollo.

Clasificación técnica del trabajo: Aplicación.

Líneas de Investigación, Innovación y Desarrollo

Principal: Sistemas de Información y/o Nuevas Tecnologías de la Información y Comunicación y sus aplicaciones.

2.2. Descripción del problema

Tubasec C.A. es una empresa que está dedicada a la fabricación de materiales de construcción, para elaborar sus productos cuenta con diferentes líneas de producción y departamentos/procesos que son directa o indirectamente involucrados. Éstos procesos internos generaran información, que sirven a su vez para generar reportes y/o indicadores de cumplimiento, evaluación, eficiencia, eficacia y gestión; mismos que son reportados a la gerencia en períodos definidos. Normalmente esta información es recolectada de los sistemas satélites que interactúan con los usuarios, para ser procesada y almacenada en hojas electrónicas a manera de cuadros estadísticos y gráficos; estos datos consolidados son finalmente enviados por correo a la gerencia, para el respectivo análisis y toma de decisiones. Un caso que ejemplifica lo mencionado es el departamento de ventas, éste genera reportes semanales, mensuales y trimestrales de ventas por cada producto, este proceso se lo realiza de manera manual y lleva demasiado tiempo con los respectivos errores que pueden darse al digitar datos numéricos.

Casos como éstos han provocado que la información llegue a destiempo a la gerencia de la empresa y la toma de decisiones no sea oportuna, hasta el punto de llegar reportes con retrasos de hasta 30 días.

Hay que recordar que en el mercado hay varias empresas que ofertan productos prefabricados de fibrocemento, polipropileno y hormigón, por ello es importante tener a tiempo la información adecuada de manera que se pueda presentar cambios en políticas de marketing, descuentos, promociones, etc., y así conseguir llegar al cliente con mejores ofertas para ganar su confianza y fidelidad.

Éstos hechos hacen que sea justificable trabajar en un proyecto para la creación e implementación de *Data Warehouse*, debido a que al tener un almacén de datos en el que se incorpore toda la información vital e importante de la empresa, la tarea de recuperación de información contenida en este *Data Warehouse* será mucho más rápida, sencilla y oportuna. Con este proyecto se pretende colaborar, para que la gerencia tenga toda la información necesaria y sus decisiones sean basadas en datos e información real, con beneficio para toda la empresa.

2.3. Preguntas básicas

¿Cómo aparece el problema que se pretende solucionar?

El problema surge por la gran cantidad de información segmentada y almacenada en diferentes repositorios, esta información es generada por las diferentes áreas y procesos que conforman la empresa Tubasec C.A.

¿Por qué se origina?

Se origina porque cada área que conforma la empresa, maneja información propia y no existe un sistema que maneje todas las áreas y ámbitos, segmentando de esta manera la información, lo que ocasiona que la alta gerencia de la empresa, desconozca muchos resultados parciales y/o finales que se pueden generar.

¿Qué lo origina?

EL divorcio entre los sistemas satélites que se manejan en las diferentes áreas, por ser tecnologías diferentes muchas veces incompatibles y en otros manejos manuales, además de las limitaciones que cada una posee al momento de generar reporte gerenciales.

2.4. Formulación de meta

Implementar un *Data Warehouse* Corporativo, que contenga información consistente y completa de las diferentes áreas que compone la empresa, de manera que ésta sea útil para la toma de decisiones de la alta gerencia y mandos medios en la empresa Tubasec C.A.

2.5. Objetivos

Objetivo general

Implementar un *Data Warehouse* Corporativo para la Integración de datos de las Áreas de la Organización para la Toma de Decisiones en la empresa Tubasec C.A.

Objetivos específicos

1. Realizar un estudio comparativo entre herramientas con licencia y de libre distribución, para la implementación más adecuada del *Data Warehouse*, considerando las necesidades y requerimientos de la empresa Tubasec C.A.
2. Analizar la información a ser almacenada en el *Data Warehouse*, enfocándose en criterios gerenciales, que facilite el posterior diseño de indicadores de cumplimiento, evaluación, eficiencia, eficacia, gestión y cuadros de mando gerenciales para Tubasec C.A.
3. Desarrollar la estructura del *Data Warehouse* para la empresa Tubasec C.A.
4. Implementar el *Data Warehouse* en la empresa Tubasec C.A.
5. Validar los resultados del *Data Warehouse* implementado en la empresa Tubasec C.A.

2.6. Delimitación funcional

Pregunta 1. ¿Qué será capaz de hacer el producto final del trabajo de titulación?

- Integrará los datos que actualmente está fraccionados y reflejará resultados que se manejan de forma manual en hojas electrónicas.
- Podrá mostrar información consolidada de las diferentes áreas de la empresa, misma que servirá de herramienta para de la toma de decisiones gerenciales, esta información será visualizada en cuadros de resumen, gráficos estadísticos y cuadros de mando para controlar el flujo de información, gastos, inversiones, avance de proyectos, muestras de pruebas de nuevos productos, etc.
- Tendrá información centralizada disponible en un Datawarehouse lo que facilitará las consultas y visualizaciones de datos mediante interfaces web desarrolladas para los usuarios.

- Se manejará perfiles de usuario para que la información pueda ser visualizada por el nivel de acceso que los usuarios sean permitidos, es decir que la información puede ser vista o no, eso dependerá se sus privilegios de acceso.

Pregunta 2. ¿Qué no será capaz de hacer el producto final del trabajo de titulación?

- El sistema no podrá cambiar datos en los sistemas de origen, solo lee la información.
- El sistema mostrará la información que encuentra en los sistemas de origen, no valida si los datos ingresados previamente tienen información correcta.
- El sistemas no sacará respaldos automáticos de las bases de datos o cubos generados

Capítulo 3

Marco Teórico

3.1. Definiciones y conceptos

3.1.1. Data Warehouse

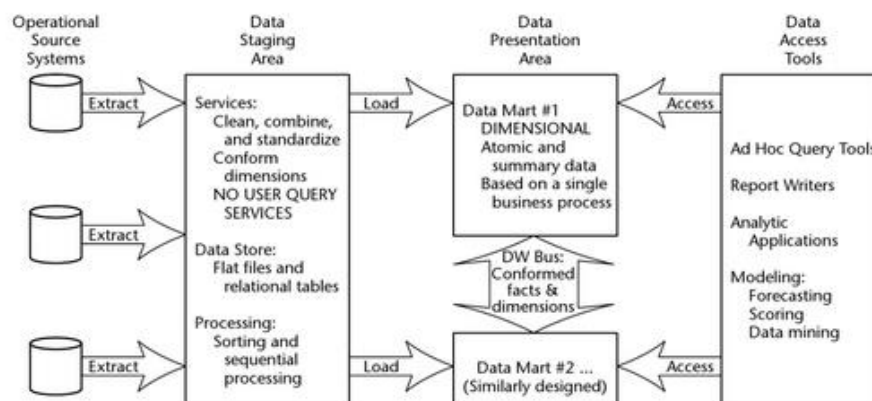
Data Warehouse o Almacén de Datos, es un repositorio que integra información de diferentes fuentes, que no necesariamente son del mismo tipo y que además pueden tener diferente tecnología.

“Un *Data Warehouse* es una colección de datos orientada al negocio, integrada, variante en el tiempo y no volátil para el soporte del proceso de toma de decisiones de la Gerencia” (Inmon, 2002).

Al analizar lo descrito por Inmon, se puede afirmar que esta definición abarca realmente la razón de ser de un *Data Warehouse* y sus principales propiedades y bondades, a lo que se puede complementar al decir que un *Data Warehouse* además es libre de las fuentes u orígenes de los datos que van a ser almacenados y de sus arquitecturas.

Un *Data Warehouse* entonces tendría una arquitectura cómo la mostrada a continuación:

Figura 1. Arquitectura del *Data Warehouse*



Fuente: (Kimball, The Data Warehouse Toolkit, 2002)

Características

Un *Data Warehouse* según (Bernabeu, 2010) posee las siguientes características:

Orientación al Negocio, al interpretar esta característica se puede indicar que la orientación al negocio es cuando *el Data Warehouse* está alineado a los objetivos de la empresa, de modo que esta se convierta en una herramienta fundamental para ayudar a conseguir cumplir dichos objetivos.

Integrada, el autor se refiere cómo integración a la validación de datos obtenidos de los *queries* o consultas, previos a su almacenamiento. En el mercado existe varias herramientas que ayudan en esta tarea conocida cómo ETL (Extracción, transformación y carga), estas herramientas son las encargadas de la depuración de datos obtenidos en la extracción, para su posterior almacenamiento, de manera que esta data sea limpia y se pueda evitar de esta manera la redundancia de información.

Variante en el Tiempo, los datos almacenados en el DW (*Data Warehouse*) deben tener asociada la variable tiempo, esto con el fin de poder navegar en datos tanto actuales como históricos, logrando de esta manera tener información que ayude a generar proyecciones a futuro en base a comportamientos históricos.

Figura 2. DW Variante en el tiempo



Fuente: (Hefesto, 2010)

No Volátil, implica que los datos sean solo para consulta es decir que no se podrán modificar, mucho menos hacer nuevas inserciones, esa parte la maneja directamente los sistemas transaccionales, el DW solo podrá hacer consulta puesto que la información no cambiará, es decir permanece estable.

Figura 3. DW no Volátil



Fuente: (Hefesto, 2010)

3.1.1.2. Metodología

“Existen muchas metodologías de diseño y construcción de DW. Cada fabricante de software de inteligencia de negocios busca imponer una metodología con sus productos.” (Rivadera, 2010)

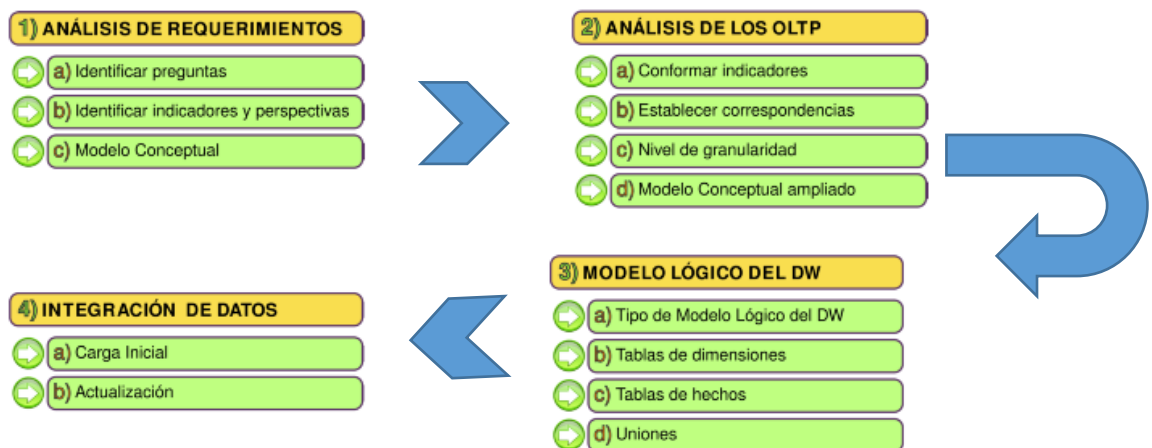
De hecho existen metodologías solo aplicables a un tipo de software, por ejemplo al hablar de herramientas de inteligencia de negocios, Oracle es una de las herramientas que maneja su propio estándar, un caso también aislado es Qlik view, ya que su manejo de cubos es un esquema completamente diferente al estándar de *Data Warehouse*. Para el caso de esta implementación se busca una metodología estándar que sea multiplataforma y posea flexibilidad a la hora de integrar datos de diversos orígenes

“Desde el punto de vista arquitectónico, la mayor diferencia entre los dos autores es el sentido de la construcción del DW, comenzar por los *Data marts* o ascendente (*Bottom-up, Kimball*) o comenzar con todo el DW desde el principio, o descendente (*Top- Down, Inmon*).” (Rivadera, 2010), sin embargo, (Hefesto, 2010) , menciona que la metodología que se debe seguir para lograr tener un *Data Warehouse* consistente debería constar de 4 fases principales

Si se considera usar la metodología *Top-Down*, se debería notar que la implementación podría ser más extensa y que no necesariamente es la más óptima, puesto que al menos en el caso de la implementación que se pretende realizar en Tubasec, se tendría que generar *Data marts* para cada área interna y en el caso de esta empresa objeto de estudio, no es necesario hacerlo.

Al considerar lo expuesto, optó por utilizar la metodología de Hefesto, en la que divide el proceso en cuatro etapas principales, que se muestran a continuación.

Figura 4. Metodología Hefesto



Fuente: (Hefesto, 2010)

Desde aquí en adelante esta será la metodología que se empleará por ser más orientada al negocio y específica para los almacenes de datos.

3.1.1.4. Elementos del *Data Warehouse*

Un DW tiene principalmente dos elementos que son las tablas (al menos una) de hechos y las tablas de dimensiones.

- **Tablas de dimensiones**

Según lo mencionado por Kimball en su libro Guía Definitiva para el modelado Dimensional, (Kimball & Ross, The Definitive Guide to Dimensional Modeling, 2013), se puede definir una tabla de dimensiones como el contexto de la tabla de hechos, éstas están íntimamente asociadas, debido a que la tabla de dimensiones contiene los datos que la tabla de hechos hace referencia en sus filtros.

Al ampliar un poco más a este concepto, se debe mencionar que una tabla de dimensiones posee también una estructura lógica y los datos que poseen son cuantitativos y sirven para mostrar lo filtrado en la tabla de hechos, relacionada con la tabla Dimensión de tiempo.

Figura 5. Tablas de Dimensiones

GEOGRAFIA	PRODUCTOS	CLIENTES	FECHAS
↗id_Geografía	↗id_Producto	↗id_Cliente	↗id_Fecha
País	Rubro	NombreCliente	Año
Provincia	Tipo		Trimestre
Ciudad	NombreProducto		Mes
Barrio			Día

Fuente: (Bernabeu, 2010)

La tabla Dimensión de tiempo, esta tabla es obligatoria en un DW, puesto que permite navegar en los datos con referencia al tiempo, ésta tabla de dimensiones de tiempo normalmente va a tener una granularidad que va a permitir filtrar la información por años, semestres, trimestres, meses, semanas y/o días, lo que permite al usuario tener la opción de realizar comparaciones, análisis y presentación de resultados según las necesidades del dato que se pretende mostrar.

- **Tablas de Hechos**

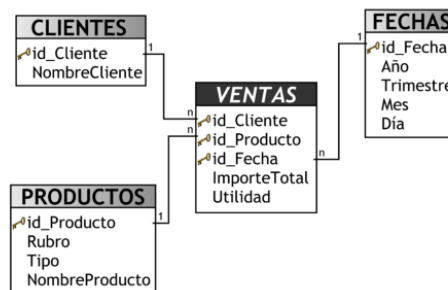
(Inmon, 2002) Menciona que una tabla de hechos es el centro de una estrella, en la que se encuentran los datos que tienen muchas ocurrencias.

Una tabla de hechos no solo tiene las relaciones con otras tablas, además de estas relaciones, una tabla de hechos puede contener información sobre todo cuantitativa, es decir campos con valores calculados que servirán para mostrar resultados de manera ágil.

Para (Fasel, 2014), Una tabla de hechos se encuentra en el centro del modelo del almacén de datos. La relación con tablas de dimensiones se realiza con la ayuda de un atributo clave.

En resumen, las tablas de hechos normalmente son las que permitirán mostrar resultados y los que llevan la relación con las tablas de dimensiones, estas están compuestas de las claves primarias de las tablas de dimensiones y datos cuantitativos útiles para la creación de indicadores y reportes.

Figura 6. Tabla de Hechos



Fuente: (Bernabeu, 2010)

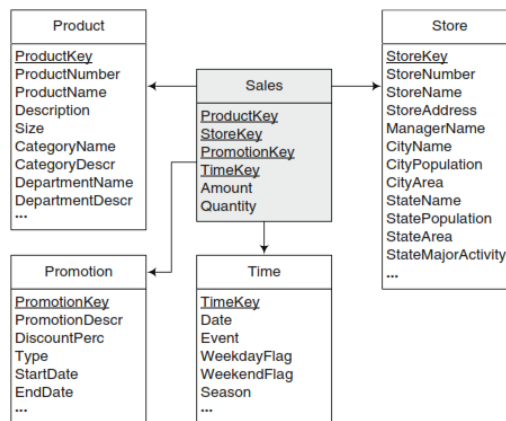
3.1.1.4. Esquemas

Existen varios tipos de modelamiento de DW, se puede aplicar el que más convenga en base a la información y relación que los datos puedan tener, los modelos usados son el esquema estrella, esquema copo de nieve y esquema constelación, que no es más que una combinación o fusión entre el esquema estrella y el esquema copo de nieve, éste es utilizado sobre todo en modelos que requieren un mayor detalle de información.

3.1.1.4.1. Esquema Estrella

Al tomar cómo referencia a (Vaisman & Zimanyi, 2014), se conoce que el esquema estrella, es aquel compuesto de una tabla de hechos central a la que están relacionados varias tablas de dimensiones y donde la tabla de hechos es la que contendrá la información importante en lo referente a datos cuantitativos, mientras que las tablas de dimensiones contendrán únicamente datos complementarios.

Figura 7. Esquema Estrella.



Fuente: (Vaisman & Zimanyi, 2014)

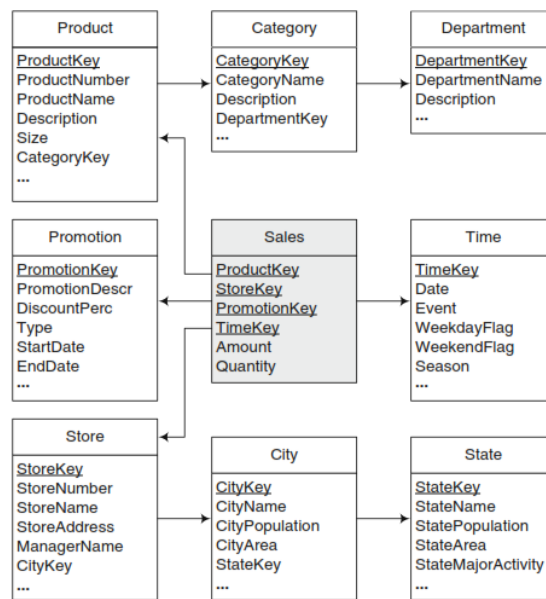
En esquemas de Base de datos relaciones es importante que esté normalizado, empero en los DW la desnormalización es importante, para evitar que las consultas sean complicadas, claro aumenta la redundancia de datos, pero las búsquedas son más eficientes.

3.1.1.4.2. Esquema Copo de Nieve

Basados en (Vaisman & Zimanyi, 2014) , este esquema permite que una tabla de dimensiones puede estar relacionada con otra tabla de dimensiones sin que esta segunda esté directamente relacionada con la tabla de hechos, es decir es una tabla secundaria en el orden jerárquico.

Otra característica importante es que aquí sus tablas están normalizadas, lo cual es útil al momento de explorar datos a más detalle. A continuación se muestra gráficamente este esquema.

Figura 8. Esquema Copo de Nieve



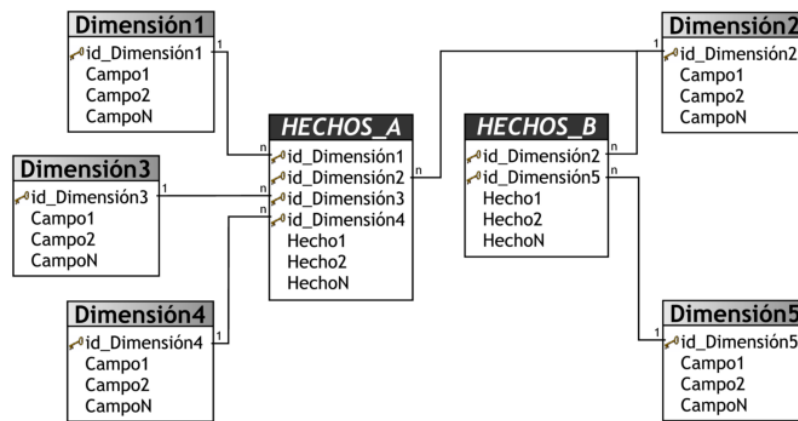
Fuente: (Vaisman & Zimanyi, 2014)

Este esquema es utilizado normalmente en casos donde sus tablas secundarias poseen gran cantidad de registros, lo que conllevaría a crecer mucho la tabla de hecho, si estas tuviesen relación directa

3.1.1.4.3. Esquema Constelación

Este esquema se puede llamar cómo la unión de varios esquemas estrella según lo menciona (Bernabeu, 2010), esto permite que se puedan tener más de una tabla de hechos, además permite la reutilización de tablas de dimensiones, este esquema es útil en casos de requerir información más detallada de una determinada categoría o dimensión, cómo es el caso de datos de direcciones en donde la normalización de las tablas es la mejor manera de tener almacenada la información. A continuación se muestra la gráfica del mencionado esquema, en donde claramente se nota que existe más de una tabla de hechos y las tablas de dimensiones pueden estar atadas a más de una tabla de hechos para lograr los resultados esperados.

Figura 9. Esquema constelación

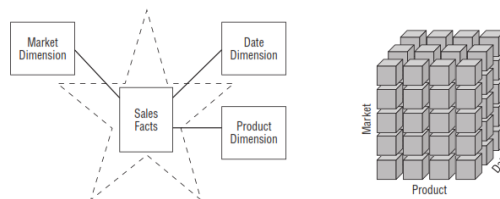


Fuente: (Bernabeu, 2010)

Cubos Multidimensionales

(Bernabeu, 2010) En su libro de la metodología de Hefesto, menciona que los cubos multidimensionales son representación de datos que están en el DW, a continuación se puede ver un esquema de tres dimensiones (Figura 10), en la que se representa un cubo de tres dimensiones en un esquema tipo estrella, pero un cubo puede tener más dimensiones. Un cubo puede tener n dimensiones cómo sean necesarias, sin embargo es importante considerar que mientras más dimensiones se tengan, los filtros pueden ser en ocasiones más perjudiciales que beneficiosos.

Figura 10. Cubo tridimensional



Fuente: (Kimball & Ross, The Definitive Guide to Dimensional Modeling, 2013)

3.1.2. Data Warehousing

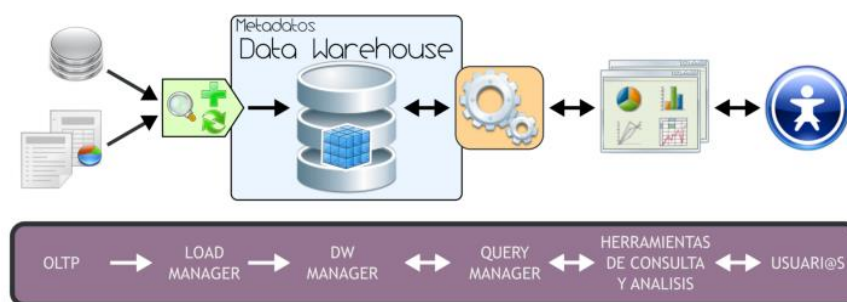
“El *Data Warehousing* posibilita la extracción de datos de sistemas operacionales y fuentes externas, permite la integración y homogeneización de los datos de toda la empresa, provee información que ha sido transformada y sumariada, para que ayude en el proceso de

toma de decisiones estratégicas y tácticas”. “Pero para que el *Data Warehousing* pueda cumplir con sus objetivos, es necesario que la información que se extrae, transforma y consolida, sea almacenada de manera centralizada en una base de datos con estructura multidimensional denominada *Data Warehouse* (DW).” (Bernabeu, 2010).

Entonces, de acuerdo a lo mencionado en el párrafo anterior, el *Data Warehousing* es el proceso de crear un *Data Warehouse*, es decir el *Data Warehouse* es el resultado final y el *Data Warehousing* es la acción o acciones que deben realizarse para obtener el producto final.

Donde cada una es complemento de la otra, es decir se debe cumplir secuencialmente todas y cada una de estas para lograr cumplir con el objetivo. La arquitectura del *Data Warehousing* se muestra a continuación

Figura 11. Arquitectura del *Data Warehousing*



Fuente: (Hefesto, 2010)

3.1.2.1. OLTP

Al citar lo descrito en (Bernabeu, 2010), OLTP (*On Line Transaction Processing*) son todas aquellas fuentes de información de donde se espera extraer datos, es decir todas aquellas fuentes en que se procesan la información a manera transaccional, ejemplos de estas fuentes son Base de datos, documentos de texto, hojas de cálculos, etc.

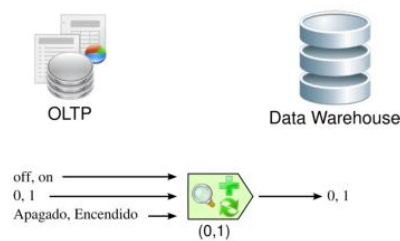
3.1.2.2. Load Manager

La Metodología de (Hefesto, 2010), describe que el Load Manager sirve para extraer la información de las fuentes de orígenes de datos y almacenarlo en el *Data Warehouse*, para ello se utiliza un proceso conocido como ETL (*Extracción, Transformación y Carga/Extraction,*

Transformation and Load) que es un conjunto de métodos y/o técnicas que ayudarán a que la información que se almacene en el DW sea consistente y esté bien estructurada.

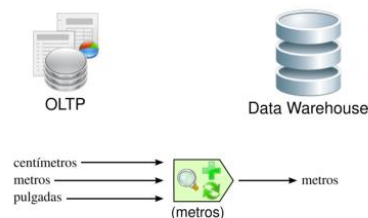
Por ejemplo en un ETL es necesario transformar la información que viene de diferentes fuentes es decir, si un dato que tiene el mismo sentido o contenido está almacenado de diferente manera, entonces en éste proceso ETL se debe que estandarizarlo y así poder comprender un solo tipo de datos, codificación o etiqueta en el DW, a continuación se muestra gráficamente varios procesos de cambio que podrían presentarse al cargar los datos desde el OLTP hacia el *Data Warehouse*.

Figura 12. Procesos de transformación de datos - Codificación



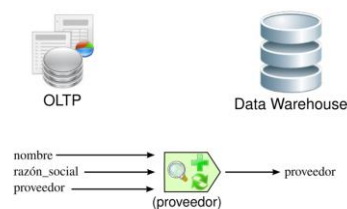
Fuente: (Hefesto, 2010)

Figura 13. Procesos de transformación de datos – Medida de Atributos



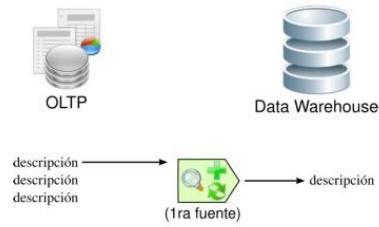
Fuente: (Hefesto, 2010)

Figura 14. Procesos de transformación de datos – Etiquetado



Fuente: (Hefesto, 2010)

Figura 15. Procesos de transformación de datos – Fuentes Múltiples



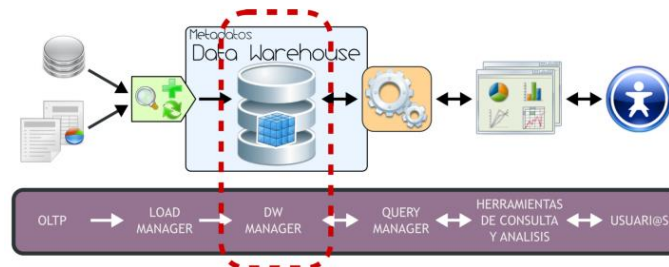
Fuente: (Hefesto, 2010)

Una vez depurada la información, esta finalmente debe ser almacenada en la base de Datos del DW y cronogramar las cargas automáticas de ser el caso.

3.1.2.3. Data Warehouse Manager

Al seguir la metodología de Hefesto y acoger lo mencionado por (Bernabeu, 2010), el *Data Warehouse Manager* es esencialmente el sistema gestor de Base de Datos que se encarga de almacenar y gestionar datos multidimensionales que son extraídos de los OLTP, además se encarga de soportar toda la arquitectura necesaria para organizar los datos en tablas de hechos y dimensiones, de manera que éstos puedan ser recuperados fácilmente por el usuario que va a realizar peticiones a través de aplicaciones BI.

Figura 16. DW Manager

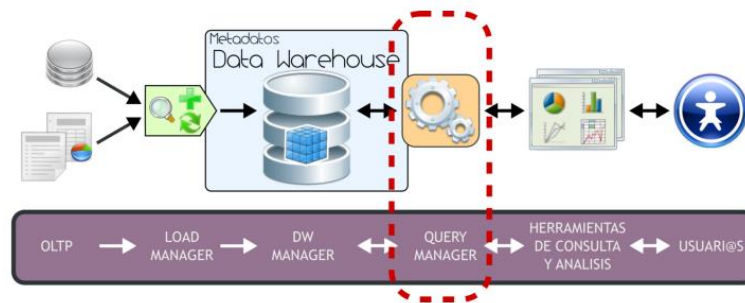


Fuente: (Hefesto, 2010)

3.1.2.4. Query Manager

“Query Manager recibe las consultas de l@s usuari@s, las aplica a la estructura de datos correspondiente (cubo multidimensional, *Business Models*, etc.) y devuelve los resultados obtenidos.” (Bernabeu, 2010), con cualquier técnica de consulta y recuperación de datos.

Figura 17. Query Manager

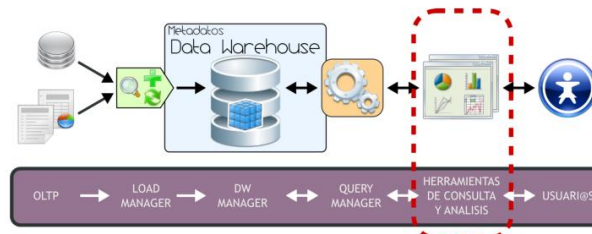


Fuente: (Hefesto, 2010)

3.1.2.5. Herramientas de consulta y análisis

Basándonos en lo que indica (Bernabeu, 2010), este tipo de herramientas se utilizan en las instancias finales del *Data Warehousing*, tienen como finalidad esencial consumir los datos y metadatos almacenados en el *Data Warehouse*, hay varias formas de conexión para que estas herramientas accedan a los datos, por ejemplo ODBC, JDBC, etc. A este tipo de herramientas son las conocidas como herramientas de BI, puesto que estas pueden no solo consultar, sino además estas pueden ya mostrar la información a manera de cuadros estadísticos, gráficos, proyecciones, predicciones. Etc.

Figura 18. Herramienta de consulta y Análisis



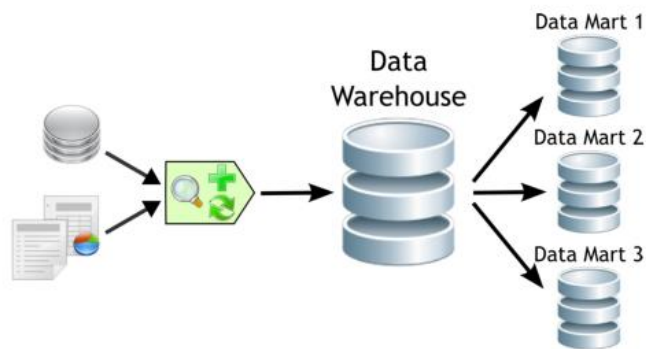
Fuente: (Hefesto, 2010)

3.1.3. Data Mart

Según los conceptos de (Ponniah, 2010), un *Data Mart* es una sección o parte de un Data Warehouse, mientras que el *Data Warehouse* unifica, almacena y procesa la información de toda la empresa, un *Data Mart* es más específico, es decir solo abarca parte de la empresa, por ejemplo, el área de finanzas o el área de ventas, o el área de Talento Humano, etc. Pero con esto no implica que abarca departamentos, sino más bien procesos, es decir los que abarca o conlleva dicho proceso.

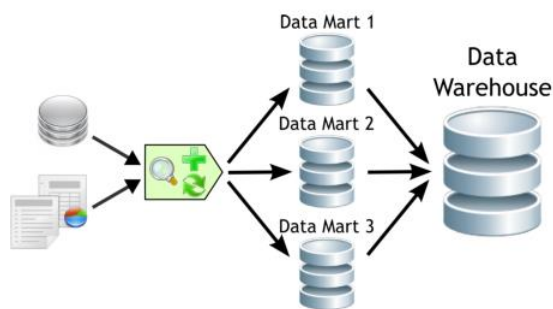
Los *Data Marts*, pueden ser parte importante en la construcción de un DW cómo el caso Bottom-up o podrían segmentarse luego de construido el DW con es el caso de los *Data Mart Top-Down*

Figura 19. Data Mart Top-Down



Fuente: (Hefesto, 2010)

Figura 20. Data Mart Top-Down



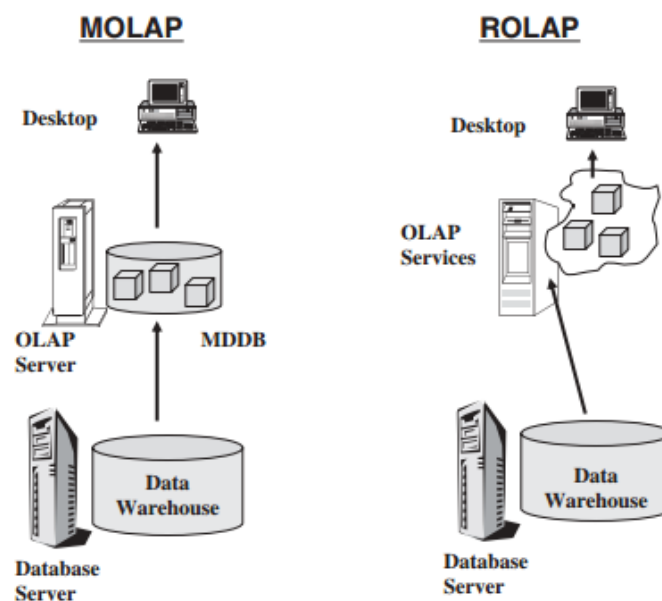
Fuente: (Hefesto, 2010)

3.1.4. ROLAP y MOLAP

De acuerdo con (Ponniah, 2010), ROLAP (*Relational On Line Analytic Processing*) es un tipo de implementación de *Data Warehouse* que forma cubos en línea, es decir no almacena la información en repositorios temporales, sino que los cubos, tablas de hechos y dimensiones, se forman al momento de la consulta, con esto se logra poca intervención y dependencia de los usuarios en mantenimiento y desarrollo, pero al mismo tiempo causa inconvenientes al depender directamente de los OLTP que dan origen a los datos, además de la tardanza en búsquedas y el hecho de consumir recursos de éstos sistemas transaccionales.

