



Tecnología Business Intelligence para tomar las mejores decisiones en tiendas Gourmet

José Ángel Alcañiz Villanueva

Grado de Ingeniería informática

Humberto Andrés Sanz

14 de junio de 2017



Esta obra está sujeta a una licencia de [Reconocimiento-NoComercial-SinObraDerivada 3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Tecnología Business Intelligence para tomar las mejores decisiones en tiendas Gourmet</i>
Nombre del autor:	<i>José Ángel Alcañiz Villanueva</i>
Nombre del consultor:	<i>Humberto Andrés Sanz</i>
Fecha de entrega (mm/aaa):	<i>06/2017</i>
Área del Trabajo Final:	<i>Business Intelligence</i>
Titulación:	<i>Grado de Ingeniería Informática</i>

Resumen del Trabajo (máximo 250 palabras):

Hoy en día, las empresas cuentan con gran cantidad de datos almacenados en diferentes bases de datos, pero que no aportan información relevante para la organización. Sin embargo, si esos datos son tratados de la manera adecuada pueden ofrecer información y conocimiento a los diferentes trabajadores. Es decir, les permitirá tomar las mejores decisiones para conseguir una ventaja competitiva.

En el presente trabajo, se realiza un proceso completo de BI a través de algunas de sus diferentes metodologías y tecnologías, para transformar los datos de una organización en información que permita responder a las necesidades del negocio. Para empezar, se selecciona la herramienta que mejor se adapta a las necesidades del trabajo. Una vez las herramientas están instaladas y configuradas para el entorno de trabajo, se desarrolla un proceso ETL para construir un *data warehouse*. Este repositorio será idóneo para obtener información, ya que los datos se integran y organizan en un modelo adecuado, para poder analizarlos y presentar la información rápidamente.

Finalmente, teniendo en cuenta los requerimientos de la empresa se crea un sistema de *reporting* y cubos OLAP. Este método, permitirá dotar a la empresa del conocimiento necesario para tomar las mejores decisiones en el cumplimiento de sus objetivos y en la mejora de sus procesos empresariales.

Abstract (in English, 250 words or less):

Nowadays, companies have a large amount of data stored in different data bases, but don't provide relevant information for the organization. However, if these data are managed in the right way, they can offer information and knowledge to different workers. That is to say, it will allow them to make the best decisions in their business to get a competitive advantage.

In the present project, a complete process of BI is performed through some of its different methodologies and technologies, to transform organizational data into information that allows answer the needs of the business. Firstly, the tool that best suits the needs of the project is selected. Once the tools are installed and configured for the working environment, an ETL process is developed for build a data warehouse. This will be a suitable repository for obtaining information, since the data are integrated and organized into an appropriate model that is able to analyze them and present the information quickly.

Finally, taking into account the requirements of the company, a reporting and OLAP cubes system is developed. This method will provide the company with the necessary knowledge to make the best decisions in the fulfillment of its objectives and in the improvement of its business processes.

Palabras clave:

Business Intelligence, Data Warehouse, ETL, Reporting, OLAP, Pentaho.

Índice

1.	Introducción.....	1
1.1	Contexto y justificación del trabajo.....	2
1.2	Objetivos del trabajo.....	4
1.3	Enfoque y método seguido.....	5
1.3.1	Costes económicos del trabajo.....	9
1.4	Planificación del trabajo	9
1.4.1	Identificación y análisis de riesgos.....	13
1.5	Breve resumen de productos obtenidos	15
1.6	Breve descripción de los otros capítulos de la memoria	16
2.	Tareas iniciales	17
2.1	Recogida de requisitos.....	17
2.2	Seleccionar las herramientas.....	17
2.3	Preparar el servidor.....	20
3.	Desarrollo del proyecto	22
3.1	Crear un <i>Data Warehouse</i> con Pentaho y MS SQL Server.....	22
3.1.1	Modelo conceptual.....	23
3.1.2	Modelo lógico.....	25
3.1.3	Diseño físico	26
3.1.4	Proceso ETL y creación del almacén de datos.....	28
3.2	Configurar la seguridad del servidor.....	33
3.3	Creación de informes	37
3.3.1	Estudio de clientes por su profesión	37
3.3.2	Estudio del proceso de ventas con ticket promedio y canasta promedio	41
3.3.3	Seguimiento de la campaña de puntos.....	45
3.3.4	Informe de ventas por tienda y día a través de un cubo OLAP	51
3.3.5	Rentabilidad de tiendas por día y hora a través de un cubo OLAP	55
3.3.6	Pedidos a proveedores servidos fuera de plazo	58

3.3.7	Pedidos con la cantidad entregada menor que la solicitada.....	61
3.4	Viabilidad del producto	65
3.4.1	Viabilidad técnica.....	66
3.4.2	Viabilidad operativa	66
3.4.3	Viabilidad Legal	67
3.4.4	Viabilidad financiera.....	67
3.4.5	Beneficios aportados	69
4.	Conclusiones.....	71
5.	Glosario.....	73
6.	Bibliografía	75
7.	Anexos	2
	Anexo 1. Preparación del entorno Windows, instalación y configuración de Pentaho	2
	Anexo 2. Configurar e inicializar MS SQL Server como la Base de Datos del repositorio de Pentaho	8
	Anexo 3. Instalar otras aplicaciones principales y de desarrollo accesorias a Pentaho.....	21
	Anexo 4. Procesos ETL con Kettle	24

Lista de figuras

Ilustración 1. <i>Enterprise Bus Architecture</i>	7
Ilustración 2. <i>Corporate Information Factory</i>	7
Ilustración 3. Diagrama de Gantt	12
Ilustración 4. Cuadrante mágico de Gartner sobre <i>Business Intelligence</i>	18
Ilustración 5. Diseño conceptual del proceso de ventas	24
Ilustración 6. Diseño conceptual del proceso de pedidos	24
Ilustración 7. Diseño físico del proceso de venta de tickets	27
Ilustración 8. Diseño físico del proceso de pedidos a proveedores	27
Ilustración 9. Orígenes de datos de Gourmet	28
Ilustración 10. Trabajo principal del proceso ETL	29
Ilustración 11. Crear la estructura del <i>data warehouse</i> con ETL.	29
Ilustración 12. Cargar los datos de las dimensiones con ETL.	30
Ilustración 13. Tabla ilustrativa con algunos registros de tienda.csv	30
Ilustración 14. Tabla ilustrativa con algunos registros de pais.csv	30
Ilustración 15. Tabla ilustrativa con algunos registros de cliente.csv	31
Ilustración 16. Solución valores vacíos	31
Ilustración 17. Tabla ilustrativa de producto.csv	31
Ilustración 18. Tabla ilustrativa de proveedor.csv	31
Ilustración 19. Cargar los datos de las tablas de hecho con ETL.	32
Ilustración 20. Tabla ilustrativa de cabeceraticket.csv	32
Ilustración 21. Tabla ilustrativa de pedido.csv	33
Ilustración 22. Permisos del rol <i>Business Analyst</i> en Pentaho	35
Ilustración 23. Permisos de las carpetas Gerencia y Barcelona	36
Ilustración 24. Diseño del <i>report</i> Estudio de clientes por profesión	38
Ilustración 25. Parámetro tienda	38
Ilustración 26. Clientes con profesión Alimentación	39
Ilustración 27. Clientes con profesión ama de casa	39
Ilustración 28. Clientes con profesión Economistas, Abogados & Admin. Empresas	40
Ilustración 29. Gráficos con los porcentajes de Importes vendidos y Puntos del ticket	40
Ilustración 30. Parámetros del informe de ticket y canasta promedio	42

Ilustración 31. Cabecera del informe del ticket y canasta promedio	43
Ilustración 32. Informe en Londres I durante el periodo navideño	44
Ilustración 33. Informe en Londres II durante el periodo navideño	45
Ilustración 34. Parámetros del informe de la campaña de puntos	46
Ilustración 35. Parámetro mes	46
Ilustración 36. Cabecera de la campaña de puntos de Barcelona en Septiembre	48
Ilustración 37. Informe de la campaña de puntos de Barcelona en Septiembre	49
Ilustración 38. Informe de la campaña de puntos de Barcelona en Diciembre	49
Ilustración 39. Cabecera de la campaña de puntos de Manhattan II en octubre	50
Ilustración 40. Cabecera de la campaña de puntos de Manhattan II en diciembre	50
Ilustración 41. Puntos de Madrid en cada compra de septiembre (primer mes de campaña)	51
Ilustración 42. Puntos de Madrid en cada compra de diciembre (último mes con datos)	51
Ilustración 43. Diseño del cubo de las ventas por tienda y fecha	52
Ilustración 44. Diseño de las dimensiones y jerarquías del cubo de las ventas por fecha y día	53
Ilustración 45. Navegación del cubo de ventas por fecha y día	53
Ilustración 46. Comparación de las ventas de las tiendas de Londres	54
Ilustración 47. Comparación de las ventas de las tiendas de Manhattan	54
Ilustración 48. Diseño del cubo de las ventas por día y hora	55
Ilustración 49. Diseño de las dimensiones y jerarquías del cubo de las ventas por día y hora	56
Ilustración 50. Rentabilidad por días de la semana	56
Ilustración 51. Horas sin ventas en Fort Lauderdale	57
Ilustración 52. Definición parámetro maxDiasEntrega	59
Ilustración 53. Ejecución del <i>report</i> Pedidos con retraso	60
Ilustración 54. Gráfico con el número de pedidos con retraso de los proveedores en las diferentes tiendas	61

Ilustración 55. Ejecución del <i>report</i> Pedidos con cantidad entregada menos que la solicitada	63
Ilustración 56. Pedidos con problemas separados por proveedores y tiendas	64
Ilustración 57. Cantidad de pedidos con problemas de cada proveedor y producto	64
Ilustración 58. Cantidades solicitadas y entregadas de productos	65
Ilustración 59. Requerimientos Pentaho.	68

1. Introducción

Desde la aparición de las Bases de Datos en las empresas, éstas han podido acceder a multitud de datos de diversa índole: tanto de su propia organización (clientes, pedidos, etc.) como de otras empresas colaboradoras (proveedores, transportistas, etc.)¹. Además, dichos datos provienen de diferentes programas (facturación, mantenimiento de clientes, etc.), por lo que se puede asegurar que las compañías cuentan con un histórico de datos de gran calidad.

A pesar de disponer de tal cantidad de datos, éstos han sido de difícil acceso y no han llegado a proporcionar la información que podían ofrecer. Sin embargo, los directivos de las empresas han sido capaces de utilizar estos datos para convertirlos en información, ya sea a través de consultas directas a la base de datos, introduciendo manualmente los datos en las populares hojas de cálculo o a través de aplicaciones más modernas que han conseguido mejorar el acceso a dichas bases de datos por parte de los usuarios.

En la actualidad, existe una gran variedad de herramientas de *Business Intelligence* (en adelante BI) capaces de juntar datos de diferentes fuentes, tanto de la propia organización como de fuentes externas. Dichos datos, pueden ser analizados en función de los objetivos que se pretenda obtener y presentarlos tanto a los directivos de la organización, como a los diferentes departamentos. De esta forma, podrán proyectar su trabajo hacia los objetivos de la organización y tomar mejores decisiones con la información recibida.

Algunas de las tecnologías que forman parte de un proyecto de BI y que pueden aplicarse al trabajo son:

- Creación de *data warehouse* o *data mart*.
- Integración de datos a través de ETL (de sus siglas en inglés *Extract, Transform and Load*).
- *Reporting*.
- Análisis OLAP (*On-Line Analytical Processing*).
- Previsiones.

- *Data mining*.
- Creación de *dashboard* o *scorecard*.

Es decir, un proyecto de BI puede incluir diferentes tecnologías, y cada una de estas tecnologías puede enfocarse para diferentes objetivos. En consecuencia, un proyecto de BI debe tener claros los objetivos a cubrir y saber aplicar el modelo de análisis que mejor se adapte al conocimiento que se pretenda obtener.

Por lo tanto, el actual trabajo debe servir para familiarizarle con las herramientas y tecnologías de BI, así como coger experiencia en el análisis de datos orientado a los objetivos de la organización. Es decir, escoger la herramienta que mejor se adapte a las necesidades y a los conocimientos del estudiante, instalarla, configurarla y construir un almacén de datos que permita poner en práctica algunas de las tecnologías, en concreto, la realización de un *data warehouse* a través de ETL, la elaboración de informes y la creación de cubos OLAP.

1.1 Contexto y justificación del trabajo

La recesión económica de los últimos años ha derivado en la búsqueda de soluciones por parte de muchas empresas, con la finalidad de volverse más competitivos o diferenciarse de la competencia. Además, el rápido avance de la tecnología y su integración en el mundo empresarial está obteniendo muy buenos resultados, ya que esta tecnología está brindando una gran cantidad de servicios que están ayudando a las empresas a mejorar sus procesos y a aumentar su productividad. Por esta razón, los directivos de las organizaciones pretenden buscar en la tecnología las herramientas que les faltan para ser más competitivos, ahorrar costes y tomar decisiones a partir de los datos de la empresa. Es decir, los directivos saben que los datos almacenados en sus empresas pueden ser analizados para responder a diferentes cuestiones sobre el negocio.

Pero, a pesar de que la mayoría los directivos de las empresas intuyen que con BI se pueden responder a muchas preguntas clave para el negocio,

desconocen el alcance real de estas herramientas. Por ejemplo, un directivo puede saber que a través de BI se puede conocer la situación económica de la empresa o cómo van las ventas de una sucursal, pero pueden desconocer que con la información que le brindan estas herramientas también podrán tomar una decisión para responder a la actual situación económica y para mejorar las ventas de sucursal.

Además, algunos directivos pueden verse amenazados con este avance informático, ya que son propensos a pensar que BI viene a sustituirles o que ellos no son capaces de tomar buenas decisiones. Sin embargo, la función de BI es precisamente ayudar a los directivos a tomar decisiones con información, lo que les permitirá ahorrar. Por un lado, el ahorro llegará en forma de tiempo, el cual lo podrán dedicar a otras funciones. Por otro lado, podrán reducir costes, lo que les permitirá destinar dinero a otros proyectos.

Pero, para alcanzar los mejores resultados, tanto los objetivos del departamento de BI como los objetivos de la organización deben estar alineados, razón por la que el CEO (director ejecutivo) y el CIO (jefe de sistemas) de una organización han de trabajar de manera conjunta y en la misma dirección. Es decir, orientando la inteligencia de negocio hacia los objetivos de la organización.

Por otro lado y, según experiencia propia, en las universidades no se profundiza demasiado en las áreas de BI, con excepción de algunas de las tecnologías utilizadas, por lo que los estudiantes universitarios solamente adquieren unos pocos conocimientos sobre dichas tecnologías. Como consecuencia, terminan la carrera sin llegar a profundizar en el funcionamiento de las diferentes herramientas y las posibilidades que éstas pueden ofrecer para alcanzar los objetivos propuestos.

Por lo tanto, las herramientas de BI son capaces de ofrecer más información, en tiempo real y con un aspecto visual, que ofrece un resultado más comprensible para ayudar a los directivos a tomar las mejores decisiones. Por esta razón, BI se ha convertido en una parte imprescindible para cualquier empresa, siendo capaz de ofrecer el conocimiento necesario para alcanzar los

objetivos de la organización, ya sea con la finalidad de mejorar las ventas, de ahorrar en costes o de tomar mejores decisiones. Esta tecnología, permitirá obtener valor para el negocio o diferenciarse de la competencia para obtener una ventaja competitiva.

En definitiva, en el presente trabajo se realizará un estudio de diferentes herramientas de BI para la construcción de un data warehouse de una cadena de tiendas Gourmet, la cual tiene sedes repartidas por varios países del mundo. Una vez construido dicho almacén de datos se crearán diferentes informes, tanto para los directivos de la organización como para los responsables de las diferentes tiendas. Así, podrán analizar diferentes situaciones y tomar las mejores decisiones para el buen funcionamiento de la organización, ya sea mejorando las ventas o ahorrando en costes, según dicten los objetivos de la organización. No obstante, no se partirá de una herramienta ya definida a priori, sino que se estudiarán algunas alternativas que se adapten a las necesidades del estudiante y a los objetivos del trabajo, ya que en el mercado existen muchas alternativas diferentes.

1.2 Objetivos del trabajo

Por un lado, los objetivos del presente trabajo están vinculados a alcanzar metas personales, seguramente por el interés del autor de adquirir nuevos conocimientos para encajar en un mundo laboral, ya que el departamento tecnológico debe estar cada vez más orientado a ofrecer valor al negocio. Por otro lado, están orientados a construir la tecnología necesaria para poder satisfacer los objetivos de negocio de la cadena de tiendas Gourmet.

En cuanto a satisfacer las metas personales, se pueden destacar los siguientes objetivos:

- El objetivo prioritario del estudiante es familiarizarse con las herramientas y tecnologías de BI, ya que prácticamente no ha tenido contacto con este tipo de tecnologías.

- En segundo lugar, se quiere aprender a crear un proceso ETL con el que diseñar y construir un *data warehouse* que resulte efectivo para convertir los datos en información relevante para la organización.
- En tercer lugar, se busca adquirir los conocimientos y habilidades necesarios para explotar las tecnologías de *reporting* y cubos OLAP, de manera que los resultados obtenidos permitan a los directivos tomar las mejores decisiones de una manera ágil y efectiva.

En lo que respecta a aplicar los objetivos de negocio de manera efectiva sobre las tecnologías de BI, se pueden destacar los siguientes objetivos, ordenados de menor a mayor importancia:

- Escoger unas herramientas de BI que se adapten a los recursos disponibles por el estudiante y que mejor puedan responder a las necesidades y objetivos del trabajo.
- Instalar y configurar correctamente el software necesario para poder implementar las tecnologías requeridas y satisfacer los objetivos de tiendas Gourmet.
- A partir de los datos proporcionados por tiendas Gourmet, realizar un proceso ETL con el que construir un *data warehouse* que pueda explotarse de manera efectiva para mejorar los procesos de ventas y de pedidos.
- Crear un sistema de informes a través de *reports* y cubos OLAP, que permitan ofrecer valor a la organización ayudando a los directivos y empleados de las sucursales a tomar las mejores decisiones.

1.3 Enfoque y método seguido

El trabajo está orientado a analizar los datos disponibles en una organización, con la intención de obtener información que permita ofrecer recomendaciones a los directivos de la empresa, ya sea para mejorar los procesos o para tomar las mejores decisiones. Para llevarlo a cabo, se necesita instalar, configurar y utilizar diferentes herramientas de BI, con las cuales el autor del presente trabajo no está familiarizado. Todo ello desde un enfoque meramente académico, ya que ni los datos ni la empresa citada en el trabajo son reales.

Cuando se inicia un proyecto orientado a objetivos, hay que tener en cuenta que dichos objetivos pueden cambiar a lo largo del tiempo, en especial a medio y largo plazo. Por ejemplo, puede que en el presente ejercicio, el objetivo de la organización sea aumentar las ventas, pero que en el año siguiente, el objetivo sea reducir costes. Por lo tanto, es posible que algunos de los modelos de análisis utilizados este año no sirvan para el siguiente. Sin embargo, hay partes del proyecto que sí que pueden ser de utilidad, por ejemplo, el proceso ETL para la creación de un *data warehouse*.

Además, puede que al cabo de dos o tres años el objetivo del negocio vuelva a ser aumentar las ventas, lo que provocará que los modelos que se habían construido con anterioridad vuelvan a ser de utilidad. Por esta razón, habrá que centrarse en una metodología de trabajo que permita adaptarse a los cambios que puedan surgir sobre los objetivos organizacionales. Es decir, se buscarán modelos de análisis terminados y de utilidad, aunque el proyecto no llegue a finalizarse completamente, para minimizar el riesgo de obtener un producto inservible.

En otras palabras, la estrategia a seguir se basa en tomar dos decisiones principales. Por una lado, la elección de la arquitectura del *data warehouse*, ya que será clave para organizar los modelos de análisis. Por otro lado, la elección de la metodología de trabajo, porque permitirá adaptar la forma de trabajar al actual trabajo y a los cambios que puedan surgir.

En cuanto al almacén de datos, existen diferentes tecnologías, unas más básicas y otras más completas y complejas. Además, teniendo en cuenta la dedicación que se le puede dar al proyecto, la experiencia del estudiante y los objetivos del presente trabajo, el estudio se ha limitado a dos arquitecturas básicas, las cuales se presentan a continuación²:

- *Enterprise Bus Architecture*: Se trata de una arquitectura que cuenta con varios *data marts* independientes.

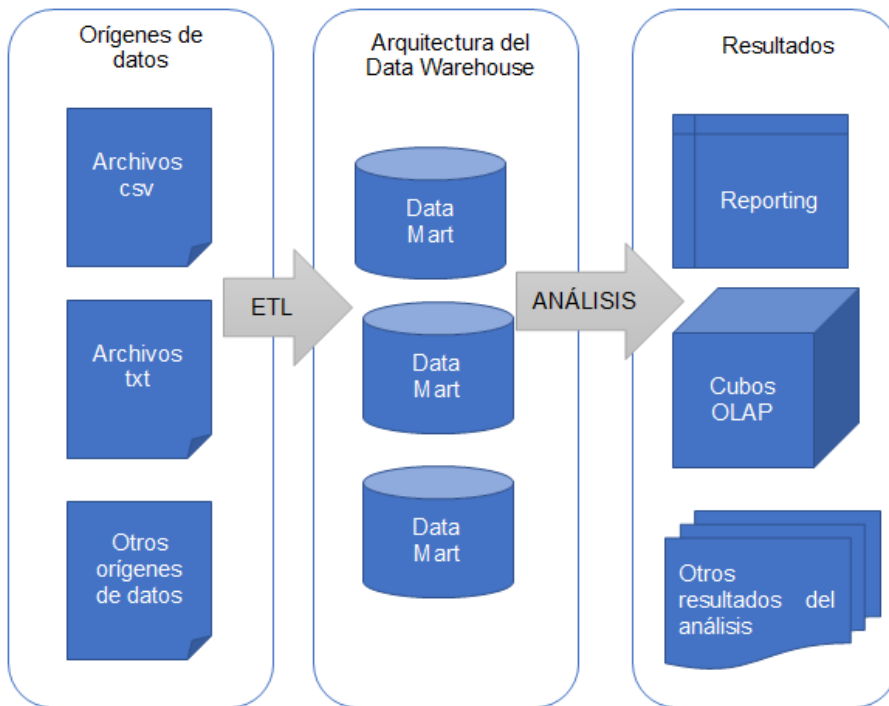


Ilustración 1. *Enterprise Bus Architecture*

- *Corporate Information Factory*: Es una arquitectura en la que existe un *data warehouse* y varios *data marts* dependientes del mismo.

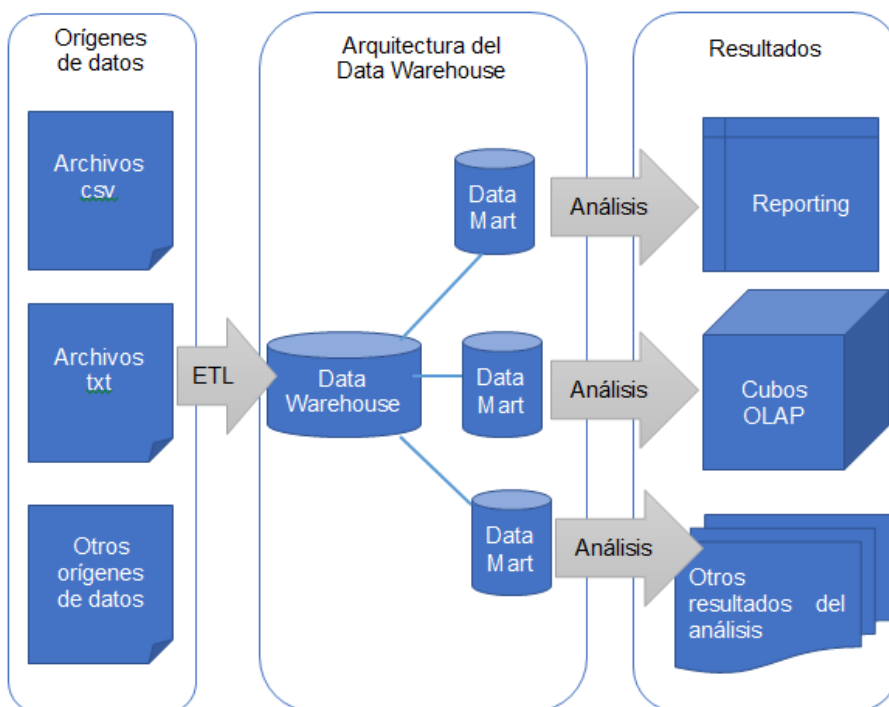


Ilustración 2. *Corporate Information Factory*

La ventaja que se obtiene con la arquitectura *Corporate Information Factory* sobre *Enterprise Bus Architecture* es que la primera tiene un almacén principal con los datos transformados y preparados. Es decir, que está orientado a la explotación analítica de la organización. Así, la creación de nuevos *data marts* se orientarán hacia un modelo de análisis concreto, pero con los datos ya transformados y preparados para el análisis. En resumen, para el presente trabajo resulta más conveniente utilizar la arquitectura *Corporate Information Factory*, pues con los datos ya preparados, se pueden crear nuevos *data marts* rápidamente para los nuevos modelos de análisis.