MOLAP (*Multidimensional On Line Analytic Processing*) a diferencia del ROLAP, éste trabaja en dos etapas, la primera es procesar y cargar los datos en repositorios que servirán para luego ser usados y consultados, sin tener dependencia de los OLTP, puesto que se realiza la carga una sola vez y no necesita realizar las transacciones en línea, con esto se logra mayor eficiencia en consultas y no se consume constantemente recursos de los sistemas que lo alimentan, pero por el contrario conlleva que se pierda flexibilidad, puesto que los datos no son en línea, sino que corresponden a la última carga realizada y si los sistemas transaccionales realizan modificaciones sobre cantidades, valores u otros, el DW no tendrá manera de conocerlo hasta que no se cargue nuevamente la información.

Figura 21. MOLAP vs ROLAP



Fuente: (Ponniah, 2010)

3.2. Estado del Arte

Al hablar de Data Warehouse o Almacén de Datos, no se debe imaginar que es solo una gran base de datos, actualmente un *Data Warehouse* es parte de toda una inteligencia de negocios, muchas veces se confunde o relaciona con un software que genera reportes y visualiza información, pero esto no es del todo cierto puesto que para llegar al resultado final que es otorgar la información necesaria para ser utilizado en herramientas de BI, es necesario realizar todo un proceso que incluye a muchas personas y áreas de una organización, tal como se menciona en el libro "*The profit impact of business intelligence*" (Williams, 2007), donde se indica que Inteligencia de negocios no es un simple producto, metodología o tecnología, a pesar que existen muchas herramientas muy útiles en todos éstos procesos, estas por si mismas no se manejan solas y no pueden ser implementadas cómo un estándar o producto semi elaborado que esté listo para ser usado en cualquier empresa, es necesario realizar todo el proceso para llegar a tener los resultados esperados.

El autor indica claramente su posición con respecto a los sistemas de inteligencia de negocios y el proceso que se debe cumplir con respecto a una implementación, pero otro punto que se debe considerar al momento de diseñar y desarrollar un Data Warehouse es la cantidad de información que una empresa u organización posee, este factor es determinante a la hora de escoger el tipo de implementación ideal para lograr un efecto óptimo.

Al investigar casos de implementación a nivel regional, se puede tomar en cuenta a la Escuela Superior Politécnica de Chimborazo, dónde se implementó el año pasado un modelo de Inteligencia de negocios basado en cubos de datos, el programa que utilizaron es Qlikview, con muy buenos resultados. Si bien es cierto es necesario una inversión inicial en servidores, el mantenimiento posterior es bastante liviano, además para la obtención de datos utilizan fuentes ODBC, lo que hacen que sea compatible con cualquier base de datos y ésta tarea se sea muy sencilla, también es posible obtener información de archivos de texto plano y hojas electrónicas, esta herramienta es bastante útil porque facilita la generación de cubos de datos que son más complejas de implementar en otra herramientas.

Un caso manejado a nivel nacional con Qlickview es el Grupo TVCable, esta empresa tiene alto nivel de transacciones y bases de datos con millones de registros, la parte interesante de esta empresa fue la manera de obtener información, ya que por la gran cantidad de datos manejados no se podía obtener información el línea, se optó por realizar recolección de datos programados a partir de la media noche, hasta las 6 de la mañana, se logra así tener la información en indicadores al día siguiente para las altas gerenticas y vicepresidencias de esta corporación, los cubos de datos se segmentaban por períodos de tiempo y gracias a estas adecuaciones, las

consultas de los cuadros de mando fueron bastante rápidas y eficientes. Parte de esta técnica podría aplicarse en Tubasec, al menos en la segmentación de períodos de tiempo, ya que si bien es cierto no se maneja la gran cantidad de datos de TVCable, pero si se maneja períodos de tiempo, sobre todo en la creación de nuevos productos.

Un caso bastante interesante a nivel internacional es la transnacional BAMESA quienes están posicionados en siete países de Europa. Al implementar un Datawarehouse lograron integrar datos de todas sus sucursales que hasta antes de la implantación de su proyecto de almacén de datos, tenía segmentados los datos por país y hasta por ciudad dentro de cada país, según sea el caso, esto debido al volumen de datos que cada sucursal generaba, al implementar el Datawarehouse, lograron reducir el volumen de datos, tiempos de respuesta y aprovecharon los recursos que se manejaban tanto a nivel tecnológico cómo en talento humano.

El presente proyecto en su parte aplicativa está orientado a utilizar todos los recursos con los que cuenta Tubasec C.A., que a diferencia de las empresas citadas, no cuenta con una gran infraestructura ni personal dedicado a tiempo completo a esta actividad, por lo que se implementará una solución que además de ser funcional, no requerirá de hardware adicional, puesto que se reutilizará el existente mediante virtualización de servidores.

La parte investigativa del proyecto está enfocada en dos aristas principales, la primera es la generación de indicadores para el manejo gerencial, es decir, analizar e investigar la manera de relacionar los datos que se obtendrán de las diferentes fuentes y plasmarlos en los cuadros de mando que desplieguen datos estadísticos e indicadores que la alta gerencia requiere para tener una visión global.

La segunda arista consiste en investigar y determinar herramientas útiles que puedan sincronizarse y complementarse para dar solución a la problemática de la empresa objeto de estudio, esto no se observó en los casos de estudio analizados, puesto que dichos ejemplos, se utiliza una sola herramienta para todo el proceso de *Data Warehousing* y *Business Intelligence*.

Una vez implementado este proyecto, la alta gerencia tendrá un herramienta importante que será una guía para la toma acertada de decisiones empresariales, basados en datos actualizados, consistentes y relacionados entre sí, que muestren una radiografía de la empresa en todos sus ámbitos, para destacar los puntos neurálgicos y evitar complicaciones futuras en la organización con posibles pérdidas económicas.

Capítulo 4

Metodología

4.1. Diagnóstico

Para la obtención de información necesaria para la implementación del *Data Warehouse*, se utilizó investigación empírica, el instrumento aplicado fue la entrevista, misma que se enfocó en los directivos de la empresa y líderes departamentales.

Una vez aplicadas las entrevistas y analizados los resultados, se encontraron problemas principalmente en temas como:

- Deficiente manejo de indicadores económicos, producción y ventas.
- Retardo en entrega de información necesaria para la toma de decisiones
- Información entre departamentos y procesos no relacionados que impide un análisis de situación real de la empresa.
- Diferentes orígenes de información que dificultan el trabajo de obtención de información.

4.2. Método(s) aplicado(s)

En el presente proyecto se aplicó métodos empíricos de investigación, las herramientas utilizadas fueron la entrevista, observación y experimentación.

Las entrevistas fueron utilizadas como instrumento de obtención de información inicial, para conocer principalmente a donde se pretende llegar y que se pretende solucionar al implementar el *Data Warehouse*. El tipo de entrevista utilizada fue una de tipo semi estructurada, esto con el fin de conocer de manera abierta la problemática con la que cada entrevistado cuenta y lograr determinar similitudes y particularidades que serán de utilidad en la estructuración del *Data Warehouse*.

En la investigación se aplicó también la observación participante, esta método de recolección de datos fue muy útil a la hora de conocer el o los procedimientos utilizados por los encargados de obtener y procesar la información necesaria para la elaboración de reportes gerenciales, éste método ayudó también para medir los tiempos que los usuarios tardan para generar dichos reportes. Estos datos fueron importantes para cuantificar y medir resultados que el *Data Warehouse* puede optimizar con su implementación.

4.2.1. Investigación

En el presente trabajo al momento de escoger las herramientas más adecuadas para la implementación del Data Warehouse, se aplicó el método de investigación experimental, esto debido a las diferentes pruebas que se realizaron al momento de instalar demos y probar la funcionalidad de cada una de ellas, hasta lograr determinar la más adecuada para implementarla finalmente en el proyecto.

Para lograr escoger las herramientas adecuadas, se planteó un flujo de toma de decisión que dio paso al diseño la matriz resumen con el análisis que permito tomar una decisión adecuada.

Figura 22. Flujo de decisión para



Fuente: elaboración propia

La búsqueda de herramientas está basada en parámetros de medición a nivel internacional, con esto se logra reducir en gran manera la búsqueda entre herramientas que no están entre la mejores, un referente importante es el nombrado cuadro mágico de Gartner, al utilizar esta referencia, se logró determinar las herramientas más utilizadas y cotizadas a nivel mundial.

Una vez escogidas dos herramientas para el análisis de factibilidad y costos fue necesario tener parámetros de evaluación, por lo que el análisis será basado en una matriz planteada en el modelo OSI/IEC 9126, este permite realizar una evaluación objetiva de cualquier tipo de software, los pilares principales de evaluación son:

Funcionalidad: Capacidad del software para cumplir lo que se espera de si, dentro de este se encuentran varias subcategorías cómo:

Idoneidad: Indica si es programa hace lo que se requiere que haga

Exactitud: Los resultados son los que se espera que sean

Interoperabilidad: Si se puede conectar con otros sistemas

Seguridad: Los perfiles de seguridad que maneja

Fiabilidad: Sirve para medir que el software en condiciones no favorables, sus subcategorías son:

Recuperabilidad: se refiere a la rapidez en que se puede recuperar el sistema

Tolerancia a Fallos: cómo se maneja los fallos o errores

Usabilidad: La complejidad a la hora de usar el software y sus categorías son:

Aprendizaje: indica si el software es fácil de aprender

Comprensión: una vez utilizado, sus resultados son comprensibles

Soporte y Entrenamiento: información de cómo manejar el software

Interface Gráfica: si la interface gráfica es amigable con el usuario.

Eficiencia: el rendimiento que el software va a tener

Comportamiento en el tiempo: velocidad de procesamiento

Comportamiento con los recursos: consumo de recursos

Mantenibilidad: cómo el software se adapta al paso del tiempo

Estabilidad: indica si es estable a pesar de los cambios

Facilidad de análisis: para ver si los errores son fáciles de interpretar.

Facilidad de cambio: cambios necesarios son fáciles de realizar.

Facilidad de Pruebas: para ver si es fácilmente probado.

Portabilidad: la capacidad del software para migrar o cambiarse de entorno.

Capacidad de Instalación: sirve para determinar la facilidad de instalación.

Capacidad de reemplazamiento: en caso de cambios de versión.

Adaptabilidad: para medir los cambios de ambiente.

Co-existencia: mide si el software co-existe con otros sistemas

Conocidos éstos parámetros, se procede con la elaboración de la Matriz para dicha medición, se deberá llenar en base a los resultados encontrados, una vez que se haya probado cada software seleccionado, dicha matriz tendrá las siguientes características y se debe elaborar una por cada tipo de software, pero si el tipo de programa es el mismo, se puede medir más de dos, a continuación lo mencionado:

Tabla 1. Matriz de Medición y Análisis de software

CARACTERÍSTICA	SUB CARACTERÍSTICA	DESCRIPCIÓN	CALIFICACIÓN	
			SW1	SW2
Funcionabilidad	Idoneidad			
	Exactitud			
	Interoperabilidad			
	Seguridad			
Usabilidad	Aprendizaje			
	Comprensión			
	Soporte y Entrenamiento			
	Interface Gráfica			
Fiabilidad	Recuperabilidad			
	Tolerancia a Fallas			
Eficiencia	Comportamiento en el tiempo			
	Comportamiento de recursos			
Manteniabilidad	Estabilidad			
	Facilidad de análisis			
	Facilidad de cambio			
	Facilidad de pruebas			
Portabilidad	Capacidad de instalación			
	Capacidad de reemplazamiento			
	Adaptabilidad			
	Co-Existencia			
Requerimientos Técnicos	Capacidad del Hardware			
	Flexibilidad			
	Criterio multitarea			
Aspectos Financieros	Costo directo del programa			
	Costo de licencias adicionales			
	TOTAL			

Fuente: elaboración propia

4.2.2. Desarrollo

La metodología aplicada para el desarrollo del *Data Warehouse* fue la de HEFESTO 2.0, dicha metodología consta de 4 fases que llevan a la construcción del *Data Warehouse* desde las recolección de información hasta la implementación final, dichas fases son:

4.2.2.1 Análisis de Requerimientos

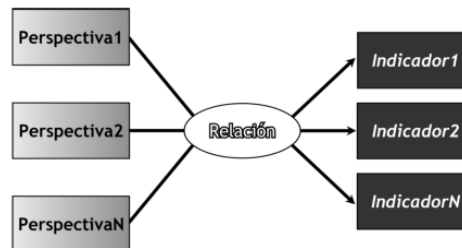
La etapa de análisis de requerimientos será la encargada de recolectar la información necesaria y básica para iniciar la implementación del *Data Warehouse*, es también el proceso inicial para poder comenzar con la definición de las posibles relaciones entre orígenes de datos.

Para la recolección de información será necesario formular una serie de preguntas que permita delimitar y formular los indicadores que son necesarios

Esta información permite identificar indicadores y perspectivas, para lograr determinar un modelo conceptual inicial que en las etapas posteriores serán ampliadas con el análisis OLTP.

El modelo conceptual podrá plasmarse en graficas que relacionen las perspectivas con los indicadores.

Figura 23. Ejemplo de ilustración de modelo conceptual



Fuente: (Hefesto, 2010)

4.2.2.2 Análisis de los OLTP (Online Transaction Processing)

Esta etapa del proceso del *Data Warehousing*, analiza las fuentes OLTP que serán de utilidad para el cálculo de indicadores, además se podrá determinar la manera en cómo se relacionan los datos para ampliar el modelo conceptual.

4.2.2.1.1 Conformar Indicadores

Los indicadores se van a conformar al definir la relación que va a existir entre los datos y las operaciones que se deben realizar para llegar a obtenerlos. Es decir si a un atributo hay se acumularlo antes de dividirlo o multiplicarlo por otro atributo hasta llegar a obtener el valor deseado

Indicador x: operación 1(atributo1, atributo2,...) ... operación n(atributo1,atributo2,...)

Esta información será almacenada en la matriz de conformación de indicadores que se muestra a continuación.

Tabla 2. Matriz de conformación de Indicadores.

Indicador	Forma de Cálculo	Aclaración

Fuente: elaboración propia

Donde, “Indicador” será el nombre del indicador que se desea obtener, “Forma de cálculo” determina la operación y los atributos que intervienen, y “aclaración” contiene una breve descripción del indicador.

4.2.2.1.2 Establecer correspondencias

En este paso se va a determinar y analizar los datos que los OLTP poseen, para de esta manera poder relacionarlos y conformar correspondencias en el modelo conceptual ampliado.

Para ello es necesario conocer los atributos que pueden encontrarse en las tablas de los OLTP o fuentes de datos, estos datos deberán ser tomados desde su origen tal como están almacenados, o podrán sufrir transformaciones que permitan relacionarse con atributos de otras OLTP.

4.2.2.3 Modelo Lógico del Datawarehouse

El modelo lógico se deberá construir basado en el modelo conceptual tratado en la fase anterior, aquí se define el tipo de modelo a implementarse y se deberá definir las tablas de hechos y tablas de dimensiones con sus respectivas relaciones.

Es necesario definir los campos que van a ser tomados en cuenta para el armado del *Data Warehouse* y de las relaciones se obtendrá las tablas de hechos, que surjan en base a la necesidad de relación de datos necesario para el desarrollo de los indicadores.

Para ello se elaboró una matriz que relaciona los datos de las tablas orígenes y las tablas destino en el *Data Warehouse*.

Tabla 3. Matriz de relación de Datos

Tipo	Origen	Campos	Destino	Campos

Fuente: elaboración propia

Esta matriz contiene campos como “Tipo” donde se identifica si es una tabla de hecho o de dimensiones, el “Origen” que sirve para etiquetar la fuente u origen de la información, “Destino” que mostrará el nombre de la tabla en la que va a ser almacenada la información y los respectivos “Campos” tanto de origen como destino.

4.2.2.4 Integración de Datos

La integración de datos es el proceso que consiste en cargar información al modelo diseñado en la base de datos, es decir probar lo que desarrollo en el modelo lógico y conceptual para ver su funcionalidad, esta carga se la realiza mediante técnicas y herramientas de limpieza carga y extracción de datos conocido como ETL. Posterior a la carga inicial de datos, se deberá configurar la actualización futura de información.

4.2.2.4.1 Carga Inicial

La carga inicial se la realizará con la ayuda de la herramienta ETL seleccionada en el análisis. Lo que primero debe cargarse en la base de datos son las tablas de dimensiones y posteriormente realizar el proceso con las tablas de hechos.

Es importante identificar y cronogramar los procesos de carga para no sobrecargar el equipo y evitar de esta manera posibles fallas de hardware y la consiguiente caída del sistema.

Un factor que determina el rendimiento y éxito de la carga inicial será el armado de las sentencias SQL que van a permitir obtener la información de los orígenes de datos, puesto que si una sentencia SQL está mal estructurada, el proceso puede convertirse en un bucle infinito. Para ello se deberá hacer uso de la pre visualización de la herramienta ETL utilizada.

4.2.2.4.1 Actualización

Este proceso va a consistir en realizar un agendamiento de tareas ETL para que la información cargada inicialmente sea actualizada, de manera que el Data Warehouse tenga datos actuales para su análisis.

Esta tarea o programación de tareas deberá realizarse de manera que se ejecute en horarios no convencionales, es decir que la carga de datos no interrumpa con el normal funcionamiento de los sistemas que son orígenes de datos y no se tenga problema de rendimiento en éstos.

La herramienta ETL escogida, brinda también esta facilidad, pero es necesario generar flujos de trabajo y reutilizar los procesos de carga inicial y en el proceso de actualizaron de información, pero se debe programar en fechas y horas no paralelas.

Capítulo 5

Resultados

En Tubasec C.A. el principal problema que se presenta es la segmentación de información y el retardo en la entrega de dicha información, índices y/o reportes. Esto causa que la gerencia general y mandos medios no tengan las herramientas necesarias para poder tomar decisiones respecto al funcionamiento de un proceso en particular o en términos generales en el funcionamiento de la empresa.

Estos argumentos dieron pie para realizar el proyecto, y los resultados se van a exponer en base al proceso que se realizó durante todo el desarrollo.

5.1. ANÁLISIS DE RESULTADOS

El proceso de desarrollo e implementación de Data Warehouse, tuvo dos fases, la primera es la terminación de las mejores herramientas a utilizarse en el proceso y en las diferentes etapas de la metodología empleada.

5.1.1. Comparación entre herramientas de Extracción, Carga y Depuración de Datos.

Una herramienta de Extracción, carga y Depuración de Datos conocida como ETL, es sumamente importante a la hora de tener información consistente y actualizada en un Data Warehouse, existen varias herramientas que ofrecen grandes facilidades para esta tarea.

Entre las herramientas más importantes que figuran en el mercado, según (Bustillos, 2014) están:

- Pentaho Kettle.
- Talend
- Informatica PowerCenter
- Inabplex Inaport
- IBM Cognos Data Manager
- Oracle Warehouse Builder
- Microsoft Integration Services

De entre todas estas herramientas, muchas tienen costo de licenciamiento, esto es una limitante a la hora de tomar una decisión puesto que en el mercado existen varias herramientas gratuitas con gran capacidad, de hecho el presupuesto que se maneja en Tubasec C.A. para TI es limitado, por lo que se optará por herramientas de libre distribución.

En base al Cuadrante mágico de Gartner a Julio de 2015, se puede identificar a Talend como una herramienta visionaria y de crecimiento continuo. Ésta es una herramienta que ha tomado bastante fuerza en el mundo de la Inteligencia de Negocios con licenciamiento gratuito por un tiempo limitado, por otro lado Pentaho es bastante similar a Talend, posee una interface gráfica bastante fácil e intuitiva y además su rendimiento es bastante aceptable y su licenciamiento no es con límite de tiempo, lo que le da una gran ventaja.

Otra herramienta que tiene mucha fuerza y es

Figura 24. Cuadrante Mágico de Gartner Data Integration



Fuente: (BI-Sapin.com, 2015)

Luego de considerar éste estudio y ver las posibles opciones entre herramientas de software libre, se realizaron pruebas con instalación de demos de Pentaho Data Integration y Talend, los resultados obtenidos fueron bastante satisfactorios en rendimiento y manejo de interface, otras características importantes que ayudó a escoger la herramienta fueron:

- Instalación es sencilla,
- No tiene costo de licencia,
- Permite calendarizar las cargas y actualización de forma automática.
- Permite conexión con múltiples fuentes de datos,

- Manejo es muy sencillo e intuitivo,

Estas características la que la convierte en una herramienta flexible y funcional útil para el desarrollo del proyecto.

Finalmente se evaluaron a Pentaho y Talend en la matriz de selección de software planteada en la sección 4.2.1, se obtuvieron los siguientes resultados:

Tabla 4. Evaluación software de Integración de Datos

CARACTERÍSTICA	SUB CARACTERÍSTICA	DESCRIPCIÓN	CALIFICACIÓN	
			PENTAHO	TALED
Funcionabilidad	Idoneidad	El software realiza las tareas ETL	10	10
	Exactitud	Los resultados son los esperados	8	7
	Interoperabilidad	Tiene conexión con múltiples bases de datos	9	7
	Seguridad	Maneja perfiles de usuario	0	0
Usabilidad	Aprendizaje	Facilidad de auto aprendizaje	9	7
	Comprensión	Comprensión del funcionamiento del sistema	5	5
	Soporte y Entrenamiento	Información de Soporte para aprender en línea	6	4
	Interface Gráfica	Buena Interface gráfica del programa	5	5
Fiabilidad	Recuperabilidad	El sistema se recupera con facilidad	8	4
	Tolerancia a Fallas	Manejo de errores	6	7
Eficiencia	Comportamiento en el tiempo	Rapidez en procesos	9	7
	Comportamiento de recursos	Manejo de recursos de Hardware	9	7
Manteniabilidad	Estabilidad	Es estable a pesar de los cambios	8	8
	Facilidad de análisis	Es fácil de interpretar los errores	6	5
	Facilidad de cambio	Se realizan fácilmente los cambios	9	5
	Facilidad de pruebas	Puede ser probado fácilmente	9	7
Portabilidad	Capacidad de instalación	Fácil instalación	9	6
	Capacidad de reemplazamiento	Cambio de versión	9	7
	Adaptabilidad	Migración fácil a otro ambiente	9	6
	Co-Existencia	Co-existe con otros sistemas	9	9
Requerimientos Técnicos	Capacidad del Hardware	Requerimientos hardware	8	6
	Flexibilidad	Es adaptable en diferentes fabricantes	10	10
	Criterio multitarea	Puede ejecutar varias instancias en paralelo	10	7
Aspectos Financieros	Costo directo del programa	Costo de programa	9	3
	Costo de licencias adicionales	Licencias adicionales por equipo	10	5
	TOTAL		199	154

Fuente: elaboración propia

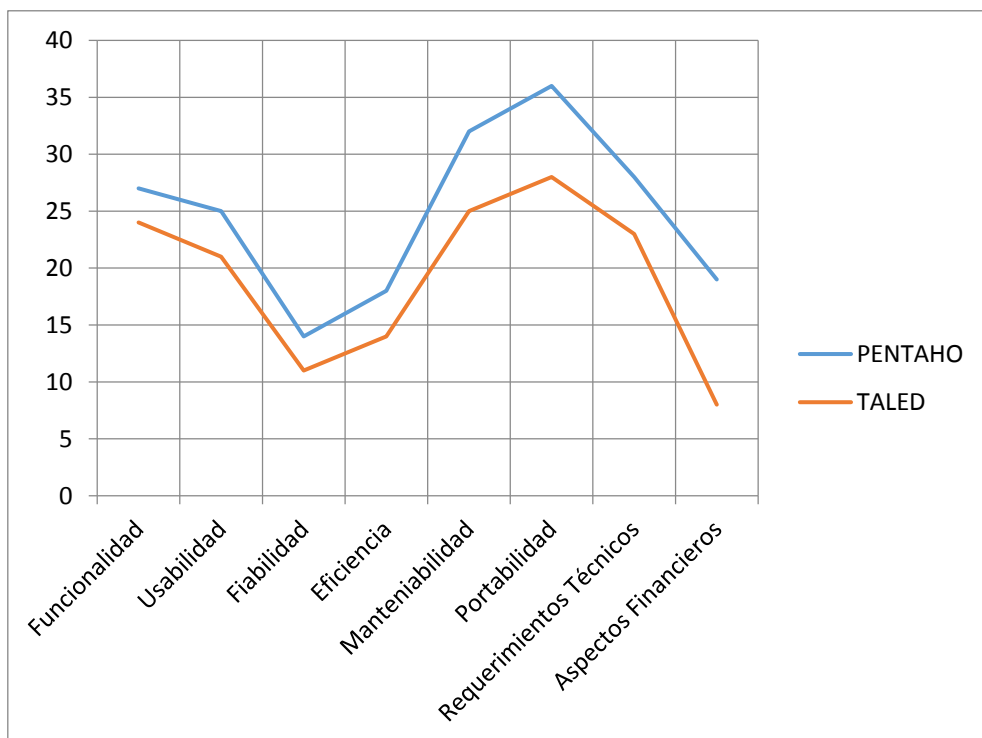
Con los valores expresados en la matriz, se puede elaborar una tabla resumen para poder graficarlos y apreciar de mejor manera los resultados obtenidos.

Tabla 5. Puntuación software de Integración de Datos

Característica	PENTAHO	TALED
Funcionalidad	27	24
Usabilidad	25	21
Fiabilidad	14	11
Eficiencia	18	14
Manteniabilidad	32	25
Portabilidad	36	28
Requerimientos Técnicos	28	23
Aspectos Financieros	19	8

Fuente: elaboración propia

Figura 25. Análisis software de Integración de Datos



Fuente: elaboración propia

INTERPRETACIÓN Y RESULTADO: En base a los resultados obtenidos, la herramienta seleccionada para la integración de datos fue *Pentaho Data Integration*, por acoplarse a las necesidades de la implementación y además por su factor económico que cómo se había mencionado tiene bastante peso en esta selección.

5.1.2. Determinación herramienta SGBD (Sistema Gestor de Base de Datos)

A la hora de escoger un motor de base de datos, es importante considerar la aplicabilidad que se va a tener, es decir, si se va a gestionar gran cantidad de datos o lo que se desea es eficiencia al momento de realizar consultas.

Al igual que en las herramientas de ETL, también es importante considerar ciertos parámetros importantes, en los sistemas gestores de Base de Datos, se debe analizar a la hora de tomar una decisión el rendimiento y el caché que se maneja para medir el volumen de información que puede procesar en consulta.

El costo de licencia es otro factor importante puesto que en el mercado existen DBMS (*Data Base Manager System*) con gran efectividad y con licencia gratuita, ese es el caso de MySQL y PostgreSQL que en éstos últimos años han sabido ganar terreno frente a otros competidores.

Éstos dos potentes sistemas no son muy fáciles de comparar, de hecho son competidores pero en categorías diferentes es decir su aplicabilidad es utilizada de acuerdo a la aplicación que vaya a desarrollarse, ese es el pilar fundamental que hizo que se tome una decisión al escoger una Base de Datos para el *Data Warehouse*.

Para este proyecto se determinó que la mejor opción es PostgreSQL a pesar que su operación sea algo más complicada, pero es más robusto y fiable en el manejo de grandes cantidades de datos, su estructura posee un excelente planificador de consultas, lo que es efectivo para realizar consultas de tablas con grandes cantidades de datos (Martinez D. , 2014). Mientras que por su lado MySQL maneja velocidad en consultas pero tiene problemas cuando las tablas son muy grandes, por lo que este DBMS es útil a la hora de desarrollar aplicaciones de alta velocidad de consulta, por ejemplo aplicaciones cliente servidor en la Web (Martinez D. , 2014). Éste factor podría ser determinante a la hora de seleccionar una herramienta, pues al considerar que los Data Warehouse normalmente manejarán gran cantidad de información y que además esta información no solo es actualizada, sino también que contendrá datos históricos que generan un crecimiento constante en la base de datos a través del tiempo.

Una vez determinadas y seleccionadas las dos posibles herramientas que al parecer se justan a los requerimientos tanto técnicos como económicos, se procede con la evaluación en la matriz planteada en el apartado 4.21., misma que permitirá evaluar la más apropiada para

los fines consiguientes y que se ajusten a las necesidades empresariales sin disminuir el rendimiento del *Data Warehouse*.

Tabla 6. Evaluación Motor de Base de Datos

CARACTERÍSTICA	SUB CARACTERÍSTICA	DESCRIPCIÓN	CALIFICACIÓN	
			MYSQL	POSTGRESQL
Funcionabilidad	Idoneidad	El software soporta grandes cantidades de datos	6	9
	Exactitud	Los resultados son los esperados	5	10
	Interoperabilidad	Tiene conexión con múltiples sistemas	8	10
	Seguridad	Maneja perfiles de usuario	5	9
Usabilidad	Aprendizaje	Facilidad de auto aprendizaje	10	10
	Comprensión	Compresión del funcionamiento del sistema	10	9
	Soporte y Entrenamiento	Información de Soporte para aprender en línea	10	10
	Interface Gráfica	Buena Interface gráfica del programa	5	10
Fiabilidad	Recuperabilidad	El sistema se recupera con facilidad	10	9
	Tolerancia a Fallas	Manejo de errores	8	8
Eficiencia	Comportamiento en el tiempo	Rapidez en procesos	5	9
	Comportamiento de recursos	Manejo de recursos de Hardware	6	9
Mantiabilidad	Estabilidad	Es estable a pesar de los cambios	8	9
	Facilidad de cambio	Se realizan fácilmente los cambios	9	9
	Facilidad de pruebas	Puede ser probado fácilmente	9	9
Portabilidad	Capacidad de instalación	Fácil instalación	10	10
	Capacidad de reemplazamiento	Cambio de versión	9	10
	Adaptabilidad	Migración fácil a otro ambiente	10	10
	Co-Existencia	Co-existe con otros sistemas	10	10
	TOTAL		153	179

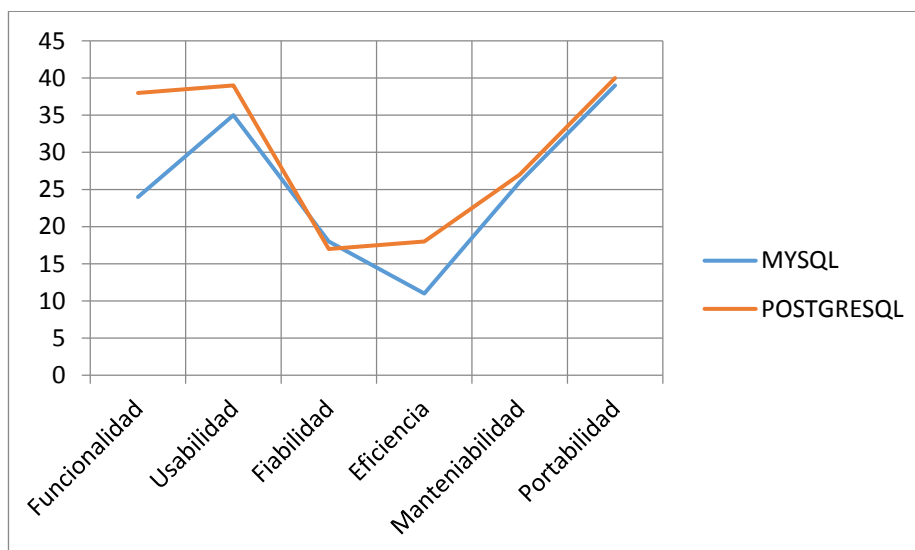
Fuente: elaboración propia

Tabla 7. Puntuación Motor de Base de Datos

Característica	MYSQL	POSTGRESQL
Funcionalidad	24	38
Usabilidad	35	39
Fiabilidad	18	17
Eficiencia	11	18
Mantiabilidad	26	27
Portabilidad	39	40

Fuente: elaboración propia

Figura 26. Análisis Motor de Base de Datos



Fuente: elaboración propia

INTERPRETACIÓN Y RESULTADO: Tras analizar los resultado obtenidos, PostgreSQL posee características que se busca, si se considera que en un *Data Warehouse* se va a manejar grandes cantidades de datos y muchas veces es necesario unir varias tablas para procesar consultas de datos desglosados con el objetivo de mostrar informes resumidos, también es importante conocer que la memoria caché que maneja PostgreSQL es mayor que en MySQL, con lo que se mejora el rendimiento en volúmenes altos.

5.1.3. Comparación entre herramientas BI

En la actualidad existen múltiples herramientas BI que solventan fácilmente nuestras necesidades de visualización de datos, esto orienta a realizar un análisis de las principales herramientas, sin olvidar tener en cuenta las mismas consideraciones que se tuvo con el gestor de Base de Datos y con las herramientas ETL.

Una consideración importante es conocer que herramientas BI son líderes en el mercado, para ello el análisis se basó en el Cuadrante Mágico de Gartner, este cuadro brinda una una visión global de las principales herramientas y su impacto a nivel mundial, con este cuadro se logra referenciar a estudios elaborados entre múltiples plataformas y tecnologías orientadas a este proceso.

Figura 27. Cuadrante Mágico de Gartner BI



Fuente: (Gartner, 2016)

Otra factor determinante al igual que en las herramientas de Integración de Datos y el motor de Dase de Datos, es el costo de licencias, puesto que si bien es cierto las mejores herramientas prestan muchas facilidades y al mismo tiempo tienen gran flexibilidad y versatilidad, su costo es considerable también.

Del cuadro de Gartner, se pudo revisar que las herramientas líderes en la actualidad son Tableau y Qlik, los costos de licencias son variados, ya que depende de la funcionalidad y características que se desee contratar, sin embargo oscilan entre los \$10.000 y \$20000 para la instalación de servidor y licencias para usuario (Datos obtenidos de una proforma enviada a Tubasec C.A. por parte del Grupo Novatech), todo esto cómo un solo paquete, esto sin contar con el soporte que es adicional al costo de licencias. Otro factor que está por fuera del costo de la licencia es el valor por capacitación, que también está enfocado a varios niveles, es decir, para principiantes y programadores con experiencia, esto también determina el costo por este factor.

A continuación un cuadro comparativo de las dos más importantes en el que se puede un enfoque amplio de criterios a la hora de elegir una herramienta

Figura 28. Comparación entre Tableau vs Qlikview

Comparison of Tableau vs. Qlikview					
Business Criteria					
Business Criteria	Tableau		Qlikview		Comment
	Assessment	score	Assessment	score	
Time to implement	fast	7	shortest	8	Both increase productivity
Scalability	Good	6	RAM Limited	6	Tableau uses virtual RAM
Price for Developer	\$1,999	6	about \$4000	4	Tableau is less expensive
Server License/user	\$1K, min 10 users	5	> \$1000	3	
Support fees / year	20%	5	20%	5	
SaaS Platform	Core or Digital	9	Special License	8	Requires negotiation
Overall Cost	Above Average	5	High, needs R&D	4	Tableau costs less
Enterprise Ready	Good for SMB	6	Good for SMB	6	
Long-term viability	Fastest growth	6	Public company	7	Both 1 product company
Mindshare	Tableau Public	7	Growing Fast	6	Tableau Leads
Big Data Support	Above Average	6	Average	4	
Partner Network	Below Average	3	Large	7	Qlikview: 1000+ partners
Visualization Criteria					
Visualization Criteria	Tableau		Qlikview		Comment
Data Interactivity	Very Good	8	Excellent	9	Qlikview is more mature
Visual Drilldown	Very Good	8	Excellent	9	
Offline Viewer	Free Reader	8	Personal Edition	6	Tableau has better option
Analyst's Desktop	Tableau Pro	9	Qlikview Desktop	9	
Dashboard Support	Excellent	9	Excellent	9	
Web Client	Excellent	8	Very Good	7	
Mobile Clients	Very Good	7	best variety	8	
Visual Controls	Very Good	8	Very Good	8	
UI Interactivity	Good, no MDI	8	Good, MDI support	8	Multi-Document Interface!
Technical Criteria					
Technical Criteria	Tableau		Qlikview		Comment
Data Integration	Excellent	8	Very Good	7	Tableau is a winner
Development	Tableau Pro	5	Qlikview Developer	7	Qlikview has Scripting
64-bit in-memory DB	Very Good	7	Excellent	9	Qlikview is faster
Integration with GIS	Excellent	8	Average	5	Tableau has Mapping
Modeling, Analytics	Below Average	4	Below Average	4	Both have calculated Fields
Data Mining	Limited	3	None	2	Tableau reads SSAS
Multidimensional	Very Good	7	Limited	3	
xVelocity Support	Good	7	None	1	Tableau is a winner
PowerPivot Support	Good	7	None	1	
API	None	1	Limited	3	
Best Feature	Easy Visualization		Visual Drilldown		
		201		183	

Fuente (Sanz, 2013):

Este cuadro muestra la diferencia entre las dos herramientas, sin embargo los criterios utilizados aquí inclinarían la balanza por Tableau, a pesar de esa inclinación la elección fue por Qlik, debido a que Qlick sense y Qlick view Personal Edition en sus versiones gratuitas, poseen las facilidades de Tableau (también tiene una versión gratuita), sin embargo, Qlik es más potente a la hora de realizar análisis de datos para toma de decisiones (Sanz, 2013)

Con las dos herramientas preseleccionadas, ahora se analizará en la matriz de evaluación de software de la sección 4.2.1 para determinar la mejor opción e implementarla con la finalidad de mostrar los datos almacenados en el *Data Warehouse*.

Tabla 8. Evaluación herramientas *Business Intelligence*.

CARACTERÍSTICA	SUB CARACTERÍSTICA	DESCRIPCIÓN	CALIFICACIÓN	
			QLIK	TABLEAU
Funcionabilidad	Idoneidad	El software realiza las tareas BI	10	8
	Exactitud	Los resultados son los esperados	10	8
	Interoperabilidad	Tiene conexión con múltiples bases de datos	10	8
	Seguridad	Maneja perfiles de usuario	8	7
Usabilidad	Aprendizaje	Facilidad de auto aprendizaje	9	8
	Comprensión	Comprensión del funcionamiento del sistema	10	10
	Soporte y Entrenamiento	Información de Soporte para aprender en línea	8	4
	Interface Gráfica	Buena Interface gráfica del programa	10	9
Fiabilidad	Recuperabilidad	El sistema se recupera con facilidad	10	8
	Tolerancia a Fallas	Manejo de errores	10	7
Eficiencia	Comportamiento en el tiempo	Rapidez en procesos	9	8
	Comportamiento de recursos	Manejo de recursos de Hardware	9	8
Manteniabilidad	Estabilidad	Es estable a pesar de los cambios	10	9
	Facilidad de análisis	Es fácil de interpretar los errores	9	8
	Facilidad de cambio	Se realizan fácilmente los cambios	10	9
	Facilidad de pruebas	Puede ser probado fácilmente	10	7
Portabilidad	Capacidad de instalación	Fácil instalación	10	10
	Capacidad de reemplazamiento	Cambio de versión	10	7
	Adaptabilidad	Migración fácil a otro ambiente	10	9
	Co-Existencia	Co-existe con otros sistemas	10	10
Requerimientos Técnicos	Capacidad del Hardware	Requerimientos hardware	9	9
	Flexibilidad	Es adaptable en diferentes fabricantes HW	9	9
	Criterio multitarea	Puede ejecutar varias instancias en paralelo	5	5
Aspectos Financieros	Costo directo del programa	Costo de programa	10	3
	Costo de licencias adicionales	Licencias adicionales por equipo	10	5
	TOTAL		235	193

Fuente: elaboración propia

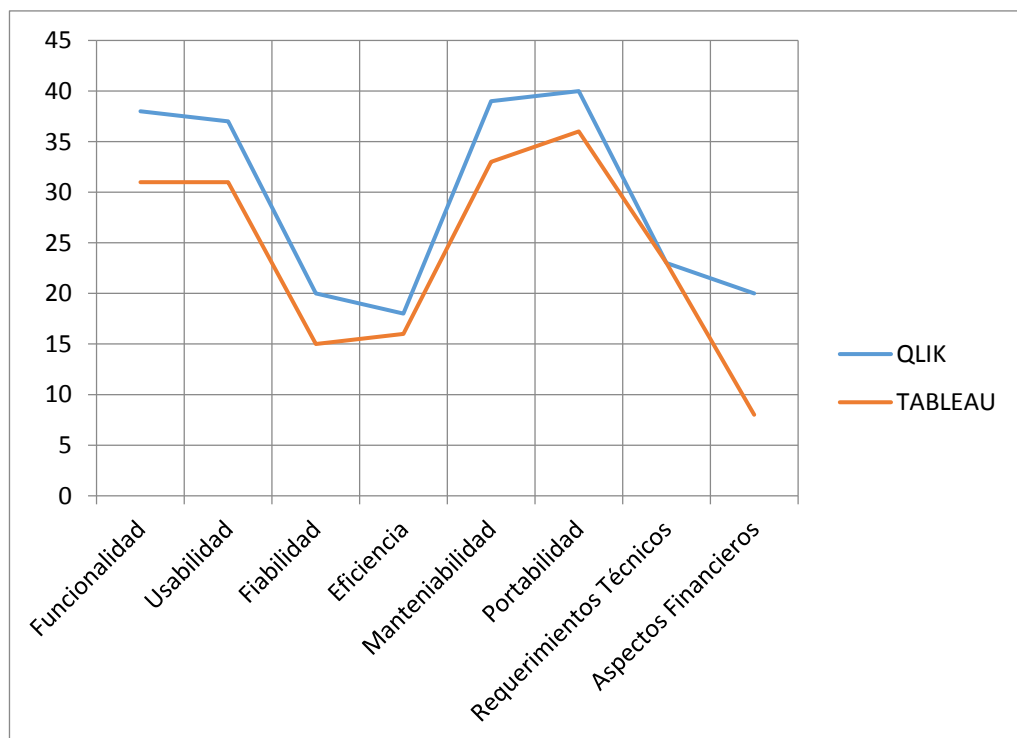
Dados los parámetros de evaluación y llenos los campos con la calificación respectiva, es procedente realizar un cuadro resumen que ayuda a graficar los valores tabulados y lograr de esta manera visualizar los resultados obtenidos, para tener una visión panorámica y poder elegir acertadamente la herramienta que más se ajuste a las necesidades tanto empresariales como técnicas.

Tabla 9. Puntuación herramientas Business Intelligence.

Característica	QLIK	TABLEAU
Funcionalidad	38	31
Usabilidad	37	31
Fiabilidad	20	15
Eficiencia	18	16
Mantiabilidad	39	33
Portabilidad	40	36
Requerimientos Técnicos	23	23
Aspectos Financieros	20	8

Fuente: elaboración propia

Figura 29. Análisis herramientas de Business Intelligence



Fuente: elaboración propia

INTERPRETACIÓN Y RESULTADO: Luego de revisar las consideraciones planteadas y analizadas en cada una de las citas expuestas,, se puede emitir un criterio más acertado y afirmar que existen herramientas muy potentes y con características importantes, que pueden ser utilizadas para cumplir con el objetivo planteado, sin embargo por las características propias de la empresa donde va a ser implementado el *Data Warehouse*, l

alanza se inclina por Qlik, debido a que esta herramienta cumple con lo que se necesita para la visualización de datos y además esta posee una versión gratuita que no está limitada por el tiempo de uso y es bastante flexible a la hora de generar reportes, indicadores y cuadros de mando gerenciales.

5.1.2 Desarrollo de la Aplicación

El desarrollo estuvo basado en la metodología de Hefesto 2.0 que consta de cuatro pasos o fases principales, hasta llegar a conseguir la implementación del Data Warehouse, posterior a esto se implementó la visualización de datos almacenados en el Data Warehouse en una herramienta de Inteligencia de Negocios, para el apoyo de toma de decisiones.

A continuación se va a revisar los resultados obtenidos en cada fase de la metodología aplicada y sus respectivos resultados.

5.1.2.1. Análisis de requerimientos

Esta etapa fue trascendental para la elaboración del Data Warehouse, debido a que es la base sobre la que las demás fases serán implementadas.

Con las entrevistas realizadas a los mandos medios y Gerencias, se logró identificar el rumbo que se debe tomar a la hora de diseñar el modelo lógico, entre las preguntas principales que se aplicó a los entrevistados fueron:

Líderes:

- Qué datos les serían útiles a la hora de tomar una decisión importante en su proceso?
- Dónde almacena la información concerniente a su proceso?
- Qué indicadores maneja actualmente?
- Cómo mide el rendimiento de su proceso?

Gerencias:

- Cómo analiza el estado de su empresa actualmente?
- Qué información considera importante para tomar decisiones trascendentales para la empresa?
- Qué nivel de granularidad le gustaría manejar para analizar la información?
- Qué áreas de la empresa le gustaría integrar?
- Cómo mide el rendimiento de su empresa en eficacia y eficiencia?

Gracias a estas preguntas se logró identificar los indicadores y perspectivas que cada uno tiene y lo que se espera que el DW almacene en las diferentes tablas, así como también las

posibles relaciones que se va a encontrar en las diferentes áreas y/o procesos con las que cuenta Tubasec C.A. Los indicadores encontrados son:

Ventas

- Ventas por producto, tiempo, conjunto, centro de costo.
- Ventas por peso en el tiempo, por producto y conjunto de artículos.
- Ventas vs Devoluciones por tiempo.
- Cartera por clientes en el tiempo, región, producto.

Producción

- Producción mensual por producto.
- Pedidos vs Producción, por tiempo, producto.
- Toneladas por producción, por tiempo y producto.

Talento Humano

- Horas Extras Pagadas por proceso y tiempo.
- Índices de ausentismo por motivo y tiempo.
- Sueldos y Salarios por proceso, tiempo y centro de costo.

Proveedores

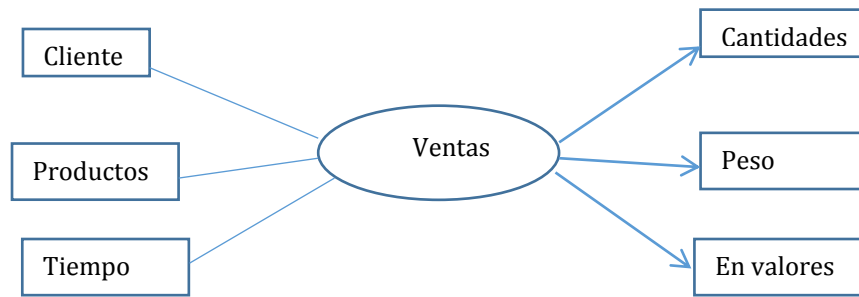
- Deudas a proveedores por tiempo.
- Pagos a proveedores por tiempo.

Salud Ocupacional

- Índice de ausentismo por enfermedades laborales.
- Índice de enfermedades recurrentes en el tiempo.
- Índice de rotación de puestos de trabajo por proceso y tiempo.

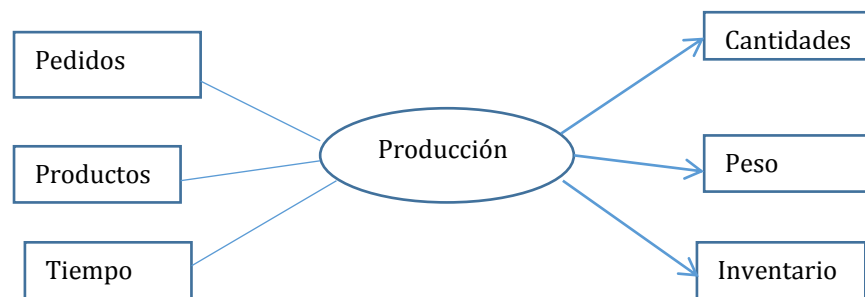
El modelo conceptual inicial luego de las entrevistas y procesos de investigaciones realizadas, estará representado por una ilustración gráfica donde se identifican las entradas y las salidas que se desea obtener en cada indicador, tal cómo se muestra a continuación:

Figura 30. Modelo Lógico Ventas



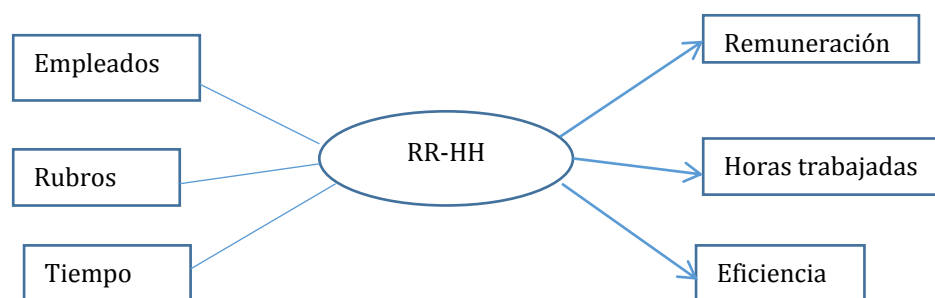
Fuente: elaboración Propia

Figura 31. Modelo Lógico Producción



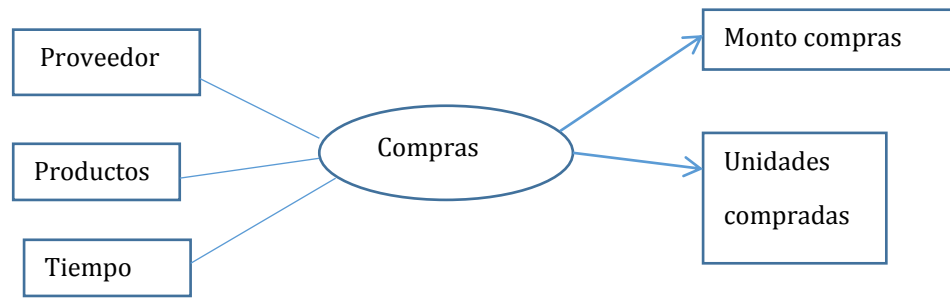
Fuente: elaboración propia

Figura 32. Modelo Lógico Talento Humano



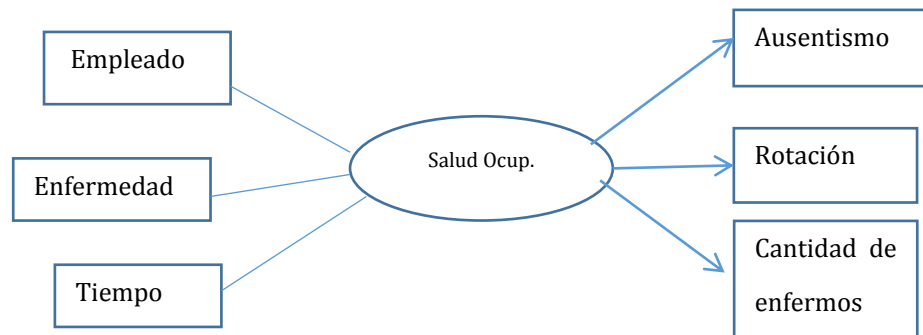
Fuente: elaboración propia

Figura 33. Modelo Compras



Fuente: elaboración propia

Figura 34. Seguridad y Salud Ocupacional



Fuente: elaboración propia

5.1.2.2 Análisis de los Orígenes de Datos

Del análisis orígenes de datos conocidos también como OLTP, por sus siglas en inglés (*OnLine Transaction Processing*) se logró determinar que los datos para lograr obtener los indicadores y verificar que éstos sean alcanzables, además confirmar que exista una relación entre ellos, a continuación se muestra los indicadores y orígenes de la información, además de la manera en cómo se realizó el cálculo de cada indicador, para lograr plasmar lo que la gerencia necesita analizar.

Tabla 10. Indicadores de Ventas

Indicador	Forma de Calculo	Aclaración
Unidades expedidas	$(\text{Unidades Vendidas}) - (\text{Unidades Devueltas})$	Cantidad de unidades vendidas y restadas de las unidades devueltas en un período de tiempo determinado.
Valor Facturado	$(\text{Valor Facturado}) - (\text{Valor en Notas de Crédito}) + (\text{Valor en Notas de Débito})$	Valor procedente de la venta de productos, quitándole el valor de Notas de Crédito y se suma el valor de Notas de Débito, en un determinado período de tiempo.
Expediciones por Peso	$(\text{Unidades expedidas}) * (\text{Peso del producto})$	Valor en Kilos de cada producto vendido en un período de tiempo determinado
Porcentaje de ventas	Sumatoria del valor Vendido / sumatoria del valor vendido de un producto.	Indica el porcentaje de ventas de un determinado producto
Porcentaje de ventas por peso	Expediciones por peso general / Expediciones por peso del producto	Porcentaje que representa un producto sobre el total de ventas en un período de tiempo determinado
Porcentaje de devoluciones	Cantidad de devoluciones totales / Cantidades de devoluciones del producto	Porcentaje de devoluciones de un producto en relación con las devoluciones totales determinado en un período de tiempo establecido.

Fuente: elaboración propia

Tabla 11. Indicadores de Producción

Indicador	Forma de Calculo	Aclaración
Producción	Sumatoria de Toneladas Producidas	Sumatoria de Toneladas Producidas en un período de tiempo determinado, clasificado por producto.

Inventario	Toneladas producidas / peso del producto	Cantidad de productos en patios aptos para la venta, por producto o por conjunto de artículos, en un período de tiempo determinado
Bajas	Inventarios – Bajas de producción	Cantidad de productos que no son aptos para la venta al público, por producto en un período de tiempo determinado
Porcentaje de producción	Producción total / producción del producto	Ver el porcentaje de producción de un determinado producto, comparándolo con la producción total, en un rango de fechas definidas por el usuario.
Porcentaje de bajas	Total de producción de un producto / total de bajas de ese producto	Determina el porcentaje de bajas de un producto en una producción determinada y en un tiempo establecido por el usuario para poder realizar análisis de producción.
Porcentaje de inventarios	Total de órdenes de un producto / inventario de un producto	Porcentaje de productos disponibles para ordenes ingresadas de un producto en un período de tiempo determinado

Fuente: elaboración propia

Tabla 12. Indicadores de Proveedores

Indicador	Forma de Calculo	Aclaración
Cantidad de Compras por proveedor	Suma de la Cantidad de productos comprados	Suma de compras realizadas a un proveedor específico
Compras por proveedor	(Valor comprado) – (Valor devuelto)	Total comprado menos el total devuelto a un proveedor en un tiempo determinado

Porcentaje de compras	de	Valor total de compras / valor de compras del producto	Determina el porcentaje de la compra de un producto vs el total de compras en un tiempo definido
-----------------------	----	--	--

Fuente: elaboración propia

Tabla 13. Indicadores de Talento Humano

Indicador	Forma de Calculo	Aclaración
Cantidad de Horas Extras	Suma la cantidad de Horas extras	Sumar la cantidad de horas extras en un tiempo de terminado
Valor Horas Extras.	Suma el valor pagado por horas extras	Obtiene el valor de horas extras de uno o varios empleados en un período de tiempo establecido
Remuneración	Suma el valor de la remuneración	Obtiene el valor por concepto de remuneración de uno o varios empleados en un período establecido
Porcentaje de Ausentismo	Suma horas de ausentismo / Horas trabajadas	Muestra el porcentaje de ausentismo de un empleado / trabajador en un período de tiempo determinado y clasificado por motivos.
Horas extras por sexo	Total de horas extras / cantidad por sexo	Muestra el porcentaje de horas extras por sexo en un período de tiempo determinado
Porcentaje de Remuneración por sexo	Remuneración / cantidad de empleados por sexo	Muestra el porcentaje de remuneración por sexo en un período de tiempo determinado
Empleados por sexo	Cantidad total de empleados / (cantidad de empleado por sexo)	Porcentaje de empleados por sexo
Tipo de empleados	Empleados por sexo / Cantidad de empleados por tipo	Muestra el porcentaje de empleado por sexo y por tipo, este tipo se refiere a administrativo u obrero.

Fuente: elaboración propia

Tabla 14. Seguridad y Salud Ocupacional

Indicador	Forma de Calculo	Aclaración
Porcentaje de enfermedades que causan ausentismo	Suma horas no asistidas por enfermedad totales/ Numero horas no asistidas por empleado	Encuentra el porcentaje de ausencia por cada enfermedad.
Porcentaje de enfermedades que no causan ausentismo laboral	Suma total empleados con cualquier enfermedad / Numero empleados con una enfermedad especifica	Índice del porcentaje que representa una enfermedad frente al total de enfermedades.
Índice de rotación de puesto de trabajo	Empleados que se cambian de puesto / Total de empleados	Muestra el porcentaje de cambio de lugar de trabajo. Frente al total de empleados.

Fuente: elaboración propia

5.1.2.3 Modelo Lógico de Data Warehouse

El modelo lógico del *Data Warehouse*, fue elaborado en base sobre un modelo tipo constelación, puesto que el Data Warehouse consta de varias tablas de hecho y la dimensión tiempo está relacionada con todas las FACT, así mismo cada tabla de hechos está diseñada en un esquema copo de nieve o estrella según sea el caso, de manera que este DW es una mezcla de los tres esquemas.

Los datos que van a ser utilizados están descritos en la matriz de transformación presentada en la metodología y que se muestra a continuación, esta matriz será la base que presente el modelo lógico y tendrá una por cada tabla de hechos.

Modelo Lógico de Inventarios y Producción.

El modelo lógico que contempla los datos de la producción e inventarios de los productos elaborados por la empresa, esto implica que se debe tener en cuenta para este modelo tanto los datos de producción cómo los datos de los productos que están inventariados y listos para ser comercializados, es decir el stock de inventarios. Los datos principales y necesarios son los detallados en la siguiente matriz:

Tabla 15. Matriz de Relación de datos de inventarios

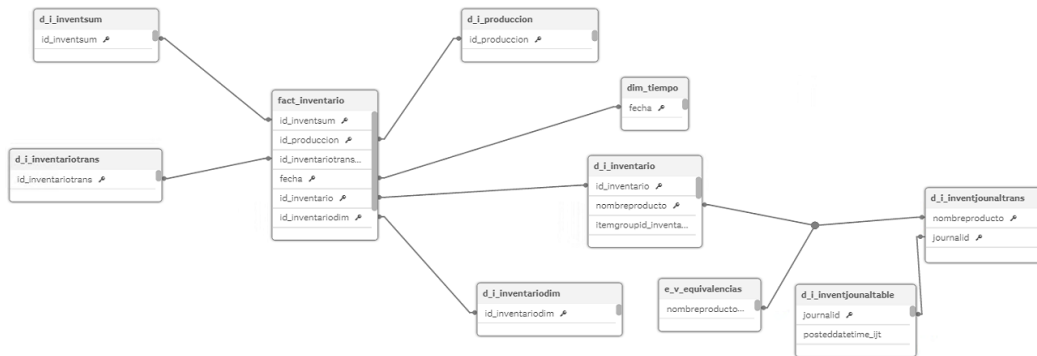
Tipo	Origen	Campos	Destino	Campos
Tabla de Dimensiones	ERP (Inventsum)	Id	Data Warehouse (d_i_inventsum)	Id_inventsum
		itemid		itemid_inventsum
		onorder		onorder_inventsum
		availorder		availorder_inventsum
		availphysical		availphysical_inventsum
		physicalinvent		physicalinvent_inventsum
Tabla de Dimensiones	ERP (Inventariotrans)	Id	Data Warehouse (d_i_inventariotrans)	Id_inventariotrans
		itemid		itemid_inventariotrans
		inventdimid		inventdimid_inventariotrans
		transrefid		transrefid_inventariotrans
		dateexpected		dateexpected_inventariotrans
		datefinanciam		datefinanciam_inventariotrans
		bodega		bodega_inventariotrans
		qty		qty_inventariotrans
Tabla de Dimensiones	ERP (prodtable)	Id	Data Warehouse (d_i_produccion)	Id_produccion
		itemid		itemid_produccion
		name		name_produccion
		qtystup		qtystup_produccion
		realdate		realdate_produccion
		dimension		dimension_produccion
		dimension3		dimension3_produccion
Tabla de Dimensiones	ERP (inventdim)	id	Data Warehouse (d_i_inventariodim)	id_inventariodim
		inventdim		inventdim_inventariodim
		inventlocationid		inventlocationid_inventariodim
Tabla de Dimensiones	ERP (inventtable)	Itemgroupid	Data Warehouse (d_i_inventario)	Itemgroupid_inventario
		itemid		itemid_inventario
		itemname		itemname_inventario
		dimension		dimension_inventario
		dimension3		dimension3_inventario
		netweight		netweight_inventario

Fuente: elaboración Propia

A partir de la información presentada en la matriz anterior, se procedió con la tabla de hechos para inventarios y el respectivo diagrama del modelo.

fac_inventario. Las relaciones directas de la tabla de hechos de inventarios y producción son las tablas de dimensiones d_i_inventsum, d_i_inventariotrans, d_i_produccion, d_i_inventariodim y d_i_inventario, esta última tiene relaciones independientes con tres tablas d_i_invetjopurnaltrans, e_v_equivalencias y d_i_inventjournaltable; éstas permiten conocer las transferencias a bodegas internas, que finalmente servirá para generar indicadores de movimientos y rotación de inventario.

Figura 35. Modelo lógico fact_inventario



Fuente: elaboración propia.

Modelo Lógico Proveedor. La tabla de hechos denominada fact_proveedor, contiene la relación de las tablas de dimensiones donde se encuentra contenidos datos de los proveedores de la empresa, aquí se puede consultar valores de pagos y compras que la empresa ha realizado a cada proveedor por fecha y tipo de compra.

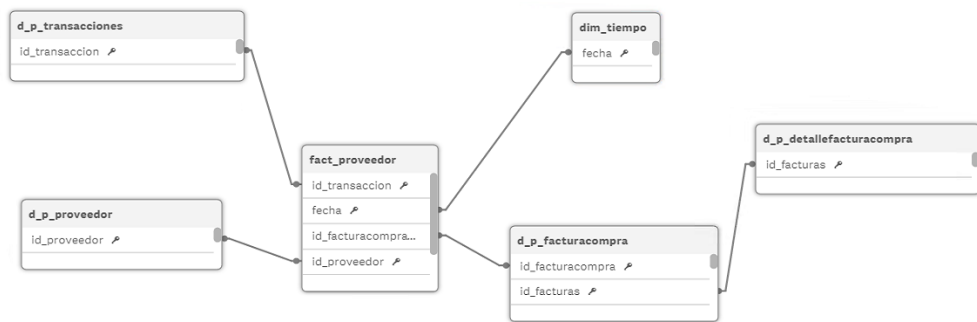
Tabla 16. Matriz de Relación de datos de inventarios

Tipo	Origen	Campos	Destino	Campos
Tabla de Dimensiones	ERP (Vendtable)	accounum	Data Warehouse (d_p_proveedor)	accounum_proveedor
		name		name_proveedor
		address		address_proveedor
		city		city_proveedor
		vendgroup		vendgroup_proveedor
Tabla de Dimensiones	ERP (Vendtrans)	accountnum	Data Warehouse (d_p_proveedor)	accountnum_transacciones
		transdate		transdate_transacciones
		voucher		voucher_transacciones
		invoice		invoice_transacciones
		txt		txt_transacciones
		amountcur		amountcur_transacciones
Tabla de Dimensiones	ERP (vendinvoicejour)	purchid	Data Warehouse (d_p_facturacompra)	purchid_facturacompra
		orderaccount		orderaccount_facturacompra
		invoicedate		invoicedate_facturacompra
		qty		qty_facturacompra
		invoiceamount		invoiceamount_facturacompra
		Internalinvoiceid		internalinvoiceid_facturacompra
Tabla de Dimensiones	ERP (vendinvoicestrans)	invoiceid	Data Warehouse (d_p_detallefacturacompra)	invoiceid_detallefacturacompra
		invoicedate		invoicedate_detallefacturacompra
		itemid		itemid_detallefacturacompra
		qyt		qyt_detallefacturacompra
		lineamount		lineamount_detallefacturacompra
		name		name_detallefacturacompra

Fuente: elaboración propia.

fact_proveedor: Las tablas de dimensiones que se relacionan son d_p_transacciones, d_p_proveedor, d_p_facturacompra y la dim_tiempo, además la tabla d_p_facturacompra se relaciona únicamente con la tabla d_p_detallefacturacompra, esta relación servirá para ver los detalles de las facturas en caso de ser requerido.

Figura 36. Modelo lógico fact_proveedor



Fuente: elaboración propia.

Modelo Lógico Empleados.

El modelo lógico de empleados fue concebido con el fin de asociar al Data Warehouse, la información de los empleados y trabajadores, de esta manera se logra conocer la situación económica que cada uno o en su defecto cada departamento o proceso genera para los costos y gastos de producción, además.

Este modelo es de tipo estrella y sus conexiones están ligadas aun sola tabla de hechos mimas que se relaciona finalmente con las otras tablas de hechos.

Tabla 17. Matriz de Relación de datos de Personal

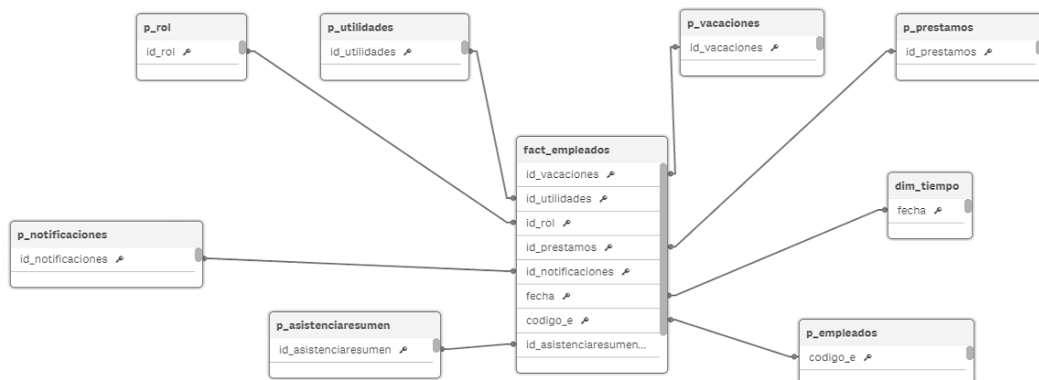
Tipo	Origen	Campos	Destino	Campos
Tabla de Dimensiones	THUT (empleados)	codigo	Data Warehouse (p_empleados)	codigo_e_empleados
		nombres		nombres_empleados
		apellidos		apellidos_empleados
		proceso		proceso_empleados
		ccosto		ccosto_empleados
		fecha_i_tub		fecha_i_tub_empleados

Tabla de Dimensiones	THUT (tvacaciones)	id	Data Warehouse (p_vacaciones)	id_vacaciones
		fecha_i		fecha_i_vacaciones
		fecha_f		fecha_f_vacaciones
		período		período_vacaciones
		días		días_vacaciones
		mes		mes_vacaciones
		año		año_vacaciones
		comentario		comentario_vacaciones
Tabla de Dimensiones	THUT (trol)	id	Data Warehouse (p_rol)	id_rol
		codigo_e		codigo_e_rol
		mes		mes_rol
		año		año_rol
		rubro		rubro_rol
		valor		valor_rol
Tabla de Dimensiones	THUT (tasistenciaresumen)	codigo_e	Data Warehouse (p_asistenciaresumen)	codigo_e_aistenciaresumen
		horas		horas_aistenciaresumen
		he25		he25_aistenciaresumen
		he50		he50_aistenciaresumen
		he100		he100_aistenciaresumen
		mes		mes_aistenciaresumen
		año		año_aistenciaresumen

Fuente: elaboración propia.

Fact_empleados: La tabla de hechos fact_empleados es aquella que fue formada para relacionar información del manejo de talento humano de la empresa, contiene la relación de las tablas de empleados, pagos, asistencias, vacaciones y la dimensión tiempo, este modelo es un modelo estrella puesto que todas se relacionan únicamente con una tabla de hechos.

Figura 37. Modelo lógico fact_empleados



Fuente: elaboración propia.

Modelo lógico Ventas: contiene información que relaciones a las tablas de dimensiones que almacenan información de las ventas y de los clientes que la empresa tiene registrado, con estas relaciones se puede consultar datos de compras realizadas a través del tiempo, determinar también que producto se ha vendido más, así como las devoluciones que se han efectuado y los clientes que más han consumido los productos de la empresa.