Se analizarán tres de las metodologías de trabajo más utilizadas:

- Método ágil: Flexible ante los cambios, con pocos roles y con procesos menos controlados.
- Método tradicional: Basado en normas, rígida ante los cambios y con un seguimiento estricto del plan inicial de desarrollo.
- Método iterativo incremental: Que consiste en dividir el proyecto en mini-proyectos, en los que se realizan iteraciones con un proceso similar. De esta manera, se puede entregar al cliente dichos mini-proyectos terminados.

Ante la necesidad de que la metodología debe permitir tanto cambiar la estrategia como obtener modelos de análisis completos y terminados, se seguirá un método ágil e incremental. Este modelo, permitirá adaptarse a los cambios de estrategia y poder disponer de mini-proyectos terminados que podrán reutilizarse en un futuro. Aun así, para la realización del trabajo se seguirán las recomendaciones de PMBOK. Pues, aunque no garantiza el éxito del proyecto, sí que permite tener una guía de buenas prácticas para encauzarlo tanto hacia los intereses del negocio como hacia los intereses del departamento de BI.

Por lo tanto, para la realización de cada modelo de análisis se realizará un mini-proyecto, que incluirá el análisis de requerimientos, tanto de las necesidades como de los datos, el desarrollo del modelo, las pruebas y la

documentación para el entregable, llevando en todo momento y de forma paralela un control de la calidad.

1.3.1 Costes económicos del trabajo

El trabajo se pretende realizar aprovechando el software y el hardware que el autor tiene disponible. No obstante, para la realización de un proyecto de esta magnitud no se requiere un gasto elevado, en especial si se escogen soluciones *open source* o gratuitas. Por lo que el coste del trabajo se calculará con el precio de un ordenador de coste medio para un estudiante y con un Sistema Operativo Windows actual. En cuanto al gestor de base de datos y a las herramientas de BI, se calculan sin coste inicial, ya que se pretende buscar herramientas *open source* o gratuitas. Además, no se contarán las horas de dedicación del autor ni otros gastos legales adicionales. Por lo tanto, el coste estimado para la realización del trabajo es:

Recurso	Coste mínimo
Servidor	700€
S.O. Servidor	135€
Herramientas B.I.	0€
Gestor de Base de datos	0€
Horas proyecto (286 horas)	0€
TOTAL	835€

El coste total puede variar, en función del precio de las licencias de software que finalmente se escojan.

1.4 Planificación del trabajo

En el presente trabajo se estudiará y documentará un proceso completo de BI: instalar y configurar las herramientas, dotar de seguridad al servidor de BI, crear un almacén de datos y analizar las necesidades de la organización para diseñar modelos de análisis que permitan a la organización tomar las mejores decisiones.

Para la realización del trabajo, el autor dedicará 3 horas diarias entre semana y 8 horas los sábados, sumando un total de 23 horas semanales. Los domingos, festivos y puentes, no se realizará ninguna tarea, mientras no se dé el caso en el que el proyecto corra algún riesgo y necesite dedicar más tiempo al proyecto. En consecuencia, contando los días laborables se obtiene un total de 286 horas para la realización del trabajo.

En cuanto a la planificación de los entregables, hay de destacar ciertos aspectos, ya que algunas tareas parecen planificadas con muy poco tiempo, como por ejemplo, dotar de seguridad al servidor, mientras que otras parece que se dediquen demasiados días, como es el caso de la creación de cubos OLAP. Sin embargo, no solamente se ha tenido en cuenta la complejidad de la tarea, sino que también se ha tenido en cuenta los conocimientos que el estudiante tiene sobre el área para la realización de la tarea.

Además, con el objetivo de ahorrar tiempo y mejorar la calidad de la memoria se han planificado tareas de documentación de la memoria en paralelo con cada una de las tareas, de manera que se pueda elaborar la memoria con la información más reciente de cada supuesto. A su vez, servirá para revisar si la tarea se está realizando con la suficiente calidad. Por ejemplo, el almacén de datos debe tener correctamente relacionadas las tablas para que las consultas sean efectivas, mientras que los modelos de análisis deben tener la suficiente calidad como para poder extraer información útil para el negocio.

A continuación, se muestra una tabla con la planificación y el diagrama de Gantt, donde se han definido las tareas a realizar en orden secuencial, y divididas en los cuatro entregables ya citados. En la tabla, también se indica la duración de cada tarea, las tareas que son dependientes de otras (previas y posteriores) y el supuesto responsable de la misma (en el caso de que se tratara de un proyecto real en una empresa). Además, se ha valorado la complejidad de 1 a 10, siendo 1 la menos compleja y 10 la más compleja, teniendo en cuenta las características de la tarea y los conocimientos del estudiante:

PLANIFICACIÓN TFG	Inicio	Final	Tareas Previas	Tareas Posteriores	Responsable	Complejidad 1 a 10
1. PAC1	22/02/2017	15/03/2017		2, 3, 4		
1.1. INICIO DEL PROYECTO	22/02/2017	05/03/2017			Jefe proyecto	5
Contexto y justificación del trabajo	22/02/2017	02/03/2017				
Establecer los objetivos del trabajo	28/02/2017	02/03/2017				
Gestionar los riesgos	02/03/2017	04/03/2017				
Documentar en la memoria	22/03/2017	05/03/2017				
1.2. PLANIFICAR	06/03/2017	15/03/2017		2, 3, 4	Jefe proyecto	5
Identificar y secuenciar las actividades	07/03/2017	12/03/2017				
Documentar en la memoria	06/03/2017	15/03/2017				
2. PAC2	16/03/2017	19/04/2017				
2.1. REVISIÓN PAC1	16/03/2017	19/03/2017	1		Jefe proyecto	3
Revisión y corrección PAC1	16/03/2017	19/03/2017				
2.2. RECOGER LOS REQUISITOS	20/03/2017	21/03/2017		2.3		5
2.3. SELECCIONAR LAS HERRAMIENTAS	21/03/2017	24/03/2017	2.2	2.4, 2.5	Arquitecto	6
Estudiar alternativas	21/03/2017	23/03/2017				
Documentar en la memoria	21/03/2017	24/03/2017				
2.4. PREPARAR EL SERVIDOR	22/03/2017	31/03/2017	2.3	2.5		6
Instalar las herramientas	22/03/2017	27/03/2017				
Configurar las herramientas	22/03/2017	29/03/2017				
Comprobar el funcionamiento	27/03/2017	30/03/2017				
Documentar en la memoria	22/03/2017	31/03/2017				
2.5. DATA WAREHOUSE	31/03/2017	11/04/2017	2.3, 2.4	3.2, 3.3	Arquitecto	8
Analizar los requerimientos	31/03/2017	03/04/2017				
Analizar los orígenes de datos	31/03/2017	05/04/2017				
Modelizar (ETL)	03/04/2017	11/04/2017				
Pruebas y control de calidad	04/04/2017	11/04/2017				
Documentar en la memoria	31/03/2017	11/04/2017				
2.6. SEGURIDAD	10/04/2017	19/04/2017			Arquitecto	5
Análisis y requerimientos	10/04/2017	12/04/2017				
Configurar servicio	11/04/2017	18/04/2017				
Pruebas y control de calidad	12/04/2017	18/04/2017				
Documentar en la memoria	10/04/2017	19/04/2017				
3. PAC3	20/04/2017	24/05/2017				
3.1. REVISIÓN PAC2	20/04/2017	21/04/2017	2		Jefe proyecto	3
Revisión y corrección PAC2	20/04/2017	21/04/2017				
3.2. REPORTING	21/04/2017	03/05/2017	2.5		Analista funcional	7
Análisis y requerimientos	21/04/2017	23/04/2017				
Desarrollo	22/04/2017	02/05/2017				
Pruebas y control de calidad	23/04/2017	02/05/2017				
Documentar en la memoria	21/04/2017	03/05/2017				
3.3. CUBO OLAP	02/05/2017	24/05/2017	2.5		Analista funcional	9
Análisis y requerimientos	02/05/2017	10/05/2017				
Desarrollo	08/05/2017	23/05/2017				
Pruebas y control de calidad	09/05/2017	23/05/2017				
Documentar en la memoria	03/05/2017	24/05/2017				
4. MEMORIA	25/05/2017	14/06/2017				
4.1. REVISIÓN Y CORRECCIÓN PAC3	25/05/2017	26/05/2017	3		Jefe proyecto	3
4.2. COMPLETAR LA MEMORIA	25/05/2017	05/06/2017		4.3	Resp. área	6
4.3. ELABORAR PRESENTACIÓN	05/06/2017	13/06/2017	4.2		Jefe proyecto	6
4.4. FINALIZACIÓN Y CIERRE	13/06/2017	14/06/2017			Jefe proyecto	1

Diagrama de Gantt:

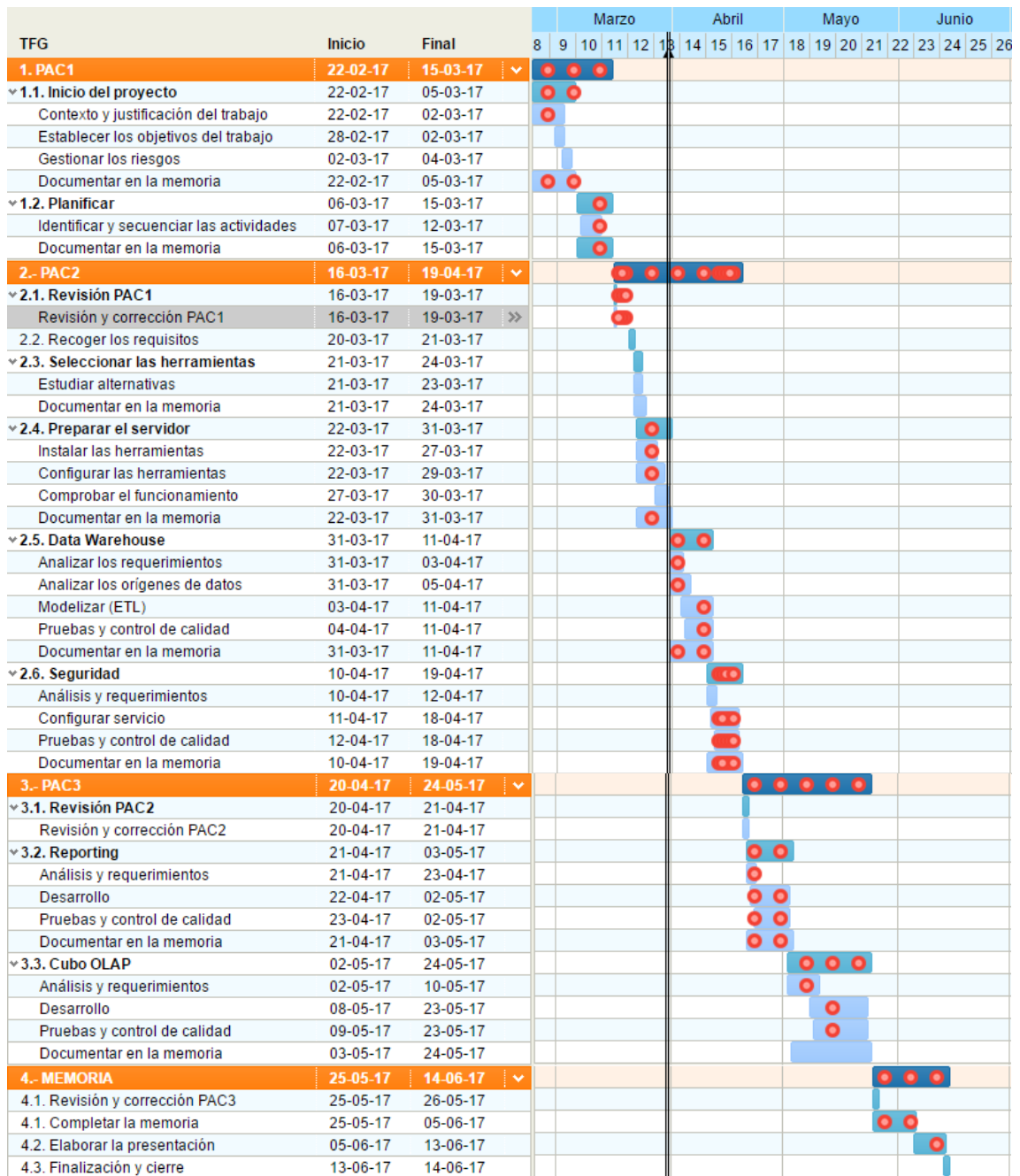


Ilustración 3. Diagrama de Gantt

Legenda:  Festivos y puentes

1.4.1 Identificación y análisis de riesgos

Todo proyecto tiene unos riesgos asociados que generan cierta incertidumbre y que se deben identificar antes de iniciar el trabajo, ya sea para evitar un impacto negativo o para aprovechar la oportunidad. Por lo tanto, una buena identificación de riesgos es crucial para el buen desarrollo de la planificación y término del proyecto:

Riesgo 1

Nombre	Retraso en la configuración de las herramientas
Causa	Se desconocen las herramientas a utilizar El estudiante desconoce las herramientas de BI
Consecuencia	Pone en peligro todo el proyecto, ya que el resto de tareas dependen de esta configuración
Probabilidad	Media
Impacto	Alto
Acción	Buscar una herramienta <i>open source</i> con una gran comunidad de apoyo Dar un plazo más amplio a la realización de la tarea

Riesgo 2

Nombre	No saber aplicar correctamente los modelos de análisis
Causa	El estudiante desconoce la utilización de tecnologías BI
Consecuencia	Se pone en peligro la realización de un modelo concreto
Probabilidad	Media
Impacto	Medio
Acción	Ejecutar un modelo de desarrollo incremental por mini-proyectos

Riesgo 3

Nombre	Modelo conceptual incorrecto para los posibles <i>Data Marts</i>
Causa	El estudiante está poco familiarizado con el diseño de bases de datos
Consecuencia	Los modelos dispondrían de datos bien relacionados para realizar un análisis correcto
Probabilidad	Baja
Impacto	Alto
Acción	Realizar un diseño lo más simple posible Valorar la necesidad de <i>Data Marts</i> para cada modelo de análisis

Riesgo 4

Nombre	Tiempo establecido sobredimensionado
Causa	El estudiante ha sido muy cauteloso con la planificación y sobra demasiado tiempo
Consecuencia	Se trata de una oportunidad para mejorar el proyecto
Probabilidad	Baja
Impacto	Bajo
Acción	Posibilidad de probar otras herramientas

Por otro lado, es posible encontrarse con otros riesgos que ya no dependen tanto del proyecto. Por un lado están los riesgos asociados al material utilizado. Por ejemplo, el hardware utilizado corre el riesgo de estropearse, por lo que será necesario recurrir a otros ordenadores y el software puede corromperse ante una subida de tensión, por lo que se realizarán copias de seguridad. Por otro lado, están los riesgos asociados a factores externos al proyecto. Por ejemplo, motivos laborales o personales que impidan dedicar la totalidad de las horas planificadas. En este caso, habrá que modificar la programación, por ejemplo, trabajando los días festivos.

1.5 Breve resumen de productos obtenidos

Tras la finalización del trabajo, se han obtenido diferentes documentos y productos. Por lo tanto, el estudiante ha realizado varios entregables durante la realización del trabajo, como son las tres entregas parciales (PAC1, PAC2 y PAC3), así como la memoria y la presentación del trabajo. Sin embargo, se prefiere enfocar los productos obtenidos desde un punto de vista más amplio, es decir, dirigido a los intereses que pueda tener el lector del trabajo:

- Guías de instalación de los productos necesarios para la instalación y configuración de Pentaho:
 - Preparación del entorno Windows, instalación y configuración de Pentaho (ver Anexo 1).
 - Configuración e inicialización de MS SQL Server como repositorio de Pentaho (ver Anexo 2).
 - Instalación y configuración de otras aplicaciones accesorias a Pentaho (ver Anexo 3).
 - Creación del proceso ETL con Kettle (ver Anexo 4).
- Configuración de la seguridad del servidor.
- Construcción de un *data warehouse* destinado a satisfacer las necesidades de mejorar los procesos de ventas y de pedidos de la organización. Para conseguir dicho repositorio de datos se ha realizado un proceso ETL con la herramienta Pentaho Data Integration (Kettle).
- Realización de diferentes reports y cubos OLAP (mondrian) sobre los procesos de ventas y pedidos a proveedores, que es lo que al cliente final le permite tomar las mejores decisiones de negocio:
 - Estudio de clientes por profesión.
 - Estudio del proceso de ventas con ticket promedio y canasta promedio.
 - Seguimiento de la campaña de puntos.
 - Informe de ventas por tienda y día (OLAP).
 - Rentabilidad de tiendas por día y hora (OLAP).
 - Pedidos a proveedores servidos fuera de plazo.
 - Pedidos con la cantidad entregada menor que la solicitada.

1.6 Breve descripción de los otros capítulos de la memoria

En el resto del trabajo se pueden encontrar los siguientes capítulos:

- Tareas iniciales: Capítulo donde se seleccionan las herramientas necesarias para la realización del trabajo y se configura el servidor para el correcto desarrollo del proyecto. Es decir, el servidor quedará preparado para crear el *data warehouse*, el cual servirá posteriormente como repositorio para los informes y cubos OLAP.
- Desarrollo del proyecto: En este capítulo tiene lugar el grueso del trabajo de BI, ya que se realiza un proceso ETL para crear el repositorio, se configura la seguridad del servidor de Pentaho y se crean los informes y cubos OLAP que permitirán a la organización tomar las mejores decisiones de negocio.
- Conclusiones: Capítulo en el que se realiza una reflexión crítica sobre las lecciones aprendidas por el estudiante y si se han alcanzado los objetivos propuestos. Además, se analiza el seguimiento de la planificación y la metodología seguida, sin dejar de lado aquellas líneas de trabajo que han quedado pendientes de explorar.
- Glosario: En este capítulo se definen o comentan las palabras del trabajo que el autor ha considerado necesario comentar.
- Bibliografía: Capítulo donde se relacionan los textos y fuentes consultadas, que provienen de diferentes soportes.
- Anexos: Capítulos agregados que quedan fuera del trabajo principal, pero que quedan como referencia de consulta. En ellos, se incluye la preparación del entorno para Windows, la instalación y configuración de Pentaho y otras herramientas, la configuración de MS SQL Server como repositorio de Pentaho y el proceso ETL con Kettle.

2. Tareas iniciales

2.1 Recogida de requisitos

Teniendo en cuenta la justificación del trabajo y los objetivos propuestos, se buscarán aplicaciones y herramientas que no supongan un gasto extra para el estudiante, ya sean *open source* o gratuitas. Además, se intentará que el software sea compatible tanto con el hardware como con el Sistema Operativo que el estudiante tiene disponible.

En primer lugar y como requisito principal, habrá que buscar las herramientas necesarias para responder a las necesidades del proyecto. Es decir, que permita construir almacenes de datos, crear *reports* y construir cubos OLAP.

En segundo lugar, las herramientas seleccionadas deben ser apropiadas para cumplir con los objetivos del proyecto. De una forma muy general, los objetivos pasarían por aprender a utilizar las herramientas y las tecnologías de BI a la vez que se aprende a orientarlas hacia los objetivos del negocio.

Para la selección del gestor de bases de datos, la intención del autor es buscar una herramienta de BI que permita utilizar SQL Server, ya que es único gestor de bases de datos con el que el estudiante ha trabajado. No obstante, si fuera necesario se podría utilizar un gestor de bases de datos alternativo.

En cuanto a la formación, se buscarán herramientas que tengan un alto grado de aceptación en el mercado y cuenten con gran apoyo de la comunidad, lo que permitirá al estudiante formarse con la utilización de la herramienta y encontrar soluciones a problemas que puedan surgir.

2.2 Seleccionar las herramientas

Dados los escasos conocimientos del estudiante con aplicaciones de BI, la selección de las herramientas será un factor clave para la correcta ejecución del proyecto. Por esa razón, para comenzar con la selección habrá que estudiar diferentes alternativas, con la finalidad de seleccionar la que mejor se adapte al trabajo.

Para ello, se puede partir del cuadrante mágico de Gartner³ para BI, líder mundial en consultoría e investigación de tecnologías de la información. A partir de dicho cuadrante, el cual se muestra a continuación, se podrá obtener una visión rápida tanto de las herramientas más utilizadas como de las tendencias del mercado, lo que facilitará la selección de la herramienta. En el cuadrante, se puede apreciar que los líderes del mercado son Tableau, Microsoft y Qlik, mientras que otras compañías fuertes como SAP e IBM se quedan en el cuadrante de visionarios. En cuanto a los jugadores de nicho, podemos apreciar que existen compañías como Pentaho y Oracle. Por otro lado, se echa en falta la aparición de más herramientas *open source* en el cuadrante, como puede ser el caso de Jaspersoft.



Ilustración 4. Cuadrante mágico de Gartner sobre *Business Intelligence*

Viendo el cuadrante mágico, solamente se encuentran dos alternativas que cumplan los requisitos marcados, en concreto, las herramientas de Qlik y Pentaho, por lo que se analizarán ambas herramientas con la finalidad de escoger la que mejor se adapte al proyecto.

Pentaho es una herramienta desarrollada bajo la filosofía del software libre, la cual ofrece soluciones completas para las diferentes tecnologías existentes de BI, mientras que Qlik parte de la filosofía de que el usuario final debe tener la autonomía necesaria para poder crear sus soluciones, por lo que apuesta fuerte por el aspecto visual y la facilidad de uso.

Si se analizan las herramientas buscando cual será mejor para satisfacer los objetivos del negocio, es fácil darse cuenta de que se trata de un objetivo que no depende tanto de la herramienta, sino de los conocimientos de la persona que crea el modelo. Todas las herramientas de inteligencia de negocio están pensadas para tomar decisiones, por lo que ambas herramientas servirán para este propósito. No obstante, se analizarán teniendo en cuenta las necesidades sobre las diferentes tecnologías que se utilizarán en el trabajo:

- Creación de un *data warehouse*: Para la creación del almacén de datos, Pentaho cuenta con una herramienta que permite crear almacenes de datos en diferentes gestores bases de datos, entre los que se incluye MS SQL Server. Sin embargo, Qlikview prefiere utilizar los diferentes orígenes de datos para que sea el mismo usuario el que haga un modelo para presentar los datos rápidamente en pantalla. Por lo tanto, para la creación del almacén de datos con Qlik se debería recurrir a otra herramienta, que podría incluso ser el mismo gestor de bases de datos.
- Creación de *reports*: Para la creación de informes, tanto Pentaho como Qlik disponen de las herramientas necesarias para poder realizarlos. La diferencia reside en que Pentaho está más orientado hacia el personal técnico o personal informático, mientras que Qlik sigue buscando la facilidad para el usuario final.
- Creación de cubos OLAP: En cuanto a la creación de cubos OLAP, en ambos casos se aprecia la tendencia a recurrir a tablas de hecho y diferentes dimensiones para su creación. Sin embargo, Pentaho precisa conocimiento específico del tema, mientras que Qlik sigue orientado hacia una presentación llamativa y una construcción asequible para los usuarios finales del producto.

Por lo tanto, queda demostrado que Qlik posee unas herramientas mejor diseñadas para que sean los propios usuarios los que puedan crear los modelos que necesiten en cada momento. Es decir, tiene mayor visión de futuro que Pentaho. Sin embargo y, a pesar de que Pentaho ha bajado del cuadrante de visionario al cuadrante de jugadores de nicho, el estudiante busca una herramienta más orientada hacia personal técnico. Además, uno de los objetivos principales del presente trabajo es aprender sobre la inteligencia de negocio a través de las herramientas de BI, por lo que Pentaho cumple mejor con dicho propósito, en especial con el diseño de *data warehouse*. Finalmente, la herramienta seleccionada para llevarlo a cabo será Pentaho Community 7.0.