Tabla 18. Matriz de Relación de ventas

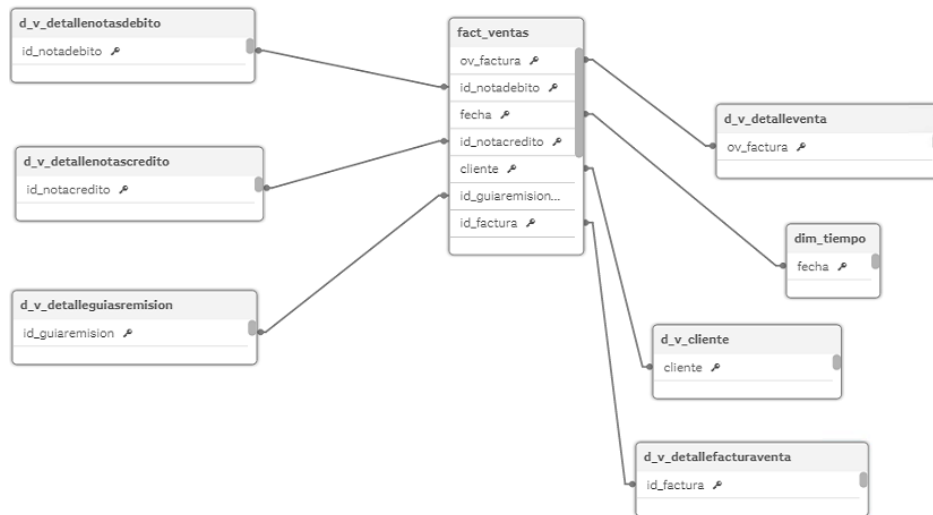
Tipo	Origen	Campos	Destino	Campos
Tabla de Dimensiones	ERP (leccreddoclinetable)	itemid	Data Warehouse (d_v_detallenotascredito)	itemid_detallenotascredito
		invoicedate		invoicedate_detallenotascredito
		num		num_detallenotascredito
		name		name_detallenotascredito
		qty		qty_detallenotascredito
		salesprice		salesprice_detallenotascredito
Tabla de Dimensiones	ERP (lecdebdoclinetable)	itemid	Data Warehouse (d_v_detallenotasdebito)	itemid_detallenotasdebito
		invoicedate		invoicedate_detallenotasdebito
		num		num_detallenotasdebito
		name		name_detallenotasdebito
		qty		qty_detallenotasdebito
		salesprice		salesprice_detallenotasdebito
Tabla de Dimensiones	ERP (salesline)	itemid	Data Warehouse (d_i_detalleventas)	itemid_detalleventas
		invoicedate		invoicedate_detalleventas
		num		num_detalleventas
		name		name_detalleventas
		qty		qty_detalleventas
		salesprice		salesprice_detalleventas
Tabla de Dimensiones	ERP (custtable)	accountnum	Data Warehouse (d_i_cliente)	accountnum_cliente
		name		name_cliente
		address		address_cliente
		city		city_cliente
		identificationnumber		identificationnumber_cliente
		country		country_cliente

Fuente: elaboración propia.

fact_ventas: la tabla de hechos de ventas llamada “fact_ventas” es una tipo estrella, las tablas de dimensiones que se relacionan son d_v_detalleventa, d_v_detallenotascredito,

d_v_detallenotasdebito, d_v_detalleguiasremision, d_v_cliente y la dimensión tiempo, todas estas relacionadas en un modelo tipo estrella.

Figura 38. Modelo lógico fact_ventas



Fuente: elaboración propia.

Finalmente el modelo lógico de todo el Data Warehouse relaciona los modelos lógicos parciales para lograr relacionar toda la información de la empresa, tal como se muestra en la siguiente figura.

5.1.2.4 Integración de Datos

Cómo parte del diseño de *Data Warehousing*, la integración de datos es un proceso necesario para obtener los datos que son llenados en el *Data Warehouse*, dicha información fue extraída mediante sentencias SQL en la Herramienta Pentaho Data Integration (Instalación y configuración en Apéndice B), estas tareas se lograron programar de manera que la carga y actualización de datos sea automática en la noche y madrugada, en eventos diarios y sin intervención manual.

A continuación se muestra los procesos mencionados.

Carga Inicial

La carga inicial en la *Pentaho Data Integration* consta de varios procesos, primero es necesario crear una transformación, luego la selección y finalmente la inserción de información en la base de datos del *Data Warehouse*.


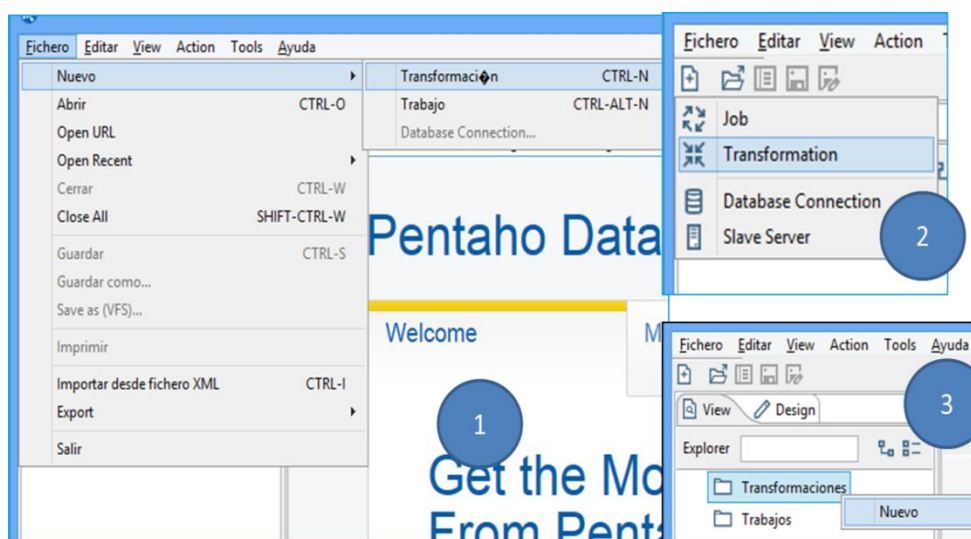
Creación de una transformación: La creación de una nueva transformación se puede realizar de tres formas distintas, una de ellas es desde la barra de menú en la opción  file/nuevo/transformación, la segunda por la barra de herramientas en el icono “nuevo”, al clicar en esta opción aparecerá el submenú *job, transformation, connection* y servidor. Y la tercera forma se la puede realizar a través del panel de vista, al clicar sobre la opción de transformación.

Figura 40. Creación de una nueva transformación

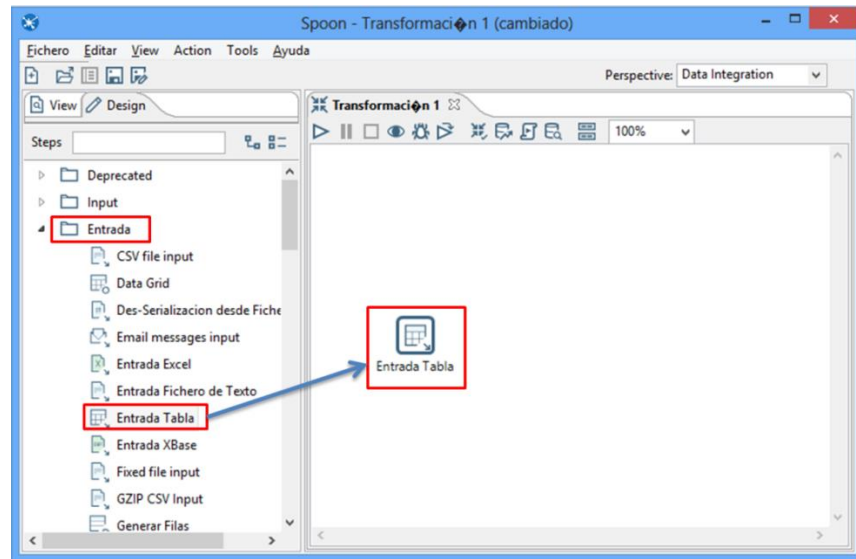


Fuente: elaboración propia.

Entrada

Toda transformación debe contar con una entrada de fuente de datos, sea esta una hoja de cálculo, tabla de una base de datos, documento de texto, etc. Se debe seleccionar la opción “Entrada/Entrada de tabla”. Para el caso de esta implementación la extracción va a ser desde tablas de bases.

Figura 41. Entrada de una fuente de datos



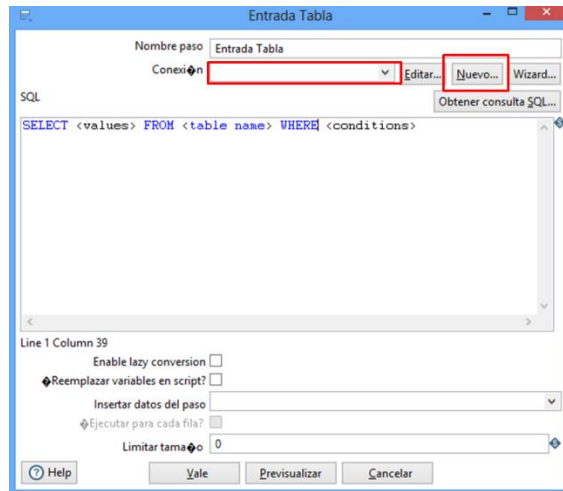
Fuente: elaboración propia.

Para la carga de datos se insertó un *Query* con el que extrajo datos de los orígenes. Cabe mencionar que en esta sección se puede cargar consultas tan simples como selecciones directas de una tabla o consultas con varias tablas con uniones y transformaciones incluidas.

Una vez insertado el objeto para la entrada de la tabla, es necesario configurar la conexión con la base de datos, este proceso se lo debe realizar tanto en la entrada tanto como en la inserción, pero una sola vez, puesto que ya creada la conexión, esta se puede reutilizar en las entradas de las otras cargas.

La conexión de entrada y salida son independientes y diferentes, pues en la entrada se conecta con el o los orígenes de datos, mientras que en el proceso de inserción la conexión se va a realizar con el *Data Warehouse*, para poder insertar la información en las tablas seleccionadas.

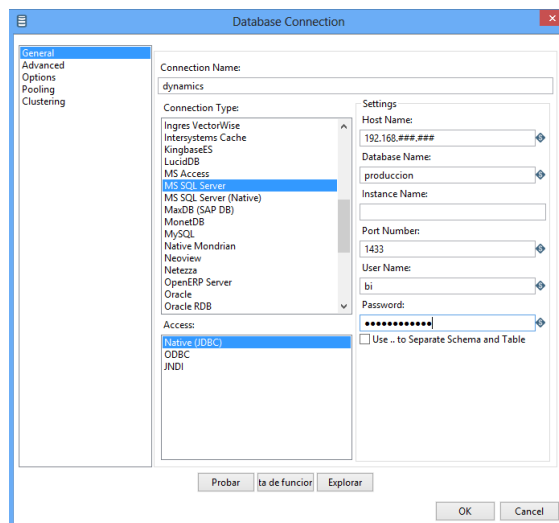
Figura 42. Carga de Datos.



Fuente: elaboración propia.

En la ventana de creación de la conexión, aparecen varios campos que deben ser llenados obligatoriamente, tales como nombre a la conexión, la dirección del servidor de base de datos, ya sea localhost o una dirección IP (caso de servidores externos), el nombre de la base de datos, el número del puerto (viene por defecto por el que generalmente escucha o se puede escribir otro número de puerto en caso de ser necesario), un nombre de usuario y la contraseña, este usuario debe tener privilegios elevados en la base de datos para que no se tenga problema con los queries de extracción.

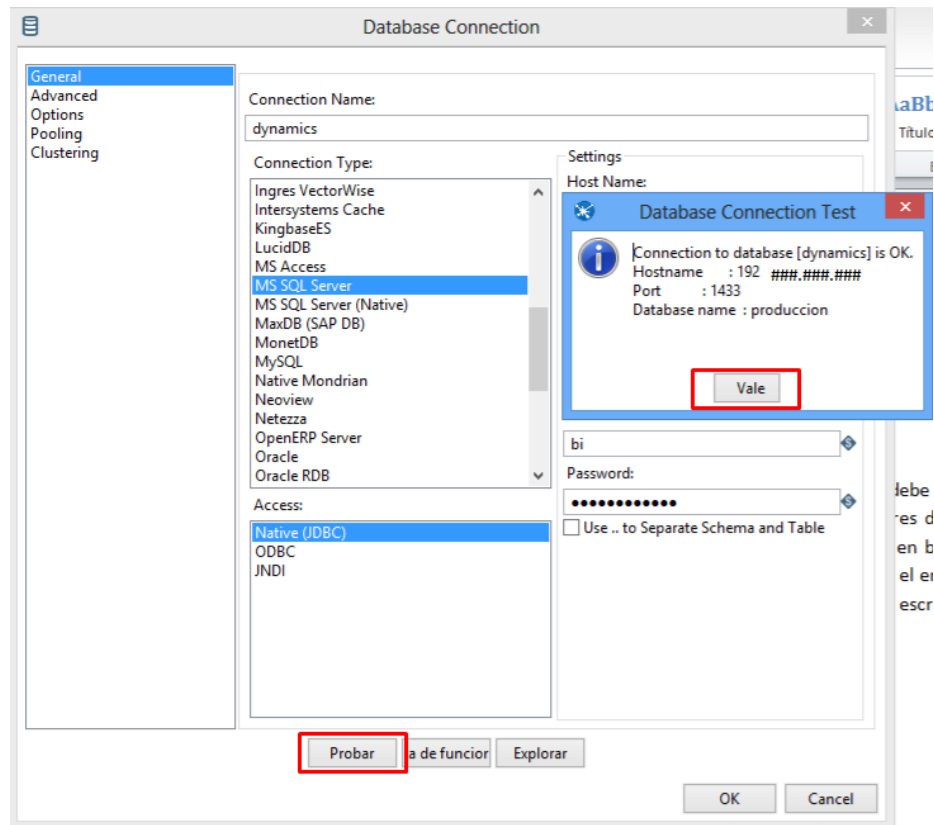
Figura 43. Conexión a la Base de Datos



Fuente: elaboración propia.

Si la conexión se la realizó adecuadamente, el test será exitoso. En ocasiones pudieren existir errores debido a que no se cuenta con los diferentes drivers de conexión, por lo general en bases de datos PostgreSQL y MySQL, este driver se lo puede descargar e instalar para solventar este error. Pueden existir otros tipos de erros que pueden originarse por los datos de conexión mal ingresados, lo importante es identificar el error con los mensajes que la herramienta provee.

Figura 44. Error de conexión

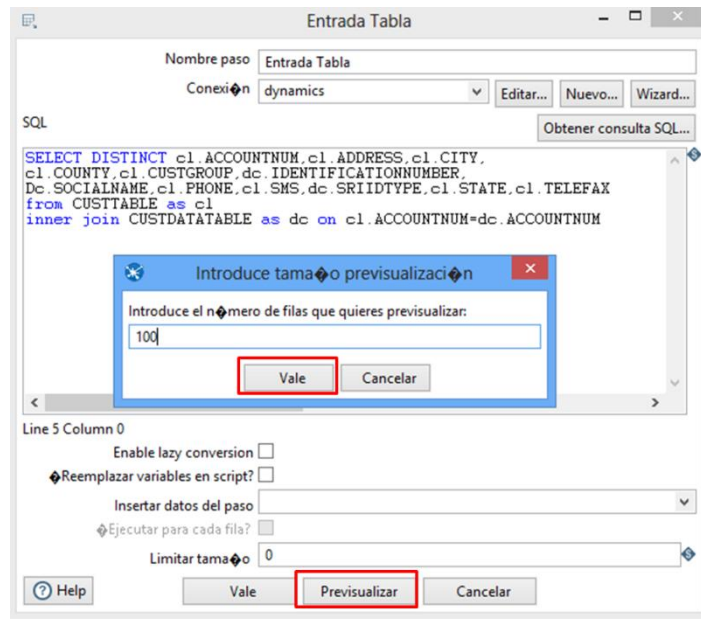


Fuente: elaboración propia

Una vez configurada la conexión, se deberá dar clic en **VALE** y en **OK** para terminar el proceso de la conexión.

El siguiente paso es insertar una sentencia SQL con el fin de extraer los datos que se requieren extraer, si el query correcto, se puede realizar una pre visualización de los resultados y verificar si la información que se está va a extraer es correcta. Los queries utilizados en las extracciones están descritos en el apéndice “D”.

Figura 45. Pre visualización de resultados

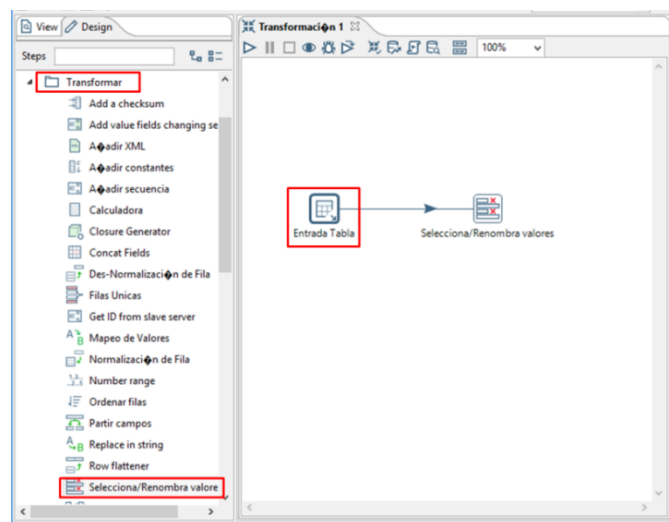


Fuente: elaboración propia

Transformación

En el proceso de transformación se puede personalizar o cambiar los tipos de datos, nombres e inclusive estructuras, se lo realiza en la opción “transformar”, en la subcategoría “selecciona/renombra valores”, se inserta ésta tarea y se enlaza a la tarea de “carga de la tabla”, puesta anteriormente.

Figura 46. Filtrado de datos

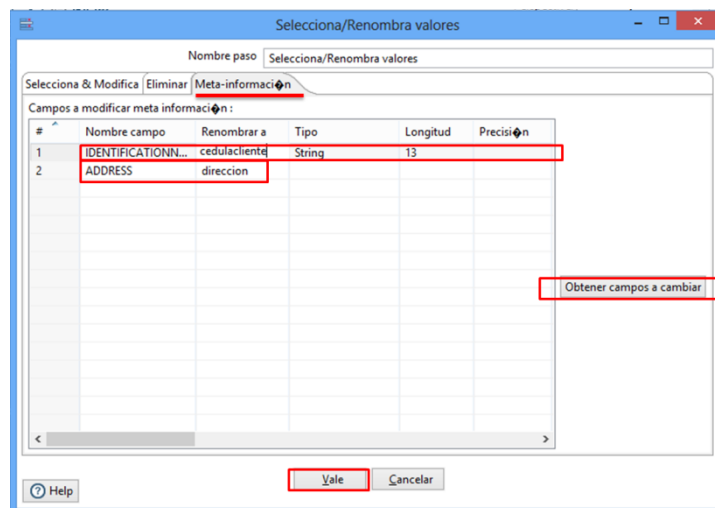


Fuente: elaboración propia

En la pestaña de meta-información se renombra los datos que se desea transformar, es posible cambiar el nombre, tipos de datos y longitud del mismo. Generalmente esta

opción es usada en fechas, donde los orígenes de datos están guardados en un formato de fecha y el destino tiene otro formato, por las características propias del Data Warehouse. El cambio de nombres es útil para poder reconocer campos por la tabla en la que va a ser almacenado y la longitud ayuda para optimizar la memoria.

Figura 47. Transformación de tipo de datos

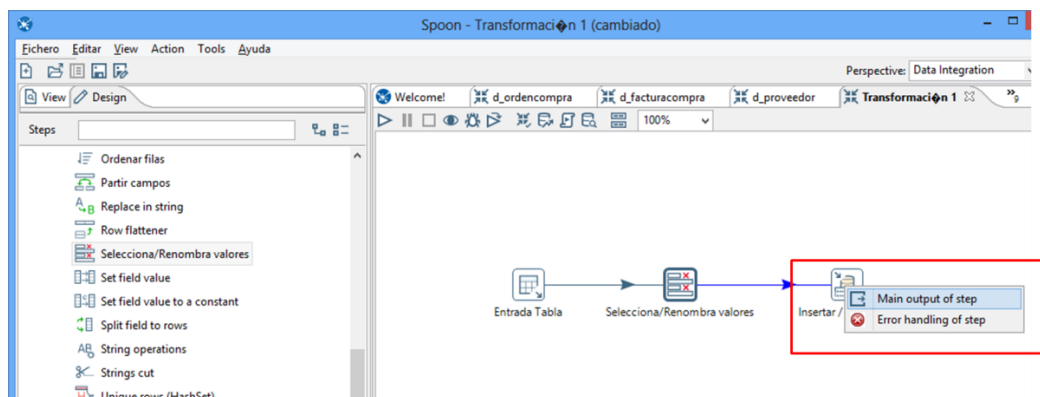


Fuente: elaboración propia,

Inserción de Datos

Finalmente para la inserción, se agrega el destino de la base que será quien almacene la información del *data Warehouse*, este proceso se le encuentra en la categoría “**salida**”, dentro de esta se encuentra la sub categoría “**insertar/actualizar**”, se debe agregar esta tarea y vincularla tal como se hizo con las dos anteriores.

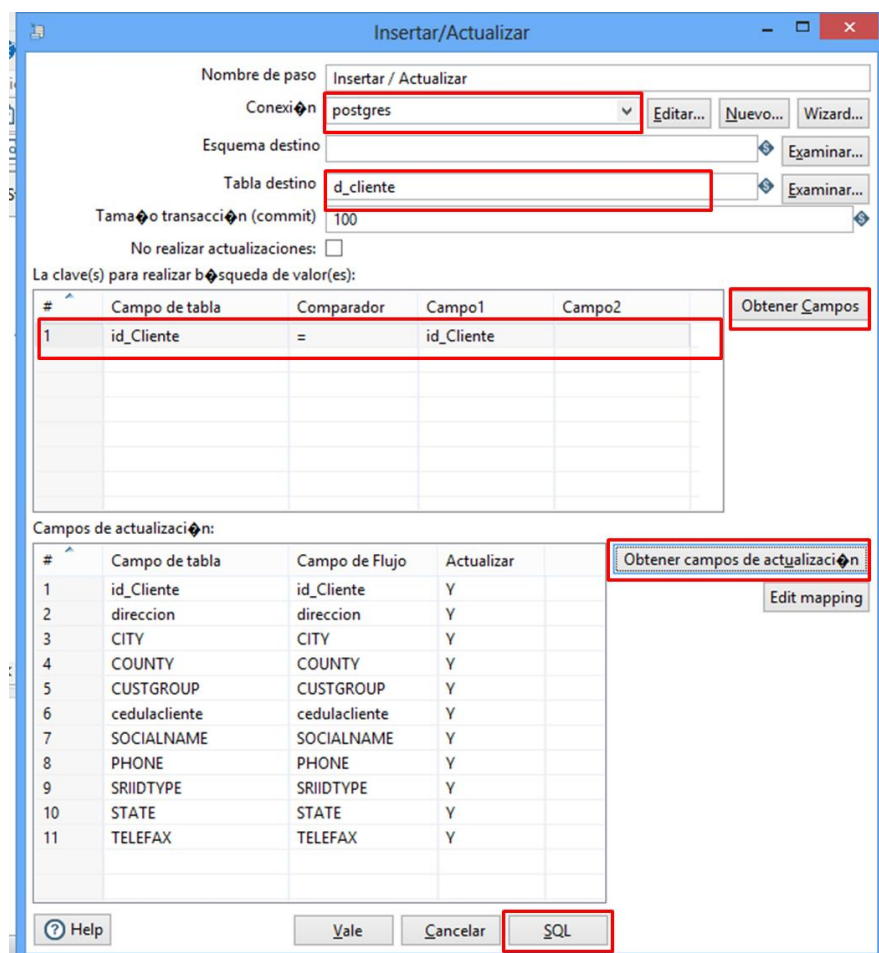
Figura 48. Carga de Datos



Fuente: elaboración propia

En el proceso de Insertar/Actualizar cómo ya se dijo, es necesario crear una conexión hacia la base de datos, una vez realizada esta conexión (proceso igual al de la carga, solo cambia la base de datos de destino) y sin olvidar que la base de datos destino debe estar creada, es decir el *Data Warehouse*, se procede a seleccionar la tabla en la que se desea almacenar la información extraída y transformada. Otro campo obligatorio a llenar son los id's o claves foráneas, donde se debe indicar que campo va a ser la clave, para evitar redundancias, lo siguiente es clicar en el botón "Obtener campos de actualización", esta acción extrae los campos del select insertado en la entrada y que van a ser grabados en la base del *Data Warehouse*. Finalmente se da clic en el botón "SQL" para que se valide la inserción.

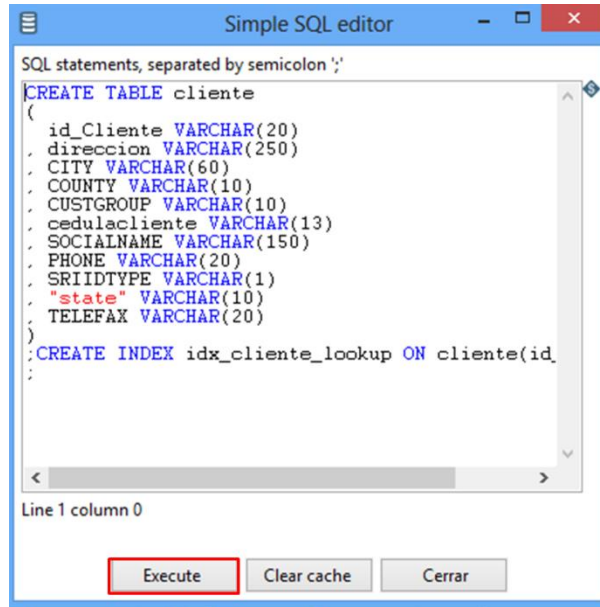
Figura 49. Creación de la Tabla desde Spoon



Fuente: elaboración propia.

El paso "SQL" se debe ejecutar una sola vez para cada tabla, si se lo hace más de una vez, empezará a aparecer errores debido a que la instancia ya fue creada.

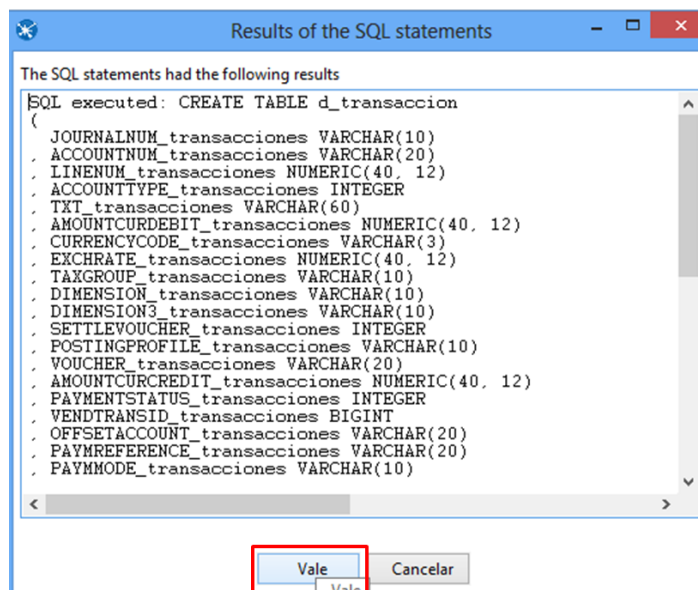
Figura 50. Ejecución Script creación tabla



Fuente: elaboración propia.

Una vez ejecutado el script aparecerá un mensaje el mismo que indicará si el script se ejecutó satisfactoriamente.

Figura 51. Mensaje de ejecución de Script.



Fuente: elaboración propia.


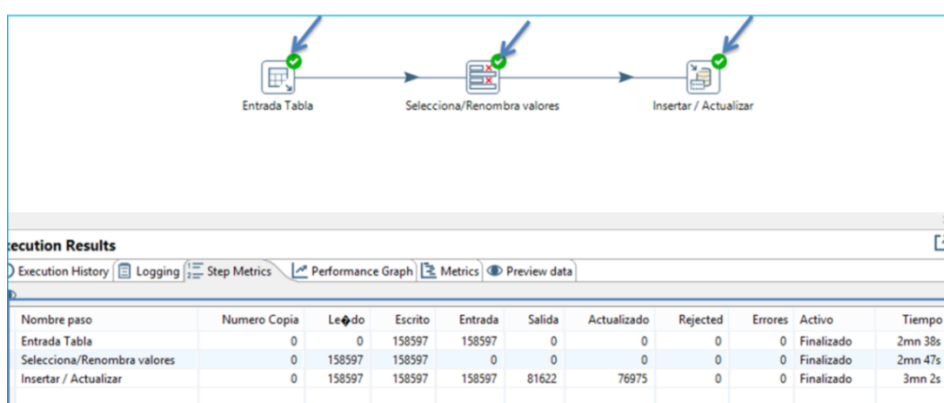
Una vez concluida la configuración se debe ejecutar la transformación al dar clic sobre el ícono , en la barra de herramientas principal, el mismo que iniciará la transformación. El proceso es exitoso cuando aparece un visto en la entrada, transformación y en la barra de resultados indica el tiempo que se demoró en realizar cada uno de los pasos y el número de registros que fueron afectados.

Figura 52. Tabla de resultados de la extracción de datos



Fuente: elaboración propia

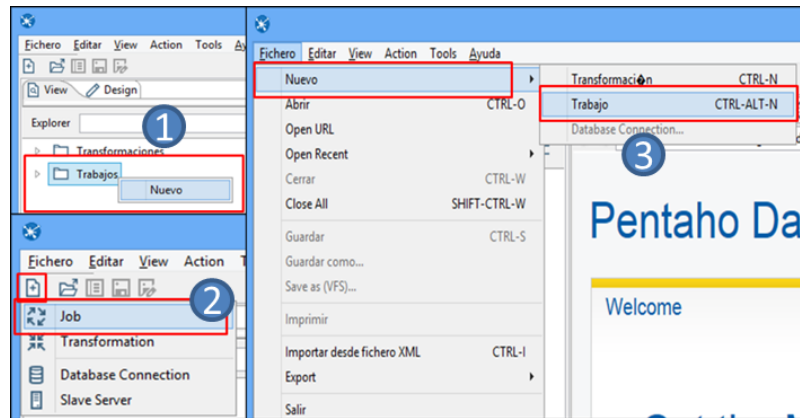
Conclusión: Éstos pasos deben crearse por cada carga, una sola vez, posterior a esto solo se los reutiliza en la actualizaciones.

Actualización Programada de Datos

Para la actualización de datos una vez que se creó por primera vez las tablas y estructura del *Data Warehouse*, resta únicamente crear el proceso de actualización de datos, para ello se creó trabajos programados, que se mantiene en horarios que no interrumpen en normal desempeño de los sistemas orígenes de información.

El primer paso es configurar un trabajo para ello se utilizará la misma aplicación de Kettle de Pentaho, de la misma forma en que se creó una transformación se debe crear un trabajo, tal como se muestra en la siguiente imagen:

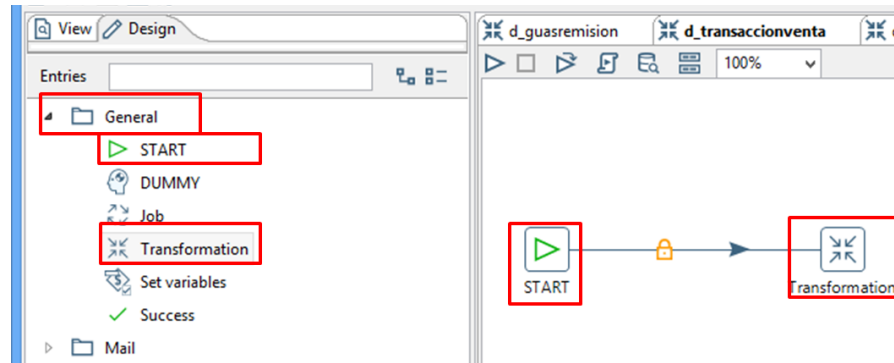
Figura 53. Creación de un Trabajo en Kettle.



Fuente: elaboración propia.

Se procede a insertar los objetos “*Start*” y “*Transformation*”, estos dos objetos van a ser necesarios para programar las tareas automáticas.

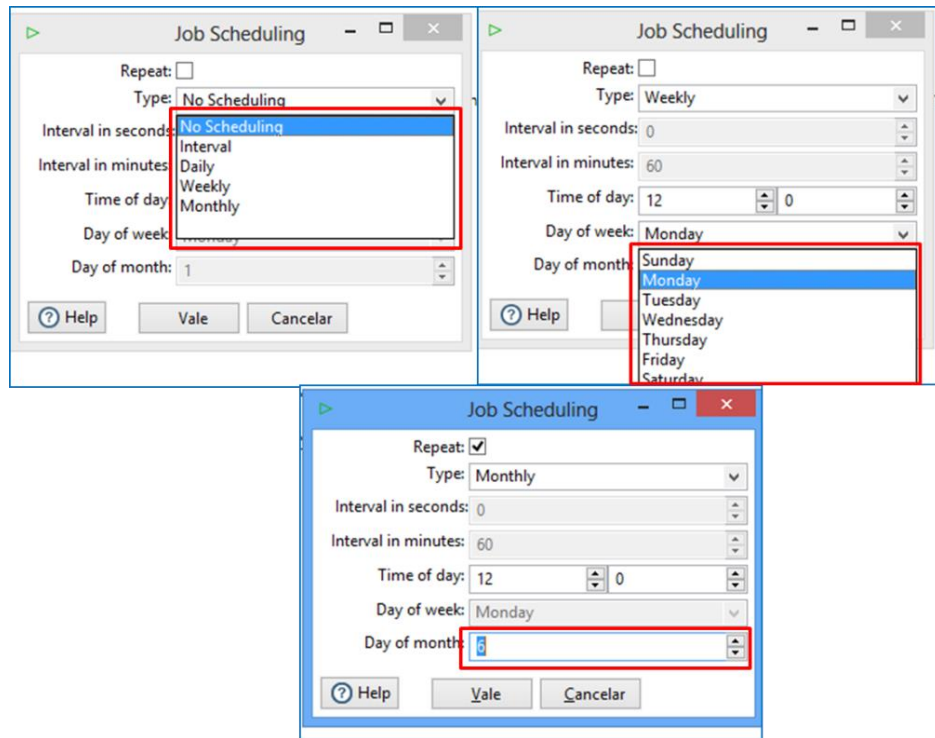
Figura 54. Objetos para crear un trabajo automático



Fuente: elaboración propia

En el objeto **START** permitirá que se programe con qué frecuencia se desea actualizar el Data Warehouse, sea esta diariamente, semanal, mensual etc., o es su defecto es posible ponerlo en días específicos. También se puede escoger la hora que se desea iniciar la carga, esto cómo se había mencionado, para no interferir con los OLTP transaccionales y el rendimiento de los sistemas orígenes y la red interna de datos. A continuación la programación que se hizo en uno de los trabajos programados.