Una vez seleccionada la herramienta de BI, ya se puede asegurar que el gestor de bases de datos a utilizar será MS SQL Server Express 2014, pues se trata de la herramienta que mejor conoce el estudiante. Por otro lado, el Sistema Operativo del ordenador será Microsoft Windows 7x64, pues es el que el autor tiene instalado en su ordenador y se adapta a las necesidades del proyecto.

2.3 Preparar el servidor

Para la instalación de Pentaho Server se necesita que tanto el hardware como el Sistema Operativo sean de 64 bits. Por un lado, se recomienda utilizar un Sistema Operativo para servidor (Microsoft Windows 2008 Server R2 o 2012 Server). Sin embargo, se ha utilizado Windows 7, ya que es el sistema operativo más accesible con el que se contaba. Por otro lado, se recomienda un mínimo de 8 GB de RAM, con 4 GB dedicados para el servidor. Pero, una vez más se encuentran limitaciones, ya que el PC donde se instalará el servidor no cuenta más que con 4 GB de RAM. No obstante, las pruebas realizadas con posterioridad a la instalación dan un resultado aceptable para la correcta ejecución del trabajo.

Para realizar una correcta instalación y configuración del servidor se ha seguido la guía de instalación de Pentaho⁴. Así, se ha dividido la instalación en tres apartados:

- Preparación del entorno Windows, instalación y configuración de Pentaho: En un primer momento, hay que crear la estructura de directorios donde se instalará Pentaho y las diferentes herramientas de análisis. Además, será necesario instalar el gestor de base de datos escogido (MS SQL Server) y realizar la instalación de Pentaho Community, la cual incluye Apache Tomcat, un servidor web donde correrá Pentaho. Además, hay que realizar la instalación y configuración de java, ya que es necesario modificar las variables de entorno para Pentaho. Puede verse el proceso completo en el [Anexo 1](#).
- Configurar e inicializar MS SQL Server como la Base de Datos del repositorio de Pentaho: Una vez preparado el entorno Windows y el servidor web con Pentaho, hay que configurar MS SQL Server como repositorio, donde habrá que realizar ciertas operaciones. Entre otras, será necesario habilitar MS SQL Server para recibir conexiones remotas y habilitar TCP/IP para SQL, ya que Pentaho debe ser accesible desde otras estaciones de trabajo de la red local. Además, hay que ejecutar ciertos scripts para habilitar los repositorios y comprobar el correcto funcionamiento del servidor Tomcat, el cual ya viene instalado y configurado con Pentaho. Para ver el proceso completo de configuración, se invita al lector a seguir el enlace al [Anexo 2](#).
- Instalar otras aplicaciones principales y de desarrollo accesorias a Pentaho: Además de tener instalada la aplicación de Pentaho, son necesarias otras herramientas del mismo fabricante para realizar el proyecto. En concreto, será necesario instalar:
 - Data Integration, que permitirá realizar el proceso de ETL.
 - Report Designer, para realizar diferentes informes personalizados, los cuales pueden incluir gráficos para darle un aspecto más llamativo y facilitar la comprensión.
 - Schema Workbench, que permitirá la creación de cubos OLAP, lo que permitirá analizar diferentes variables al mismo tiempo y, si se desea, con una dimensión temporal.

La guía de instalación y configuración de las aplicaciones accesorias puede verse en el [Anexo 3](#).

3. Desarrollo del proyecto

3.1 Crear un *Data Warehouse* con Pentaho y MS SQL Server

Un *data warehouse* es el núcleo de un sistema de inteligencia de negocio. Se trata del repositorio del que se nutrirán el resto de tecnologías de BI, el cual funcionará como un almacén de datos estratégico para resolver sus necesidades. Por lo tanto, debe estar orientado a cubrir sus objetivos de negocio y a los diferentes aspectos y datos que se pueden cuantificar en dichos procesos. Así, un *data warehouse* está compuesto por:

- Tablas de hecho⁵: Es la tabla central de un esquema dimensional que representa a los procesos de negocio de la compañía y contiene los valores de negocio.
- Dimensiones⁶: Son los parámetros de los que dependen los hechos.
- Métricas: Se trata de los indicadores con los que se pueden medir los procesos de negocio.

Además, un *data warehouse* puede estructurarse con dos tipos diferentes de esquemas:

- Esquema en estrella: Se trata de un esquema en el que la tabla de hecho (proceso) se sitúa en el centro, mientras que las dimensiones (parámetros) parten de la tabla de hecho, por lo que si se representa gráficamente, su estructura se asemeja a una estrella.
- Esquema en copo de nieve: Tiene una estructura parecida al esquema en estrella, con la peculiaridad de que las tablas de dimensión se implementan con más de una tabla, con la finalidad de eliminar redundancia de datos. A pesar de esta ventaja, se trata de una estructura con peor rendimiento.

Para el presente trabajo, se utilizarán los datos disponibles de una cadena de tiendas Gourmet, de la que solamente se tienen datos de unos pocos meses, es decir, no se dispone de una cantidad elevada de datos. Por lo tanto, se ha

decidido crear una estructura en estrella, ya que la redundancia de datos será poco significativa.

Los datos que llegan desde tiendas Gourmet vienen en archivos con formato de texto CSV, siendo necesario procesarlos para integrarlos en una base de datos con el formato adecuado. Dicho proceso se realizará a través de *Pentaho Data Integration*, que permite crear procesos de ETL con la herramienta *Kettle* y almacenar los datos en una base de datos, que en este caso será MS SQL Server. Además, como el tiempo para la entrega del trabajo está limitado, no se desea alargar el proyecto en exceso. En consecuencia, solamente se realizará un pequeño almacén de datos que permita realizar algunos modelos de análisis, aunque con la intención de que sean de ayuda para tomar decisiones y de que sean relevantes para la organización.

3.1.1 Modelo conceptual

Para la creación del modelo conceptual se identificarán los objetivos de la organización y los procesos sobre los que se pretende actuar. Por esta razón, se supondrá que uno de los objetivos de la organización es mejorar las ventas, por lo que se cubrirá este proceso de negocio a partir de los tickets de venta que las diferentes sucursales generan cada día.

No obstante, se pretende que el almacén de datos pueda servir para futuras necesidades, ya que los objetivos de las organizaciones cambian con el tiempo. De ahí, que también se cubra el proceso de pedidos a proveedores, partiendo de los datos sobre pedidos, productos y proveedores que ha facilitado la empresa.

Por lo tanto, se diseñará un modelo conceptual para cada uno de los procesos, uno para el proceso de ventas y otro para el proceso de pedidos. Cada modelo contendrá su tabla de hechos, de la que partirán el resto de dimensiones del modelo. Además, se crearán dos tablas de dimensión tiempo, ya que son de gran utilidad en los cubos OLAP, como se explicará en el siguiente apartado.

El proceso de tickets de venta incluirá las siguientes dimensiones:

- Tienda
- Cliente
- Fecha (Dimensión Tiempo para fechas).
- Hora (Dimensión Tiempo para horas)

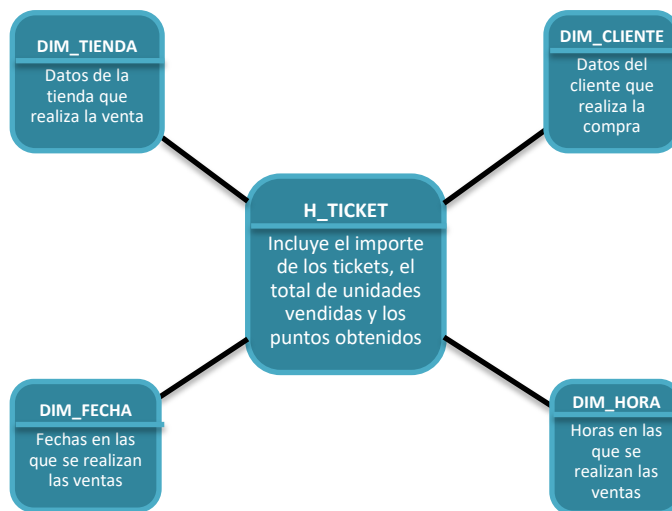


Ilustración 5. Diseño conceptual del proceso de ventas

Mientras que el proceso de pedidos incluirá:

- Tienda
- Producto
- Fecha

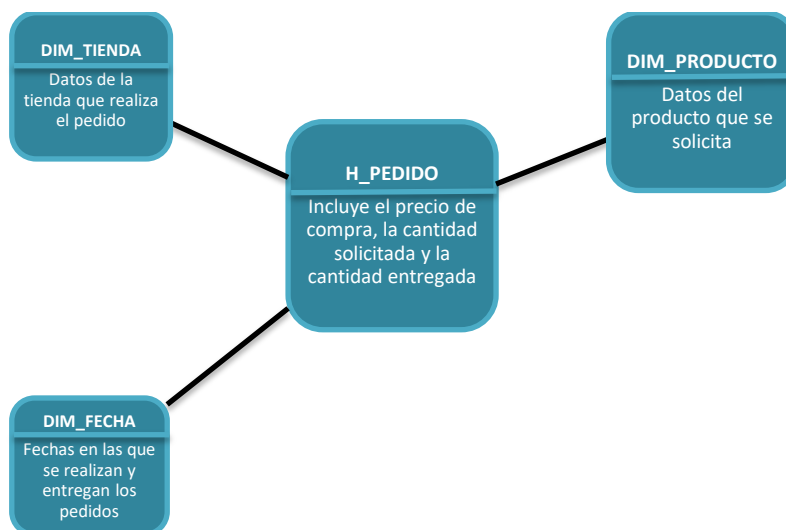


Ilustración 6. Diseño conceptual del proceso de pedidos

3.1.2 Modelo lógico

En el modelo lógico se incluirán las métricas que forman parte de las tablas de hechos y los atributos de las dimensiones. A su vez, cada tabla de hechos contiene una clave primaria, una clave foránea a cada una de las dimensiones y las métricas con las que medir los procesos.

Merecen especial atención las dos tablas de dimensión “tiempo”, ya que en las bases de datos se suelen almacenar los hechos conforme van ocurriendo a lo largo del tiempo. Es decir, contienen fechas (variables) para estudiar. La primera, la tabla dim_fecha contendrá los años, meses, nombres de los meses, días y días de la semana. Esta dimensión, permitirá a los usuarios escoger los meses por su nombre o los días de la semana cuando estén trabajando en un cubo OLAP. Por ejemplo, observando las ventas que ocurren los miércoles. La segunda, la tabla dim_hora, contendrá las horas. Lo que permitirá, por ejemplo, que el usuario pueda recorrer las diferentes horas en las que ha habido ventas. Además, la clave primaria se ha declarado de tipo entero, ya que resulta menos costoso hacer búsquedas sobre enteros que sobre tipos de formato *date*.

A continuación, se representa el modelo lógico de los procesos:

Proceso de ventas

TABLA DE HECHO	CLAVES FORÁNEAS	MÉTRICAS
H_TICKET Clave primaria: CODVENTA	NOMBRETIENDA CODCLIENTE FECHA HORA	IMPORTETOTAL TATALUNIDADES PUNTOSTICKET

TABLA DE DIMENSIÓN	CLAVE PRIMARIA	ATRIBUTOS
DIM_TIENDA	NOMBRE	NOMBRE DIRECCION SUPERFICIE FORMATO PAIS EXTENSIONPAIS POBLACIONPAIS
DIM_CLIENTE	CODCLIENTE	NOMBRECLIENTE SEXO FECHANACIMIENTO ESTADOCIVIL PROFESION NUMEROHIJOS NACIONALIDAD

DIM_FECHA		TOTALCOMPRAS PUNTOSACUMULADOS
	FECHA	AÑO MES SEMANA DIA DIASEMANA NMES NDIA NDIASEMANA
DIM_HORA	HORA	

Proceso de pedidos

TABLA DE HECHO	CLAVES FORÁNEAS	MÉTRICAS
H_PEDIDO Clave primaria: CODPEDIDO	NOMBRETIENDA CODPRODUCTO FECHASOLICITUD FECHAENTREGA	PRECIOCOMPRA CANTIDADSOLICITADA CANTIDADENTREGADA

TABLA DE DIMENSIÓN	CLAVE PRIMARIA	ATRIBUTOS
DIM_TIENDA	NOMBRE	NOMBRE DIRECCION SUPERFICIE FORMATO PAIS EXTENSIONPAIS POBLACIONPAIS
DIM_PRODUCTO	CODPRODUCTO	DESCRIPCION NOMBREPAIS COSTE PRECIOVENTA NOMBRESUBFAMILIA MARCA NOMBREPROVEEDOR
DIM_FECHA	FECHA	AÑO MES SEMANA DIA DIASEMANA NMES NDIA NDIASEMANA

3.1.3 Diseño físico

Finalmente, se creará el diseño físico a través de un proceso ETL, partiendo de los ficheros de la empresa Gourmet y de diferentes scripts en SQL, lo que permitirá ajustar los datos a las necesidades de los modelos, rellenando los datos faltantes o transformando los datos al formato adecuado, según las

necesidades. Así, se podrán crear las tablas necesarias para el almacén de datos. Tras la ejecución de la herramienta de ETL, quedará completado el *data warehouse*, por lo que el diseño físico de la base de datos con los procesos de venta de tickets y pedidos a proveedores deben ser los siguientes:

Proceso de venta de tickets

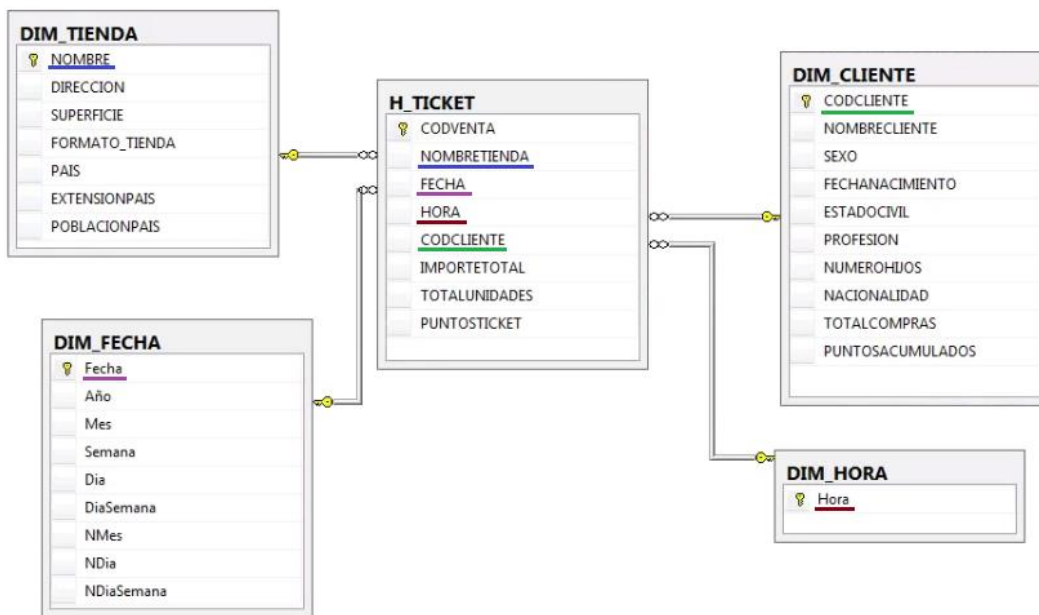


Ilustración 7. Diseño físico del proceso de venta de tickets

Proceso de pedidos a proveedores

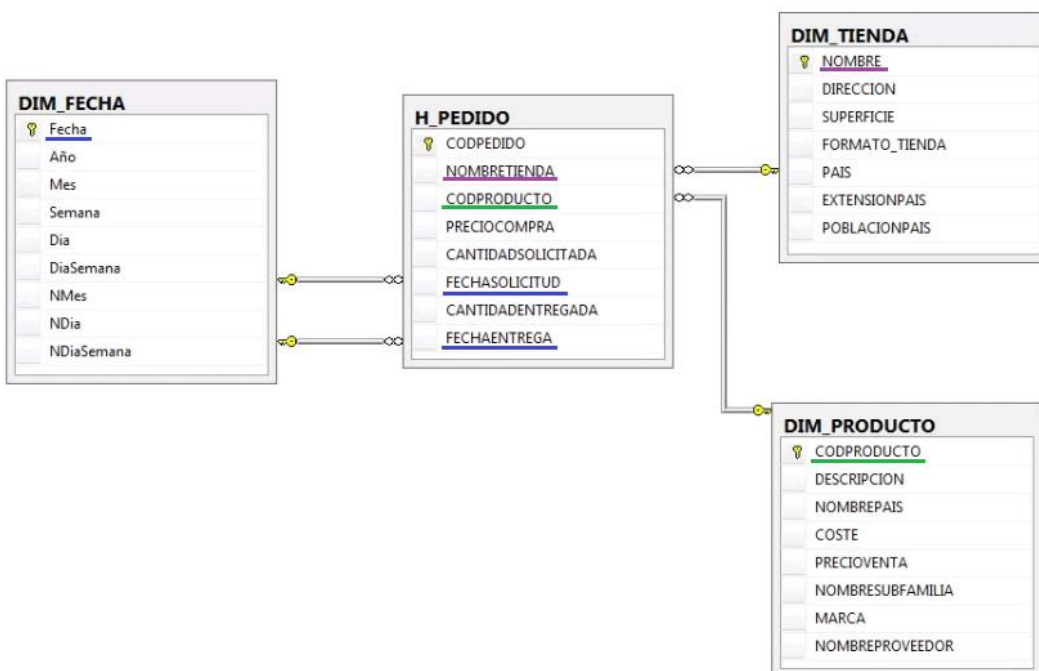


Ilustración 8. Diseño físico del proceso de pedidos a proveedores

3.1.4 Proceso ETL y creación del almacén de datos

Para la creación del almacén, se han utilizado los archivos suministrados por tiendas Gourmet con la información de sus procesos de negocio, con la que se han realizado los procesos ETL necesarios para construir el *data warehouse*. Además, se han creado otras tablas de dimensiones de tiempo, ya que son de gran utilidad para crear series temporales de la variable a estudiar en la construcción de cubos OLAP.

Como orígenes de datos, se han utilizado diferentes ficheros de datos para la creación de las tablas de hechos y de dimensiones:

Fichero de origen	Descripción
Cabeceraticket.csv	Contiene la información de un ticket de venta
Cliente.csv	Información sobre el cliente que realiza la compra
Tienda.csv	Diferentes sucursales de la cadena Gourmet
Pais.csv	Países en los que existen sucursales
Pedido.csv	Pedido de productos que realiza una tienda
Producto.csv	Productos que utiliza la cadena Gourmet
Proveedor.csv	Diferentes proveedores de los productos

Ilustración 9. Orígenes de datos de Gourmet

Se ha seguido un proceso ETL con Kettle, para lo cual se han configurado varios trabajos enlazados, de manera que con la ejecución del primer trabajo se realiza todo el proceso.

Trabajo principal del proceso ETL

El trabajo principal es el encargado de llamar al resto de trabajos para crear el almacén de datos de manera correcta:

- En primer lugar, comprueba la conexión con SQL Server, para verificar que se pueden crear las tablas y cargar los datos en la base de datos Gourmet.
- En segundo lugar, llama a un trabajo que crea la estructura de la base de datos, donde posteriormente se almacenarán los datos.
- Posteriormente, ejecuta un trabajo para cargar los datos en las tablas de dimensiones, ya que las tablas de hechos contienen las claves foráneas a las tablas de dimensiones y han de cargarse con posterioridad.

- Concluye con un trabajo para cargar los datos en las tablas de hechos

Si durante el proceso se encuentra algún error, el sistema enviará un correo electrónico al responsable de ejecución de la tarea:

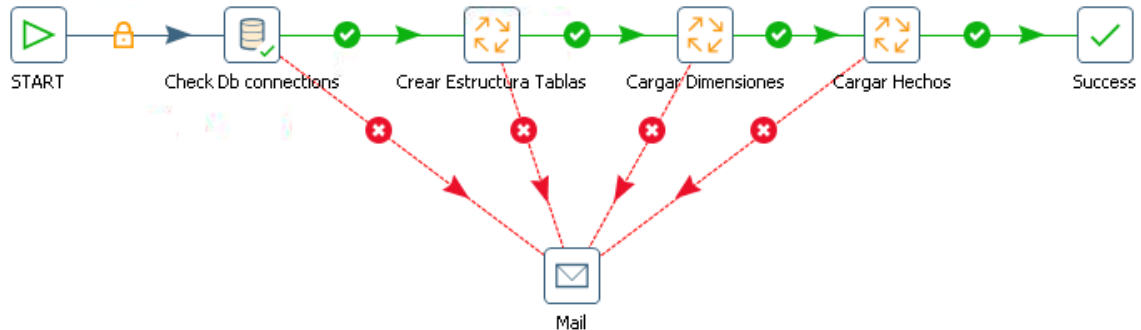


Ilustración 10. Trabajo principal del proceso ETL

Crear la estructura de tablas a través del proceso ETL

En primer lugar y, antes de cargar los datos, hay que crear la estructura del almacén, para poder insertar posteriormente los datos. Primero, se han creado las tablas necesarias para los procesos de ventas y de pedidos. Seguidamente, se han creado las dimensiones *dim_fecha* y *dim_hora* a través de sendos *scripts* en SQL, ya que se trata de unas tablas que permiten el estudio de diferentes variables en un espacio temporal:

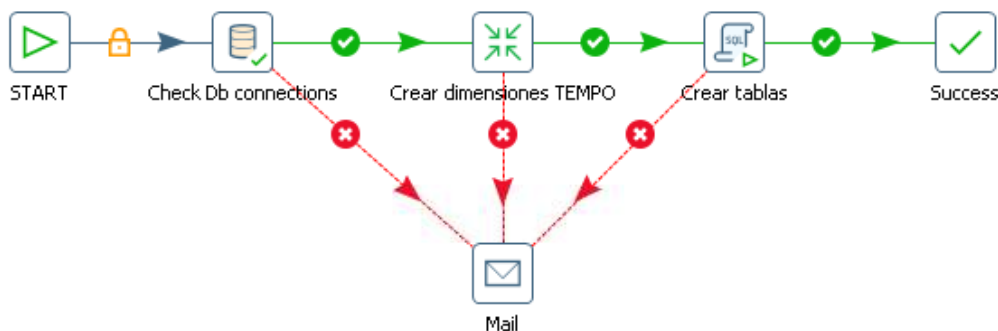


Ilustración 11. Crear la estructura del *data warehouse* con ETL.

Cargar las dimensiones a través del proceso ETL

En este trabajo, se cargan los datos de las dimensiones de todos los procesos, tanto de ventas como de pedidos. En él, se realizarán diversas transformaciones de datos y/o unión de varios orígenes de datos, en función de las necesidades de los modelos de análisis.

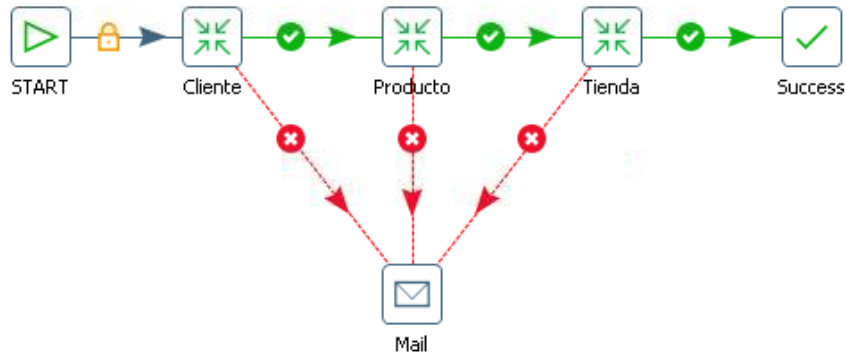


Ilustración 12. Cargar los datos de las dimensiones con ETL.

Creación de la dimensión DIM_TIENDA

La dimensión dim_tienda se crea a partir de los ficheros tienda.csv y país.csv:

Nombre	Dirección	Superficie	Ubicación	País	Tipo
Barcelona	Castellet, 119	+223.00	Centro comercial	España	XRTX
Florenca	Venize, 56	+275.00	Tienda de barrio	Italia	XRTS
Fort Lauderdale	23, Vyn Ness Street	+135.00	Centro comercial	Estados Unidos	BRTX

Ilustración 13. Tabla ilustrativa con algunos registros de tienda.csv

País	Extensión	Población	Región
Italia	+0000301230.	+0058384321.	Sur Europa
Estados Unidos	+0009372610.	+0266504935.	Norteamérica
España	+0000504750.	+0039510740.	Sur Europa

Ilustración 14. Tabla ilustrativa con algunos registros de pais.csv

De las tiendas se ha eliminado el tipo de tienda, mientras que del país se ha eliminado la región a la que pertenecen, ya que se trata de datos que carecen de interés para el estudio. Al juntar ambos orígenes de datos, se crea una estructura más sencilla para un esquema en estrella, pero también se crean valores duplicados, lo cual no es relevante debido a que hay un número bajo de sucursales. El proceso seguido se puede ver en el apartado correspondiente del [Anexo 4](#).