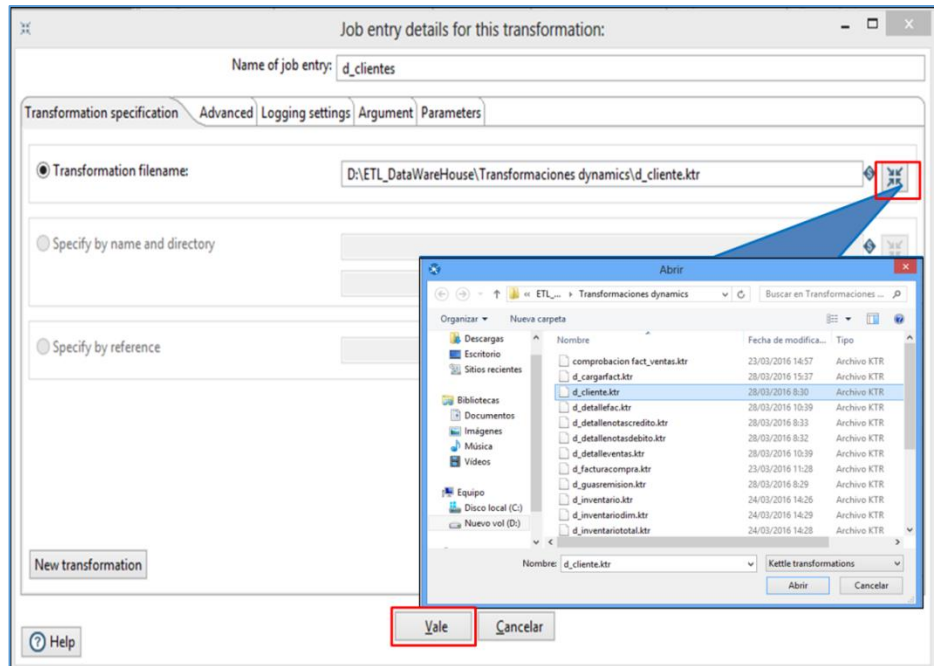
Figura 55. Programación de las transformaciones.



Fuente: elaboración propia.

En el objeto transformación se debe colocar un nombre que referencie a la transformación a la cual va a ejecutar o al conjunto de transformaciones que ejecutará, el mismo que se puede crear en ese instante o se lo puede traer del lugar donde se almacenó, para éste caso las transformaciones están en archivos independientes así en caso de dañarse la tarea, la transformación se mantendrá intacta y todos los pasos previos de selección, etiquetado y carga. Para poder realizar esta acción aparecerá una ventana de diálogo con la cual se podrá navegar por el computador, una vez seleccionado el archivo de carga, se acepta la selección.

Figura 56. Selección de la transformación a programar



Fuente: elaboración propia.


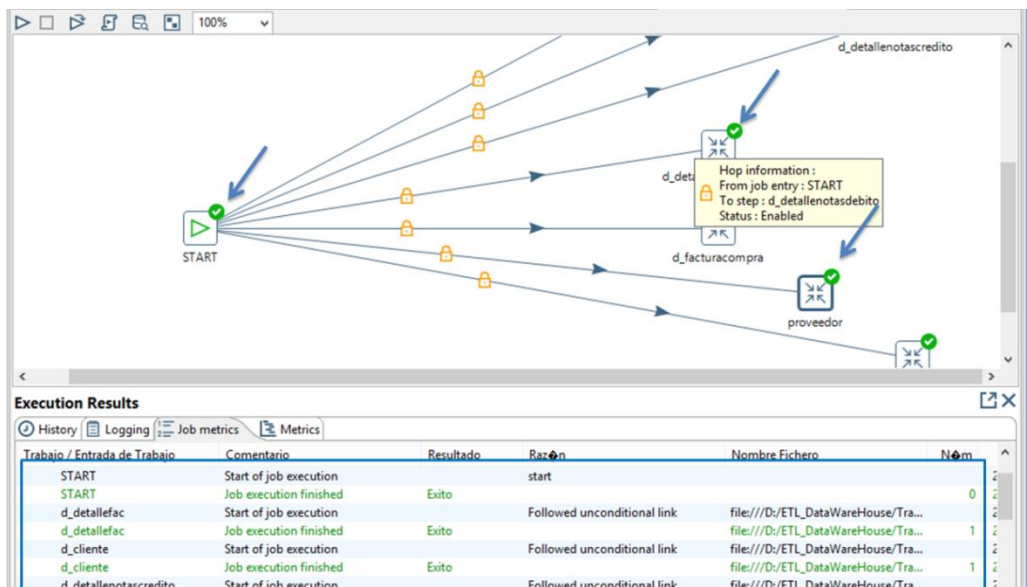
Finalmente luego de haber configurado el proceso de carga automática, se lo debe iniciar al dar clic en el ícono  una vez que se haya ejecutado aparecerá con un visto verde sobre el proceso o se marca con un X roja en caso de existir, tal como se muestra en la siguiente figura.

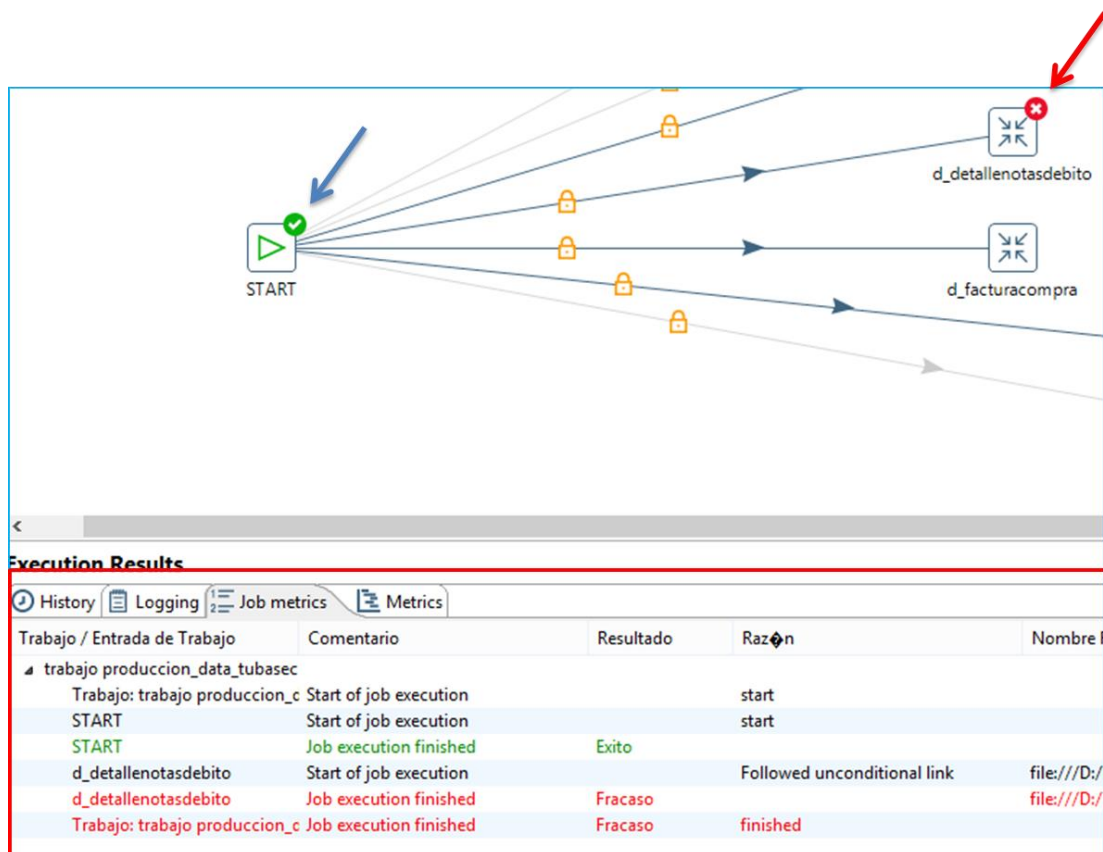
Figura 57. Ejecución del trabajo de forma exitosa



Fuente: elaboración propia

Conclusión: Este trabajo puede incluir uno o varios procesos, es decir se pueden agregar varias cargas, sin embargo es importante hacerle en paralelo para conocer en qué parte se produce un error en caso de existir alguno.

Figura 58. Errores de tarea inconclusa de estación



Fuente: elaboración propia

Presentación de Información en la herramienta de BI

Cómo se mencionó anteriormente los reportes fueron generados en Qlik, específicamente Qlik Sense que es la versión gratuita de Qlik View, la herramienta es bastante fácil de operar, a continuación se muestra cómo se implementa un indicador para posteriormente analizar los resultados de todos los indicadores presentados.

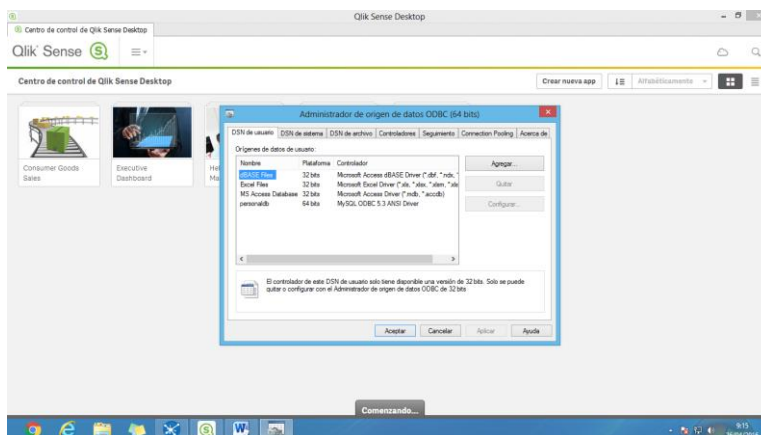
Creación de Indicadores

Los indicadores, gráficos estadísticos, tablas y demás fueron implementados en Qlik sense, esta herramienta fue descargada de manera gratuita de la página oficial de Qliksoft¹.

¹ http://www.cliksoft.com.ec/seccion/16/descarga_gratuita

Luego de descargar e instalar Qlik Sense (ver apéndice C), primero se debe empezar con la creación de la conexión con la base de datos, estas conexiones se las realiza mediante un ODBC, por lo que se procede a crear una conexión con fuentes ODBC de Windows.

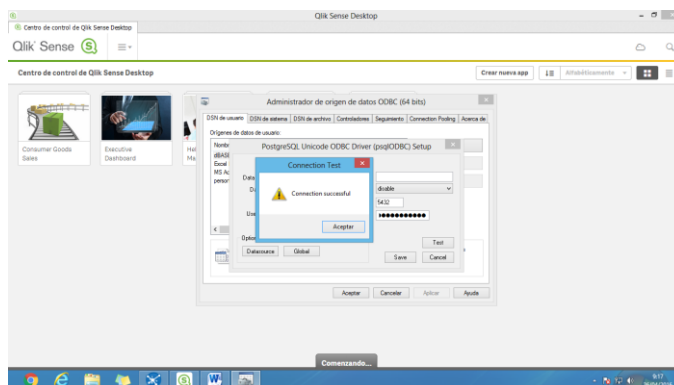
Figura 59. Conexión ODBC



Fuente: elaboración propia

Si la conexión se creó correctamente se mostrará un cuadro de mensaje cómo el que se presenta a continuación

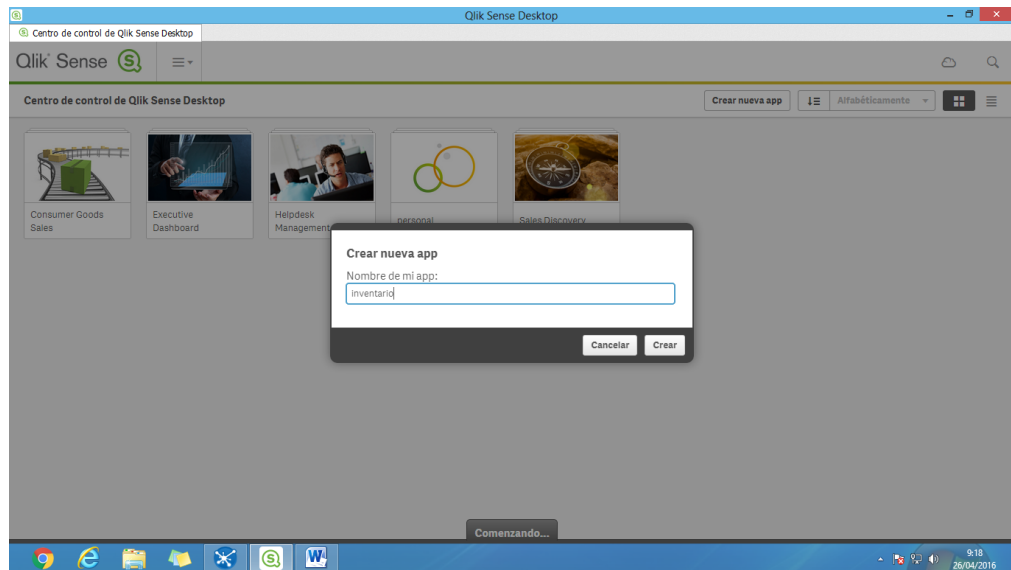
Figura 60. Conexión de ODBC exitosa



Fuente: elaboración propia

Posterior a esto se abre el programa Qlik Sense para verificar que esté funcional, la pantalla inicial muestra la creación de una nueva app sobre la que se debe empezar a trabajar el reporte o indicador deseado.

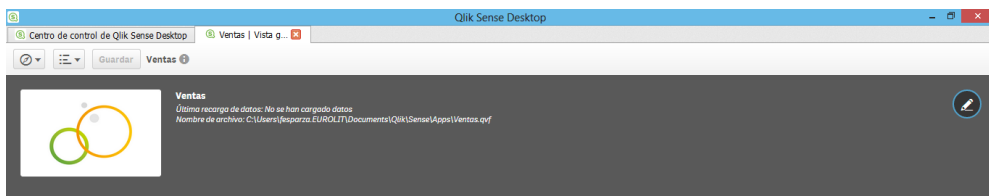
Figura 61. Aplicación Qlik Sense





Fuente: elaboración propia

Al escribir un nombre y clicar en el botón crear, se presenta la pantalla de bienvenida, tal como la que se aprecia en la siguiente gráfica:

Figura 62. Aplicación creada en Qlik Sense.



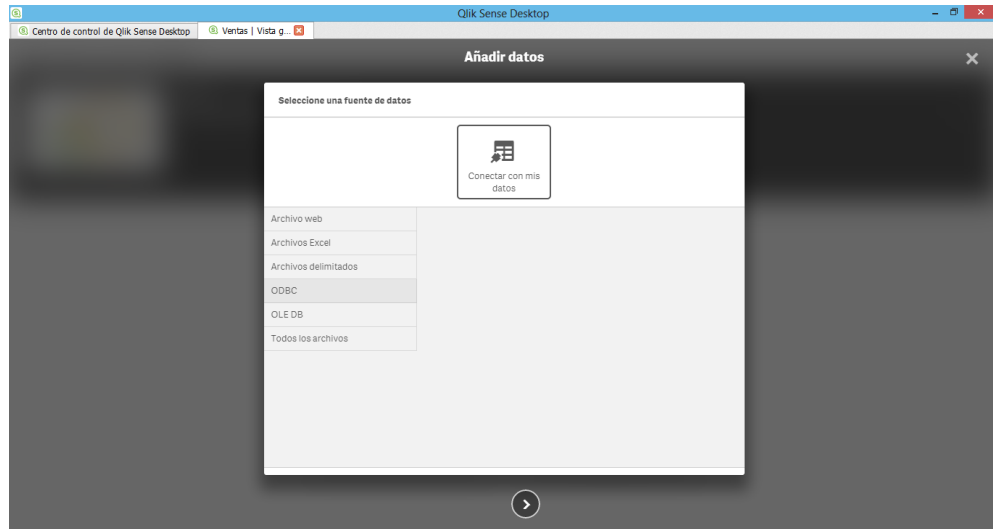
Comience añadiendo datos a su app.

 <p>Añadir datos Añada datos de un archivo, de una base de datos o de Qlik DataMarket.</p>	 <p>Editor de carga de datos Cargue datos desde archivos o bases de datos y proceda a la transformación de los datos mediante el script de carga de datos.</p>
--	--

Fuente: elaboración propia

Al ingresar en la opción “Añadir datos”, aparece la configuración para insertar la conexión a la base de datos, que para el caso es ODBC.

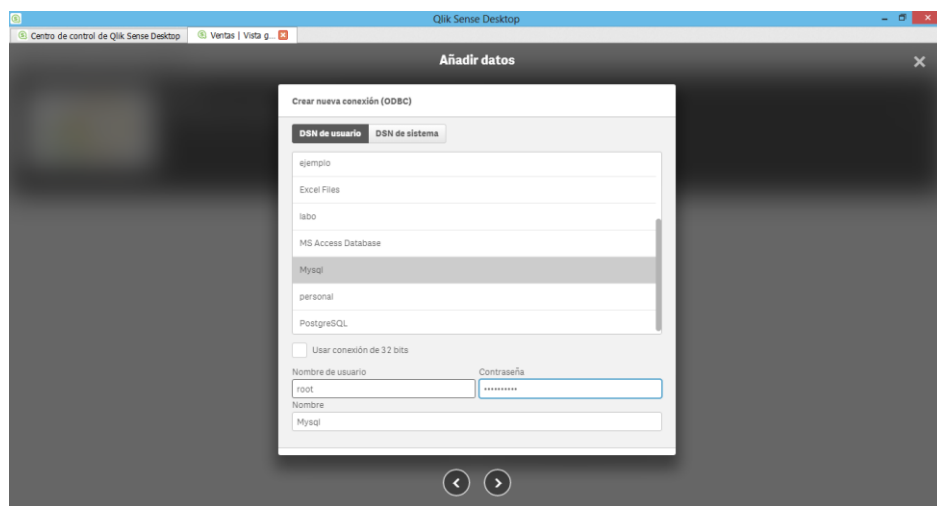
Figura 63. Añadir datos Qlik Sense.



Fuente: elaboración propia

Lo que el programa solicita para continuar es la configuración del ODBC creado en pasos anteriores, aquí se debe llenar los datos respectivos.

Figura 64. Configuración ODBC en Qlick Sense

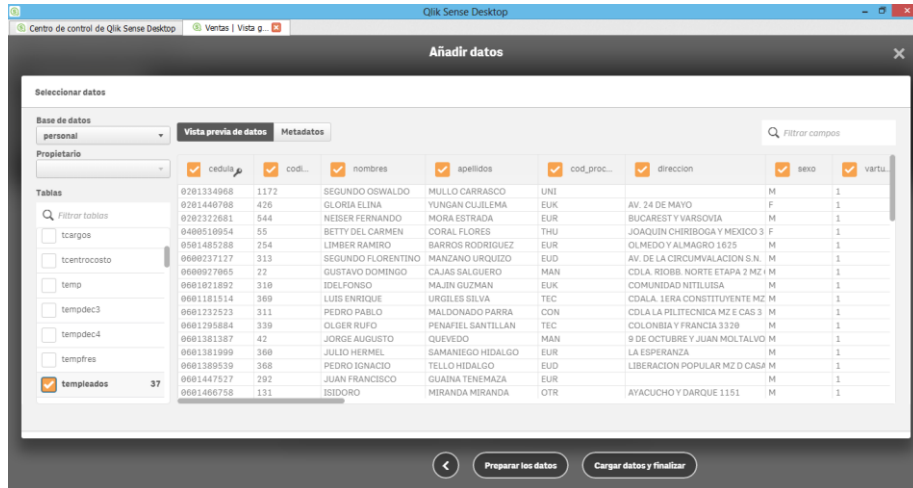


Fuente: elaboración propia

En la siguiente configuración se debe asignar la base de datos a la que se desea conectar y la tabla que corresponda, para poder seleccionar los campos respectivos y empezar a

formular los indicadores y reportes. Una vez seleccionado lo necesario se debe clicar en el botón “Cargar datos y finalizar”

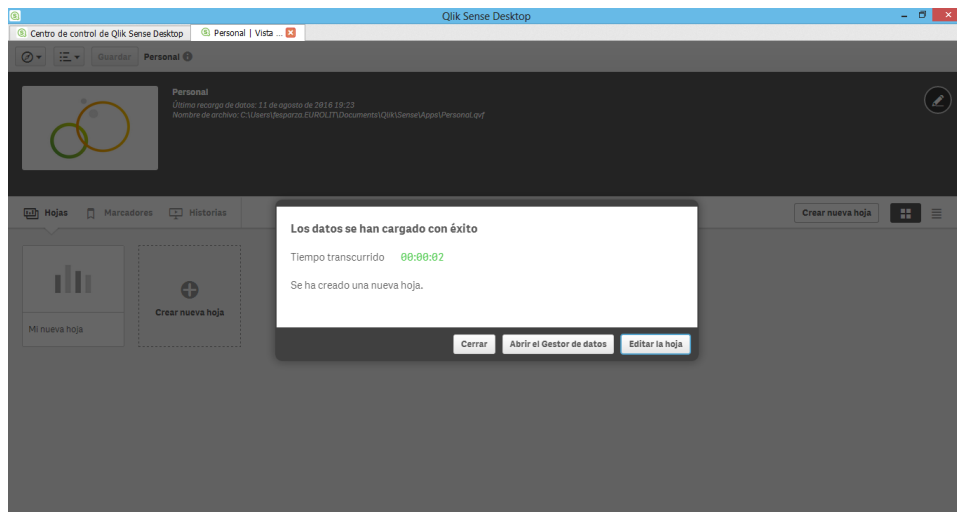
Figura 65. Selección Base de Datos Qlik Sense.



Fuente: elaboración propia

Con esta acción se presenta una pantalla cómo la mostrada a continuación y se debe clicar en la opción “Abrir Gestor de datos”.

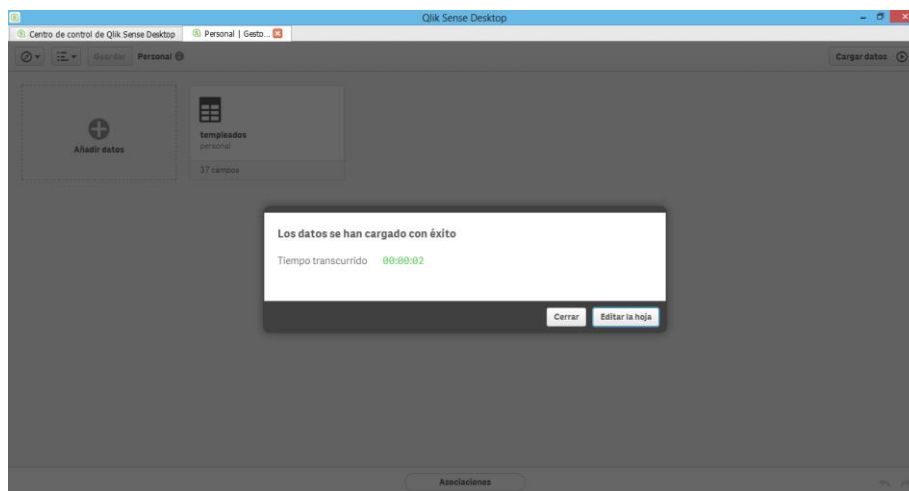
Figura 66. Abrir Gestor de datos Qlik Sense



Fuente: elaboración propia

La siguiente acción que se debe realizar es cargar datos en el botón que aparecerá en la parte superior derecha de la pantalla y una vez concluido este proceso se tiene algo similar a la siguiente figura.

Figura 67. Carga de datos Qlik Sense.



Fuente: elaboración propia

Finalmente se clikea en la opción "Editar la hoja", para empezar a ingresar datos y formar tablas, cuadros estadísticos, indicadores, etc.

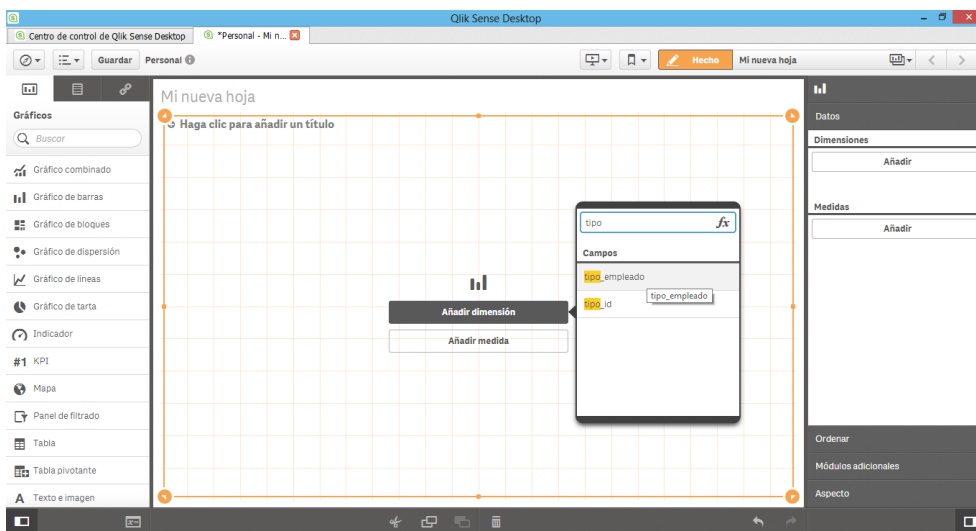
Figura 68. Editar Hoja en Qlik Sense



Fuente: elaboración propia

Lo siguiente es arrastrar los objetos necesarios y comenzar a dimensionarlos, relacionarlos y agruparlos, en este caso se va a arrastrar un objeto de barras y la dimensión añadida es “tipo empleado”

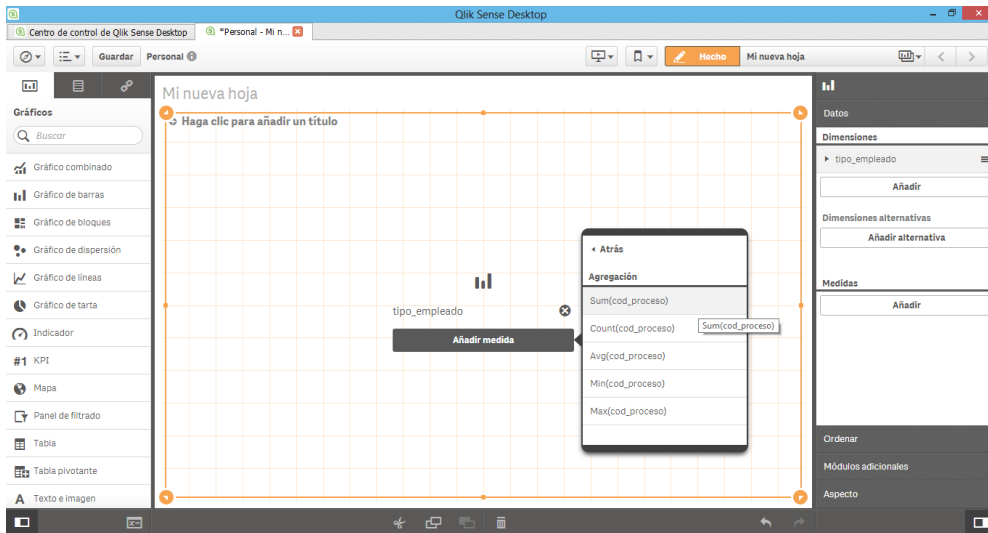
Figura 69. Añadir dimensión Qlik Sense



Fuente: elaboración propia

Luego se añade la dimensión y el tipo de agrupación que se desea, en este caso se va a sumar la cantidad de empleados por proceso.

Figura 70. Añadir agregación Qlik Sense.

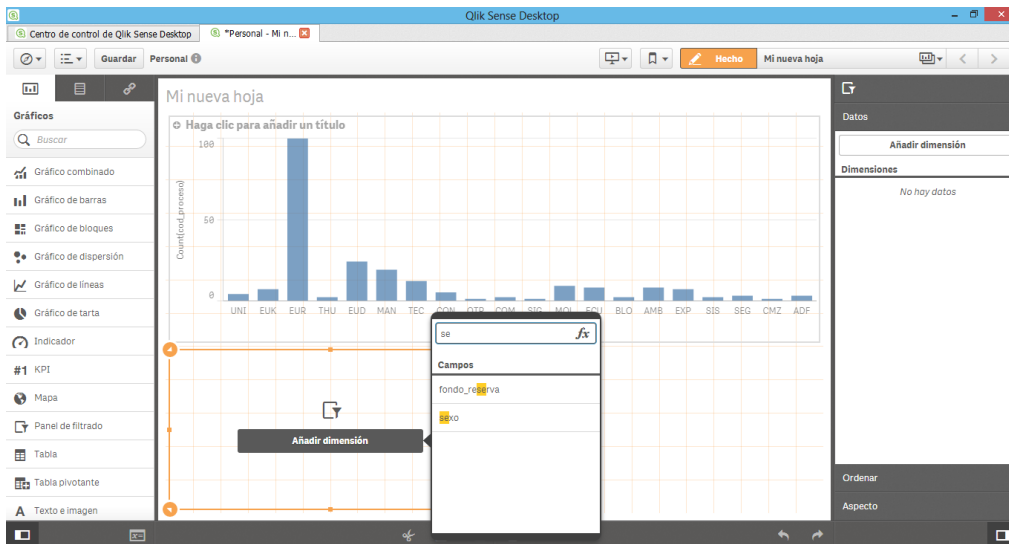


Fuente: elaboración propia

Una vez que se tiene lo deseado, se clikea en el botón “Hecho” para visualizar los resultados y verificar si está bien elaborado el indicador / reporte.

Los filtros para segregar información se pueden agregar por el objeto “Panel de Filtrado”, este permite dar opción al usuario de filtrar información y mostrarla en el cuadro presentado de estadísticas.

Figura 71. Añadir filtros Qlik Sense.



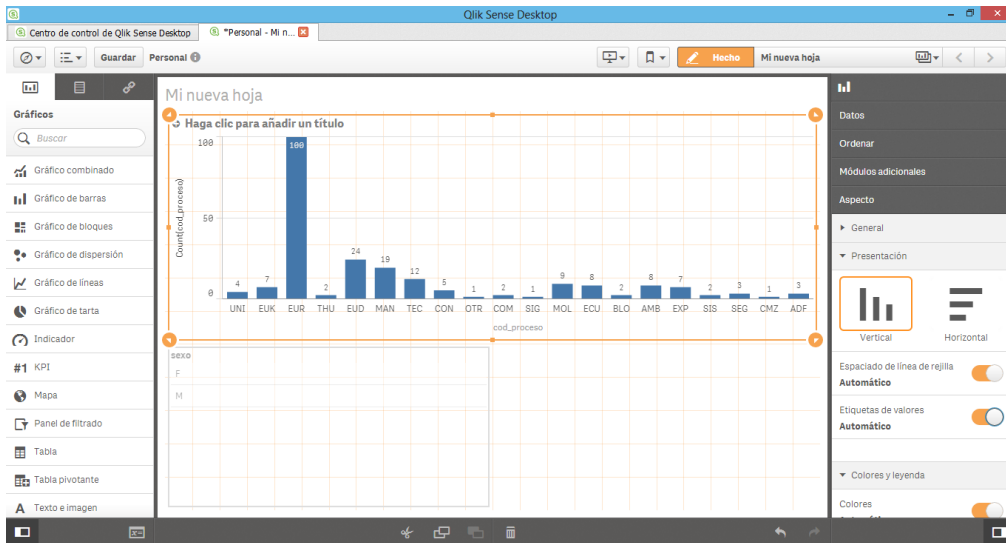
Fuente: elaboración propia

Se puede colocar la cantidad de filtros que se consideren necesarios con el fin de mejorar la usabilidad que se le pueda dar al reporte.

Para personalizar los datos que se muestran, en la parte derecha de la pantalla se puede personalizar la presentación de los gráficos, esto se encuentra en la opción “Aspecto”, para el

ejemplo se activará la opción de etiquetas de valores que normalmente vienen desactivadas, también se puede personalizar tipo de gráficos, colores, etiquetas, etc.

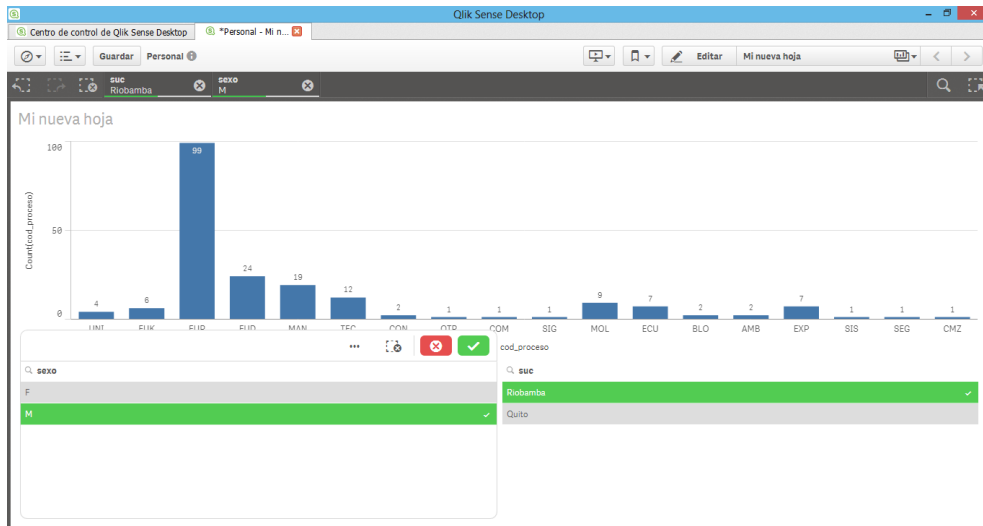
Figura 72. Aspecto gráficas Qlik Sense.



Fuente: elaboración propia

Una vez incluidos los filtros y personalizados se tendrá un aspecto cómo el que se puede observar a continuación, en donde se ha filtrado por ciudad y por el sexo del empleado.

Figura 73. Personal por ciudad y por sexo



Fuente: elaboración propia

Implementación de Indicadores

Cómo se planteó inicialmente, se implementaron varios indicadores, algunos de ellos ya se los ha utilizado de manera manual y otros se los creó con la implementación del Data Warehouse. A continuación los detalles de cada uno.