Creación de la dimensión DIM_CLIENTE

Para la creación de esta tabla, se ha utilizado solamente un origen de datos. Tras estudiar el archivo cliente.csv, podemos encontrar que los clientes que son empresas tienen los valores estadocivil y numerohijos sin datos:

Estado Civil	Dirección	Profesión	Hijos
Divorciado/a	68 Avenue de Villiers , 75015 Paris	Economistas, Abogados & Admin. Empresas	0
Casado/a	Senefelderstrasse 12, München	Gerentes & Directivos	3
	28 E 12th St New York, NY 10004	Catering	

Ilustración 15. Tabla ilustrativa con algunos registros de cliente.csv

Por lo tanto, se ha decidido incluir valores a estos atributos, ya que en algunos modelos de inteligencia de negocio puede crear problemas. Por ejemplo, podrían dar valores imprecisos al buscar clientes con hijos o errar en los cálculos de porcentajes. Para conseguirlo, se ha insertado una operación llamada *replace null value*, la cual ha permitido incluir un valor en dichos atributos para permitir que los informes puedan presentar resultados correctos. Los valores que se han introducido en los atributos son:

ATRIBUTO	VALOR INCLUIDO
ESTADOCIVIL	Sin Estado
NUMEROHIJOS	0

Ilustración 16. Solución valores vacíos

Se puede ver el proceso completo siguiendo el enlace al [Anexo 4](#).

Creación de la dimensión DIM_PRODUCTO

La creación de la dimensión *dim_producto* se ha realizado a partir de los datos de productos y de proveedores, ya que se pretende simplificar el modelo para obtener los datos de una manera más rápida y efectiva:

Algunos registros de producto.csv

Código	Descripción	País	Coste	Precio Venta	Tipo	Subfamilia	Marca	Proveedor
001CH	Cheshire	Reino Unido	+003.30	+004.69	Porción (250 g)	Vaca Curado	Smith Farmer	101
002CO	Cotswold	Reino Unido	+003.47	+004.98	Porción (250 g)	Vaca Curado	Old Hill	101
003CL	Crumbly Lancashire	Reino Unido	+003.93	+005.59	Porción (250 g)	Vaca Semicurado	The Farmhouse	101

Ilustración 17. Tabla ilustrativa de producto.csv

Algunos registros de proveedor.csv

Código	Nombre	Contacto	Dirección	Teléfono	Periodo Pago
101	The Farmhouse Company	Richard Addams	122 Brackbury Rd., London SW3	76589900	60
102	Cooperative de Produits Laitiers	Jean François Duval	34 Rue de Raspail , Brussels, Belgium	90090011	60
103	Compagnie Laitier	Florence Deveroix	96 Rue Vauvilliers , 75003 Paris, France	79034678	60

Ilustración 18. Tabla ilustrativa de proveedor.csv

Para crearla, se han cruzado los datos de ambos ficheros y se han eliminado los campos que no son necesarios, como el contacto o el teléfono del

proveedor, ya que no ofrecen información relevante para un estudio de BI. Sin embargo, se han dejado algunos datos que, aunque no serán utilizados a corto plazo, sí que pueden ofrecer resultados interesantes en futuros modelos de análisis. Por ejemplo, el coste y precio de venta, que pueden mostrar información de los márgenes de los diferentes productos. El proceso se puede ver con más detalle en el apartado que le corresponde del [Anexo 4](#).

Cargar las tablas de hechos a través del proceso ETL

Una vez cargados los datos de las tablas de dimensiones, ya se pueden cargar los datos de las tablas de hechos. Por tanto, se ha procedido a realizar las transformaciones necesarias para las dos tablas de hechos del almacén:

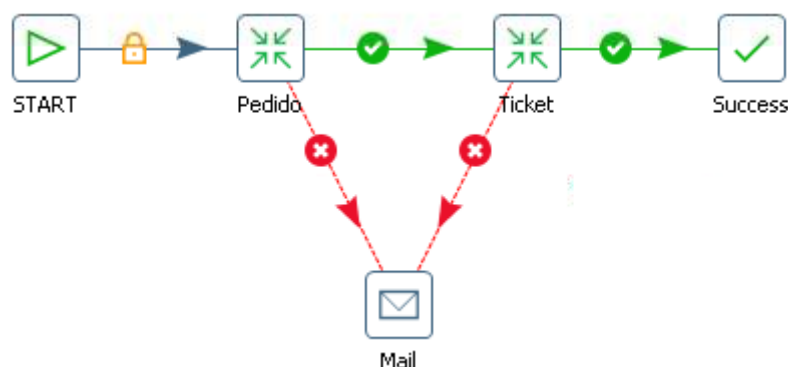


Ilustración 19. Cargar los datos de las tablas de hecho con ETL.

Creación de la tabla de hechos H_TICKET

La creación de la tabla de hechos h_ticket se realiza a partir del fichero cabeceraticket.csv:

Código	Tienda	Fecha Venta	Hora venta	Forma Pago	Cliente	Importe	Unidades	Puntos
LI0065538	Liverpool	20001012	13	Cheque		159.500	3	0
LI0065539	Liverpool	20001117	12	Tarjeta Débito		290.700	3	0
MI0065536	Milán	20001103	11	Efectivo	0000001R	37.250	1	0

Ilustración 20. Tabla ilustrativa de cabeceraticket.csv

Tras analizar los datos, se observa que hay muchos tickets sin código de cliente, seguramente por tratarse de clientes no habituales que no están registrados, por lo que se ha decidido rellenar estos campos con el código "00000000". De esta manera, se tendrá información de todas las ventas

realizadas a clientes no fidelizados. Además, será necesario crear un nuevo cliente en la dimensión `dim_cliente`, al cual llamaremos “indeterminado”, ya que no puede existir una clave foránea que apunte a un cliente inexistente. El detalle del proceso se puede seguir siguiendo el enlace al [Anexo 4](#).

Creación de la tabla de hechos H_PEDIDO

La creación de la tabla se realiza a partir del fichero `pedido.csv`, el cual viene con un formato correcto y sin campos vacíos. Por lo tanto, el proceso ETL se limita a asegurar que los tipos numéricos cuenten con la misma cantidad de decimales que el resto de tablas y que los atributos de fechas tengan el mismo formato que en las otras tablas. De ese modo, tras ordenar las filas resulta sencillo incorporar los datos en el almacén:

Código	Tienda	Producto	Precio Compra	Cantidad Solicitada	Fecha compra	Cantidad Entregada	Fecha Entrega
82207	Florenia	137TR	160.000	12	20000801	12	20000821
70595	Milán	219SY	182.800	12	20000801	12	20000821
93196	Miami Beach	335GB	93.100	12	20000801	12	20000821

Ilustración 21. Tabla ilustrativa de `pedido.csv`

El proceso detallado se puede ver en el apartado correspondiente del [Anexo 4](#).

3.2 Configurar la seguridad del servidor

Como medida preventiva en cualquier organización, resulta necesario dotar de seguridad el servidor, impidiendo que la información que contiene pueda ser utilizada incorrectamente, borrada o incluso robada. Por lo tanto, para evitar poner en riesgo la información, en la consola de Pentaho se pueden crear nuevos usuarios y roles, lo que permitirá:

- Salvaguardar la confidencialidad de la información que ofrecerán los diferentes *reports* y otros modelos de datos.
- Proteger la integridad de los datos y de la configuración del Servidor de Pentaho, ya sea en carpetas o archivos del mismo.

A cada uno de los roles, se le pueden asignar diferentes operaciones. En concreto, las operaciones permitidas en cada rol son:

Operación	Definición
Administer security	<ul style="list-style-type: none"> • Da acceso a la consola de administración de Pentaho (Perspectiva <i>Administration</i>). • Permite acceder y administrar las carpetas y ficheros (Perspectiva <i>Browse</i>). • Permite ver y trabajar con las programaciones de horarios de los usuarios (Perspectiva <i>Schedule</i>). • Ofrece la posibilidad de crear horarios de bloqueo del servidor (Perspectiva <i>Schedule</i>).
Schedule Content	<ul style="list-style-type: none"> • Permite al usuario programar los horarios de los <i>reports</i> y su contenido. • Permite ver, editar o borrar sus programaciones de la perspectiva <i>Schedule</i>.
Read Content	<ul style="list-style-type: none"> • Permite al usuario ver el contenido de carpetas y archivos. • Permite al usuario ver el contenido a través del menú <i>File>Open</i>.
Publish Content	<ul style="list-style-type: none"> • Incluye herramientas como <i>Report Designer</i> y <i>Schema Workbench</i>. • Permite que las herramientas almacenen informes o modelos de datos en el repositorio de Pentaho.
Create Content	<ul style="list-style-type: none"> • Permite al usuario crear, importar, borrar y guardar reports en el repositorio de Pentaho. • Ofrece al usuario la posibilidad de ver una lista de orígenes de datos para poder crear <i>reports</i> o <i>dashboards</i>.
Execute	<ul style="list-style-type: none"> • Habilita los botones <i>Run</i>, <i>Preview</i>, <i>Debug</i>, <i>Reply</i> y <i>Verify</i>. Además de las entradas en <i>Spoon</i>, <i>Kitchen</i>, <i>Pan</i> y <i>Carte</i>. • Permite al usuario guardar, copiar o ejecutar transformaciones y trabajos. • Ofrece al usuario la posibilidad de usar los botones de exportar y las entradas asociadas del menú.
Manage Data Sources	<ul style="list-style-type: none"> • Permite al usuario crear, editar o borrar nuevos orígenes de datos. • Ofrece la posibilidad de ver una lista de orígenes de datos para crear <i>reports</i> o <i>dashboards</i>.

Es más, se aprovecharán los cuatro roles que ya están creados en Pentaho, aunque modificados para adaptarlos a la cadena de tiendas Gourmet:

Rol	Descripción	Operaciones permitidas	Grupo de Trabajadores
Administrator	En este grupo se incluirá a todos aquellos profesionales de TI que tengan que administrar el servidor	<ul style="list-style-type: none"> • Administer security • Schedule Content • Read Content • Publish Content • Create Content • Execute • Manage Data Sources 	<ul style="list-style-type: none"> • Administrador del servidor de Pentaho
Business Analyst	Grupo al que pertenecerán todos los directivos o jefes intermedios que necesiten tomar decisiones con los resultados de los modelos	<ul style="list-style-type: none"> • Read Content • Publish Content 	<ul style="list-style-type: none"> • Directivos del grupo Gourmet • Jefes de tienda

Power user	Grupo al que pertenecen los usuarios más avanzados, con un perfil cercano al del departamento de Tecnologías de la Información	<ul style="list-style-type: none"> • Schedule Content • Read Content • Publish Content • Create Content • Execute 	Engloba a diferentes usuarios de la cadena
Report Author	Se trata del grupo de trabajadores con conocimientos tanto del negocio como de BI, que pueden ofrecer un mayor valor a la organización a través de modelos de conocimiento	<ul style="list-style-type: none"> • Schedule Content • Publish Content 	Engloba a diferentes usuarios de la cadena. Normalmente directivos y jefes intermedios con conocimientos de BI

Por ejemplo, el rol de *Business Analyst* quedaría como sigue:

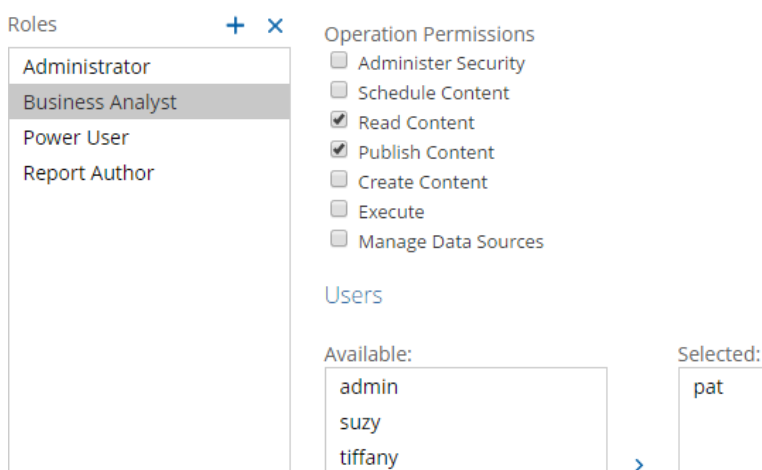


Ilustración 22. Permisos del rol *Business Analyst* en Pentaho

Por otro lado, habría que crear las carpetas necesarias donde incluir los *reports* y otros modelos de datos, compartiéndolos únicamente con los usuarios que puedan acceder al contenido. Por ese motivo, se ha creado una carpeta para almacenar los modelos que afectan a toda la compañía, que será utilizado por los directivos de Gourmet, mientras que para cada una de las tiendas se creará una carpeta específica, a la cual también tendrán acceso los responsables de cada tienda. Algunas de las carpetas que podremos encontrar son:

Carpeta	Descripción
Gerencia	Carpeta a la que solo tienen acceso los directivos de Gourmet. En esta carpeta se almacenarán todos los modelos de inteligencia de negocio que sirvan para tomar decisiones que afecten a toda la compañía.
Barcelona	Carpeta a la que tienen acceso los directivos de Gourmet y los responsables de la tienda de Barcelona.

Florencia	Carpeta a la que tienen acceso los directivos de Gourmet y los responsables de la tienda de Florencia.
Liverpool	Carpeta a la que tienen acceso los directivos de Gourmet y los responsables de la tienda de Liverpool.
Londres I	Carpeta a la que tienen acceso los directivos de Gourmet y los responsables de la tienda de Londres I.
Londres II	Carpeta a la que tienen acceso los directivos de Gourmet y los responsables de la tienda de Londres II.

Además, en cada carpeta se asignarán permisos a los usuarios que necesiten tener acceso a los informes que se guarden en ella, de forma que solo las personas autorizadas puedan ver el contenido de las mismas:

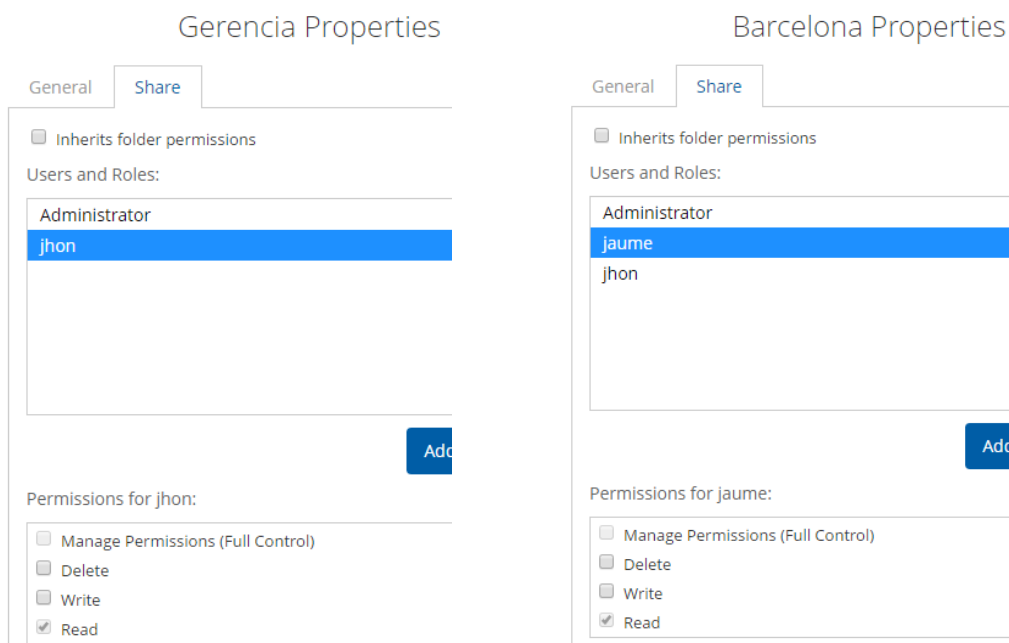


Ilustración 23. Permisos de las carpetas Gerencia y Barcelona

Por ejemplo, si se crea un informe y se guarda en la carpeta Gerencia, se puede comprobar que solamente Jhon tiene acceso a dicha carpeta y a los informes que hay en ella. De esta manera, al iniciar sesión con el usuario Jhon, que es el gerente de la organización, se aprecia que puede acceder tanto a la carpeta Gerencia como al resto de carpetas de las sucursales. Sin embargo, no puede acceder a las carpetas personales de los usuarios. Por el contrario, al iniciar sesión con el usuario Jaume, que es el responsable de la sucursal en Barcelona, el usuario solamente tiene acceso a la carpeta pública, a su carpeta personal y a la carpeta de Barcelona.

3.3 Creación de informes

En la cadena de tiendas gourmet se han fijado como objetivo mejorar el proceso de ventas de las diferentes sucursales, con la finalidad de obtener más ganancias⁷. Las buenas prácticas dictan que para dar valor a un negocio, los objetivos del departamento de tecnologías de la información deben estar alineados con los objetivos de la organización, por lo que ya se ha creado un *data warehouse* enfocado a dicho objetivo. Por lo tanto, es el momento de crear informes que permitan cumplir los objetivos dictados por la organización.

3.3.1 Estudio de clientes por su profesión⁸

Con la finalidad de mejorar las ventas, la dirección de la empresa ha pensado en ofertar productos personalizados, enviar publicidad y ofrecer los productos gourmet visitando diferentes centros de trabajo profesionales. Con la finalidad de crear un informe que pueda ayudar a decidir qué centros de trabajo pueden ser más rentables, se ha decidido crear un *report* con la información de las ventas realizadas a los clientes. Pero, estarán agrupados según su profesión, ya que es la característica de los clientes que más se acerca a las necesidades de la campaña. Así, a través del informe se pretende conocer tanto la profesión de los clientes que compran en las tiendas como qué profesiones gastan más dinero en las tiendas. Por lo tanto, se trata de un informe dirigido a los supervisores de tiendas y al equipo directivo, ya que muestra información sobre las ventas realizadas a los clientes agrupadas por profesión.

Diseño del informe

En la creación del informe se han utilizado las tablas *h_ticket* y *dim_cliente*, ya que son las tablas que contienen la información con la que se puede responder a estas preguntas. Primero, se ha creado un listado con la fecha de la compra, el cliente que la realiza, las unidades compradas, los puntos obtenidos en la compra y el importe gastado, pues con estos atributos se podrá estudiar a los clientes con diferentes métricas. Posteriormente, para facilitar la comprensión del informe, se han incluido en el pie dos gráficos con el importe vendido a

cada profesión y con los puntos obtenidos por los mismos, de manera que se pueda decidir qué profesiones son más rentables o más propensas a fidelizar:

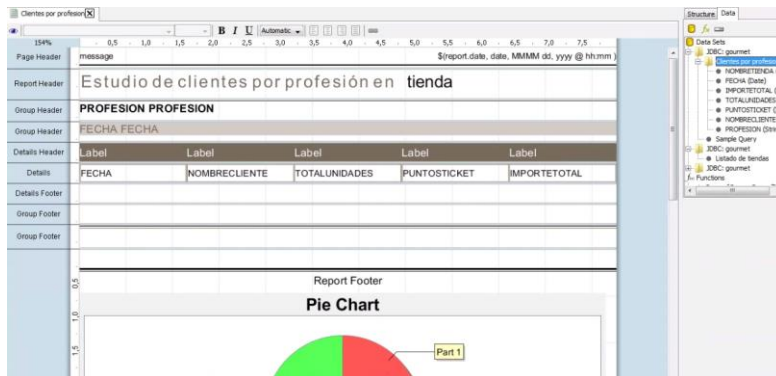
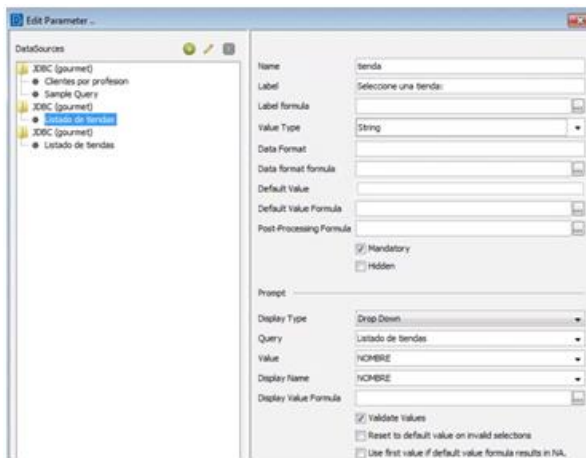


Ilustración 24. Diseño del *report* Estudio de clientes por profesión

En cuanto a los parámetros utilizados se ha decidido incluir únicamente la sucursal, ya que se pretende conocer los detalles de una sola tienda en cada estudio. Además, lo más probable es que los colectivos profesionales que comprenden en cada sucursal sean muy diferentes:



```
1 SELECT NOMBRE
2 FROM gourmet.dbo.DIM_TIENDA
```

Ilustración 25. Parámetro tienda

Finalmente la consulta del *report* queda como se muestra a continuación:

```
SELECT
    "dbo"."H_TICKET"."NOMBRETIENDA",
    "dbo"."H_TICKET"."FECHA",
    "dbo"."H_TICKET"."IMPORTETOTAL",
    "dbo"."H_TICKET"."TOTALUNIDADES",
    "dbo"."H_TICKET"."PUNTOSTICKET",
    "dbo"."DIM_CLIENTE"."NOMBRECLIENTE",
    "dbo"."DIM_CLIENTE"."PROFESION"
```

```

FROM
    "dbo"."DIM_CLIENTE" INNER JOIN "dbo"."H_TICKET"
    ON "dbo"."DIM_CLIENTE"."CODCLIENTE" =
        "dbo"."H_TICKET"."CODCLIENTE"
WHERE
    "dbo"."H_TICKET"."NOMBRETIENDA" = ${tienda}
GROUP BY
    "dbo"."H_TICKET"."NOMBRETIENDA",
    "dbo"."H_TICKET"."FECHA",
    "dbo"."H_TICKET"."IMPORTETOTAL",
    "dbo"."H_TICKET"."TOTALUNIDADES",
    "dbo"."H_TICKET"."PUNTOSTICKET",
    "dbo"."DIM_CLIENTE"."NOMBRECLIENTE",
    "dbo"."DIM_CLIENTE"."PROFESION"
ORDER BY
    "dbo"."DIM_CLIENTE"."PROFESION"

```

La ejecución del informe solicitará la tienda que se quiere consultar, mostrando a continuación un listado de todas las compras realizadas por los clientes en la tienda seleccionada. Además, el listado estará agrupado por profesión y ordenado por la fecha de compra:

Estudio de clientes por profesión en Barcelona				
PROFESION Alimentación				
FECHA 01-sep-2000				
FECHA	NOMBRECLIENTE	TOTALUNIDADES	PUNTOSTICKET	IMPORTETOTAL
01-sep-2000	Delight Restaurant	2	0	18
01-sep-2000	Mexican Pizzeria	3	0	18
01-sep-2000	Valletto House	1	0	19
01-sep-2000	Tamura Steakhouse	2	0	26
01-sep-2000	Noodle Steakhouse	3	0	36

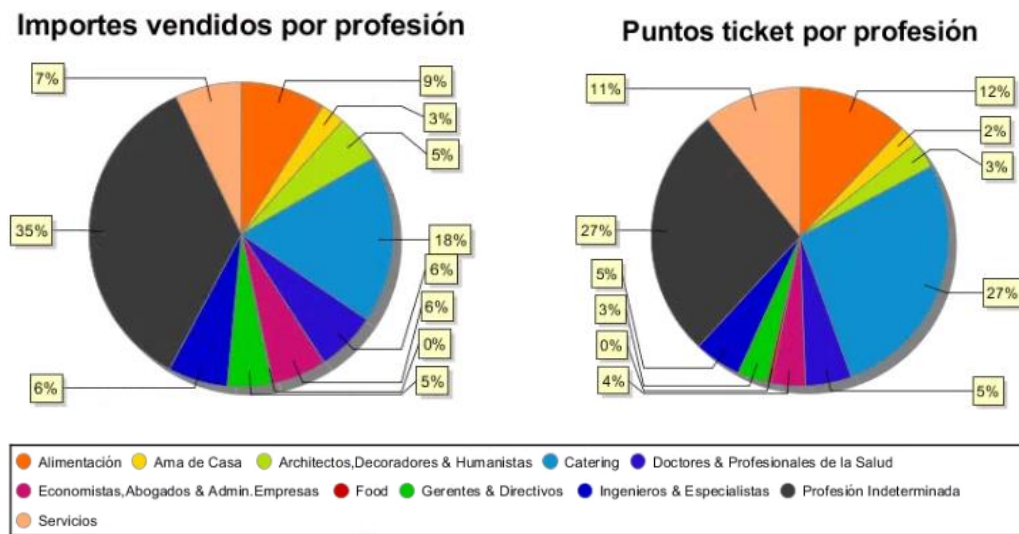
Ilustración 26. Clientes con profesión Alimentación

PROFESION Ama de Casa				
FECHA 02-sep-2000				
FECHA	NOMBRECLIENTE	TOTALUNIDADES	PUNTOSTICKET	IMPORTETOTAL
02-sep-2000	Serralta Rodrigo Carlos	1	0	7
02-sep-2000	Rovira Soriano Inés	1	0	11
02-sep-2000	Callado Calpe Victoria	2	1	55
FECHA 04-sep-2000				
FECHA	NOMBRECLIENTE	TOTALUNIDADES	PUNTOSTICKET	IMPORTETOTAL
04-sep-2000	Diez Durán Patricia	2	0	9
FECHA 06-sep-2000				
FECHA	NOMBRECLIENTE	TOTALUNIDADES	PUNTOSTICKET	IMPORTETOTAL
06-sep-2000	Boada Larrea Lucía	1	0	11
06-sep-2000	Piqueras Nieto Itziar	4	0	27
FECHA 08-sep-2000				
FECHA	NOMBRECLIENTE	TOTALUNIDADES	PUNTOSTICKET	IMPORTETOTAL
08-sep-2000	Diez Durán Patricia	1	0	5
08-sep-2000	Boada Larrea Lucía	4	1	76

Ilustración 27. Clientes con profesión ama de casa

PROFESION Economistas, Abogados & Admin. Empresas

FECHA 01-sep-2000				
FECHA	NOMBRECLIENTE	TOTALUNIDADES	PUNTOSTICKET	IMPORTETOTAL
01-sep-2000	Marqués Acereda Juana	1	0	4
01-sep-2000	Almirall Castrillo Mauricio	2	0	28
FECHA 02-sep-2000				
FECHA	NOMBRECLIENTE	TOTALUNIDADES	PUNTOSTICKET	IMPORTETOTAL
02-sep-2000	Marqués Acereda Maria del Pilar	1	0	7
02-sep-2000	Puertas Alquézar Inés	5	0	25
02-sep-2000	Ferrero Bautista Alejandra	3	0	30
FECHA 04-sep-2000				
FECHA	NOMBRECLIENTE	TOTALUNIDADES	PUNTOSTICKET	IMPORTETOTAL
04-sep-2000	Puertas Alquézar Jaime	1	0	6
04-sep-2000	Padilla Valverde Meritxell	2	0	14

Ilustración 28. Clientes con profesión Economistas, Abogados & Admin. Empresas**Ilustración 29. Gráficos con los porcentajes de Importes vendidos y Puntos del ticket**

Este informe permite hacer un seguimiento de las profesiones que más veces acuden a la tienda a comprar y cómo se reparte el porcentaje de ventas entre las distintas profesiones.

Por un lado, se observa que el mayor porcentaje está en los clientes de los que no se tienen datos, por lo que se supone que son clientes poco habituales. No obstante, para mejorar la información de los informes de BI sería recomendable tener una ficha de estos clientes, por lo que se puede recomendar fidelizarlos con algún tipo de tarjeta de cliente. Por otro lado, en cuanto a los clientes de los que sí que se tienen datos, se aprecia que los servicios de catering son los que más compran, seguidos por el sector Alimentación, Doctores & Profesionales de la Salud, Ingenieros & Especialistas & Economistas y Abogados & Admin. Empresas. Por lo tanto, estos son los sectores

profesionales a los que habría que darles un trato especializado por medio de visitas.

En lo que respecta al gráfico de puntos obtenidos con el ticket, se observa que los clientes del sector servicios son el que más puntos de compra obtienen, por lo que se entiende que les resulta interesante acumular puntos. Así pues, se les podría recomendar completar la compra con algún producto que les permita obtener puntos extra.

Además, en el listado se observa que muchos clientes acuden a la tienda y vuelven al día siguiente a por algún producto, probablemente porque se les ha olvidado comprarlo. Por esta razón, sería conveniente ofrecer productos a los clientes cuando han realizado la compra. Para eso, podría utilizarse de un modelo de datos de *data mining*, aunque se trata de un estudio que queda fuera de este trabajo.

3.3.2 Estudio del proceso de ventas con ticket promedio y canasta promedio⁹

La cadena de tiendas Gourmet pretende conocer mejor el proceso de ventas de cada una de las sucursales. Igualmente, saben que con la información que contienen los tickets se pueden dar las indicaciones pertinentes a cada una de sus sucursales para mejorar las ventas.

Una de las mejores formas de conocer un proceso es midiéndolo y, en este caso, se trata de medir las ventas. Una forma de hacerlo es por medio de los KPI (*Key Performance Indicator*), que pueden medir los resultados de las ventas del negocio, por lo que con este informe se pretende utilizar dos indicadores: el ticket promedio y la canasta promedio.

Por un lado, el ticket promedio es un KPI que permite conocer cuánto gastan los clientes de promedio en cada venta. Por otro lado, la canasta promedio es un KPI que mide cuántos productos está comprando el cliente cada vez que visita la tienda. Así, con este informe se podrá comprobar si con cada venta realizada se ha alcanzado tanto el promedio del importe como el promedio de

artículos de cada venta. La finalidad es que en cada venta, el dependiente intente que el cliente alcance tanto el ticket como la canasta promedio.

Se trata de un informe que va dirigido tanto a los responsables de las sucursales como a los directivos de la cadena, ya que se trata de un informe que permitirá, no solamente comprobar el ticket promedio y la canasta promedio de la sucursal, sino que también podrá compararlo con un periodo de ventas de la misma tienda. Además, con los permisos adecuados se podrá seleccionar otra sucursal, con la finalidad de que se pueda comparar con los datos con los de otras tiendas. Esta comparación, permitirá saber si la tienda está funcionando mejor o peor que otras y, a la vez, fomentará la rivalidad entre sucursales con datos parecidos, lo que permitirá aumentar las ganancias.

Diseño del informe

Para el diseño del informe se ha utilizado únicamente la tabla h_ticket, pues todos los datos necesarios están en dicha tabla. En especial, contiene las métricas con el importe y el número de artículos

Con la finalidad de personalizar el informe de una manera más eficiente, al ejecutarlo se solicita la tienda a consultar y las fechas en las que se quiere comprobar si se alcanza dicho objetivo, por lo que se han utilizado tres parámetros en el informe:

Ilustración 30. Parámetros del informe de ticket y canasta promedio

Una decisión que se ha tenido que tomar es de qué fechas debía seleccionarse el promedio objetivo. Lo ideal, sería poder comparar con los datos del año anterior para poder superarlos, pero como no se cuenta con esos datos, se ha

decidido calcularlo con los datos acumulados de todo el año, pues corresponderá con un promedio que se puede intentar superar. Además, se ha indicado también el porcentaje de ventas que no han alcanzado el objetivo en el periodo indicado, lo que facilitará la comprensión del estudio que se está realizando:

Importe y nº de artículos en cada venta

Ticket promedio objetivo en cada venta: 44	Fuera del objetivo: 67.61%
Canasta promedio objetivo en cada venta: 3	Fuera del objetivo: 58.61%

NOMBRE TIENDA Munich

Ilustración 31. Cabecera del informe del ticket y canasta promedio

La consulta SQL se ha realizado con diferentes variables, ya que cuenta con muchas columnas. De esta manera, se facilita la comprensión del código en caso de que en un futuro se deba modificar:

```

DECLARE @NUM_VENTAS INT;
DECLARE @IMP_VENTAS INT;
DECLARE @NUM_ARTICULOS INT;
DECLARE @TICKETPROMEDIO INT;
DECLARE @CANASTAPROMEDIO INT;
DECLARE @NUM_VENTAS_PERIODO INT;
DECLARE @NUM_VENTAS_PERIODO_BAJO_TICKET FLOAT;
DECLARE @NUM_VENTAS_PERIODO_BAJO_CANASTA FLOAT;

SET @NUM_VENTAS = (SELECT COUNT(*) FROM H_TICKET WHERE
NOMBRETIENDA = ${tienda});

SET @IMP_VENTAS = (SELECT SUM(IMPORTETOTAL) FROM H_TICKET WHERE
NOMBRETIENDA = ${tienda});

SET @NUM_ARTICULOS = (SELECT SUM(TOTALUNIDADES) FROM H_TICKET
WHERE NOMBRETIENDA = ${tienda});

SET @TICKETPROMEDIO = (SELECT @IMP_VENTAS / @NUM_VENTAS);

SET @CANASTAPROMEDIO = (SELECT @NUM_ARTICULOS / @NUM_VENTAS);

SET @NUM_VENTAS_PERIODO = (SELECT COUNT(*) FROM H_TICKET WHERE
NOMBRETIENDA = ${tienda} AND "dbo"."H_TICKET"."FECHA" BETWEEN
${fechaInicio} AND ${fechaFin});

SET @NUM_VENTAS_PERIODO_BAJO_TICKET = (SELECT COUNT(*) FROM
H_TICKET WHERE NOMBRETIENDA = ${tienda} AND
"dbo"."H_TICKET"."FECHA" BETWEEN ${fechaInicio} AND ${fechaFin}
AND IMPORTETOTAL < @TICKETPROMEDIO);

```

```

SET @NUM_VENTAS_PERIODO_BAJO_CANASTA = (SELECT COUNT(*) FROM
H_TICKET WHERE NOMBRETIENDA = ${tienda} AND
"dbo"."H_TICKET"."FECHA" BETWEEN ${fechaInicio} AND ${fechaFin}
AND TOTALUNIDADES < @CANASTAPROMEDIO);

SELECT
    "dbo"."H_TICKET"."CODVENTA",
    "dbo"."H_TICKET"."NOMBRETIENDA",
    "dbo"."H_TICKET"."FECHA",
    "dbo"."H_TICKET"."HORA",
    "dbo"."H_TICKET"."IMPORTETOTAL",
    "dbo"."H_TICKET"."TOTALUNIDADES",
    @TICKETPROMEDIO AS TICKETPROMEDIO,
    @CANASTAPROMEDIO AS CANASTAPROMEDIO,
    (@NUM_VENTAS_PERIODO_BAJO_TICKET/@NUM_VENTAS_PERIODO) * 100
    AS PORCENTAJEPORDEBÁJOTICKET,
    (@NUM_VENTAS_PERIODO_BAJO_CANASTA/@NUM_VENTAS_PERIODO) *100
    AS PORCENTAJEPORDEBÁJOCANASTA
FROM
    "dbo"."H_TICKET"
WHERE
    "dbo"."H_TICKET"."NOMBRETIENDA" = ${tienda} AND
    "dbo"."H_TICKET"."FECHA" BETWEEN
        ${fechaInicio} AND ${fechaFin}
ORDER BY
    "dbo"."H_TICKET"."FECHA",
    "dbo"."H_TICKET"."HORA"

```

Tras la ejecución del informe, éste devolverá un listado de las ventas realizadas en la tienda seleccionada, de manera que pueda comprobarse si se ha alcanzado el objetivo de ticket y canasta promedio en el periodo solicitado:

Importe y nº de artículos en cada venta

Ticket promedio objetivo en cada venta: 54 Fuera del objetivo: 70.49%
 Canasta promedio objetivo en cada venta: 4 Fuera del objetivo: 64.03%

NOMBRE TIENDA Londres I				
CODVENTA	FECHA	HORA	IMPORTETOTAL	TOTALUNIDADES
L11048584	18-dic-2000	10	109	6
L13276904	18-dic-2000	10	108	12
L14784150	18-dic-2000	11	419	18
L15373995	18-dic-2000	12	27	2
L15636118	18-dic-2000	12	88	10

Ilustración 32. Informe en Londres I durante el periodo navideño

En la ilustración se observa que el ticket promedio de la tienda de Londres I es de 54€, mientras que la canasta promedio es de 4 artículos. Además, se puede apreciar que un 70,49% de las ventas quedan por debajo del ticket promedio y que el 64,03% por debajo de la canasta promedio. Por lo tanto, en la tienda de Londres I se aconsejará ofrecer más productos, con el objetivo de alcanzar

tanto un importe mayor como más cantidad de artículos vendidos. A ser posible, los productos ofrecidos estarán relacionados con la venta realizada.

Sin embargo, al observar el resultado del mismo informe en Londres II:

Importe y nº de artículos en cada venta

Ticket promedio objetivo en cada venta: 35 Fuera del objetivo: 66.58%

Canasta promedio objetivo en cada venta: 2 Fuera del objetivo: 34.36%

NOMBRE TIENDA Londres II				
CODVENTA	FECHA	HORA	IMPORTE TOTAL	TOTAL UNIDADES
L22490430	18-dic-2000	10	57	3
L20852042	18-dic-2000	12	30	4
L21376288	18-dic-2000	13	92	7
L21441887	18-dic-2000	13	113	12
L20786495	18-dic-2000	13	45	4

Ilustración 33. Informe en Londres II durante el periodo navideño

Se observa que el ticket promedio de la tienda es más bajo que en Londres I, pasando de 54€ a 35€. Además, la canasta promedio también es más baja, ya que se reduce en 2 unidades por venta. Por otro lado, el objetivo del ticket promedio es algo mejor que en Londres I, aunque no tan bueno como se desea. Pero, el objetivo del ticket promedio está claramente superado, ya que solamente un 34,36% de las ventas no alcanzan dicho objetivo. Por lo tanto, la recomendación para esta tienda será ofrecer productos con un importe más elevado con la finalidad de mejorar el porcentaje del ticket promedio objetivo.

Cabe mencionar que se puede realizar un estudio de *data mining* para ofrecer productos personalizados a los clientes de cada venta, aunque se trata de un estudio que queda fuera del presente trabajo.

3.3.3 Seguimiento de la campaña de puntos

En tiendas Gourmet se puso en marcha hace unos meses una campaña en la que los clientes obtenían puntos por las compras, por lo que tanto la dirección como los responsables de las sucursales quieren ver cómo están respondiendo los clientes a dicha campaña. Observando detenidamente el resultado del informe de cada tienda, se podrá ver si se están fidelizando nuevos clientes y si los clientes están acudiendo a la tienda varias veces al mes.

Diseño del informe

A la hora de diseñar el informe se ha utilizado la tabla de cabecera h_ticket y las tablas de dimensiones dim_fecha y dim_cliente. Gracias a la tabla de cabecera se pueden sacar los datos de la venta y utilizar la métrica puntosticket, ya que es la que se necesita para el grueso del informe. En cuanto a las tablas de dimensiones, éstas únicamente se utilizarán para extraer el nombre del cliente y el nombre del mes que se quiere consultar.

Para personalizar la consulta, se han utilizado dos parámetros que ayudarán a filtrar los resultados. Se trata de la tienda sobre la que se realizará la consulta y el mes en el que se quiere comprobar cómo está funcionando la campaña. De esta manera, se podrán realizar comparativas de diferentes meses para ver la evolución de la misma:

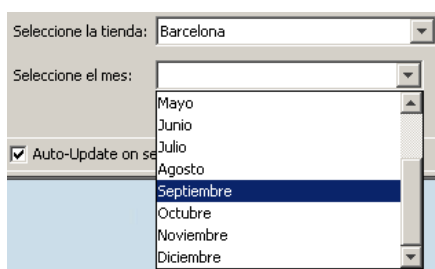


Ilustración 34. Parámetros del informe de la campaña de puntos

El parámetro mes se obtiene de una consulta en la que se utilizan las columnas Mes y NMes. La primera, permite ordenar los meses, mientras que la segunda es el valor que se muestra en el parámetro, es decir, el nombre del mes:

```

1 select distinct Mes, NMes
2 from Dim_Fecha
3 order by mes

```

Ilustración 35. Parámetro mes

Para la ejecución de la consulta se ha incluido un alias en cada una de las columnas que incluyen un código más largo del habitual, pues ayudará a identificar el valor al que representa. Finalmente, la sentencia SQL utilizada ha sido la siguiente:

```

SELECT
    "dbo"."H_TICKET"."nombretienda",
    "dbo"."dim_fecha"."nmes",
    "dbo"."H_TICKET"."CODVENTA",
    "dbo"."H_TICKET"."FECHA",
    "dbo"."H_TICKET"."codcliente",
    "dbo"."dim_cliente"."nombrecliente",
    "dbo"."H_TICKET"."PUNTOSTICKET",

    (
    select count("dbo"."H_TICKET"."codventa")
    from "dbo"."H_TICKET" inner join "dbo"."dim_fecha"
    on "dbo"."H_TICKET"."fecha" = "dbo"."DIM_FECHA"."FECHA"
    where PUNTOSTICKET <> '0'
    and "dbo"."H_TICKET"."nombretienda" = ${tienda}
    and "dbo"."DIM_FECHA"."NMES" = ${mes}
    ) as VentasConPuntos,

    (
    select count("dbo"."H_TICKET"."codventa")
    from "dbo"."H_TICKET" inner join "dbo"."dim_fecha"
    on "dbo"."H_TICKET"."fecha" = "dbo"."DIM_FECHA"."FECHA"
    where "dbo"."H_TICKET"."nombretienda" = ${tienda}
    and "dbo"."DIM_FECHA"."NMES" = ${mes}
    ) as VentasTienda,

    (
    select count("dbo"."H_TICKET"."codventa")
    from "dbo"."H_TICKET" inner join "dbo"."dim_fecha"
    on "dbo"."H_TICKET"."fecha" = "dbo"."DIM_FECHA"."FECHA"
    where PUNTOSTICKET <> '0'
    and "dbo"."H_TICKET"."nombretienda" = ${tienda}
    and "dbo"."DIM_FECHA"."NMES" = ${mes}
    and "dbo"."H_TICKET"."CODCLIENTE" = '00000000'
    ) as VentasPosiblesNuevosClientes,

    (
    select count("dbo"."H_TICKET"."codventa")
    from "dbo"."H_TICKET" inner join "dbo"."dim_fecha"
    on "dbo"."H_TICKET"."fecha" = "dbo"."DIM_FECHA"."FECHA"
    where
    "dbo"."H_TICKET"."nombretienda" = ${tienda}
    and "dbo"."DIM_FECHA"."NMES" = ${mes}
    and "dbo"."H_TICKET"."CODCLIENTE" = '00000000'
    ) as VentasNuevosClientesNoFidelizados

FROM "dbo"."H_TICKET" inner join "dbo"."dim_fecha"
on "dbo"."H_TICKET"."fecha" = "dbo"."DIM_FECHA"."FECHA"
inner join "dbo"."dim_cliente"
on "dbo"."dim_cliente"."codcliente" =
"dbo"."H_TICKET"."codcliente"

WHERE
    PUNTOSTICKET <> '0'
    and nombretienda = ${tienda}
    and "dbo"."DIM_FECHA"."NMES" = ${mes}

```



```

GROUP BY
    "dbo"."H_TICKET"."nombretienda",
    "dbo"."dim_fecha"."nmes",
    "dbo"."H_TICKET"."CODVENTA",
    "dbo"."H_TICKET"."FECHA",
    "dbo"."H_TICKET"."codcliente",
    "dbo"."dim_cliente"."nombrecliente",
    "dbo"."H_TICKET"."PUNTOSTICKET"

ORDER BY
    "dbo"."H_TICKET"."nombretienda",
    "dbo"."dim_fecha"."nmes",
    "dbo"."dim_cliente"."nombrecliente",
    "dbo"."H_TICKET"."FECHA"

```

Así, tras la ejecución del *report* se obtendrá un informe en el que el encabezado ya ofrece información de interés:

Seguimiento campaña de puntos: Barcelona - Septiembre
 Total ventas tienda: 1202 Ventas que han obtenido puntos: 160
 Clientes nuevos que han comprado: 575
 Clientes nuevos que se han interesado en los puntos: 43 (Se podrían fidelizar)

Ilustración 36. Cabecera de la campaña de puntos de Barcelona en Septiembre

En la cabecera se identifica la tienda y el mes seleccionado para el informe, indicando además que en dicho mes se han realizado un total de 1202 ventas, de las cuales, han obtenido puntos 160 ventas. Por lo tanto, se percibe un cierto interés por parte de los clientes, ya que se trata del primer mes de campaña.

Siguiendo con la cabecera, se muestra que existes 575 clientes de los que no se tiene ficha de cliente, pero que 43 de ellos han mostrado interés en los puntos. Es decir, si en la próxima visita el cliente utiliza los puntos, se le haría una ficha de cliente, por lo que se podría fidelizar a un total de 43 clientes.

En cuanto al listado, aparece una columna con las fechas en las que los clientes han obtenido puntos con una compra. Además, van ordenadas por el nombre del cliente, por lo que puede apreciarse cuántas veces ha acudido cada cliente a la tienda en el mes seleccionado. Por último, aparecerán los puntos que el cliente ha obtenido en cada compra, lo que reflejará si cada venta está creciendo gracias a la campaña.

nombretienda: Barcelona

mes: Septiembre

FECHA	nombrecliente	PUNTOSTICKET
09-sep-2000	Alvarez Arenas Mauricio	1
01-sep-2000	Aparicio Arqués Azucena	1
21-sep-2000	Aparicio Arqués Azucena	1
16-sep-2000	Asensio Laseca Claudio	1
23-sep-2000	Ayana	2
08-sep-2000	Boada Larrea Lucía	1
23-sep-2000	Boada Larrea Lucía	1
08-sep-2000	Bosch Pérez Eduard	1
02-sep-2000	Burladero	2
08-sep-2000	Burladero	2
08-sep-2000	Burladero	2
13-sep-2000	Burladero	1
16-sep-2000	Burladero	2
20-sep-2000	Burladero	1
23-sep-2000	Burladero	1
23-sep-2000	Callado Calpe Ignacio	1
02-sep-2000	Callado Calpe Victoria	1

Ilustración 37. Informe de la campaña de puntos de Barcelona en Septiembre

En este informe se aprecia que existen varios clientes que vienen más de una vez al mes a la sucursal, incluso alguna empresa especializada que acude todas las semanas, por lo que obtendrán un buen número de puntos y acudirán a tiendas Gourmet con más asiduidad. De hecho, al mirar el mes de diciembre:

nombretienda: Barcelona

mes: Diciembre

FECHA	nombrecliente	PUNTOSTICKET
23-dic-2000	Acedo Gómez Rosa	1
23-dic-2000	Almirall Castrillo Joan	1
23-dic-2000	Alvarez Arenas Mauricio	1
16-dic-2000	Aparicio Arqués Azucena	1
23-dic-2000	Asensio Laseca Agustín	1
02-dic-2000	Asensio Laseca MiguelÁngel	1
23-dic-2000	Austin	1
08-dic-2000	Bagatto Casa	1
31-dic-2000	Blasco Heredia Alberto	1
08-dic-2000	Bosch Pérez Eduard	1
05-dic-2000	Bosch Pérez Narciso	1
23-dic-2000	Bosch Pérez Narciso	1
01-dic-2000	Burladero	2
05-dic-2000	Burladero	1
15-dic-2000	Burladero	1
16-dic-2000	Burladero	1
21-dic-2000	Burladero	1
23-dic-2000	Burladero	1
23-dic-2000	Burladero	1
23-dic-2000	Burladero	4
24-dic-2000	Burladero	1
31-dic-2000	Burladero	1

Ilustración 38. Informe de la campaña de puntos de Barcelona en Diciembre

Se puede apreciar que la empresa Burladero acude con más frecuencia desde que está la campaña de puntos, por lo que se aprecia que al menos algunos clientes están respondiendo positivamente.

Sin embargo, si no fijamos en los datos de la cabecera de Manhattan II:

Seguimiento campaña de puntos: Manhattan II - Octubre

Total ventas tienda: 470 Ventas que han obtenido puntos: 74
 Clientes nuevos que han comprado: 275
 Clientes nuevos que se han interesado en los puntos: 37 (Se podrían fidelizar)

Ilustración 39. Cabecera de la campaña de puntos de Manhattan II en octubre

Se observa que de un total de 470 ventas, al menos 275 son de nuevos clientes. Es decir, más de la mitad de los clientes de Manhattan II son temporales (no tienen ficha de cliente). Como se verá en el siguiente *report*, Manhattan II tiene una facturación mucho más pequeña que la que tiene Manhattan I. Además, es una tienda de barrio, por lo que no se trata de una sucursal a la que vayan muchos turistas de Manhattan. Por consiguiente, habrá que comprobar cómo evoluciona la situación en meses posteriores:

Seguimiento campaña de puntos: Manhattan II - Diciembre

Total ventas tienda: 597 Ventas que han obtenido puntos: 110
 Clientes nuevos que han comprado: 307
 Clientes nuevos que se han interesado en los puntos: 43 (Se podrían fidelizar)

Ilustración 40. Cabecera de la campaña de puntos de Manhattan II en diciembre

En esta imagen, se puede apreciar que la fidelidad de los clientes ha mejorado en el mes de diciembre. Por lo tanto, sería recomendable realizar un seguimiento más detallado de la campaña en Manhattan II.