Indicadores de Ventas

Los indicadores permiten a la dirigencia conocer la cantidad de productos que se venden para en base a los resultados aumentar o disminuir la producción de determinados productos, los indicadores implementados son:

- **Valor Facturado**

Objetivo

Valor procedente de la venta de productos, quitándole el valor de Notas de Crédito y se suma el valor de Notas de Débito, en un determinado período de tiempo

Medidas

Para generar el indicador de ventas fue necesario sumar el monto facturado con el valor de notas de débito y restarlo del valor de las notas de crédito

Valor real = $\text{sum}(\text{montofac}) + \text{sum}(\text{montonc}) + \text{sum}(\text{montond})$

Filtros

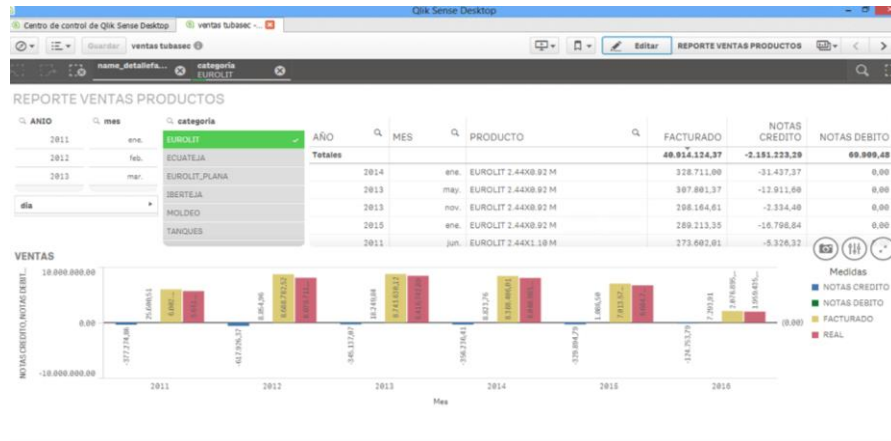
Se agregaron filtros para delimitación temporal y de productos

- Categoría de producto
- Mes
- Año

Dimensiones

La dimensión que se ocupó para comparación de ventas fueron los años
 $\text{year}(\text{fecha_relacion})$

Figura 74. Indicador de Ventas – Valor Facturado.



Fuente: elaboración propia

- **Expediciones por Peso**

Objetivo

Valor en Kilos de cada producto vendido en un período de tiempo determinado.

Medidas

Para generar el indicador de expediciones por peso fue necesario sumar la cantidad facturada con el valor de notas de débito y restarlo del valor de las notas de crédito, y todo esto multiplicarlo por el peso del producto seleccionado.

$$(\text{sum}(\text{cantidadfactura})+\text{sum}(\text{cantidadnc})+\text{sum}(\text{cantidadnd}))\cdot\text{peso}$$

Filtros

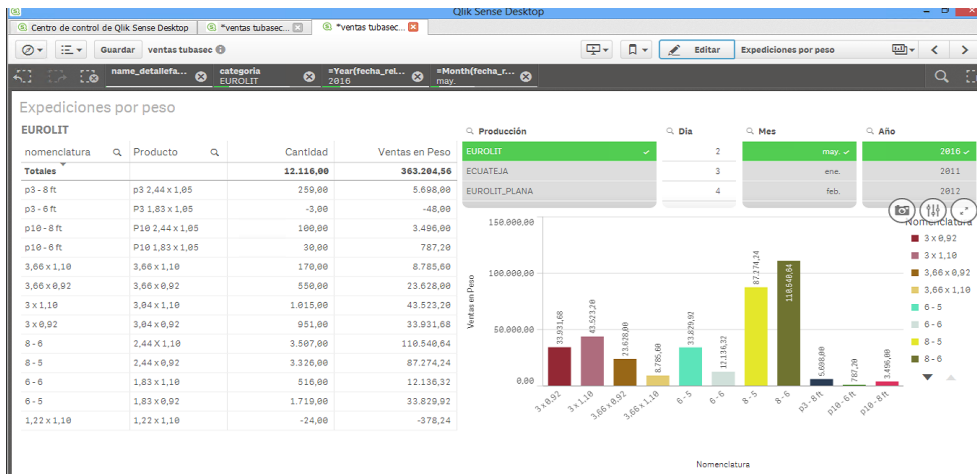
Se agregaron filtros para delimitación temporal y de productos

- Categoría de producto
- Mes
- Año

Dimensiones

La dimensión que se ocupó para comparación de expediciones por peso fue la nomenclatura, es decir el producto.

Figura 75. Indicador de Ventas – Expediciones por peso.



Fuente: elaboración propia

- **Porcentaje de ventas por peso**

Objetivo

Porcentaje que representa un producto sobre el total de ventas en un período de tiempo determinado.

Medidas

Para generar el indicador de porcentaje de ventas por peso fue necesario se sumó la cantidad en peso de lo vendido con las notas de débito y restarlo de la cantidad en peso de las notas de crédito

$$\text{sum(cantpeso)} + \text{sum(cantpesonc)} + \text{sum(cantpesond)}$$

Filtros

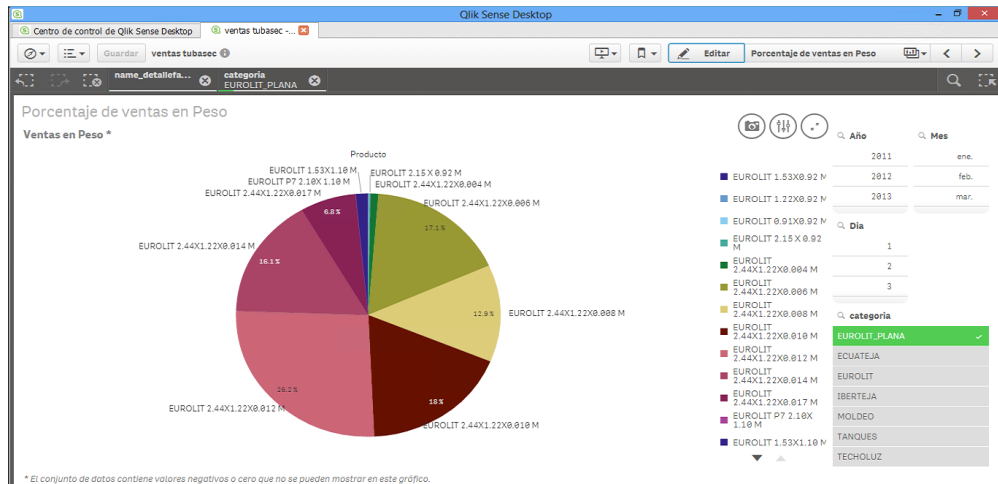
Se agregaron filtros para delimitación temporal y de productos

- Categoría de producto
- Mes
- Año

Dimensiones

La dimensión que se ocupó para comparación de ventas fue el producto “nombreproducto”

Figura 76. Indicador de Ventas – Porcentaje de Ventas por peso.



Fuente: elaboración propia

- **Porcentaje de ventas**

Objetivo

Indica el porcentaje de ventas de un determinado producto.

Medidas

Para generar el indicador de porcentaje de ventas fue necesario se sumó la cantidad vendida con la cantidad de notas de débito y restarlo de la cantidad de las notas de crédito.

$$\text{sum}(\text{cantidadfactura})+\text{sum}(\text{cantidadnc})+\text{sum}(\text{cantidadnd})$$

Filtros

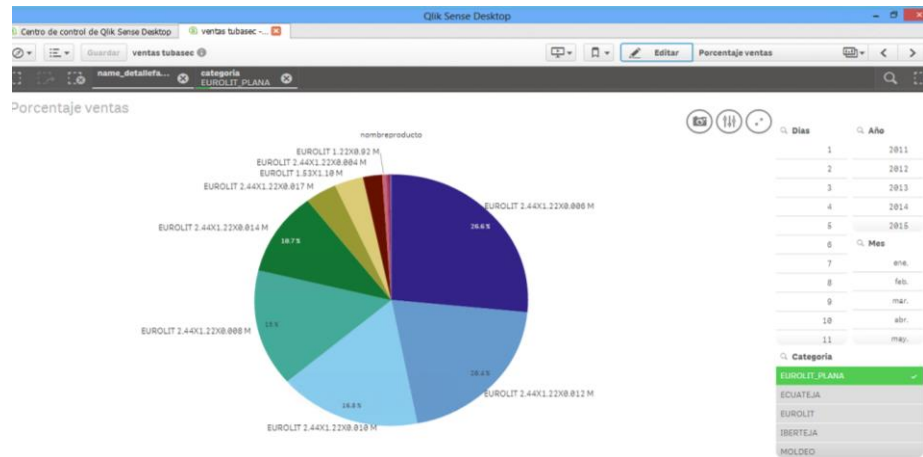
Se agregaron filtros para delimitación temporal y de productos

- Categoría de producto
- Mes
- Año

Dimensiones

La dimensión que se ocupó para comparación de ventas fue el producto “nombreproducto”

Figura 77. Indicador de Ventas – Porcentaje de Ventas por peso



Fuente: elaboración propia

- **Porcentaje de devoluciones**

Objetivo

Porcentaje de devoluciones de un producto en relación con las devoluciones totales determinado en un período de tiempo establecido.

Medidas

Se suma el número de devoluciones generadas.
Sum(cantidadnc)

Filtros

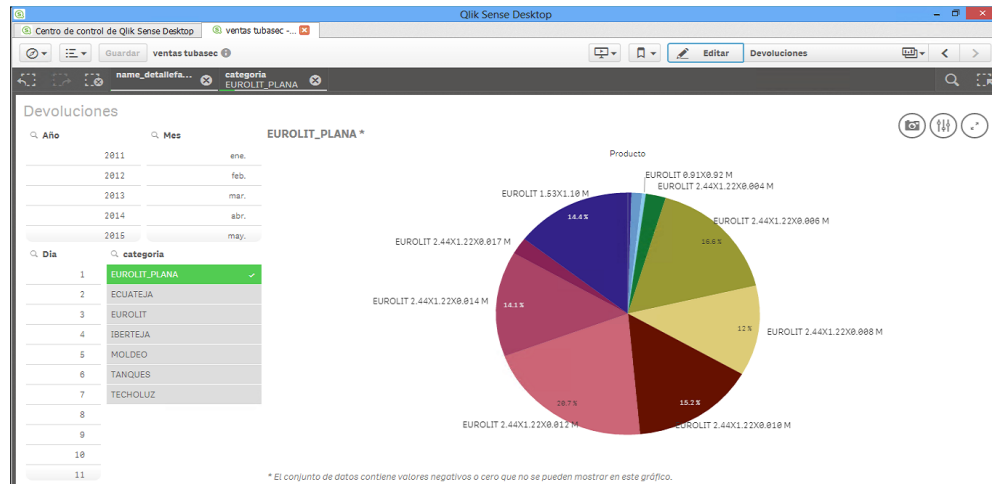
Se agregaron filtros para delimitación temporal y de productos

- Categoría de producto
- Mes
- Año
- Día

Dimensiones

La dimensión que se ocupó para comparación de ventas fue el producto “nombreproducto” y “año”

Figura 78. Indicador de Ventas – Porcentaje devoluciones



Fuente: elaboración propia

Indicadores de Producción

- **Producción e inventario.**

Objetivo

Inventarios: Cantidad de productos en patios aptos para la venta, por producto o por conjunto de artículos, en un período de tiempo determinado.

Producción: Sumatoria de Toneladas Producidas en un período de tiempo determinado, clasificado por producto.

Medidas

Inventario: Se suma el número de toneladas producidas y se divide para el peso del producto.

$\text{Sum}(\text{toneladas_produccion}) / \text{netweight_inventario}$

Producción: Se suma la cantidad de toneladas producidas

$\text{Sum}(\text{toneladas_produccion})$

Filtros

Se agregaron filtros para delimitación temporal y de productos

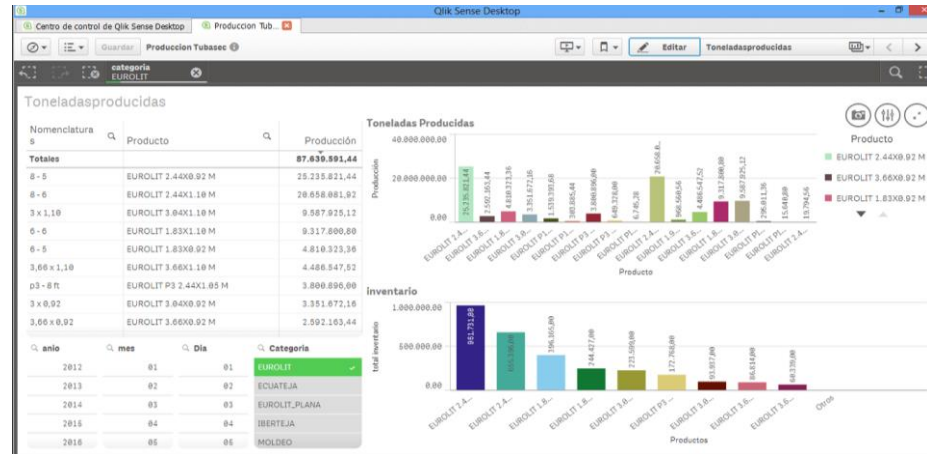
- Categoría de producto
- Mes
- Año
- Día

Dimensiones

Inventarios: La dimensión para comparación de inventarios es “nombreproducto”

Producción: La dimensión para comparación de producción es “name_produccion”

Figura 79. Indicador de Producción – Producción e Inventario



Fuente: elaboración propia

- **Porcentaje de producción.**

Objetivo

Ver el porcentaje de producción de un determinado producto, comparándolo con la producción total, en un rango de fechas definidas por el usuario.

Medidas

Se suma la cantidad de toneladas producidas, pero mostradas en porcentaje
Sum(toneladas_produccion)

Filtros

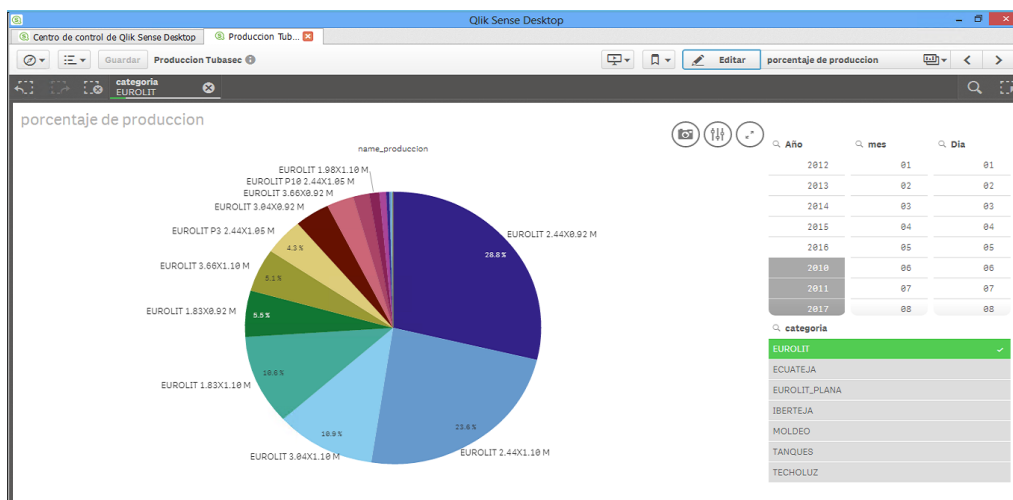
Se agregaron filtros para delimitación temporal y de productos

- Categoría de producto
- Mes
- Año
- Día

Dimensiones

La dimensión para comparación de producción es “name_produccion”

Figura 80. Indicador de Producción – Porcentaje de producción



Fuente: elaboración propia

- **Porcentaje de Inventarios.**

Objetivo

Porcentaje de productos disponibles para órdenes ingresadas de un producto en un período de tiempo determinado.

Medidas

La sumad de las cantidades en ordenes de pedido, dividido para la cantidad de productos disponibles para la venta.

$$(Sum(availordered_inventsum))/sum(availphysical_inventsum)$$

Filtros

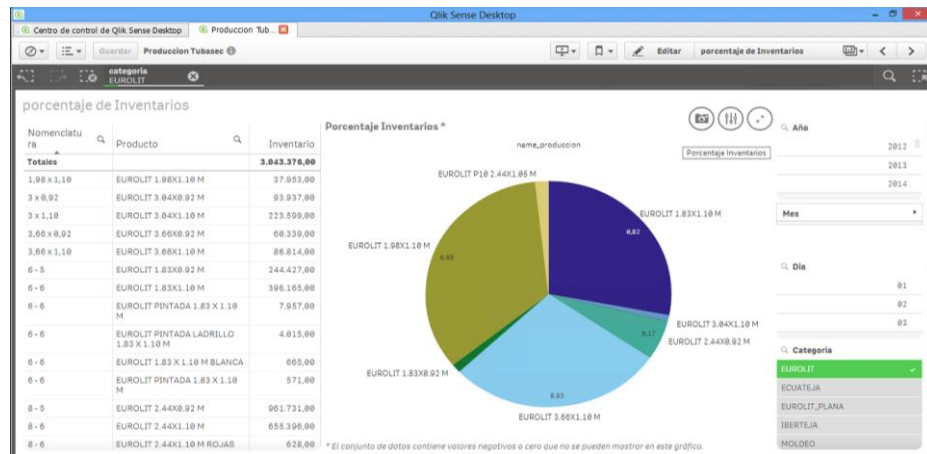
Se agregaron filtros para delimitación temporal y de productos

- Categoría de producto
- Mes
- Año
- Día

Dimensiones

La dimensión para comparación de inventarios es “name_produccion”

Figura 81. Indicador de Producción – Porcentaje de Inventario



Fuente: elaboración propia

- **Bajas**

Objetivo

Cantidad de productos que no son aptos para la venta al público, por un producto determinado, en un período de tiempo específico.

Medidas

La suma de las cantidades de bajas por el lugar de destino.

Bajas Laboratorio: sum(if(journalnameid_ijt='BAJAS-LAB',pesototal,0))

Bajas Mermas: sf sum(if(journalnameid_ijt='BAJAS MERMAS',pesototal,0))

Bajas Totales: sf sum(if(journalnameid_ijt='BAJ_TOT',pesototal,0))

Filtros

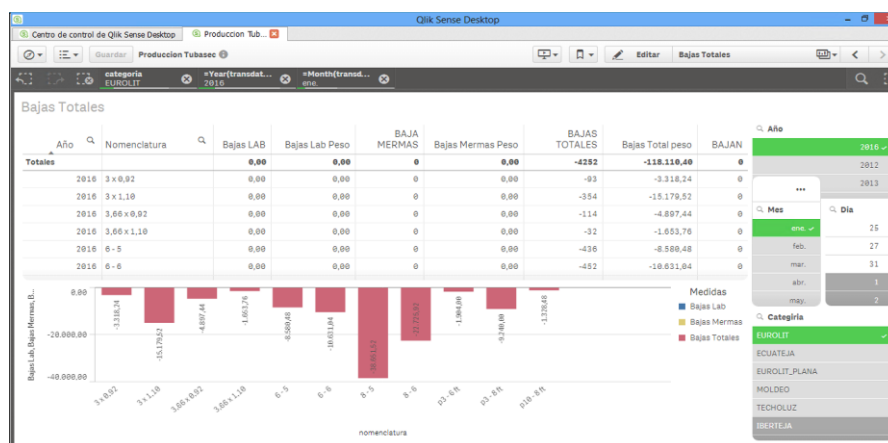
Se agregaron filtros para delimitación temporal y de productos

- Categoría de producto
- Mes
- Año
- Día

Dimensiones

La dimensión para comparación de bajas es “nomenclatura”

Figura 82. Indicador de Producción – Bajas



Fuente: elaboración propia

Talento Humano

- **Horas Extras Pagadas por proceso y tiempo.**

Objetivo

Conocer la cantidad de horas extras en un tiempo de terminado.

Medidas

La sumad de las valores agrupados por tipo de hora extra.

Horas del 100%: Suma la cantidad de horas extras del 100%

Sum(h100_asistenciaresumen)

Horas del 50%: Suma la cantidad de horas extras del 50%

Sum(h50_asistenciaresumen)

Horas del 25%: Suma la cantidad de horas extras del 25%

Sum(h25_asistenciaresumen)

Filtros

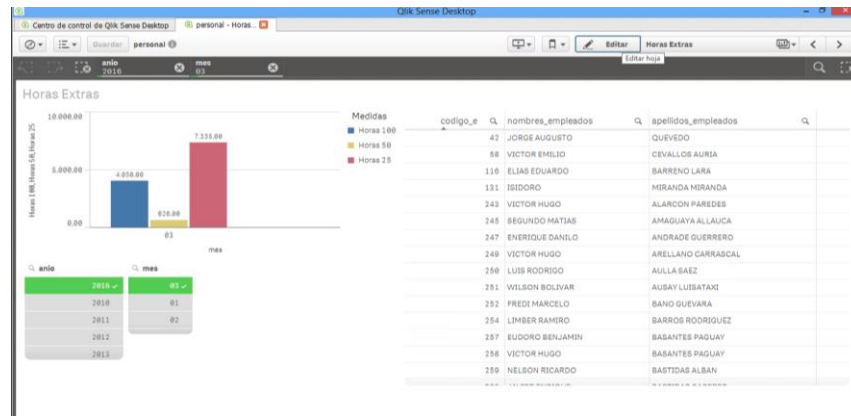
Se agregaron filtros para delimitación temporal y de productos

- Mes
- Año

Dimensiones

La dimensión para comparación de Horas extras es por el campo “mes”

Figura 83. Indicador de Personal – Horas Extras



Fuente: elaboración propia

- **Remuneración.**

Objetivo

Obtiene el valor por concepto de remuneración de uno o varios empleados en un período establecido.

Medidas

La suma de valores agrupado por el concepto.

Ingresos: suma de todos los ingresos fijos

Sum(i_bpro_rol)+Sum(i_crol_rol)+Sum(i_ddin_rol)+Sum(i_fres_rol)+Sum(i_hex1_rol)+Sum(i_hex1_rol)+Sum(i_hex2_rol)+Sum(i_hex5_rol)+Sum(i_lvac_rol)+Sum(i_movi_rol)+Sum(i_otro_rol)+Sum(i_pcom_rol)+Sum(i_pvac_rol)+Sum(i_rdec3_rol)+Sum(i_rdec4_rol)+Sum(i_sbas_rol)

Horas del 100%: Suma la cantidad de horas extras del 100%

Sum(h100_asistenciaresumen)

Horas del 50%: Suma la cantidad de horas extras del 50%

Sum(h50_asistenciaresumen)

Horas del 25%: Suma la cantidad de horas extras del 25%

Sum(h25_asistenciaresumen)

Egresos: suma de todo los egresos

Sum(e_caho_rol)+Sum(e_comi_rol)+Sum(e_dant_rol)+Sum(e_dbpr_rol)+Sum(e_dcyg_rol)+Sum(e_denf_rol)+Sum(e_dfac_rol)+Sum(e_dfal_rol)+Sum(e_dfra_rol)+Sum(e_dfru_rol)+Sum(e_fsol_rol)+Sum(e_iless_rol)+Sum(e_iren_rol)+Sum(e_otrd_rol)+Sum(e_pcel_rol)+Sum(e_pcel_rol)+Sum(e_pemp_rol)+Sum(e_phip_rol)+Sum(e_pqui_rol)

Filtros

Se agregaron filtros para delimitación temporal y de productos

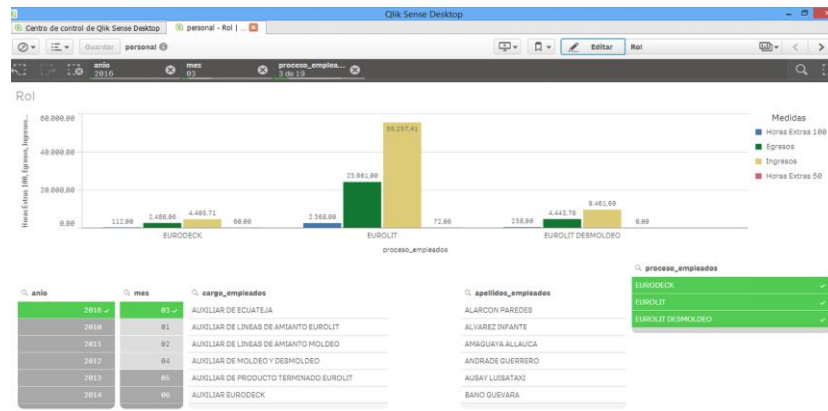
- Mes
- Año

- Cargo empleado
- Proceso empleado
- Empleado

Dimensiones

La dimensión para comparación de Remuneración de empleados es por el campo “proceso_empleados”

Figura 84. Indicador de Personal – Remuneración



Fuente: elaboración propia

Proveedores

- Cantidad de compras.

Objetivo

Obtiene la suma de compras realizadas a un proveedor específico o una agrupación de ellos.

Medidas

Cantidades: La suma de la cantidad de artículos comprados a uno o varios proveedores.

Sum(qty_detallefacturacompra)

Valor comprado: La suma del valor comprado a uno o varios proveedores.

Sum(invoiceamount_facturacompra)

Filtros

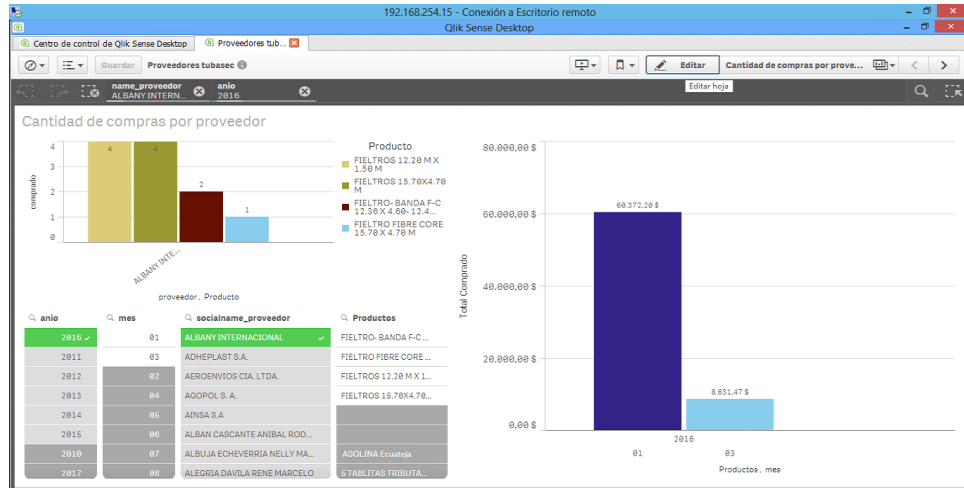
Se agregaron filtros para delimitación temporal y de productos

- Mes
- Año
- Proveedor
- Producto

Dimensiones

La dimensión para comparación de compras a proveedores es “socialname_proveedor” y “name_detallefacturacompra”.

Figura 85. Indicador de Proveedores – Cantidad de compras



Fuente: elaboración propia

- **Porcentaje de compras a proveedores.**

Objetivo

Conocer el porcentaje que representa la compra de a uno o varios proveedores, por períodos de tiempo específicos.

Medidas

Cantidades: La suma en porcentaje el valor comprado aun proveedor y comparado con el total comprado a todos los proveedores. Sum(lineamount_detallefacturacompra)

Filtros

Se agregaron filtros para delimitación temporal y de productos

- Mes
- Año
- Proveedor

Dimensiones

La dimensión para comparación de compras a proveedores es “socialname_proveedor” y “name_detallefacturacompra”.

5.2. Evaluación preliminar

Una vez que el *Data Warehouse* se implementó en Tubasec C.A., se procedió con la entrega y capacitación al personal que van a utilizar la herramienta y posterior a esto, se aplicaron entrevista al personal usuario del *Data Warehouse*.

En la entrevista planteada, se toparon varios tópicos importantes a la hora de evaluar los resultados y comparar las dos maneras de obtener reportes e indicadores, en la siguiente matriz, se recoge los resultados obtenidos.

Tabla 19. Matriz de evaluación entre Data Warehouse vs. Proceso Manual

ENFOQUE	CARACTERÍSTICA	DESCRIPCIÓN	CALIFICACIÓN	
			Data Warehouse	Proceso Manual
Funcionalidad	Idoneidad	El reporte entrega lo que se requiere	10	8
	Exactitud	Los resultados del reporte son los esperados	10	10
	Interoperabilidad	El reporte relaciona datos con otros procesos o departamentos	10	2
Usabilidad	Aprendizaje	Facilidad de aprendizaje para obtener la información	10	4
	Comprensión	El reporte es fácil de entender	10	8
Eficiencia	Comportamiento en el tiempo	Rapidez para obtener el reporte	10	3
	Comportamiento de recursos	Maneja poco recursos para obtener el reporte	10	3

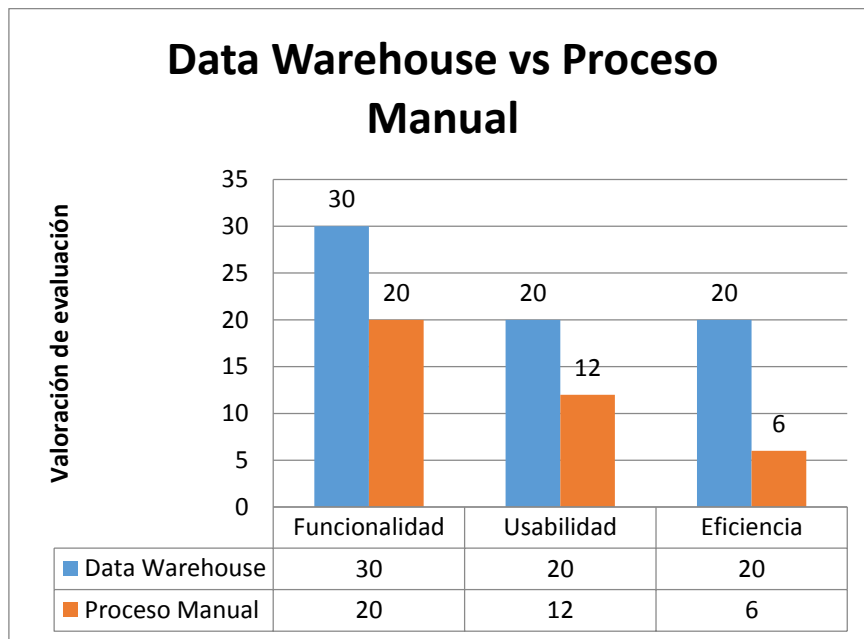
Fuente: elaboración propia

Tabla 20. Resumen evaluación entre Data Warehouse vs. Proceso Manual

Característica	Data Warehouse	Proceso Manual
Funcionalidad	30	20
Usabilidad	20	12
Eficiencia	20	6

Fuente: elaboración propia

Figura 86. Análisis Data Warehouse vs. Proceso Manual



Fuente: elaboración propia

Durante la entrevista también se pidió dar un comentario sobre la herramienta, con lo que se obtuvo los siguientes puntos relevantes:

- Alto rendimiento para obtener información (Velocidad de obtención de la información)
- Histórico de datos con alta disponibilidad, sin tener la necesidad de buscar en sus archivos la información que se generó en su tiempo específico.
- Integración entre áreas antes aisladas, pues antes de la implementación, no se podía comparar información entre procesos tal cómo ventas y producción.
- Mejores reportes y presentación de información, claramente evidenciado en los gráficos estadísticos y tablas de resumen.
- Personalización de filtros para analizar información en distintas perspectivas, con lo que se logra tener una visión panorámica del estado de la empresa sin necesidad de intervención manual con su correspondiente grado de tergiversación de la información.

Al conocer éstos resultados de la matriz presentada y con la aceptación favorable del personal usuario de la aplicación, se confirma que la implantación del *Data Warehouse* pudo resolver el problema planteado en esta investigación y por ende da solución al mismo.

Capítulo 6

Conclusiones y Recomendaciones

6.1. Conclusiones

- Al analizar las herramientas necesarias para la implementación y puesta en marcha del Data Warehouse, se determina que es importante relacionar el tipo y cantidad de información con la que cuenta la empresa, sin dejar de lado el presupuesto y la infraestructura que esta posee.
- Para implementar y desarrollar un *Data Warehouse* es imprescindible realizar un buen análisis de requerimientos y orígenes de datos, de éstos dependerá el tener la información necesaria que ayude a conseguir el enfoque gerencial que se dará a los indicadores y cuadros de mando requeridos por la alta Gerencia.
- Para desarrollar una buena estructura del *Data Warehouse*, se debe escoger una metodología y esquema apropiado, pues éste ayudara o perjudicará su desempeño al momento de mostrar resultados.
- Una vez implementado el *Data Warehouse* en Tubasec C.A., se pudo tener otra visión de la empresa, además se logró relacionar información entre procesos que antes de su implantación no fueron tomados en cuenta debido al proceso manual que se manejaba.
- Al realizar la validación de resultados, se confirmó que los tiempos de respuesta al obtener información y mostrarla en los cuadros de mando e indicadores, son considerablemente más bajos que los tiempos previos a la implementación de Data Warehouse.

6.2. Recomendaciones

- Se recomienda no escoger una herramienta del proceso del Data Warehousing únicamente por su facilidad de manejo, sino más bien por la compatibilidad con múltiples plataformas y herramientas de otros fabricantes, esto ayudará en la reingeniería que pudiese darse en un futuro.
- Al realizar la recolección inicial de información, se recomienda entrevistar previamente a los mandos medios, a fin de tener una idea del funcionamiento de la empresa y así poder definir claramente los alcances del proyecto con la alta gerencia.