Para mejorar la comprensión del seguimiento de la campaña, al final del informe se muestra un gráfico con la cantidad de puntos que obtienen los clientes en cada compra:

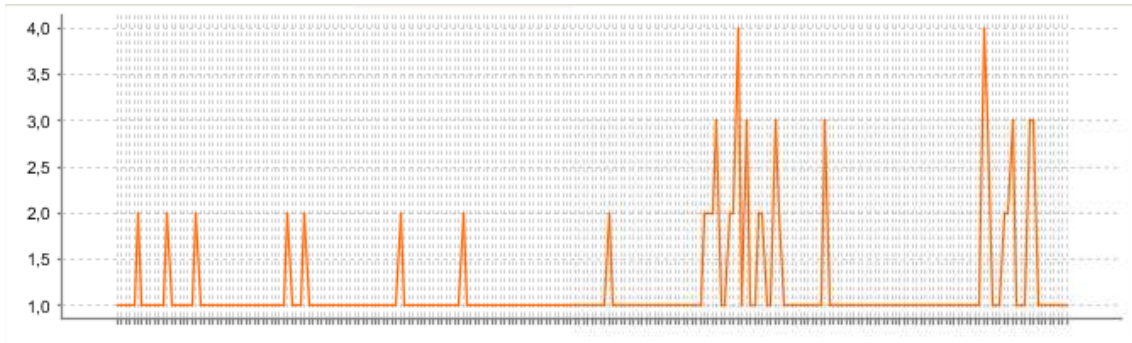


Ilustración 41. Puntos de Madrid en cada compra de septiembre (primer mes de campaña)

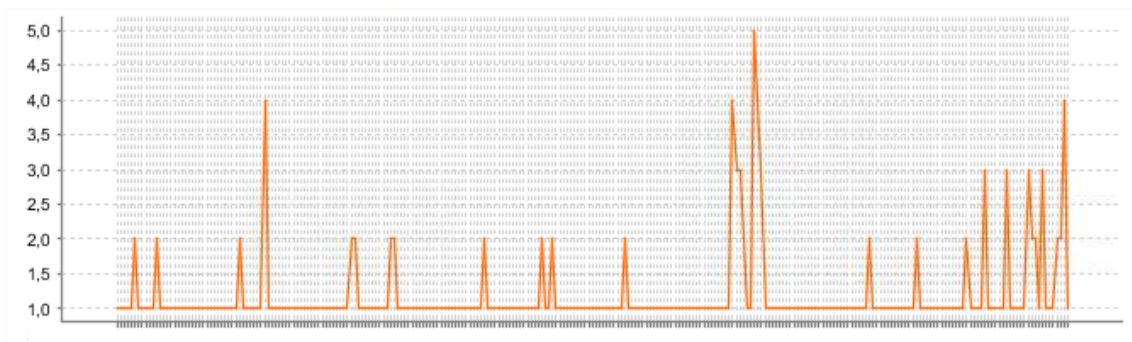


Ilustración 42. Puntos de Madrid en cada compra de diciembre (último mes con datos)

Tanto en la imagen de septiembre como de diciembre, se aprecia que la mayoría de los clientes que obtienen puntos en la compra reciben un punto, pero que hay picos con más puntos. La principal diferencia que se observa es que de septiembre a diciembre disminuye el número de compras con 3 puntos. Pero, las compras de 4 puntos mejoran, llegando una compra incluso hasta los 5 puntos, por lo que los clientes están gastando más dinero en sus compras.

3.3.4 Informe de ventas por tienda y día a través de un cubo

OLAP¹⁰

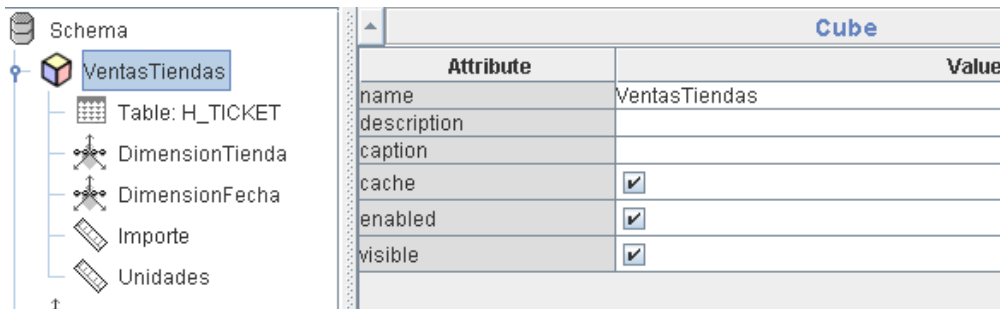
En esta ocasión, la dirección pretende conocer en qué días o en qué meses las sedes obtienen mejores resultados en las ventas, lo que le permitirá tomar ciertas decisiones. Por ejemplo, dotar de más recursos a las tiendas en los días de más ventas o estudiar cómo mejorar las ventas en los días con peores resultados.

Para dar respuesta a estas preguntas, se podría haber creado un informe convencional como los anteriores. Sin embargo, se ha pensado que un cubo

OLAP dotará al informe de más dinamismo, pues se podrá cambiar de tienda y día de una manera mucho más rápida. Por lo tanto, con la creación del cubo se pretende dar una visión general de los importes y las unidades vendidas en cada tienda, donde se podrán seleccionar diferentes tiendas, meses y días. La finalidad del informe es poder detectar en qué días o en qué meses se realizan más o menos ventas, detectar la razón por lo que ocurre y tomar una decisión. Además, se pueden comparar las ventas de diferentes tiendas en el mismo periodo de tiempo.

Diseño del informe

Para la creación del cubo, se ha recurrido a la tabla de hechos h_ticket y las tablas de dimensiones dim_tienda y dim_fecha. Por otro lado, se han utilizado las medidas importe de la venta y unidades vendidas. Con estos datos, se puede hacer un repaso o un comparativo de las ventas de cada tienda en diferentes días:



The screenshot shows a BI tool interface. On the left, a 'Schema' tree lists the components: 'VentasTiendas' (the cube), 'Table: H_TICKET' (the fact table), 'DimensionTienda' (the store dimension), 'DimensionFecha' (the date dimension), 'Importe' (the sales amount measure), and 'Unidades' (the units sold measure). On the right, a 'Cube' configuration table is displayed.

Attribute	Value
name	VentasTiendas
description	
caption	
cache	<input checked="" type="checkbox"/>
enabled	<input checked="" type="checkbox"/>
visible	<input checked="" type="checkbox"/>

Ilustración 43. Diseño del cubo de las ventas por tienda y fecha

En cuanto a las dimensiones, se ha utilizado la dimensión tienda con una sola jerarquía (nombre de la tienda), mientras que para la dimensión fecha se han utilizado dos jerarquías (el mes y el día). Además, en la dimensión fecha se ha recurrido a las columnas que contienen la descripción del mes y del día para que la lectura del cubo resulte más sencilla para el usuario:

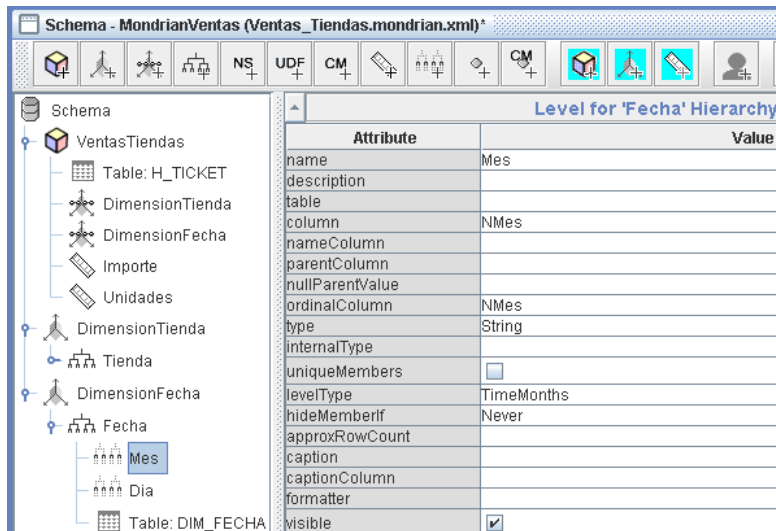


Ilustración 44. Diseño de las dimensiones y jerarquías del cubo de las ventas por fecha y día

Tras crear una nueva vista con jPivot, se pueden recorrer las diferentes tiendas e ir desplegando los meses y días para ver los importes o unidades vendidos en cada periodo. Por ejemplo, en la siguiente captura se puede apreciar que en la tienda de Barcelona se han vendido un total de 12.119 unidades con un importe total de 151.864€, mientras que en septiembre se vendieron 2.487 unidades con un importe total de 33.655€. Además, también se pueden ver las métricas de cada día del mes:

Barcelona	All DimensionFecha.Fechas	151.864	12.119
	Septiembre	33.655	2.487
	1 Sep	1.576	115
	2 Sep	2.092	149
	4 Sep	1.333	111
	5 Sep	176	19
	6 Sep	601	48
	7 Sep	1.369	95
	8 Sep	2.838	154
	9 Sep	2.311	169

Ilustración 45. Navegación del cubo de ventas por fecha y día

Si lo que se desea es comparar los datos de diferentes tiendas, es muy sencillo hacerlo a través de un cuadro OLAP, ya que basta desplegar el periodo de tiempo de cada una de las tiendas que interesa comparar:

Londres I	All DimensionFecha.Fechas	582.023	43.476
	Septiembre	129.510	9.705
	Octubre	140.283	10.464
	Noviembre	140.553	10.407
	Diciembre	171.677	12.900
Londres II	All DimensionFecha.Fechas	249.212	18.951
	Septiembre	60.262	4.515
	Octubre	59.026	4.676
	Noviembre	59.068	4.405
	Diciembre	70.856	5.355

Ilustración 46. Comparación de las ventas de las tiendas de Londres

Tras recorrer el cubo por sus diferentes tiendas se observa que en diciembre, suele aumentar mucho tanto el importe como el nº de artículos vendidos en todas las tiendas. Sin embargo, en algunas de las tiendas se reduce. Por ejemplo, las tiendas ubicadas en Madrid o Liverpool tienen más ventas en noviembre que en diciembre, por lo que se podría recurrir a la publicidad para ofertar los productos de cara a las navidades.

Por otro lado, si observamos las tiendas de Londres I y Londres II, se aprecia que, aunque Londres I tiene ventas muy superiores a Londres II, ésta última sigue siendo una tienda con buenas ventas. Sin embargo, al comparar Manhattan I y II, se observa que Manhattan II tiene muy pocas ventas:

Manhattan I	All DimensionFecha.Fechas	465.203	31.063
	Septiembre	106.174	7.036
	Octubre	111.044	7.276
	Noviembre	120.714	7.929
	Diciembre	127.271	8.822
Manhattan II	All DimensionFecha.Fechas	63.020	4.160
	Septiembre	11.475	782
	Octubre	14.270	981
	Noviembre	17.522	1.121
	Diciembre	19.753	1.276

Ilustración 47. Comparación de las ventas de las tiendas de Manhattan

Por lo tanto, se puede realizar alguna campaña para incrementar las ventas de dicha sucursal, estudiar la posibilidad de cambiar la ubicación de la tienda o estudiar la necesidad de conservar dos sucursales en Manhattan.

3.3.5 Rentabilidad de tiendas por día y hora a través de un cubo OLAP¹¹

Complementario al informe anterior, la dirección se ha dado cuenta de que en todas las tiendas existen horarios con mucha afluencia de clientes, mientras que hay otros horarios en los que la afluencia es casi nula. Por lo tanto, pretenden conocer las ventas de las sucursales según los diferentes horarios. Así, con la intención de disponer de un informe dinámico, se ha decidido crear un cubo que muestre las ventas realizadas en las diferentes tiendas, según el día de la semana y la hora escogida, de forma que se pueda analizar qué día de la semana se realizan más ventas o cual es la el horario en el que más flojean las mismas. En consecuencia, este informe permitirá saber si es necesario reforzar el personal de la tienda en algún día y horario concreto o si será necesario atraer a los clientes en algún horario especial. Además, se podrán realizar comparaciones entre las diferentes tiendas para ver si existe algún caso especial y tomar las decisiones que más convengan a la compañía.

Diseño del informe

El diseño del cubo se ha realizado a través de la tabla de hechos h_cabecera_ticket y de las dimensiones dim_tienda, dim_fecha y dim_hora. Además, se han incluido las medidas importe y unidades, ya que se trata de las métricas de la dimensión que permiten medir los datos de las ventas:

The screenshot shows a BI tool interface with a schema tree on the left and a cube attribute table on the right. The schema tree includes a 'VentasHoras' cube, a 'Table: H_TICKET' fact table, and three dimensions: 'DimensionTienda', 'DimensionFecha', and 'Dimension Hora'. There are also two measures: 'Importe' and 'Unidades'. The cube attribute table on the right lists the following attributes:

Attribute	
name	VentasHoras
description	
caption	
cache	<input checked="" type="checkbox"/>
enabled	<input checked="" type="checkbox"/>
visible	<input checked="" type="checkbox"/>

Ilustración 48. Diseño del cubo de las ventas por día y hora

En esta ocasión, se han utilizado tres dimensiones, todas ellas con una sola jerarquía: la dimensión tienda (nombre de la tienda), la dimensión día (día de la semana) y la dimensión hora (hora de la venta). El hecho de utilizar el día de la semana mejora la usabilidad del informe y da valor al mismo, ya que utilizando

las fechas de venta sería difícil darse cuenta si los clientes prefieren realizar las compras en festivo o si por el contrario prefieren realizarlas en días lectivos. Por lo tanto, el informe permitirá comprobar las ventas tanto por el día de la semana como por la hora en que se realizan.

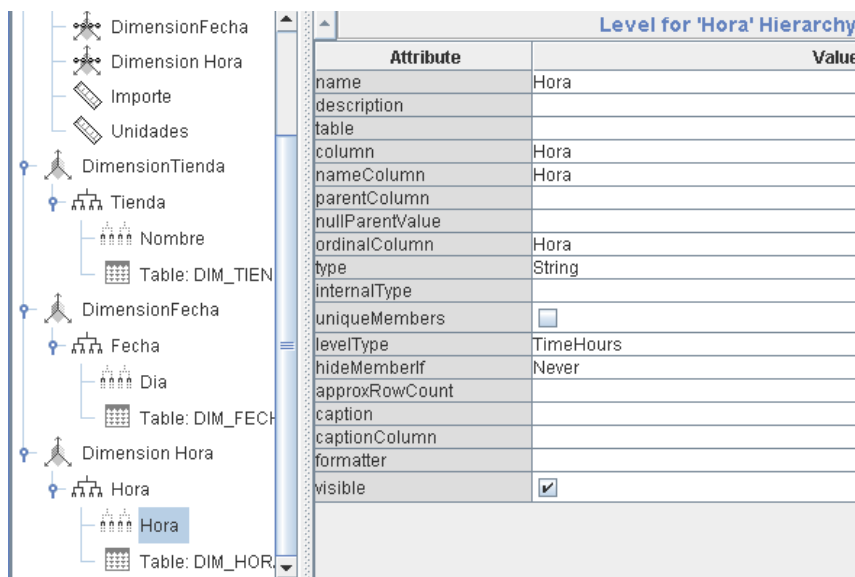


Ilustración 49. Diseño de las dimensiones y jerarquías del cubo de las ventas por día y hora

Creando una vista con jPivot se puede recorrer el cubo por las diferentes tiendas, y centrarse en los días de la semana, en las horas en las que se realizan las ventas o en ambas. Por ejemplo, se puede apreciar que en general, los días más rentables son los domingos, seguidos por los jueves y los sábados. En contra, se aprecia que los lunes son los días de la semana más flojos en ventas:

Tienda	Fecha	Hora	Medidas	
			• Importe	• Unidades
All DimensionTienda.Tiendas	All DimensionFecha.Fechas	All Dimension Hora.Horas	3.324.898	231.853
	Miércoles	All Dimension Hora.Horas	99.323	6.929
	Jueves	All Dimension Hora.Horas	125.553	8.765
	Viernes	All Dimension Hora.Horas	67.584	4.870
	Sábado	All Dimension Hora.Horas	117.833	8.131
	Lunes	All Dimension Hora.Horas	58.724	4.130
	Martes	All Dimension Hora.Horas	105.861	7.367
	Domingo	All Dimension Hora.Horas	204.900	14.408

Ilustración 50. Rentabilidad por días de la semana

Al mismo tiempo, se observa que en Fort Lauderdale y en Miami Beach, el día de menos ventas es el sábado, mientras que los lunes se trata de un día corriente de ventas. Con esta información, la dirección puede optar por reforzar

con más personal en las sucursales los días con más ventas, mientras que para los días más flojos en ventas, se podría incluir una promoción con la finalidad de atraer un mayor número de clientes y aumentar las ventas de esos días. Además, en Fort Lauderdale y en Miami Beach, donde el día de menos ventas es sábado, se podría premiar con el doble de puntos en el importe del ticket, invitando a los clientes habituales para que se acercaran en los días de menos ventas.

Por otro lado y, centrándose en que Fort Lauderdale es la sucursal con menos ventas de todas, se puede ver qué días y a qué horas, las ventas son menores:

Fort Lauderdale	All Dimension Fecha.Fecha	All Dimension Hora.Horas	46,406	3,094
	Miércoles	All Dimension Hora.Horas	1,490	99
	Jueves	All Dimension Hora.Horas	1,119	78
	Viernes	All Dimension Hora.Horas	533	38
	Sábado	All Dimension Hora.Horas	469	28
	Lunes	All Dimension Hora.Horas	894	63
			9	29
			10	50
			12	15
			13	147
			18	176
			20	147
			21	29
			22	301
	Martes	All Dimension Hora.Horas	1,140	75
			9	33
			10	133
			11	64
			16	97
			17	96
			18	90
			19	30
			20	113
			21	414
			22	70
	Domingo	All Dimension Hora.Horas	3,377	232

Ilustración 51. Horas sin ventas en Fort Lauderdale

En concreto, entre las 14h y 17h de los lunes y entre las 12h y las 15h de los martes, no existe ningún ticket de caja, por lo que conviene incrementar las ventas de dichas franjas horarias. Por ejemplo, a través de una promoción, a través de degustaciones gratuitas o a través de representaciones sencillas de cómo cocinar o servir algunos de los productos menos usuales.

Otras consideraciones

En ocasiones, los objetivos de la organización pueden cambiar con la intención de mejorar en otros aspectos. Por ejemplo, la organización puede decidir cambiar su estrategia para centrarse en el proceso de pedidos. Por ejemplo, se puede suponer que la organización ha detectado problemas en el suministro de pedidos y pretende priorizar este proceso para resolver los problemas que

están surgiendo. Esta es la razón por la que conviene crear un *data warehouse* completo, es decir, que no se centre en un solo proceso de la organización. A continuación, se muestran algunos de los informes que pueden ayudar a la organización a tomar las mejores decisiones para optimizar la calidad del proceso de pedidos:

3.3.6 Pedidos a proveedores servidos fuera de plazo¹²

Desde la dirección, se pretende conocer si los pedidos realizados a los proveedores llegan en el tiempo que se tiene previsto, ya que se han dado cuenta de que una mala gestión puede provocar problemas de falta de stock y pérdida de ventas. Por ello, se ha decidido crear un informe con el que comprobar si los pedidos de las diferentes sucursales están llegando en el tiempo marcado por la organización.

Diseño del informe

A la hora de diseñar el informe se ha utilizado la tabla de hechos *h_pedido* y la tabla de dimensiones *dim_producto*, ya que con los datos que contienen, se pueden obtener todos aquellos pedidos que han llegado fuera de plazo. También permitirá conocer los productos que contiene el pedido, así como el proveedor al que se le solicitó. Además, se han agrupado por tienda, para poder identificar si se trata de errores de suministro en una tienda concreta o si por el contrario se trata de errores habituales del proveedor. Igualmente, se han ordenado por fecha, con la intención de saber si se trata de un error en alguna época destacada, como pueda ser el periodo navideño o el vacacional, donde suele haber más problemas con los suministros de pedidos.

En informe se incluye un parámetro llamado *maxDiasEntrega*, según los días marcados por la organización. Gracias a dicho parámetro, el usuario podrá influir en el resultado, ya que indicará el periodo máximo de días en que puede llegar un pedido sin que sea un trastorno para la organización. De esta manera, el informe solamente mostrará aquellos pedidos en los que haya transcurrido más días de los que haya indicado el usuario. Este parámetro se ha establecido por defecto en 20 días, ya que es el tiempo previsto para que llegue un pedido, aunque el usuario lo puede modificar al ejecutar el informe:

Name	maxDiasEntrega
Label	Días máximos para la entrega:
Label Formula	...
Value Type	Number
Data Format	
Data format Formula	...
Default Value	20
Default Value Formula	...
Post-Processing Formula	...
<input checked="" type="checkbox"/> Mandatory <input type="checkbox"/> Hidden	
Prompt	
Display Type	Text Box
Query	RetrasoEntregaPedidos
Value	NOMBRETIENDA
<input checked="" type="checkbox"/> Validate Values <input type="checkbox"/> Reset to default value on invalid selections <input type="checkbox"/> Use first value if default value formula results in NA.	

Ilustración 52. Definición parámetro maxDiasEntrega

En otras palabras, si no se modifica el parámetro, la ejecución del informe devolverá un listado de todos los pedidos que han sufrido un retraso de más de 20 días, mientras que si se modifica el parámetro, el informe devolverá los pedidos que hayan sufrido un retraso de más de los días indicados.

La consulta SQL utilizada para obtener el *report* es la siguiente, en la que se ha incluido una columna con valor 1 en todas las filas para poder contabilizar el número de proveedores que se retrasan en sus pedidos, el cual se utilizará para mostrar el gráfico:

```

SELECT DISTINCT
    "dbo"."H_PEDIDO"."NOMBRETIENDA",
    "dbo"."H_PEDIDO"."FECHASOLICITUD",
    "dbo"."H_PEDIDO"."FECHAENTREGA",
    (datediff(day, "dbo"."H_PEDIDO"."FECHASOLICITUD",
              "dbo"."H_PEDIDO"."FECHAENTREGA")) as DIASENTREGA,
    "dbo"."DIM_PRODUCTO"."NOMBREPROVEEDOR",
    1 AS LINEA
FROM
    "dbo"."DIM_PRODUCTO" INNER JOIN "dbo"."H_PEDIDO"
ON "dbo"."DIM_PRODUCTO"."CODPRODUCTO" =
    "dbo"."H_PEDIDO"."CODPRODUCTO"
WHERE
    datediff(day, "dbo"."H_PEDIDO"."FECHASOLICITUD",
             "dbo"."H_PEDIDO"."FECHAENTREGA") > ${maxDiasEntrega}
  
```

```

GROUP BY
    "dbo"."H_PEDIDO"."NOMBRETIENDA",
    "dbo"."H_PEDIDO"."FECHASOLICITUD",
    "dbo"."H_PEDIDO"."FECHAENTREGA",
    "dbo"."DİM_PRODUCTO"."DESCRIPCION",
    "dbo"."DIM_PRODUCTO"."NOMBREPROVEEDOR"
ORDER BY
    "dbo"."H_PEDIDO"."NOMBRETIENDA",
    "dbo"."H_PEDIDO"."FECHASOLICITUD"

```

Tras la ejecución del *report* se puede observar lo que ocurre en una sola tienda, ya que están ordenadas por el nombre de la tienda, o se pueden comparar los resultados de varias tiendas:

Pedidos que se retrasan más de 20 días			
NOMBRETIENDA Barcelona			
FECHA SOLICITUD	FECHA ENTREGA	DIAS ENTREGA	NOMBRE PROVEEDOR
01-sep-2000	30-sep-2000	29	The Holland House
01-nov-2000	30-nov-2000	29	The Holland House
NOMBRETIENDA Florencia			
FECHA SOLICITUD	FECHA ENTREGA	DIAS ENTREGA	NOMBRE PROVEEDOR
01-ago-2000	04-sep-2000	34	Australian Cellars Company
01-sep-2000	30-sep-2000	29	The Holland House
01-nov-2000	30-nov-2000	29	The Holland House
01-nov-2000	05-dic-2000	34	Australian Cellars Company
NOMBRETIENDA Fort Lauderdale			
FECHA SOLICITUD	FECHA ENTREGA	DIAS ENTREGA	NOMBRE PROVEEDOR
01-ago-2000	28-ago-2000	27	Australian Cellars Company
01-ago-2000	28-ago-2000	27	Cia. De Licores, S.A.
01-sep-2000	30-sep-2000	29	The Holland House
01-nov-2000	28-nov-2000	27	Australian Cellars Company
01-nov-2000	30-nov-2000	29	The Holland House
NOMBRETIENDA Liverpool			
FECHA SOLICITUD	FECHA ENTREGA	DIAS ENTREGA	NOMBRE PROVEEDOR
01-ago-2000	04-sep-2000	34	Australian Cellars Company
01-sep-2000	30-sep-2000	29	The Holland House
01-nov-2000	30-nov-2000	29	The Holland House
01-nov-2000	05-dic-2000	34	Australian Cellars Company

Ilustración 53. Ejecución del report Pedidos con retraso

En la imagen se aprecia que en la tienda de Barcelona, todos los pedidos que sufren retraso son servidos por el mismo proveedor (*The Holland House*). Además, se trata de pedidos que no se han realizado en periodos vacacionales o navideños, por lo que sería necesario hablar con dicho proveedor para averiguar las causas del retraso, ya que pueden ser muchos los motivos. Entre otras causas, puede ser debido a falta de stock por parte del proveedor o por retrasos en el transporte.

En el primer caso, no existe forma de averiguarlo, pero en el segundo caso se puede recurrir a los seguimientos de los envíos del transportista. Sin embargo, en la tienda de Fort Lauderdale los retrasos son sufridos por parte de tres proveedores, por lo que pueden ser diferentes motivos las causas de que se sufra dicho retraso. En cualquier caso, la dirección ya dispone de información

real sobre los proveedores que están generando retrasos en sus pedidos, por lo que ya está en disposición de tomar las mejores decisiones para ofrecer una solución.

Por otro lado, en el pie del informe se presenta un gráfico donde se muestran los proveedores que se retrasan. Las barras muestran en diferentes colores las tiendas en las que los pedidos de dicho proveedor han sufrido retraso, indicando la suma de los pedidos que han llegado más tarde de lo esperado:

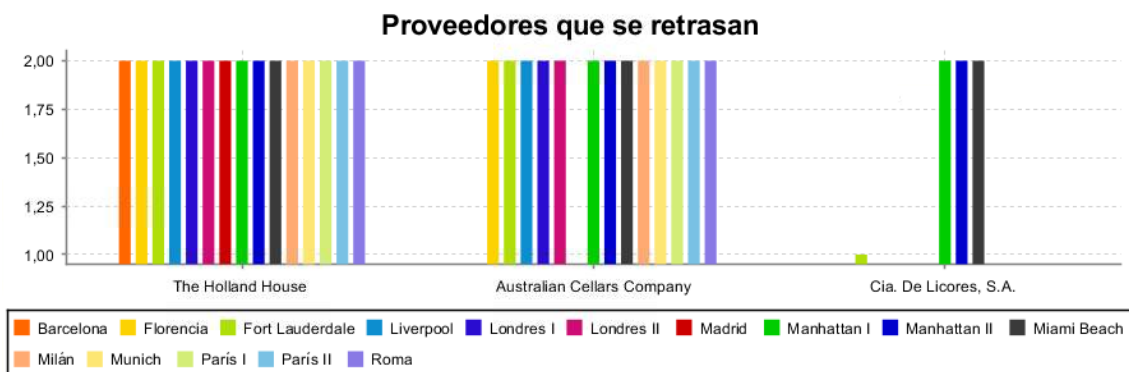


Ilustración 54. Gráfico con el número de pedidos con retraso de los proveedores en las diferentes tiendas

En el gráfico llama la atención la linealidad de los pedidos retrasados de todos los proveedores y de las tiendas, ya que en la práctica totalidad de los casos, son dos los pedidos de cada tienda y de cada proveedor los que se retrasan, por lo que parece ser un problema del propio proveedor.

Sin embargo, se observa que la Cia. De Licores S.A. solamente sufre retrasos en cuatro tiendas: Fort Lauderdale, Manhattan I, Manhattan II y Miami Beach. Es decir, las cuatro sucursales de EEUU, por lo que los motivos del retraso pueden deberse a otras causas como problemas con el transporte o problemas de aduanas, ya que se trata de una compañía de licores. Por lo tanto, la dirección de la empresa ya dispone de datos para tomar la mejor decisión.

3.3.7 Pedidos con la cantidad entregada menor que la solicitada

La dirección tiene firmados acuerdos con ciertos proveedores para garantizarse un mínimo de stock. Pero, se han dado cuenta de que no siempre se están cumpliendo estos contratos. Por esta razón, desde el departamento de BI se ha

querido mostrar a la dirección qué proveedores están incumpliendo las condiciones establecidas. A partir de ahí, se ha creado un informe que permite conocer con qué proveedores está habiendo problemas, ya que en ocasiones los proveedores no están sirviendo los pedidos con las cantidades que se habían solicitado, probablemente debido a que no tienen stock suficiente. Aunque este hecho no siempre debe suponer un problema, es necesario detectar qué proveedores tienen tendencia a no servir como es debido y con qué productos suele haber más problemas de abastecimiento.

Diseño del informe

Para diseñar el informe se han utilizado dos tablas: la tabla de hechos h_pedido y la tabla de dimensiones dim_producto. Con estas dos tablas se pueden obtener los datos necesarios para el informe, que incluyen el nombre de la tienda, el producto, la cantidad solicitada y servida, el nombre del proveedor y el porcentaje que ha sido entregado del producto en dicho pedido.

Como en el informe anterior, se ha incluido una columna con valor 1 para poder contabilizar a los pedidos que generan problemas, quedando la consulta utilizada para obtener la información de la siguiente manera:

```
SELECT
    "dbo"."H_PEDIDO"."NOMBRETIENDA",
    "dbo"."H_PEDIDO"."FECHASOLICITUD",
    "dbo"."DIM_PRODUCTO"."DESCRIPCION",
    "dbo"."H_PEDIDO"."CANTIDAD SOLICITADA",
    "dbo"."H_PEDIDO"."CANTIDAD ENTREGADA",
    (( "dbo"."H_PEDIDO"."CANTIDAD ENTREGADA" * 100) /
    "dbo"."H_PEDIDO"."CANTIDAD SOLICITADA") as PORENTEGADO,
    "dbo"."DIM_PRODUCTO"."NOMBREPROVEEDOR",
    1 AS PEDIDO
FROM
    "dbo"."DIM_PRODUCTO" INNER JOIN "dbo"."H_PEDIDO"
    ON "dbo"."DIM_PRODUCTO"."CODPRODUCTO" =
        "dbo"."H_PEDIDO"."CODPRODUCTO"
WHERE
    "dbo"."H_PEDIDO"."CANTIDAD SOLICITADA" <>
    "dbo"."H_PEDIDO"."CANTIDAD ENTREGADA"
GROUP BY
    "dbo"."H_PEDIDO"."NOMBRETIENDA",
    "dbo"."H_PEDIDO"."FECHASOLICITUD",
    "dbo"."DIM_PRODUCTO"."DESCRIPCION",
    "dbo"."H_PEDIDO"."CANTIDAD SOLICITADA",
    "dbo"."H_PEDIDO"."CANTIDAD ENTREGADA",
    "dbo"."DIM_PRODUCTO"."NOMBREPROVEEDOR"
```

ORDER BY

```
"dbo"."H_PEDIDO"."NOMBRETIENDA",
"dbo"."H_PEDIDO"."FECHASOLICITUD",
"dbo"."DİM_PRODUCTO"."DESCRIPCION"
```

Además, los resultados se han agrupado por tienda para poder tener una idea de los problemas detectados en cada una de las sucursales de Gourmet. También se han ordenado por fecha, para poder apreciar si se trata de varios productos del mismo o diferentes pedidos. Por lo tanto, si se observa el resultado de la consulta ya se pueden sacar algunas conclusiones:

Pedidos con cantidad entregada menor que la solicitada

NOMBRETIENDA: Barcelona					
FECHA SOLICITUD	PRODUCTO	CANTIDAD SOLICITADA	CANTIDAD ENTREGADA	% ENTREGADO	NOMBRE PROVEEDOR
01-ago-2000	Tinto Reserva 94	12	9	75	Bodegas Hermanos Alvarez
01-oct-2000	Chèvre	10	0	0	Compagnie Laitier
01-nov-2000	Tinto Reserva 94	24	19	79	Bodegas Hermanos Alvarez

NOMBRETIENDA: Florencia					
FECHA SOLICITUD	PRODUCTO	CANTIDAD SOLICITADA	CANTIDAD ENTREGADA	% ENTREGADO	NOMBRE PROVEEDOR
01-ago-2000	Tinto Reserva 94	12	9	75	Bodegas Hermanos Alvarez
01-oct-2000	Chèvre	10	0	0	Compagnie Laitier
01-nov-2000	Tinto Reserva 94	12	9	75	Bodegas Hermanos Alvarez

NOMBRETIENDA: Fort Lauderdale					
FECHA SOLICITUD	PRODUCTO	CANTIDAD SOLICITADA	CANTIDAD ENTREGADA	% ENTREGADO	NOMBRE PROVEEDOR
01-ago-2000	Tinto Reserva 94	12	9	75	Bodegas Hermanos Alvarez
01-oct-2000	Chèvre	10	0	0	Compagnie Laitier
01-nov-2000	Tinto Reserva 94	12	9	75	Bodegas Hermanos Alvarez

NOMBRETIENDA: Liverpool					
FECHA SOLICITUD	PRODUCTO	CANTIDAD SOLICITADA	CANTIDAD ENTREGADA	% ENTREGADO	NOMBRE PROVEEDOR
01-ago-2000	Shiraz 96	12	9	75	Vineyards of Australia Inc.
01-ago-2000	Tinto Crianza 96/97	12	9	75	Bebidas Selección
01-ago-2000	Tinto Reserva 94	12	9	75	Bodegas Hermanos Alvarez
01-oct-2000	Chèvre	10	0	0	Compagnie Laitier
01-nov-2000	Shiraz 96	24	19	79	Vineyards of Australia Inc.
01-nov-2000	Tinto Crianza 96/97	24	19	79	Bebidas Selección
01-nov-2000	Tinto Reserva 94	12	9	75	Bodegas Hermanos Alvarez

Ilustración 55. Ejecución del *report* Pedidos con cantidad entregada menos que la solicitada

En la imagen, se aprecia que las Bodegas Hermanos Álvarez tienen problemas de suministro de Tinto Reserva 94, ya que ocurre en diferentes tiendas. A pesar de esa falta de stock, en todos los casos sirven al menos un 75% de la cantidad solicitada, por lo que se podría solucionar realizando pedidos de una cantidad algo mayor.

Además, se observa que el proveedor *Compagnie Laitier* no está suministrando *Chèvre* en ninguno de los pedidos, pues en todos los pedidos de *Chèvre* a este proveedor en los que ha habido problemas, no se ha llegado a suministrar

ninguna cantidad del producto. Varias son las razones por las que puede estar ocurriendo, por lo que la dirección deberá tomar la decisión más apropiada al problema. Por ejemplo, sustituir este producto por alguno similar o buscar otro proveedor.

Por otro lado, para dar un aspecto más visual al informe, se han incluido algunos gráficos que también pueden ayudar a la toma de decisiones:

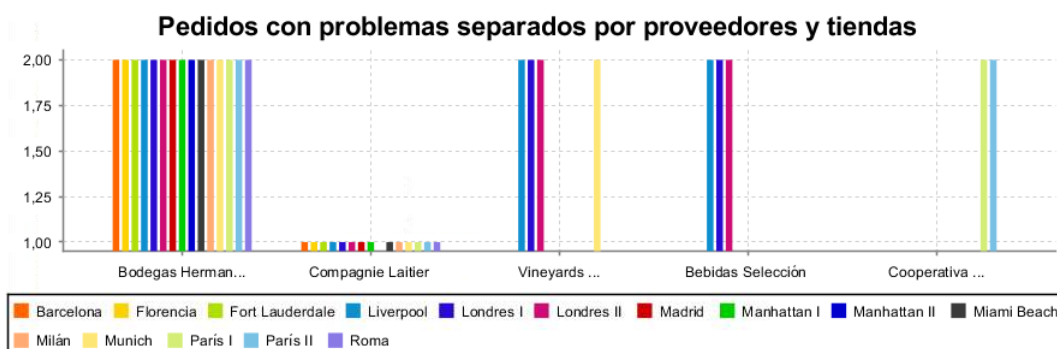


Ilustración 56. Pedidos con problemas separados por proveedores y tiendas

En la ilustración se aprecia que son cinco los proveedores que no han suministrado la cantidad total de productos solicitados, pero que cada uno no cumple debido a un solo producto. En el caso de *Compagnie Laitier* ha ocurrido en un solo pedido de cada tienda, mientras que en el resto de proveedores ha ocurrido en dos pedidos.

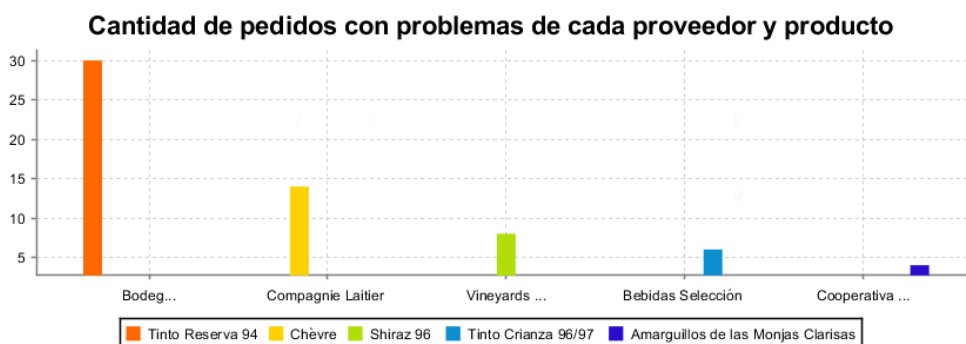


Ilustración 57. Cantidad de pedidos con problemas de cada proveedor y producto

En esta imagen se puede observar que ha habido 30 pedidos con problemas de la cantidad suministrada de Tinto reserva 94, mientras que de Amarguillos de las monjas Clarisas solamente han fallado 4. Sin embargo, si observamos el siguiente gráfico:

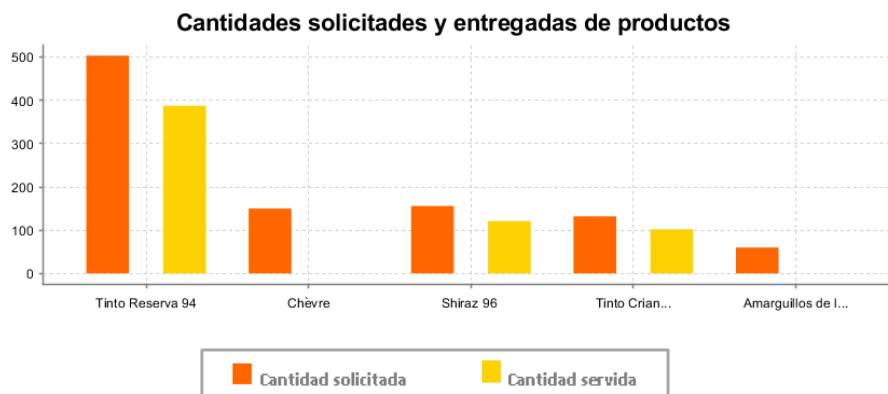


Ilustración 58. Cantidades solicitadas y entregadas de productos

Se puede observar que del producto Tinto reserva 94 se han entregado casi 400 botellas de las 500 solicitadas, mientras que de los 4 pedidos que habían fallado de Amarguillos no se ha entregado ningún pastel. Por lo tanto, se debe estudiar cada caso con el detenimiento suficiente para saber dónde está el problema de abastecimiento y el tipo de problema que supone para la empresa.

En definitiva, los informes utilizados para el proceso de compras permite determinar los proveedores que están incumpliendo habitualmente el abastecimiento de productos a la cadena de tiendas gourmet, ya sea por retrasos en las entregas o por entregar artículos con menos cantidades de las solicitadas, lo que permitirá a la dirección tomar las mejores decisiones para mejorar el proceso de pedidos a proveedores.

3.4 Viabilidad del producto

En la actualidad, las empresas se encuentran constantemente en la necesidad de tomar decisiones estratégicas. Muchas de estas decisiones se toman basadas en la experiencia de la persona que toma la decisión, es decir, sin estudiar la información con la que cuenta la empresa. En otras ocasiones, las decisiones sí que se toman en base a los datos de que disponen las empresas, sin embargo, el proceso es muy costoso y la información no llega en el momento en el que se necesita. Por lo tanto, se puede realizar un pequeño estudio sobre la viabilidad de poner en marcha el producto resultante de este trabajo sobre un entorno real en tiendas Gourmet.

3.4.1 Viabilidad técnica

Todas las sucursales de tiendas Gourmet están conectadas a un servidor principal, donde se encuentra la base de datos principal de la organización. Además, la organización no cuenta con un número elevado de sucursales, por lo que el sistema no requiere un ancho de banda elevado para obtener los informes. Es decir, la organización ya cuenta con las comunicaciones necesarias para utilizar correctamente Pentaho desde cualquier sucursal de la organización.

En cuanto a poder acceder al sistema de BI desde los puestos de trabajo que hay en la organización, no se requiere ningún cambio, ya que la organización cuenta con una red local. Sin embargo, sí que es necesario configurar la solución en un servidor actual, por lo que será necesario adquirir uno nuevo donde instalar el sistema operativo y las herramientas de Pentaho.

Además, tanto las líneas de comunicaciones como la solución de BI de Pentaho son escalables, por lo que si aumentaran las sucursales y/o el personal de tiendas Gourmet, sería sencillo poder adaptarse a dicho crecimiento sin perder la calidad de la solución.

3.4.2 Viabilidad operativa

En primer lugar, la implantación de la solución se realizará por medio del departamento de Tecnologías de la Información, que cuenta con dos perfiles diferenciados: un administrador de base de datos y un administrador de sistemas. Además, el departamento cuenta con un responsable del departamento que a su vez hace las funciones de CIO de la organización, por lo que los objetivos dictados por la organización y los objetivos del departamento de TI están alineados. Por lo tanto, el departamento cuenta con el personal necesario para la correcta puesta en marcha del servidor y de la solución, así como de la preparación de los informes de *reporting* y cubos OLAP. Además, el mantenimiento de la solución se realizaría por parte del personal del departamento, tanto el del sistema operativo como el de las herramientas de Pentaho.

Por otro lado, hay que tener en cuenta la respuesta que tendrá el personal de la organización a la hora de aceptar estos nuevos procesos de trabajo. Por lo general las personas son reacias a cualquier tipo de cambio, pues generan incertidumbre, inseguridad e implican un esfuerzo extraordinario a realizar. Esta resistencia al cambio suele venir por diversos factores: querer defender sus propios intereses, como el temor a perder poder en la empresa; falta de comprensión, debido al desconocimiento del proyecto o desconfianza provocada por el miedo a no ser capaz de desempeñar correctamente el trabajo con los nuevos procesos. Por lo tanto, es necesario minimizar esta resistencia al cambio mediante un modelo de gestión del cambio, en el que se intentará analizar a los implicados con la finalidad de comprender los intereses y las resistencias de cada una de las partes. Además, se podría minimizar dicha resistencia por medio de formación y apostando por una figura informal de líder que consiga involucrar a los trabajadores de las diferentes sucursales de la cadena. Esta figura, permitiría un diálogo fluido con los responsables de cada sucursal, con la que se obtendría la retroalimentación necesaria para optimizar los *reports* del proyecto.

3.4.3 Viabilidad Legal

En la cadena de tiendas Gourmet ya se dispone de un fichero electrónico con información sobre los clientes y ya está garantizando el cumplimiento de la Ley de Protección de Datos de Carácter Personal, por lo que no es necesario aplicar ningún otro requisito legal.

3.4.4 Viabilidad financiera

Para la correcta implantación de la solución de BI en la cadena de tiendas Gourmet, sería necesario tener en cuenta varios aspectos financieros: los gastos de adquisición del servidor, los costes de la implantación de la solución y los costes de formación de los empleados.

En cuanto a los gastos de adquisición de la solución, sería necesario contar con un servidor que cumpla con los requerimientos dictados por Pentaho, tanto en el aspecto de hardware como en el aspecto de software. Además, sería

conveniente dotarlo de Windows Server como Sistema Operativo, ya que el presente trabajo está basado en la plataforma Windows. Así, las especificaciones requeridas por Pentaho son:

Hardware—64 bit	Operating System—64 bit
<p>Processor: Intel EM64T or AMD64 Dual-Core</p> <p>RAM: 8 GB with 4 GB dedicated to Pentaho servers</p> <p>Disk Space: 20 GB free after installation</p>	<ul style="list-style-type: none"> • Microsoft Windows 2008 Server R2 & 2012 Server R2 • CentOS 6 & 7 • Red Hat Enterprise 6 & 7 • Ubuntu Server 14.04 LTS & 16.04 LTS • SUSE Linux SLES 11 (SP3+)

Ilustración 59. Requerimientos Pentaho.

Puesto que no se cuenta con un servidor de estas características, será necesario invertir aproximadamente 1552€ entre el hardware (1107€)¹³ y el software (445€)¹⁴, ya que las herramientas de Pentaho no suponen un coste en cuando a su adquisición. No obstante, será necesario realizar la instalación del servidor, del sistema operativo y de todas las herramientas accesorias de Pentaho. En el caso de realizarlo por personal de la propia empresa, sería necesario contabilizar las horas requeridas para su correcta instalación, configuración y pruebas. Por lo tanto, teniendo en cuenta las horas dedicadas en la realización del trabajo y añadiendo las horas necesarias para la instalación del Sistema Operativo se necesitarían alrededor de 110 horas para la instalación. Es decir, supondría un coste aproximado de 1980€¹⁵. Además, habría que tener en cuenta el tiempo dedicado para la creación de los reports y cubos OLAP, que siguiendo con la planificación del proyecto se necesitarían 103 horas, por lo que el coste ascendería a 1854€.

En cuanto a la formación de los usuarios sería necesario contratar una plataforma *online* para su realización, ya que la cadena tiene sucursales en diferentes países. Suponiendo que son 20 personas las que necesitarían formación y que serán necesarias 35h de formación, se calcula un coste aproximado de 1650€, donde se incluyen además los gastos de la plataforma *online*¹⁶.

Finalmente, sumando todos los elementos el coste de implantación de la solución de BI en la cadena de tiendas Gourmet sería:

Recurso	Coste
Servidor (Hardware y S.O.)	1.552€
Configuración servidor	1.980€
Preparación de los informes	1.854€
Formación	1.650€
TOTAL	7.036€

Finalmente, se puede apreciar que el coste económico de la solución es realmente bajo si se tiene en cuenta que se trata de una compañía que cuenta con 15 sucursales, pues solamente tendría una repercusión de 469€ en cada una de ellas, lo cual es perfectamente asumible por todas las sucursales.

3.4.5 Beneficios aportados

Los beneficios que la solución ofrecida podría aportar a la cadena de tiendas Gourmet son:

- Se deja de realizar especulaciones para trabajar directamente con los datos que permiten construir un informe fiable, lo que permitirá tomar las decisiones correctas.
- Las respuestas a los informes son más rápidas, con lo que las decisiones se pueden tomar en el momento oportuno para no perder oportunidades de negocio.
- Es más fácil acceder a las métricas de los hechos. Es decir, se pueden medir los procesos para mejorarlo y obtener información de mayor calidad y más precisa.
- Se pueden crear diferentes modelos que se adapten a la necesidad que se pretende cubrir, ya sea en forma tabular o gráfica. De esta manera, aumenta la capacidad de análisis de la información y se mejora la comprensión.
- Permite conseguir información valiosa, pues es capaz de detectar errores en los procesos, de optimizar las operaciones o de abrir nuevas oportunidades de negocio.

En definitiva, permite recuperar la inversión en un corto espacio de tiempo, ya que en un principio, permitirá a la cadena de tiendas Gourmet gestionar los pedidos de una manera más eficaz y mejorar el proceso de ventas para obtener mayores beneficios.