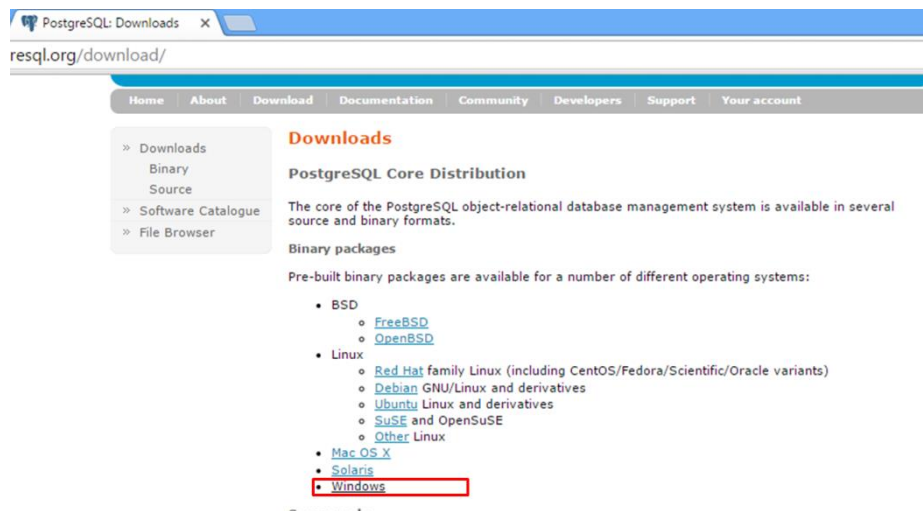
- El esquema y metodología a utilizar debería estar acorde con las características principales del proyecto, sin extenderse demasiado por proyecciones futuras, ya que esto podría conllevar en posibles falas de eficiencia del Data Warehouse.
- Es importante recomendar que la información obtenida con el nuevo enfoque que relaciona procesos antes separados, sea analizada desde distintos puntos de vista de manera que no exista ambigüedad de información.
- Una vez implementado en Data Warehouse y previa la sociabilización del proyecto, se recomienda realizar un completo control de calidad con la gente que generaba los reportes manualmente, pues ellos conocen la información histórica y tendrán un criterio más exacto de los resultados mostrados.

Apéndice A

INSTALACION POSTGRESQL

Para la instalación del gestor de bases de datos PostgreSQL es necesario poder descargar el instalador desde su página oficial la misma que es <http://www.postgresql.org/download/> donde se deberá escoger el tipo de sistema operativo sobre el cual se realizará la instalación, en este caso será Windows.

Figura 87. Descarga PostgreSQL



Fuente: elaboración propia.

La opción seleccionada anteriormente redireccionara a una página que le permita acceder a las páginas de descarga del software al dar clic en la opción de Download.

Figura 88. Seleccion opción download para redireccionamiento de página



Fuente: elaboración propia.

Una vez dentro de la página de descargas se debe seleccionar el software con la versión específica de windows que sería de 64 bits o 32 bits, en este caso aplicativo se instalaría de 64 bits, y se escogería la versión de postgres 9.5 ya que es la que soporta todas las versiones has el momento.

Figura 89. Selecciona del software de acuerdo al sistema operativo

Download PostgreSQL

Please Note: Cookies should be enabled for the download process to function correctly

Installer version **Version 9.5.1** [[Readme](#) file for customers interested in using PL/Perl, PL/Python or PL/Tcl]



Installer version **Version 9.4.6** [[Readme](#) file for customers interested in using PL/Perl, PL/Python or PL/Tcl]



Fuente: elaboración propia.

Ya una vez realizada la descarga se procede a la debida instalción donde paracera el inicio de la instalación con una bienvenida, en el cual se pondar en la opción siguiente.

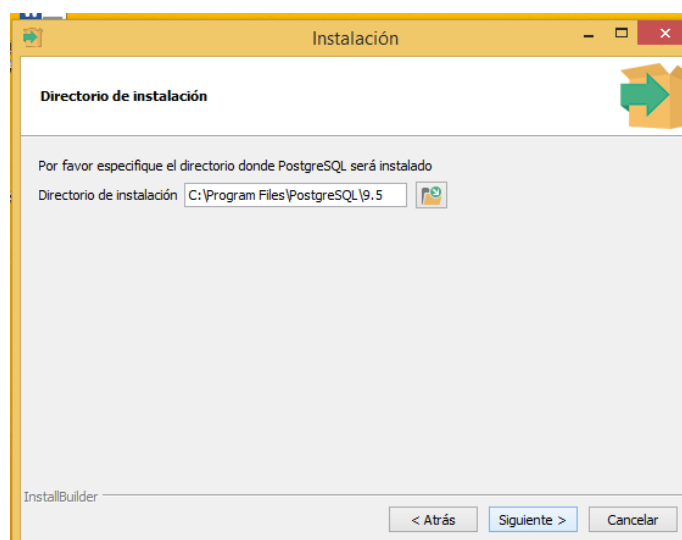
Figura 90. Bienvenida al proeso de Instalación postgresQL.



Fuente: elaboración propia.

A continuacion se mostrará la opcion para definir el lugar de instalacion, o el directorio donde se alojara postgresql. Por defecto viene C:\Program Files\PostgreSQL\9.5. La ultima carpeta en la version de postgres a instalar, luego clic en siguiente.

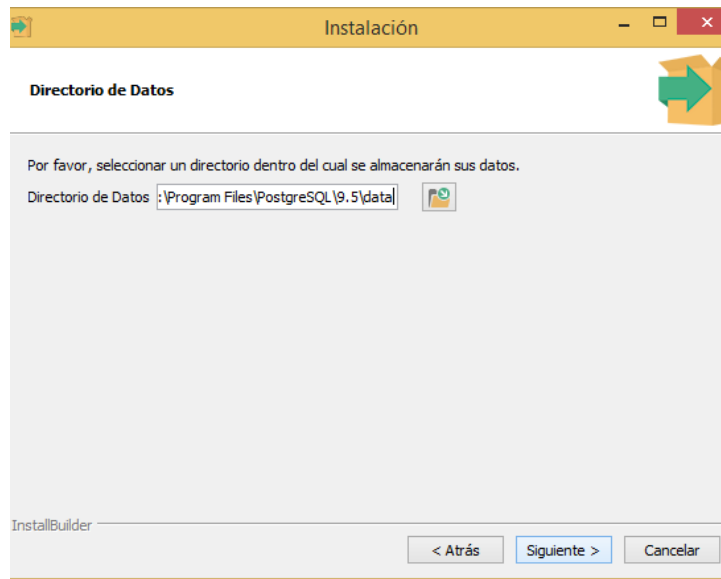
Figura 91. Direccionamiento del lugar de alojamiento de Postgresql.



Fuente: elaboración propia.

Una vez especificada la dirección de instalación se debe especificar donde se almacenarán los datos y por defecto dará la siguiente dirección C:\Program Files\PostgreSQL\9.5\data. Y clic en siguiente

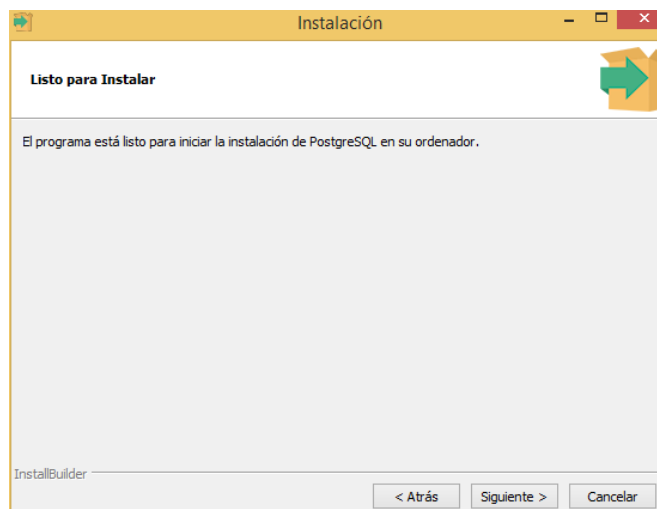
Figura 92. Directorio de localización de los datos de postgres.



Fuente: elaboración propia.

El siguiente mensaje será una de aviso de inicio de la instalación si se desea empujar se dará en siguiente caso contrario se puede cancelar en este paso al igual que en los pasos anteriores.

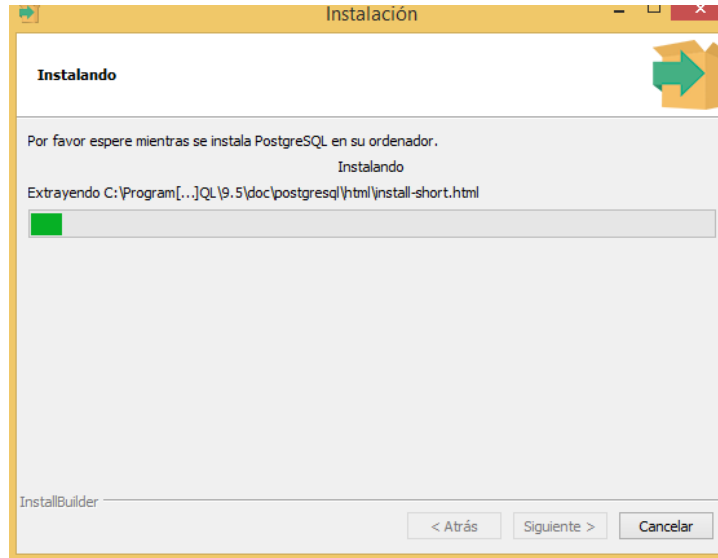
Figura 93. Inicio de la instalación.



Fuente: elaboración propia.

Se empezara a realizar la extraccion de los archivos de instalacion de Postgres que se encuentran en el instalador , el mismo que tiene una barra la que indica el avance de la instalacion.

Figura 94. Extracción de archivos de instalacion de PostgreSQL.



Fuente: elaboración propia.

Una ves finalizada la instalacion de postgres aparecera si se decea agrgar componenteas adicionales que se utilizarán para otro tipo de procesos cómo sistemas de geoposicionamiento, por tal motivo se desmarca y se selecciona terminar.

Figura 95. Extracción de archivos de instalacion de PostgreSQL.

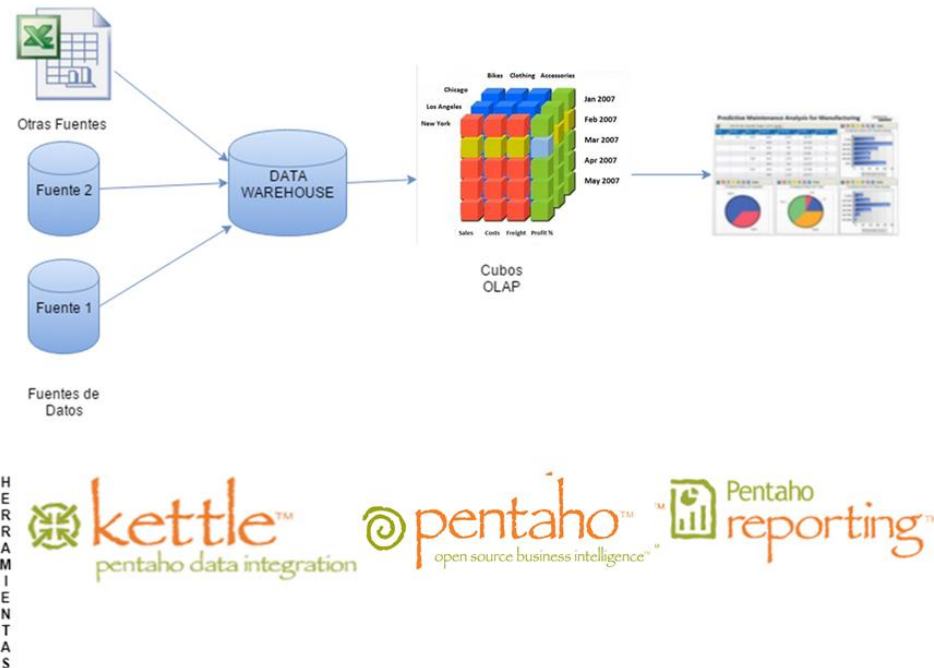


Fuente: elaboración propia.

Instalación de Pentaho Suit

Dentro de las herramientas de *Business Intelligence* se utilizará la Suit de Pentaho en la versión *Community*, ya que además Pentaho cuenta con una versión pagada, y es por ello que solo se hará uso de los módulos de interés cómo es Pentaho data Integration, para la realización de los procesos ETL, Pentaho Reporting para el diseño de los reportes el cómo se desea que éstos se los elabore para el momento de presentarlos., y por ultimo Pentaho Bi-server, este cuenta con otros módulos, los mismos que se utilizarán para la creación de los cubos y la publicación de los reportes generados mediante el cubo.

Figura 96. Diagrama de la estructura de *Business Intelligence* y sus herramientas.



Fuente: elaboración propia.

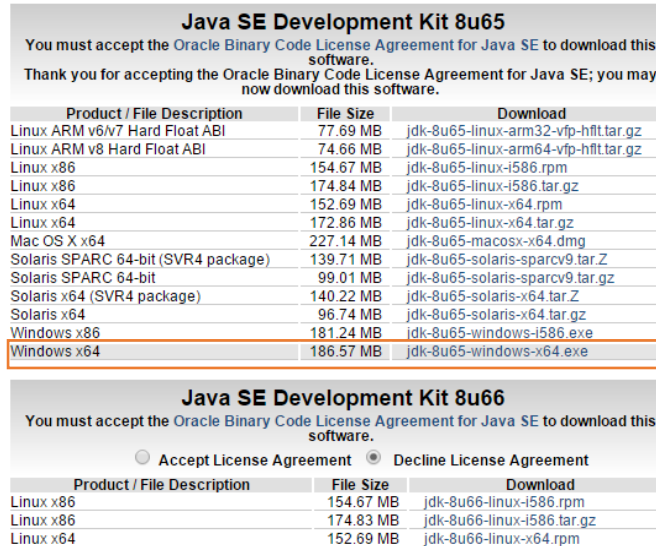
Pentaho Suit debe cumplir con una serie de requerimientos de pre-instalación cómo son Jdk de java para ello se procederá al instalación del mismo.

Instalación del Jdk de Java

En primer lugar se descarga e instala el Jdk de java. Desde la página oficial de Oracle ya que es un producto Oracle <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads->

2133151.html?ssSourceSiteId=otnes en la cual se escogerá la versión más reciente que será Jdk 8.

Figura 97. Descarga Java.



Fuente: elaboración propia.

Ya descargado se ejecuta el instalador, dicha acción mostrará la pantalla de bienvenida e inicio a la instalación del Jdk y sobre la cual se dará clic en siguiente.

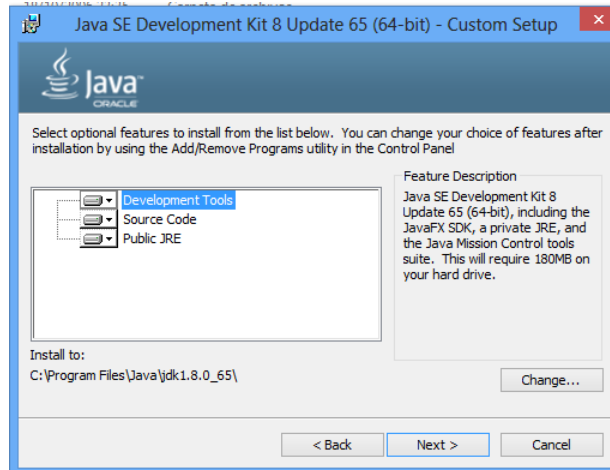
Figura 98. Pantalla de inicio de instalación del Jdk.



Fuente: elaboración Propia

La siguiente pantalla será de ubicación del directorio por defecto se instalara en los archivos de programa, si se desea se cambia caso contrario se da clic en siguiente para continuar con la instalación.

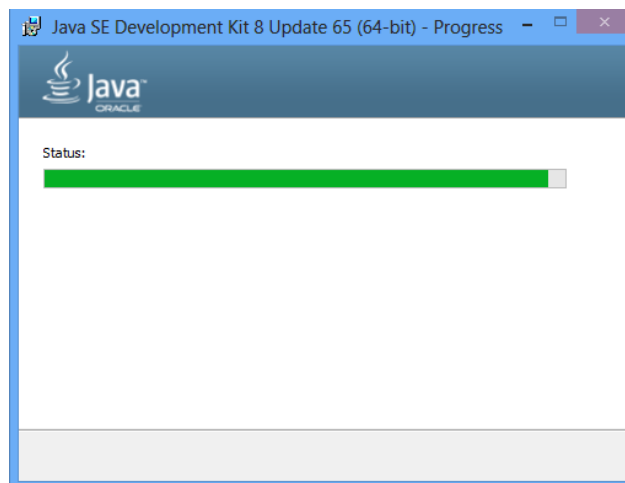
Figura 99. Ubicación del directorio de instalación



Fuente: elaboración propia.

Cómo se puede observar en la gráfica que continúa, empieza la instalación del Jdk, el mismo que contiene una barra de avance del proceso.

Figura 100. Proceso de instalación del Jdk de java



Fuente: elaboración propia.

Se espera hasta que termine la instalación y luego clic en finalizar.

Figura 101. Termino de la instalación del Jdk

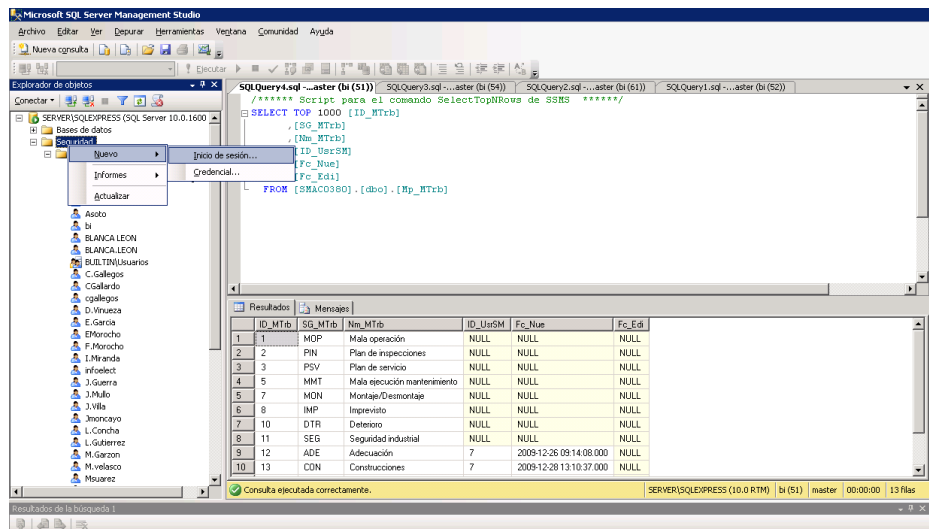


Fuente: elaboración propia.

Configuración en MS SQL Server.

Ingresar SQL Server, sobre la carpeta que dice seguridad y dar clic derecho, luego clic izquierdo en la opción Nuevo/ Inicio de sesión.

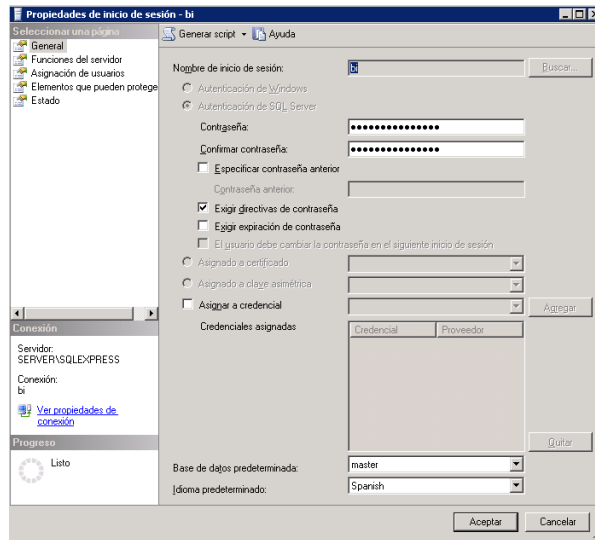
Figura 102. Creación del usuario para inicio de sesión



Fuente: elaboración propia.

En la pantalla general, llenar los campos nombre, contraseña, se escoge la primera opción que dice exigir directivas de contraseña, el nombre de la base de datos a la que se desea conectar y el idioma.

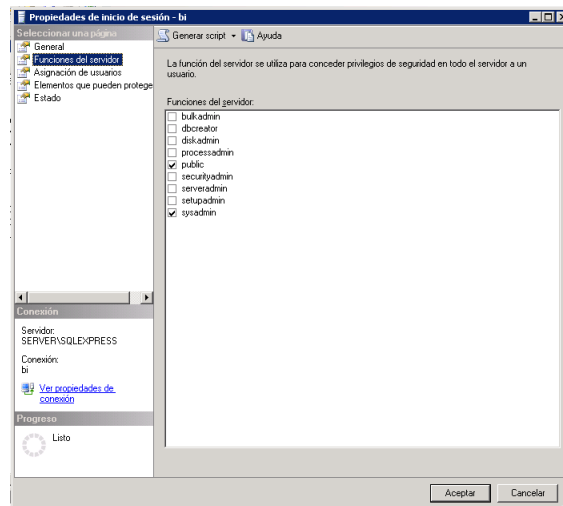
Figura 103. Pantalla genera



Fuente: elaboración propia.

En la página de funciones del servidor se observa que este seleccionado las opciones de public y sysadmin, si estas opciones estan marcadas se pasa a la siguiente.

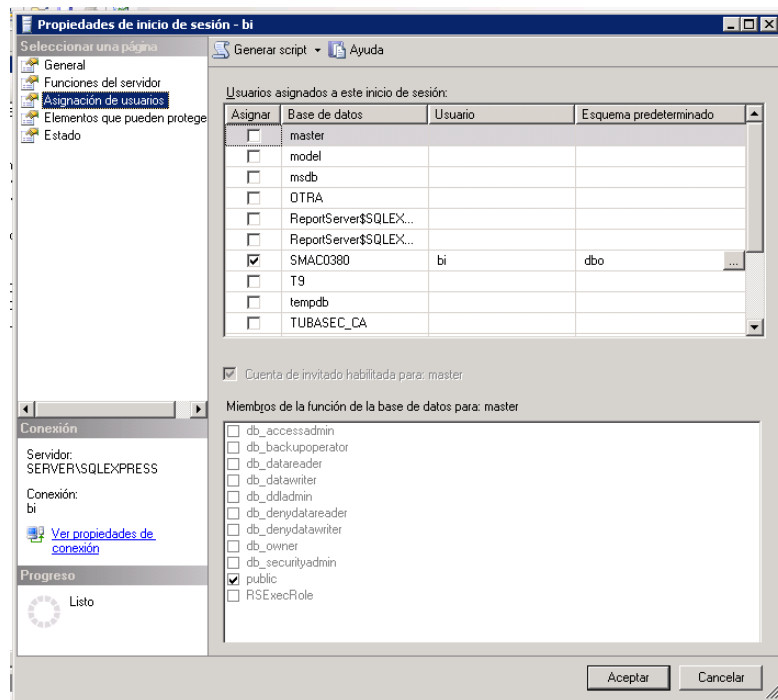
Figura 104. Pantalla de funciones del servidor



Fuente: elaboración propia.

En la página de asignación de usuario se selecciona la base de datos de la cual va a extraer los datos.

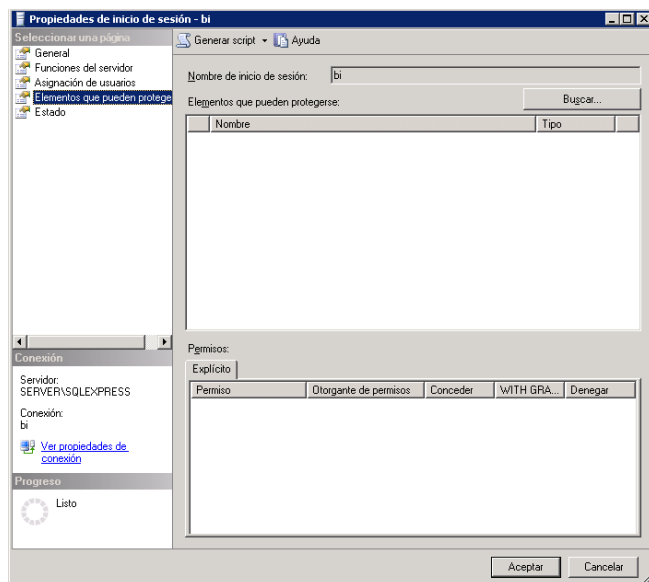
Figura 105. Pantalla de asignación de usuario



Fuente: elaboración propia.

En la página de elementos que pueden proteger se deja la configuración por defecto.

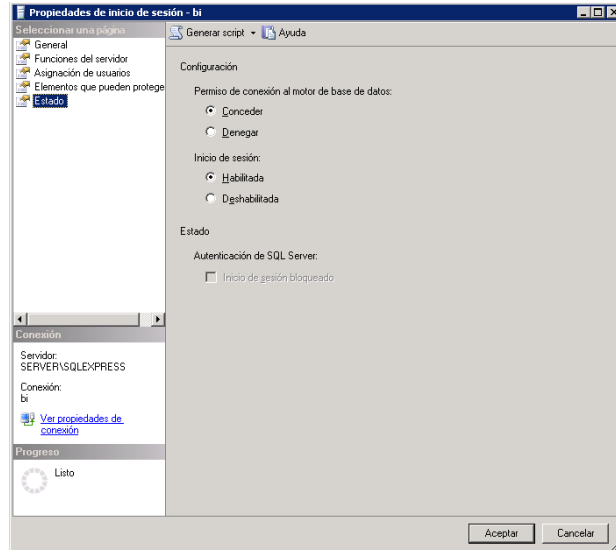
Figura 106. Pantalla de elementos que pueden proteger



Fuente: elaboración propia.

Por último en la página de estado se observa que estén habilitados los permisos de conexión a la base de datos y de inicio de sesión, clic en aceptar.

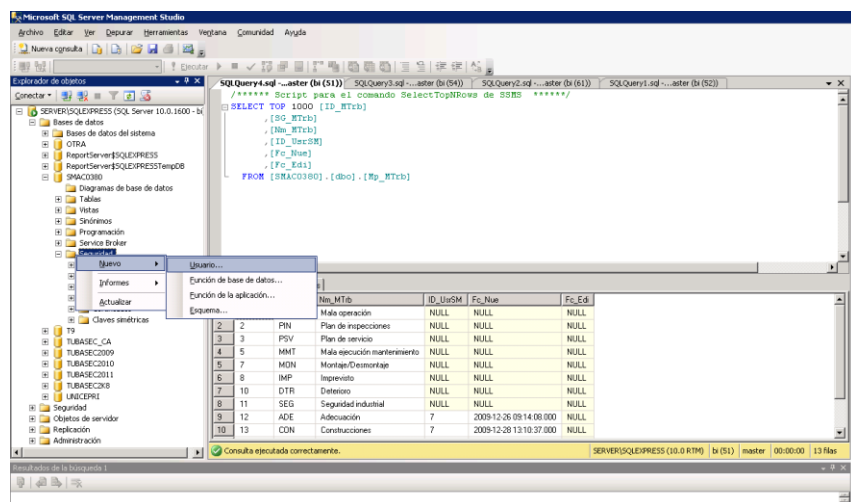
Figura 107. Pantalla de estados



Fuente: elaboración propia.

Luego de crear el usuario de inicio de sesión, se debe desplegar la base de datos a extraer y sobre la carpeta de seguridad clic derecho Nuevo/ Usuario.

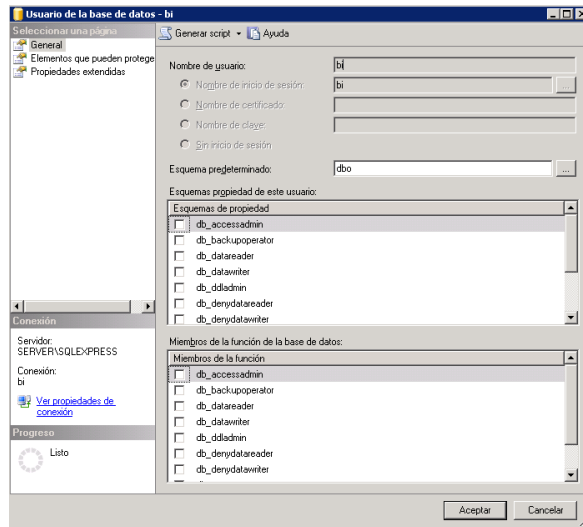
Figura 108. Creación del usuario



Fuente: elaboración propia.

Escribir el nombre de usuario antes creado y aceptar.

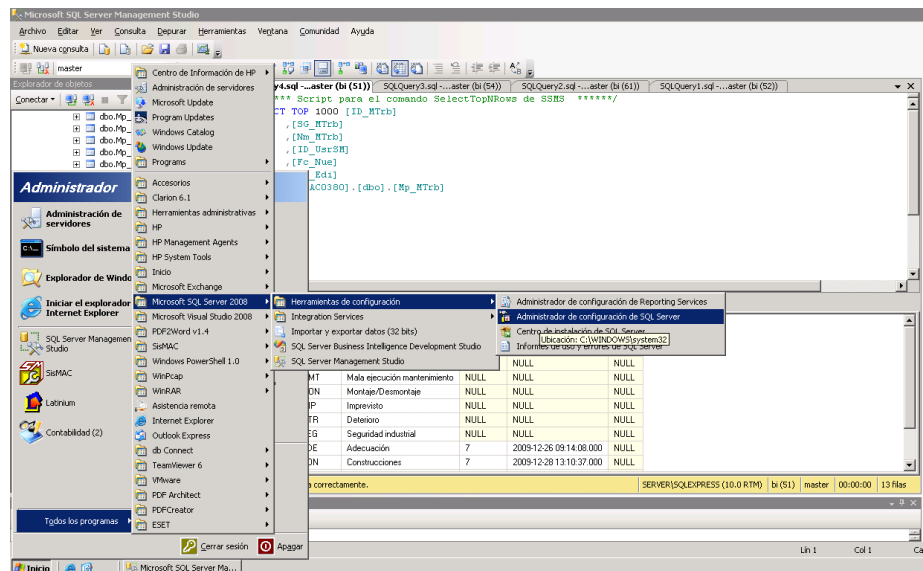
Figura 109. Agregación del usuario



Fuente: elaboración propia.

Luego se verifica si los puertos de conexión están correctos, para ello hay que dirigirse a “Inicio/ Todos los programas/ Microsoft SQL Server 2008/ Herramientas de configuración/ Administrador de configuración de SQL Server”.

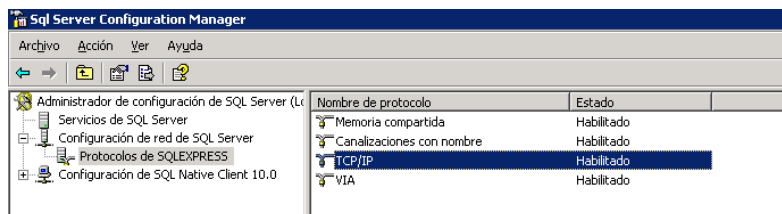
Figura 110. Ingreso a la pantalla de Configuración



Fuente: elaboración propia.

Ahora se debe configurar la red de SQL Server/ Protocolos de SQLEXPRESS, aqui se puede observar la conexión TCP/IP y esta habilitada solo falta por confirmar que el puerto se el correcto.

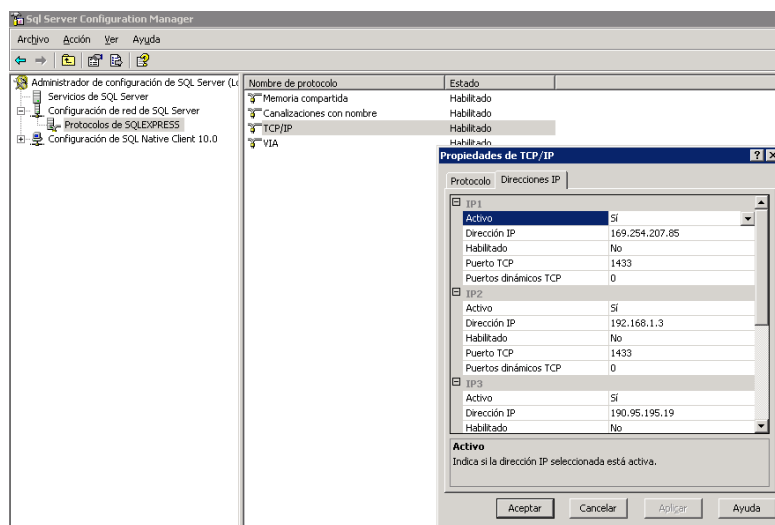
Figura 111. Verificación de la conexión



Fuente: elaboración Propia.

Sobre TCP/IP clic derecho, abrir las propiedades y se muestra todas las direcciones y puertos por los que se conecta, en éste caso el puerto TCP estaba vacío entonces se debe escribir el puerto 1433 en el casillero que dice Puerto TCP.

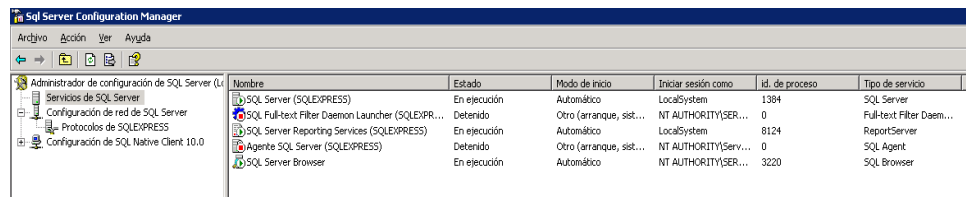
Figura 112. Verificación de puertos



Fuente: elaboración Propia

Luego de verificar los puertos se procede a reiniciar el sistema, se da clic en Servicios de SQL Server.

Figura 113. Servicios de SQL Server



Fuente: elaboración Propia

Aquí se selecciona SQL Server clic de Propiedades/ Reiniciar, luego hay que esperar a que reinicie el sistema con los cambios efectuados.

Figura 114. Reinicio del sistema



Fuente: elaboración Propia.

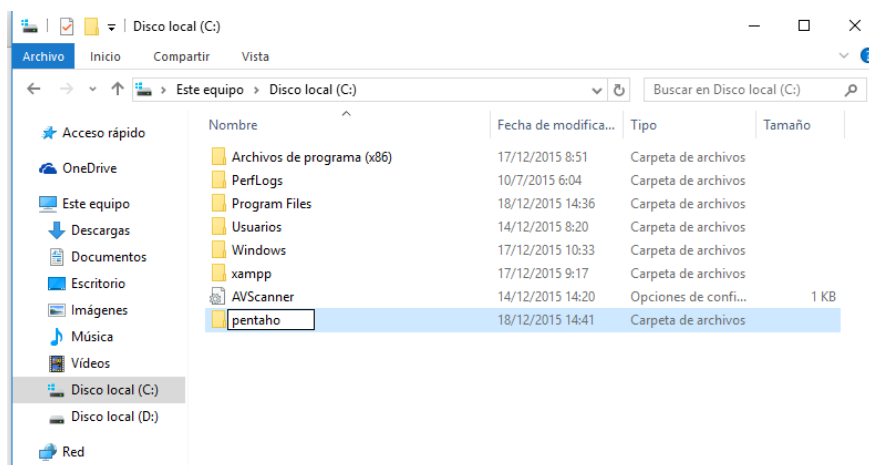
Apéndice B

INSTALACION PENTAHO SUIT

Pentaho Data Integration conocido como Kettle es una herramienta la cual permite realizar la extracción de los datos de las diferentes fuentes que tiene la empresa, es por ello que se requiere la instalación de esta herramienta ya una vez cumplido con el requerimiento de instalación cómo es Jdk 8 de java.

Luego de haber descargado Pentaho Data Integration se procede a crear en el directorio la carpeta Pentaho, en la cual se guardaran los diferentes módulos con los que se trabajara.

Figura 115. Creación del directorio Pentaho



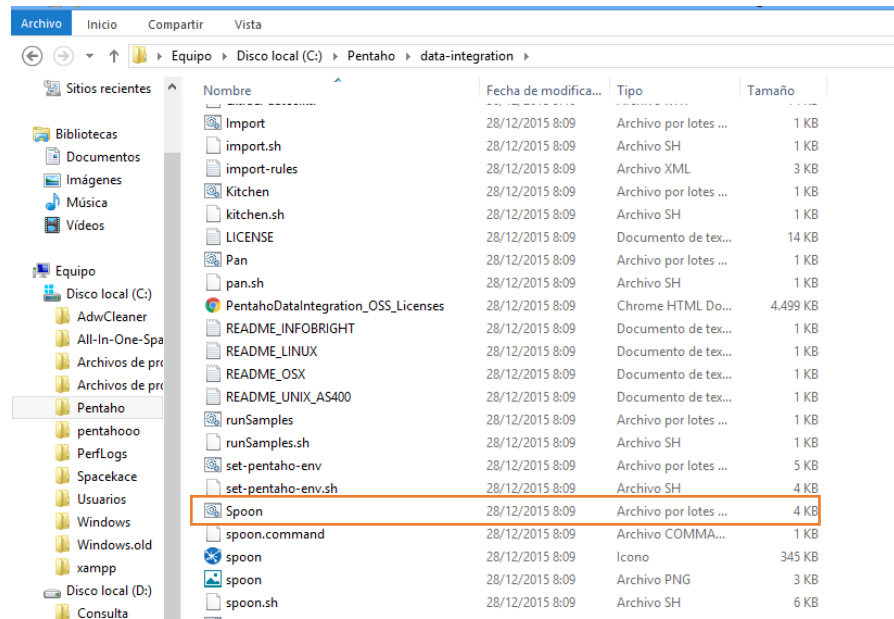
Fuente: elaboración propia

En la carpeta creada se debe descomprimir lo que se descargado cómo Data Integration, Report Designer y bi-server que se encuentran en disco D:\Instaladores

Todos los archivos se los debe descomprimir uno por uno en el directorio de Pentaho con el fin para en lo posterior utilizarlos ya en el desarrollo del *Data Warehouse*.

En la carpeta C:\Pentaho\data-integration aparece archivo Spoon.bat el mismo que se debe ejecutar para poder realizar los procesos ETL, de preferencia de lo debe ejecutar cómo administrador.

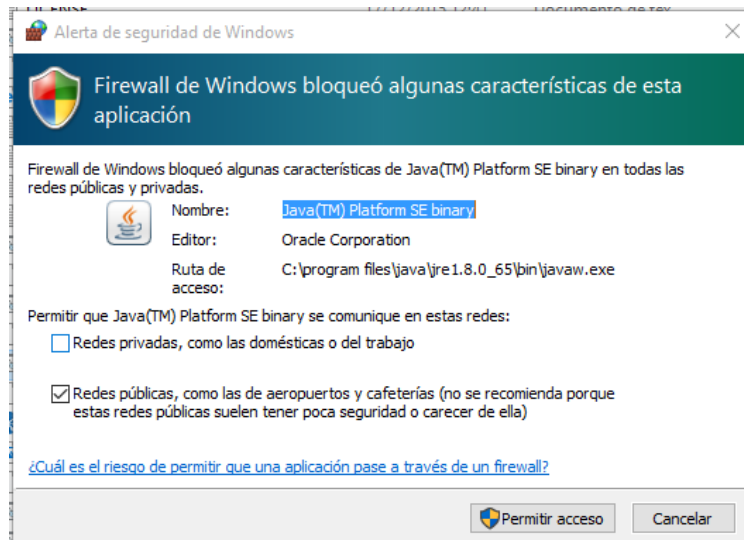
Figura 116. Ubicación Spoon



Fuente: elaboración propia

Aparecerá una pantalla de autorización del firewall, es necesario permitir el acceso.

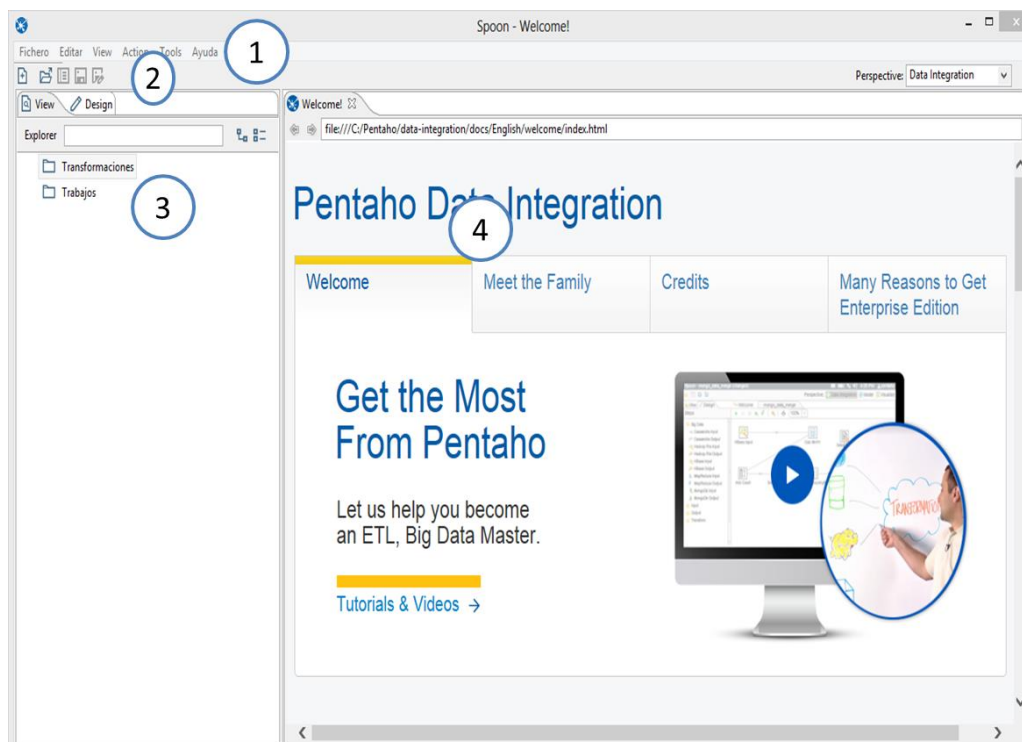
Figura 117. Autorización Firewall



Fuente: elaboración Propia

Una vez que se haya ejecutado nuestra aplicación parecerá la pantalla de inicio de la aplicación, sobre la cual se encuentra, 1. La barra de menú, 2 la barra de herramientas, 3. El Panel de trabajo ya sea vista o diseño y 4. La página de bienvenida.

Figura 118. Ubicación Spoon



Fuente: elaboración propia

Pentaho Data Integration permite crear dos tipos de archivos ya sean un trabajo o una transformación.

Transformación: la creación de una transformación se la utiliza con el fin de poder realizar las extracciones mediante una conexión de origen hacia una de destino, y en la cual también contienen las diferentes cambios que se le realizará a los datos que se extraerán cómo son, cambio de nombres, mapeo de datos, cambio de tipo de datos, etc.

Trabajo: un trabajo permite que todas las transformaciones realizadas se la ejecute sin necesidad de que una persona lo haga haya que este archivo permite realizar una programación sobre la carga de los datos.

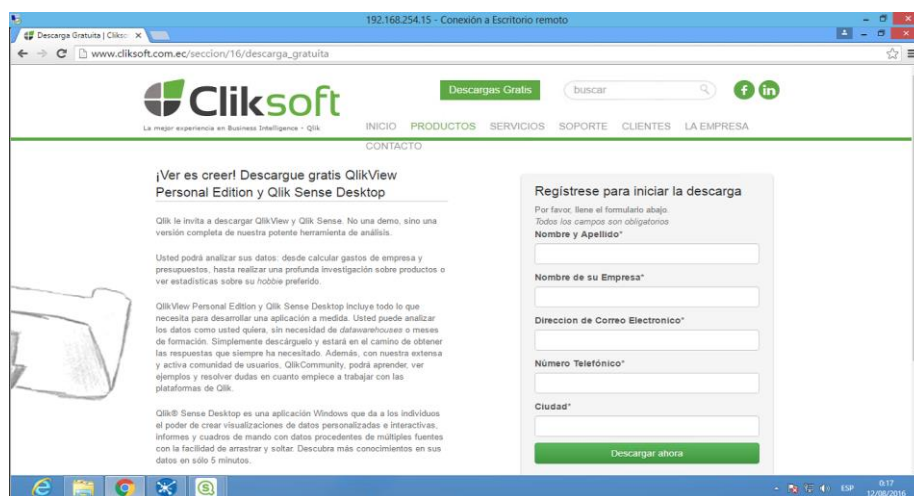
Sin embargo es necesario realizar un breve ejemplo el mismo que contendrá el proceso de realización de cada uno de ellos.

Apéndice C

INSTALACIÓN DE QLIK SENSE

Para la instalación de Qlik Sence, primero es necesario descargar el programa desde su página oficial, para ellos se debe llenar el formulario que solicita información y luego el asistente de descarga inicia el proceso de *download*.

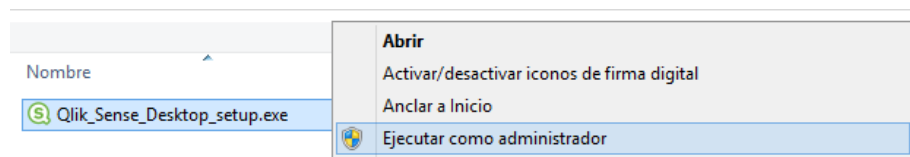
Figura 119. Descarga de Qlik Sense.



Fuente: elaboración propia

Una vez descargado se procede con la instalación, al archivo descargado se deberá iniciar como administrador

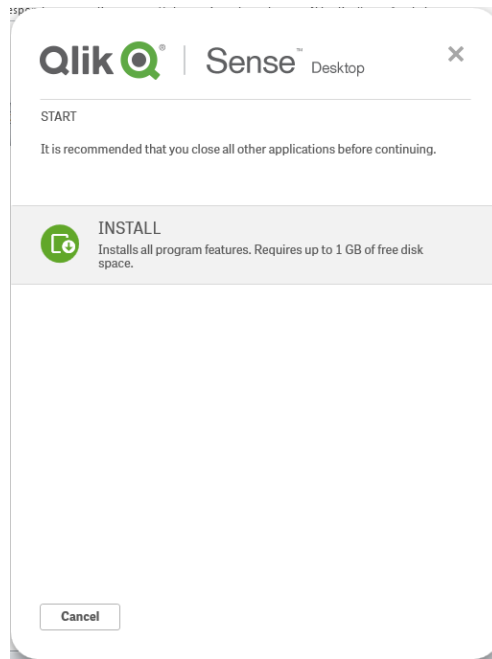
Figura 120. Inicio de instalación de Qlik Sense.



Fuente: elaboración propia

Iniciada la instalación, aparece el asistente de instalación quien será la guía durante todo el proceso.

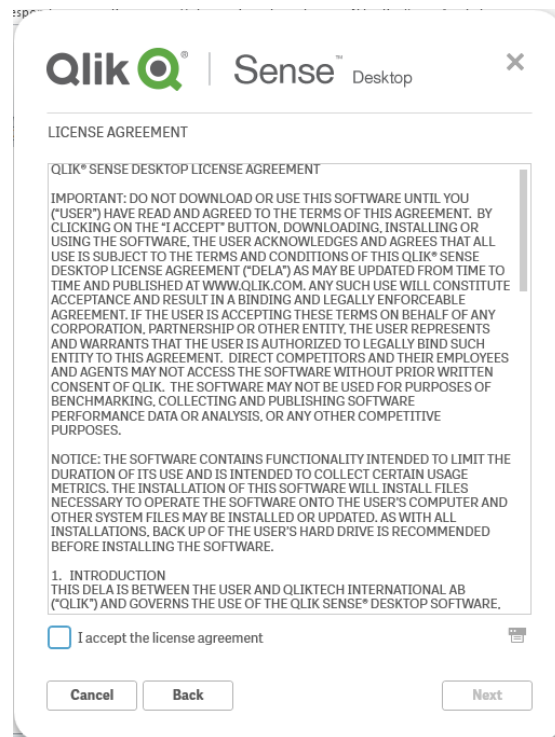
Figura 121. Inicio de asistente de instalación de Qlik Sense.



Fuente: elaboración propia

Es necesario aceptar las condiciones de uso para poder continuar con la instalación de la herramienta.

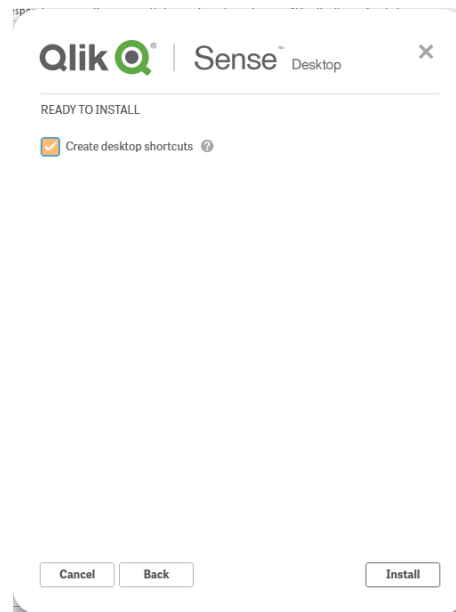
Figura 122. Condiciones de uso para instalar Qlik Sense



Fuente: elaboración propia

Una vez terminada la parte inicial, se permite instalar, para ellos se debe clicar en el botón “Install”.

Figura 123. Inicio de instalación Qlik Sense



Fuente: elaboración propia

Una vez terminado el proceso de instalación, aparece un cuadro cómo el que se muestra a continuación, donde informa que el proceso se realizó con éxito y está listo para usarse.

Figura 124. Final de la instalación de Qlik Sense



Fuente: elaboración propia

Apéndice D

SCRIPTS PARA LA EXTRACCION DE INFORMACION DE LAS BASES DE DATOS

Script para la creación de la tabla dim_tiempo, en postgresQL.

```
CREATE TABLE public.dim_tiempo
(
  fechask character varying(8) NOT NULL,
  fecha date NOT NULL,
  anio character varying(8) NOT NULL,
  trimestre character varying(8) NOT NULL,
  mes character varying(8) NOT NULL,
  semana character varying(8) NOT NULL,
  dia character varying(8) NOT NULL,
  diasemana character varying(8) NOT NULL,
  ntrimestre character varying(7) NOT NULL,
  nmes character varying(15) NOT NULL,
  nmes31 character varying(3) NOT NULL,
  nsemana character varying(10) NOT NULL,
  ndia character varying(6) NOT NULL,
  ndiasemana character varying(10) NOT NULL,
  CONSTRAINT dim_tiempo_pkey PRIMARY KEY (fecha)
)
WITH (
  OIDS=FALSE
);
```

Script para la carga de la tabla dim_tiempo en postgresQL

```
CREATE OR REPLACE FUNCTION public.fn_generar_tiempo()
  RETURNS boolean AS
$BODY$
DECLARE
FechaDesde date;
FechaHasta date;
BEGIN
--Rango de fechas a generar: del 01/01/2006 al 31/12/2030
FechaDesde := TO_DATE('20100101','YYYYMMDD');
FechaHasta := TO_DATE('20111231','YYYYMMDD');
WHILE FechaDesde <= FechaHasta LOOP
INSERT INTO DIM_TIEMPO
(
FechaSK,
Fecha,
Anio,
Trimestre,
Mes,
Semana,
Dia,
```

```

DiaSemana,
NTrimestre,
NMes,
NMes3L,
NSemana,
NDia,
NDiaSemana
)
VALUES
(
to_char(FechaDesde,'YYYYMMDD'),
FechaDesde,
to_char(FechaDesde,'YYYY'),
to_char(FechaDesde,'Q'),
to_char(FechaDesde,'MM'),
to_char(FechaDesde,'WW'),
to_char(FechaDesde,'DD'),
to_char(FechaDesde,'D'),
'T'||to_char(FechaDesde,'Q')||'/'||to_char(FechaDesde,'YY'),
to_char(FechaDesde,'MONTH'),
to_char(FechaDesde,'MON'),
'Sem '||to_char(FechaDesde,'WW')||'/'||to_char(FechaDesde,'YY'),
to_char(FechaDesde,'DD MON'),
to_char(FechaDesde,'DAY')
);
--Incremento del bucle
FechaDesde := FechaDesde + 1;
END LOOP;
return true;
END;
$BODY$
LANGUAGE plpgsql VOLATILE
COST 100;

```

Script para la carga de la tabla SALESLINE (d_v_detalleventas)

```

SELECT
SALESID
, ITEMID
, SALESSTATUS
, NAME as nombreproducto
, EXTERNALITEMID
, COSTPRICE
, SALESPRICE
, LINEPERCENT
, LINEDISC
, LINEAMOUNT
, DIMENSION
, DIMENSION3_
, INVENTTRANSID
, CUSTGROUP
, CUSTACCOUNT
, SALESQTY
, SALESTYPE
, TAXITEMGROUP
, INVENTDIMID
, RECEIPTDATEREQUESTED

```

```

, DELIVERYNAME
, DELIVERYSTREET
, DELIVERYCITY
, DELIVERYCOUNTY
, DELIVERYSTATE
, DATAAREAID
, RECID
FROM SALESLINE where DATAAREAID='tub'

```

Script para la carga de la tabla CUSTTABLE (d_v_cliente)

```

SELECT                                                                                               DISTINCT
cl.ACCOUNTNUM,cl.ADDRESS,cl.CITY,cl.COUNTY,cl.CUSTGROUP,dc.IDENTIFICATIONNUM
BER,dc.SOCIALNAME,cl.PHONE,cl.SMS,dc.SRIIDTYPE,cl.STATE,cl.TELEFAX
from CUSTTABLE as cl inner join CUSTDATATABLE as dc on
cl.ACCOUNTNUM=dc.ACCOUNTNUM

```

Script para la carga de la tabla LECINVLINETABLE (d_v_detallefacturaventa)

```

SELECT
dv.INVOICEDATE,dv.LINENUM,dv.INVENTTRANSID,dv.ITEMID,dv.NAME,dv.TAXGROUP,d
v.QTY,dv.SALESPRICE,dv.DISCPERCENT,dv.DISCAMOUNT
,dv.LINEAMOUNT,dv.SALESGROUP, dv.LEDGERACCOUNT ,dv.QTYPHYSICAL
,dv.PARTDELIVERY, dv.SALESID, dv.SALESUNIT ,dv.TAXITEMGROUP
,dv.LINEDISC,dv.LINEPERCENT, dv.ORIGSALESID ,dv.LINEHEADER ,dv.INVENTDIMID
,dv.LINEAMOUNTMST,dv.SUMLINEDISC ,dv.SUMLINEDISC MST,dv.INVNUM,
dv.LEDGERVOUCHER,(dv.LINEAMOUNT+dv.SUMLINEDISC) as sbt_sindscto
FROM LECINVLINETABLE AS DV where dv.DATAAREAID='tub'

```

Script para la carga de la tabla LECCREDDOCLINETABLE (d_v_detallenotascredito)

```

SELECT
DNC.ITEMID
, DNC.SALESID
, DNC.LEDGERVOUCHER
, DNC.NUM
, DNC.INVOICEDATE
, DNC.LINENUM
, DNC.INVENTTRANSID
, DNC.NAME
, DNC.QTY
, DNC.SALESPRICE
, DNC.DISCPERCENT
, DNC.DISCAMOUNT
, DNC.LINEAMOUNT
, DNC.QTYPHYSICAL
, DNC.SALESUNIT
, DNC.TAXITEMGROUP
, DNC.LINEDISC
, DNC.LINEPERCENT
, DNC.ORIGSALESID
, DNC.LINEHEADER
, DNC.LINEAMOUNTTAX
, DNC.LINEAMOUNTMST
, DNC.LINEAMOUNTTAXMST

```

```
, DNC.SUMLINEDISC
, DNC.SUMLINEDISC MST
FROM LECCREDDOCLINETABLE AS DNC where dataareaid='tub'
```

Script para la carga de la tabla LECDEBDOCLINETABLE (d_v_detallenotasdebito)

```
SELECT
DNd.ITEMID
, DNd.SALESID
, DNd.LEDGERVOUCHER
, DNd.NUM
, DNd.INVOICEDATE
, DNd.LINENUM
, DNd.INVENTTRANSID
, DNd.NAME
, DNd.QTY
, DNd.SALESPRICE
, DNd.DISCPERCENT
, DNd.DISCAMOUNT
, DNd.LINEAMOUNT
, DNd.QTYPHYSICAL
, DNd.SALESUNIT
, DNd.TAXITEMGROUP
, DNd.LINEDISC
, DNd.LINEPERCENT
, DNd.ORIGSALESID
, DNd.LINEHEADER
, DNd.LINEAMOUNTTAX
, DNd.LINEAMOUNTMST
, DNd.LINEAMOUNTTAXMST
, DNd.SUMLINEDISC
, DNd.SUMLINEDISC MST
FROM LECDEBDOCLINETABLE as dnd where dataareaid='tub'
```

Script para la carga de la tabla CUSTPACKINGSLIPTRANS (d_v_detalleguiasremision)

```
SELECT
PACKINGSLIPID
, DELIVERYDATE
, INVENTTRANSID
, ITEMID
, NAME as productname
, ORDERED
, QTY
, REMAIN
, DLVSTATE
, SALESID
, INVENTDIMID
, CREATEDDATETIME
, RECID FROM CUSTPACKINGSLIPTRANS
```

Script para la carga de la tabla VENDTABLE (d_p_PROVEEDOR)

```
SELECT distinct VT.ACCOUNTNUM
,VT.NAME
,VT.ADDRESS
,VT.PHONE
```

```

,VT.TELEFAX
,VT.VENDGROUP
,VT.VATNUM
,VT.COUNTRYREGIONID
,VT.TAXGROUP
,VT.CITY
,VT.STREET
,VT.PARTYID
,VDT.INVOICEEXPIRATIONDATE
,VDT.IDENTIFICATIONNUMBER
,VDT.VENDSERIALNUMBER
,VDT.VENDAATHNUMBER
,VDT.SRIIDTYPE
,VDT.SOCIALNAME
,VDT.VEHICLEPLATE
FROM VENDTABLE AS VT left join VENDDATATABLE AS VDT ON
VT.ACCOUNTNUM=VDT.ACCOUNTNUM
where vt.dataareaid='tub'

```

Script para la carga de la tabla VENINVOICETRANS (d_p_detallefacturacompra)

```

SELECT
dfc.INVOICEID as INVOICEID_detallefacturacompra,
dfc.INVOICEDATE as INVOICEDATE_detallefacturacompra,
dfc.INVENTTRANSID as INVENTTRANSID_detallefacturacompra,
dfc.ITEMID as ITEMID_detallefacturacompra,
dfc.NAME as NAME_detallefacturacompra,
dfc.QTY as QTY_detallefacturacompra,
dfc.LINEAMOUNT as LINEAMOUNT_detallefacturacompra,
dfc.PURCHID as PURCHID_detallefacturacompra,
dfc.ORIGPURCHID as ORIGPURCHID_detallefacturacompra,
dfc.INTERNALINVOICEID as id_facturas,
dfc.inventdimid as inventdimid_detallefacturacompra,
dfc.LINEAMOUNTTAX as LINEAMOUNTTAX_detallefacturacompra,
dfc.LINEAMOUNTMST as LINEAMOUNTMST_detallefacturacompra,
dfc.RETFTECONCEPT as RETFTECONCEPT_detallefacturacompra,
dfc.RETIVACONCEPT as RETIVACONCEPT_detallefacturacompra,
dfc.RETIVATAXCODE as RETIVATAXCODE_detallefacturacompra,
dfc.REFTTETAXCODE as RETFTTETAXCODE_detallefacturacompra,
(select (case when dfc.inventdimid ='AllBlank' then 'Sin Bodega' else
id.inventlocationid end ) as bod from inventdim as id where
dfc.inventdimid=id.inventdimid and id.DATAAREAID='TUB') as bodega
FROM VENDINVOICETRANS as dfc
where dfc.DATAAREAID='TUB'

```

Script para la carga de la tabla VENINVOICEJOUR(d_p_facturacompra)

```

SELECT
PURCHID,
ORDERACCOUNT,
INVOICEACCOUNT,
INVOICEID,
INVOICEDATE,
QTY,
SALESBALANCE,
INVOICEAMOUNT,
CURRENCYCODE,

```

```

LEDGERVOUCHER,
DOCUMENTDATE,
TAXGROUP,
POSTINGPROFILE,
PURCHASETYPE,
SUMTAX,
PARMID,
VATNUM,
INTERNALINVOICEID,
INVOICEAMOUNTMST
FROM VENDINVOICEJOUR where dataareaid='tub'

```

Script para la carga de la tabla VENDTRANSTABLE(d_p_tranbsacciones)

```

SELECT
ACCOUNTNUM
,TRANSDATE
,VOUCHER
,(CASE WHEN VOUCHER like 'PFAC%' THEN 'FACTURA PROVEEDOR'
WHEN VOUCHER like 'E%' THEN 'PAGADO'
WHEN VOUCHER like 'SI%' THEN 'SALDO INICIAL'
WHEN VOUCHER like 'CRUP%' THEN 'CRUCE PROVEEDOR'
WHEN VOUCHER like 'PS%' THEN 'FACTURA SERVICIO' ELSE 'NO DEFINIDO' END
)AS TipoTransaccion
,INVOICE ,TXT
,AMOUNTCUR ,SETTLEAMOUNTCUR
,AMOUNTMST ,SETTLEAMOUNTMST
,DUEDATE ,LASTSETTLEVOUCHER
,LASTSETTLEDATE ,CLOSED
,TRANSTYPE ,APPROVEDBY
,APPROVED ,DIMENSION
,DIMENSION2_ , DIMENSION3_ ,DOCUMENTDATE
,POSTINGPROFILE ,LASTEXCHADJ, COMPANYBANKACCOUNTID
,PAYMREFERENCE, PAYMMODE ,OFFSETRECID ,PROMISSORYNOTESTATUS
,EXCHRATE, APPROVEDDATE ,LASTSETTLEACCOUNTNUM
,REASONREFRECID ,MODIFIEDBY,
CREATEDBY, CREATEDTRANSACTIONID
FROM VENDTRANS

```

Script para la carga de la tabla de hechos del Data Warehouse (fact_ventas)

```

select total.*, nd.salesid as ov_notadebito,nd.num as id_notadebito,nd.total as
totalnotadebito,
(SELECT SUM (trans.AMOUNTCUR) from CUSTTRANS as trans where
trans.accountnum=total.custaccount and trans.dataareaid='tub') as saldocliente
from (select fin.* ,nc.salesid as ov_notacredito,nc.num as id_notacredito,nc.total as
totalnotacredito
from ( select v.custaccount,v.salesid as ov_factura,
(select x.invnum from LECINVHEADERTABLE as x where x.salesid=v.salesid) as
id_factura,
(select x.total from LECINVHEADERTABLE as x where x.salesid=v.salesid) as
totalfactura,
(select max( x.releasedate) as id_fecha from LECINVHEADERTABLE as x where
x.salesid=v.salesid ) as id_fecha,
(select x.packingslipid from CUSTPACKINGSLIPJOUR as x where x.salesid=v.salesid) as
id_guiaremission from SALESTABLE as v ) as fin

```

```

left join LECCREDDOCHEADERTABLE as nc on fin.id_factura=nc.modinv ) as total
left join LECDEBDOCHEADERTABLE as nd on total.id_notacredito=nd.modinv where
total.id_fecha>'20121231 00:00:01'

```

Script para la carga de la tabla de hechos del Data Warehouse (fact_ventas)

```

select p.id_proveedor as proveedor, fc.id_facturacompra as fct_or, t.id_transaccion as
pago,ti.fecha
from d_p_proveedor as p
left join d_p_facturacompra as fc on
p.accountnum_proveedor=fc.orderaccount_facturacompra
left join d_p_transacciones as t on
fc.ledgervoucher_facturacompra=t.voucher_transacciones
left join dim_tiempo as ti on (to_char(fc.invoicedate_facturacompra,
'YYYY/MM/DD'))=(to_char(ti.fecha, 'YYYY/MM/DD'))
where fc.invoiceaccount_facturacompra=t.accountnum_transacciones
and fc.invoiceid_facturacompra=t.invoice_transacciones

```

Referencias

- Bernabeu, R. D. (2010). *DATA WAREHOUSING*. Cordoba.
- BI-Spain.com. (2015). *Cuadrante Mágico para herramientas de Data Integration*. Obtenido de <http://www.bi-spain.com/articulo/74283/business-intelligence/todos/cuadrante-magico-para-herramientas-de-data-integration-julio-2015>
- Bustillos, J. (2014). *Comparativa Herramientas ETL*. Recuperado el 04 de 2016, de <http://es.slideshare.net/JorgeCarlos3/comparativa-herramientas-etl>
- Castelan Gracia, L. (s.f.). *Diseño de un almacen de datos basado en Data Warehouse* . Obtenido de <http://www.uv.mx/mis/files/2012/11/disenio-de-un-almacen-de-datos.pdf>
- Conesa, J. C. (2010). *Introducciona Businnes Intelligence*.
- Corr, L. (2011). *Agile Data Warehouse Desing*. DecisionOne Press.
- Elzbieta Malinowski, E. (2009). *Advanced Data Warehouse Desing*. San José, Costa Rica: Springer.
- Espinoza, R. (2010). *El Rincon de BI*. Obtenido de <https://churriwifi.wordpress.com/2010/06/01/comparativa-talend-vs-kettle-pdi/>
- Fasel, D. (2014). *Fuzzy Data Warehousing for Performance Measurement*. New York: Springer .
- Gartner. (Febrero de 2016). <https://www.gartner.com/doc/reprints?id=1-2XXET8P&ct=160204&st=sb>.
- Hefesto. (2010). *Metodología para la construcción de un Data Warehouse*. Córdoba: Bernabeu.
- Inmon, W. H. (2002). *Building the Data Warehouse*. New York: John Wiley & Sons, Inc.
- Kimball, R. (2002). *The Data Warehouse Toolkit*. New York: Robert Ipsen.
- Kimball, R., & Ross, M. (2013). *The Definitive Guide to Dimensional Modeling*. Indianapolis: John Wiley & Sons, Inc.
- Largo García, C. A. (2005). *Guia Técnica para la evaluacion de software*. Bogota.
- Martinez, D. (2014). *PostgreSQL vs. MySQL*. Obtenido de <https://danielpecos.com/documents/postgresql-vs-mysql/>
- Martinez, R. (2010). *PostgreSQL-es*. Obtenido de http://www.postgresql.org.es/sobre_postgresql#caracteristicas
- Moura Pires, J. (2015). *Data Warehouse - Basic Concepts*. Obtenido de <http://ssdi.di.fct.unl.pt/dw/ressources/lectures/files/DW-02-Basics.pdf>

- Oracle. (2016). *MySQL Technical Specifications*. Obtenido de <http://www.mysql.com/products/enterprise/techspec.html>
- PCW. (2008). Boletín de Asesoría Gerencial. *pcw*, 4.
- Ponniah, P. (2010). *Data Warehousing Fundamentals for IT Professionals*. New Jersey: John Wiley & Sons.
- Rivadera, G. (5 de 2010). *La metodología de Kimball para el diseño de almacenes de datos*. Obtenido de <http://www.ucasal.edu.ar/htm/ingenieria/cuadernos/archivos/5-p56-rivadera-formateado.pdf>
- Root, R., & Mason, C. (2012). *SQL Server 2012 BI Solutions*. apress.
- Sanz, C. (2013). *QLIKVIEW VS TABLEAU: COMPARATIVA DE HERRAMIENTAS DE BUSINESS INTELLIGENCE*. Obtenido de <http://porlaempresa.com/quickview-vs-tableau-comparativa-de-herramientas-de-business-intelligence/>
- Vaisman , A., & Zimanyi, E. (2014). *Data Warehouse Systems*. Buenos Aires: 3.
- Williams, S. W. (2007). *The profit impact of Business Intelligence*. Boston.

Resumen Final

Implementación de un *Data Warehouse* corporativo para la integración de datos de las áreas de la organización para la toma de decisiones en la empresa Tubasec C.A.

José Fernando Esparza Parra

106 páginas

Proyecto dirigido por: José Marcelo Balseca Manzano, Mg.

Desarrollar e implementar un Proyecto de implantación de *Data Warehouse* está basado en toda una serie de fases, unas predecesoras de otras que tienen que ser implementadas paso a paso y sin saltarse una para realizar otra, el presente trabajo se basó en la metodología de HEFESTO, misa que consta de 4 grandes fases y sub procesos dentro de cada fase.

Esta metodología parte del análisis de requerimientos, mismo que se convierte en el pilar fundamental para la siguiente fase denominada análisis OLTP, aquí se determina si lo formulado y planteado en la fase de análisis es factible, entonces se puede diseñar el modelo lógico para mediante este determinar indicadores y saber exactamente que tablas y campos van a ser utilizados para generar dichos indicadores, finalmente se procede con la última fase que se denomina carga y actualización de datos.

Este trabajo se lo realizo completo en la empresa Tubasec C.A. en la ciudad de Riobamba, ya que ésta presentaba un fraccionamiento de información por contar con varios sistemas satélites y un ERP que estaban divorciados completamente, en términos tecnológicos. Sus datos eran incompatibles, pero al implementar esta solución, los altos mandos ya pueden contar con una visión global y real de la empresa en todas sus áreas y sub áreas.