4. Conclusiones

El trabajo se inició con la intención de adquirir los conocimientos necesarios sobre BI para encajar en el mundo laboral actual, ya que el departamento de Tecnologías de la Información está cada vez más enfocado a ofrecer valor al negocio. Durante el transcurso del mismo, se ha evolucionado satisfactoriamente en varios aspectos. Por un lado, se ha conseguido configurar y poner en marcha un entorno de producción con Pentaho, permitiendo diseñar y construir un *data warehouse* e implementar los informes y cubos OLAP necesarios, con la intención de que la organización pueda tomar las mejores decisiones para los objetivos del negocio. Por otro lado, también se ha logrado orientar los informes tanto hacia los objetivos de la organización como hacia la mejora de procesos.

En cuanto a los objetivos planteados, se ha conseguido familiarizarse con las herramientas y tecnologías de BI y se ha aprendido a construir un *data warehouse* a través de un proceso de ETL. Dicho almacén de datos podría haber sido más completo. Sin embargo, se ha preferido crear un almacén sencillo para que el tiempo dedicado a su diseño y construcción no ponga en riesgo el resto del proyecto, pero que a la vez permita explotar correctamente las tecnologías planificadas en el trabajo. Además, se han adquirido nuevos conocimientos sobre *reporting* y se ha logrado instruirse en la creación de cubos OLAP orientados a la mejora de procesos.

En lo que respecta a la planificación, se han cumplido correctamente los plazos establecidos en el diagrama de Gantt, pues solamente ha habido desviaciones en el calendario para realizar mejoras en la calidad del trabajo. Además, la metodología seguida en la planificación también ha sido correcta, ya que ha permitido tanto adaptarse a los cambios de estrategia de la organización como a entregar modelos de análisis completos. Es decir, no se ha iniciado un nuevo modelo de análisis hasta que no se ha terminado el anterior, eliminando el riesgo de obtener un producto inservible.

Finalmente, el presente trabajo habría podido mejorarse con la realización de un *Scorecard* o de un *Dashboard*. El primero, permitiría mejorar la estrategia de

la organización, mientras que el segundo, lograría mejorar los procesos de ventas y pedidos. No obstante, este aspecto representa una oportunidad para que el autor del trabajo pueda explorar esta tecnología en un futuro próximo. Por otro lado, utilizando tecnología *data mining* se podrían mejorar algunos informes. Por ejemplo, estudiando al cliente potencial de tiendas Gourmet o permitiendo que los empleados de las tiendas puedan ofrecer productos personalizados a sus clientes.

5. Glosario

- *Business Intelligence*¹⁷: Conjunto de estrategias, aplicaciones, datos, productos, tecnologías y arquitecturas técnicas que están enfocados a la administración y creación de conocimiento a través del análisis de los datos existentes en una organización.
- CEO (*Chief Executive Officer*)¹⁸: Persona encargada de la máxima autoridad de la gestión y dirección administrativa de una organización.
- CIO (*Chief Information Officer*)¹⁹: Persona encargada de la tecnología de la información y de los sistemas de computación que soportan los objetivos de la organización.
- *Corporate Information Factory*²⁰: Arquitectura en la que existe un *data warehouse* corporativo y unos *data marts* dependientes del mismo.
- *Dashboard*²¹: Paneles de información que muestran valores de un proceso o las cifras de un departamento en una sola ventana.
- *Data Mart*²²: Subconjunto de datos del *data warehouse* con el propósito de ayudar a que un área específica del negocio pueda tomar decisiones.
- *Data Mining*²³: Campo de la estadística y de las ciencias de computación que intenta descubrir patrones en grandes volúmenes de datos.
- *Data Warehouse*²⁴: Colección de datos orientada a un determinado ámbito (empresa, organización, etc.), no volátil y variable en el tiempo, que ayuda a la toma de decisiones en la entidad en la que se utiliza.
- *Enterprise Bus Architecture*²⁵: Arquitectura de un almacén de datos basada en *data marts* independientes.
- ETL (*Extract, Transform and Load*)²⁶: Proceso que permite a las organizaciones mover datos desde múltiples fuentes, reformatearlos, limpiarlos y cargarlos en otra base de datos, *data mart* o *data warehouse*, para poder analizarlos y apoyar a un proceso de negocio.
- KPI: Indicadores clave de rendimiento²⁷. Se trata de un indicador para medir el rendimiento de un proceso o producto, el cual se ha fijado previamente en función de los objetivos de la organización.

- OLAP (*On-Line Analytical Processing*)²⁸: Estructuras de datos multidimensionales que contienen datos resumidos de grandes Bases de datos y permiten realizar informes con una respuesta más rápida.
- *Open source*²⁹: Software distribuido y desarrollado libremente.
- PMBOK³⁰: Conjunto de estándares, terminología y directrices para la gestión de proyectos.
- *Reporting*: Tecnología basada en la creación de informes operacionales sobre los datos de una organización.
- *Scorecard*³¹: Aplicaciones que miden el progreso hacia el logro de una estrategia u objetivo por medio de KPIs.

6. Bibliografía

- ¹ Curto Díaz, J. y Conesa i Caralt. J. Introducción al business intelligence. Editorial UOC, Barcelona (2010).
- ² Curto Díaz, J. y Conesa i Caralt. J. Introducción al business intelligence. Editorial UOC, Barcelona (2010).
- ³ Rollupconsulting: Cuadrante mágico Business Intelligence [En línea].
<http://rollupconsulting.com/wp-content/uploads/2017/02/Cuadrante-m%C3%A1gico-1.png>
(marzo de 2017)
- ⁴ Pentaho Documentation: Prepare your Windows Environment for Installation. [En línea].
<https://help.pentaho.com/Documentation/7.0/0F0/0P0/020/0A0> (marzo de 2017)
- ⁵ Wikipedia: Tabla de hechos. [En línea]. https://es.wikipedia.org/wiki/Tabla_de_hechos (abril de 2017)
- ⁶ Wikipedia: Tabla de dimensión. [En línea].
https://es.wikipedia.org/wiki/Tabla_de_dimensi%C3%B3n (abril de 2017)
- ⁷ Entrepreneur: Cómo medir las ventas. [En línea]. <https://www.entrepreneur.com/article/259331>
(abril de 2017)
- ⁸ Youtube: Curso de introducción a Pentaho (2). [En línea].
https://www.youtube.com/watch?v=elnPrPds_7c (abril de 2017)
- ⁹ WIVOBLOG: Top 11 KPIs para el éxito en retail. [En línea].
<http://blog.wivoanalytics.com/conoce-tus-tiendas-top-11-kpis-para-exito-en-retail> (abril de 2017)
- ¹⁰ Youtube. Curso de introducción a Pentaho (1). [En línea].
<https://www.youtube.com/watch?v=IQEHd27CdX4> (mayo de 2017)
- ¹¹ Youtube: Crear cubo con Pentaho y MySQL. [En línea].
https://www.youtube.com/watch?v=GwA0qebuO_M (mayo de 2017)
- ¹² La voz de Houston: Cuáles son los indicadores clave de rendimiento para un departamento de compras. [En línea]. <http://pyme.lavoztx.com/cules-son-los-indicadores-clave-de-rendimiento-para-un-departamento-de-compras-4478.html> (mayo de 2017)
- ¹³ DELL: Servidor torre PowerEdge T630. [En línea].
<http://www.dell.com/es/empresas/p/poweredge-t630/pd> (mayo de 2017)
- ¹⁴ Microsoft: Precios y licencias de Windows Server 2016. [En línea].
<https://www.microsoft.com/es-xl/cloud-platform/windows-server-pricing> (mayo de 2017)
- ¹⁵ Jobtonic: Salario: Informática en España. [En línea].
<http://espana.jobtonic.es/salary/26526/16047.html> (mayo de 2017)
- ¹⁶ Elearningfacil: Cómo poner precio a tus cursos *online*. [En línea].
<https://www.elearningfacil.com/como-poner-precio-a-tus-cursos-online/> (mayo de 2017)
- ¹⁷ Wikipedia: Inteligencia empresarial. [En línea].
https://es.wikipedia.org/wiki/Inteligencia_empresarial (marzo de 2017)

-
- ¹⁸ Wikipedia. Director ejecutivo. [En línea]. https://es.wikipedia.org/wiki/Director_ejecutivo (marzo de 2017)
- ¹⁹ Wikipedia: *Chief information officer*. [En línea]. https://en.wikipedia.org/wiki/Chief_information_officer (marzo de 2017)
- ²⁰ Curto Díaz, J. y Conesa i Caralt, J. Introducción al business intelligence. Editorial UOC, Barcelona (2010).
- ²¹ Blog Aníbal Goicoechea: *Scorecard* es a Estrategia como *Dashboard* es a Proceso. [En línea]. <https://anibalgoicochea.com/2009/10/30/scorecard-es-a-estrategia-como-dashboard-es-a-proceso/> (marzo de 2017)
- ²² Wikipedia: Data mart. [En línea]. https://es.wikipedia.org/wiki/Data_mart (marzo de 2017)
- ²³ Wikipedia: Minería de datos. [En línea]. https://es.wikipedia.org/wiki/Miner%C3%ADa_de_datos (marzo de 2017)
- ²⁴ Wikipedia: Almacén de datos. [En línea]. https://es.wikipedia.org/wiki/Almac%C3%A9n_de_datos (marzo de 2017)
- ²⁵ Curto Díaz, J. y Conesa i Caralt, J. Introducción al business intelligence. Editorial UOC, Barcelona (2010).
- ²⁶ Wikipedia: *Extract, transform and load*. [En línea]. https://es.wikipedia.org/wiki/Extract,_transform_and_load (marzo de 2017)
- ²⁷ Wikipedia: Indicador clave de rendimiento. [En línea]. https://es.wikipedia.org/wiki/Indicador_clave_de_rendimiento (abril de 2017)
- ²⁸ Wikipedia: OLAP. [En línea]. <https://es.wikipedia.org/wiki/OLAP> (marzo de 2017)
- ²⁹ Wikipedia. Código abierto. [En línea]. https://es.wikipedia.org/wiki/C%C3%B3digo_abierto (marzo de 2017)
- ³⁰ Wikipedia: *Project management Body of Knowledge*. [En línea]. https://en.wikipedia.org/wiki/Project_Management_Body_of_Knowledge (marzo de 2017)
- ³¹ Blog Aníbal Goicoechea: *Scorecard* es a Estrategia como *Dashboard* es a proceso. [En línea]. <https://anibalgoicochea.com/2009/10/30/scorecard-es-a-estrategia-como-dashboard-es-a-proceso/> (marzo de 2017)

7. Anexos

Anexo 1. Preparación del entorno Windows, instalación y configuración de Pentaho

Crear la estructura de directorios

Para la correcta instalación del programa, se crea la estructura de directorios recomendada por Pentaho:

```
pentaho\server\pentaho-server
```

Instalar Java

Instalar JDK (*Java Development Kit*) para 64 bits, ya que Pentaho viene preparado únicamente para servidores con hardware de 64 bits y que corran en sistemas operativos de 64 bits.

Instalar el servidor de aplicación web

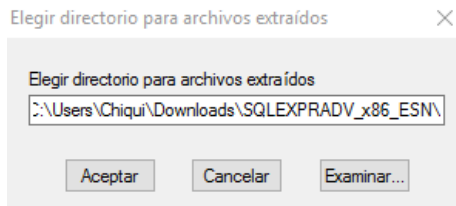
Por defecto, Pentaho viene preparado para trabajar con el servidor Tomcat, por lo que utilizaremos este servidor, ya que viene preinstalado y configurado, por lo que se ahorrará tiempo de configuración.

Instalar la base de datos que se utilizará como repositorio de Pentaho

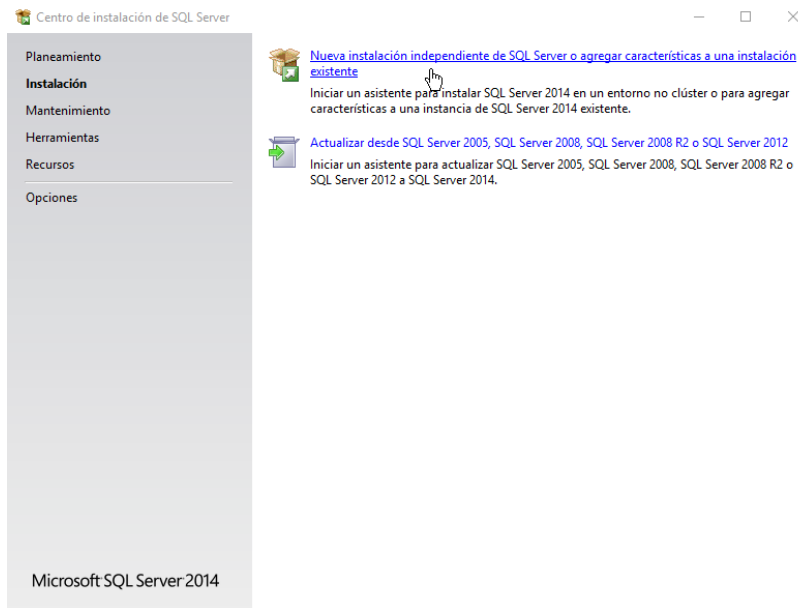
Pentaho viene preparado para trabajar con PostgreSQL. No obstante, se ha decidido trabajar con Microsoft SQL Server Express, debido a que se tiene mayor conocimiento de dicho gestor de bases de datos. La instalación Gestor de Base de Datos es sencilla, tal y como se muestra en las siguientes pantallas:

Para instalar MS SQL Server 2014, se descarga el archivo desde la web de Microsoft y se ejecuta el archivo de instalación.

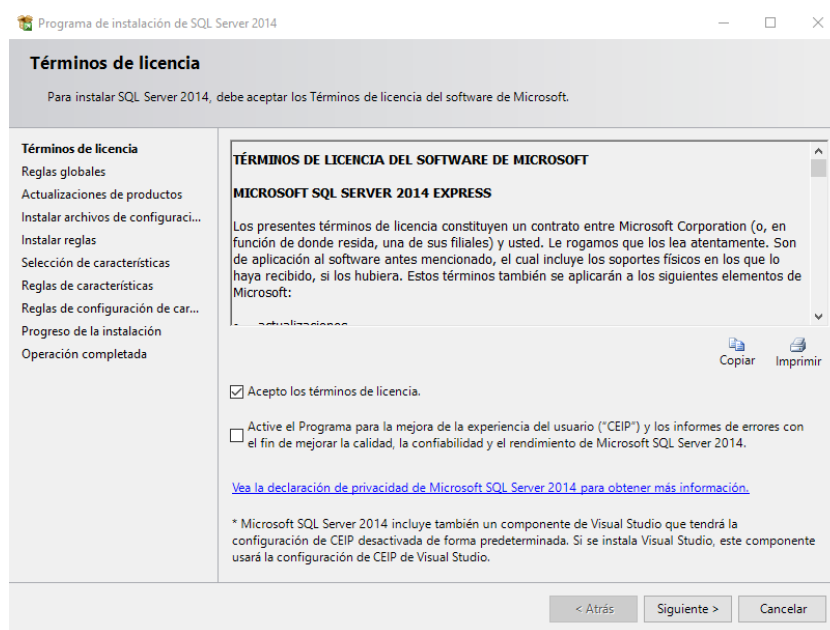
- Comenzará preguntando por la carpeta temporal donde se extraerán los archivos de instalación:



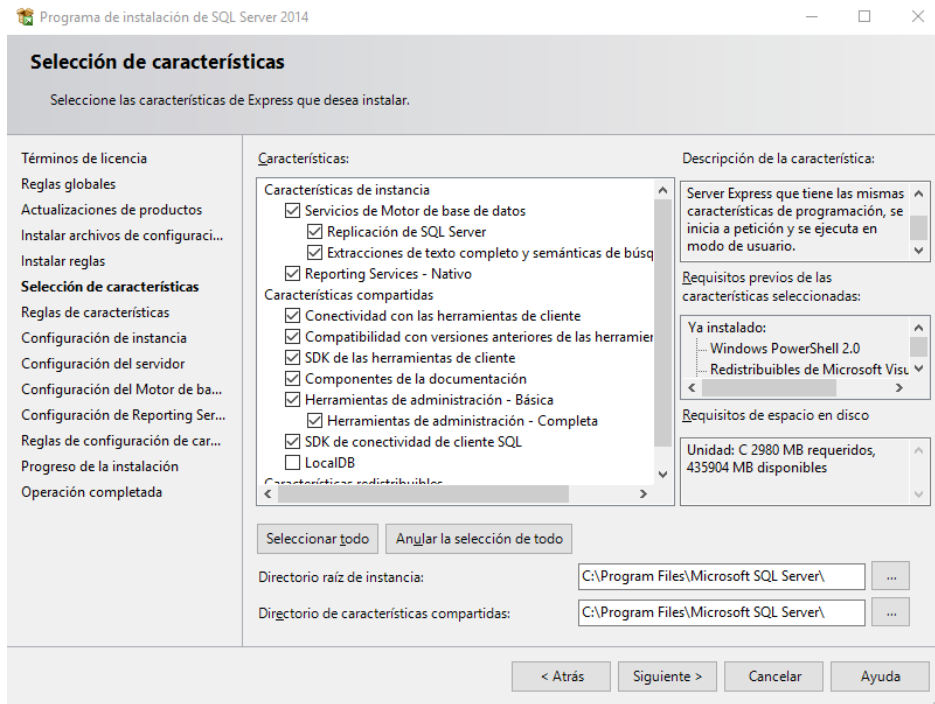
- Se selecciona una nueva instalación independiente:



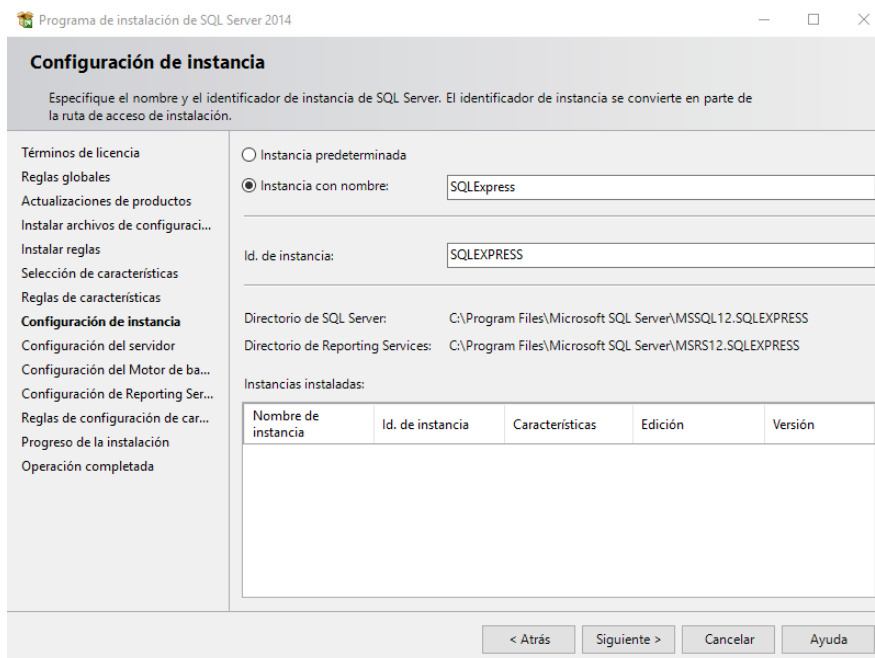
- Se aceptan los términos de licencia:



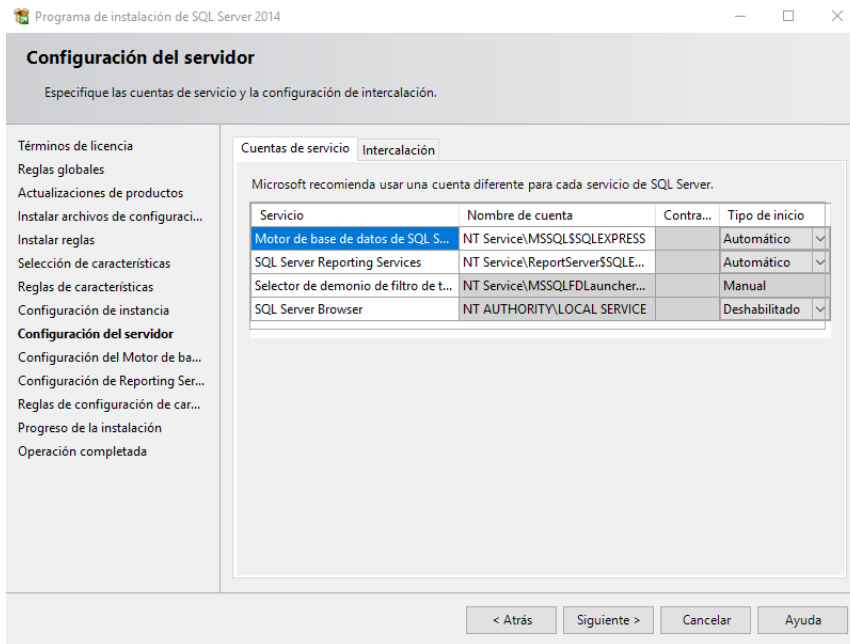
- Se seleccionan las características a instalar:



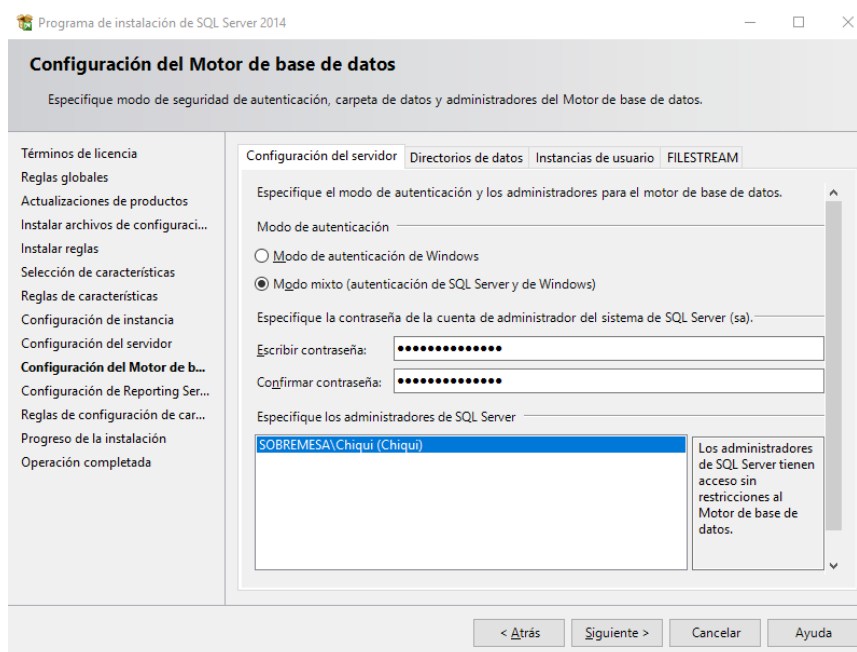
Se deja el nombre de instancia que viene por defecto, pues no afecta al proyecto:



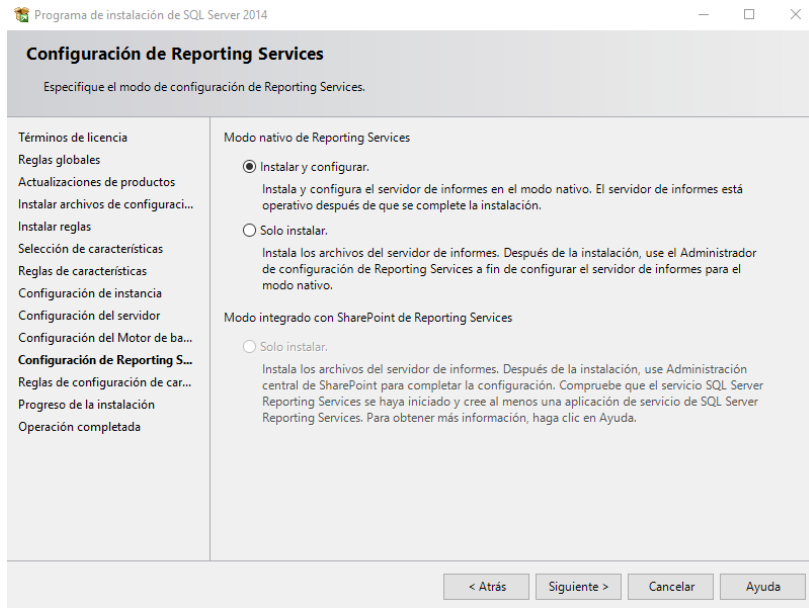
- Se dejan las cuentas de servicio que vienen por defecto, ya que son diferentes para cada uno de los servicios, lo que aporta mayor seguridad:



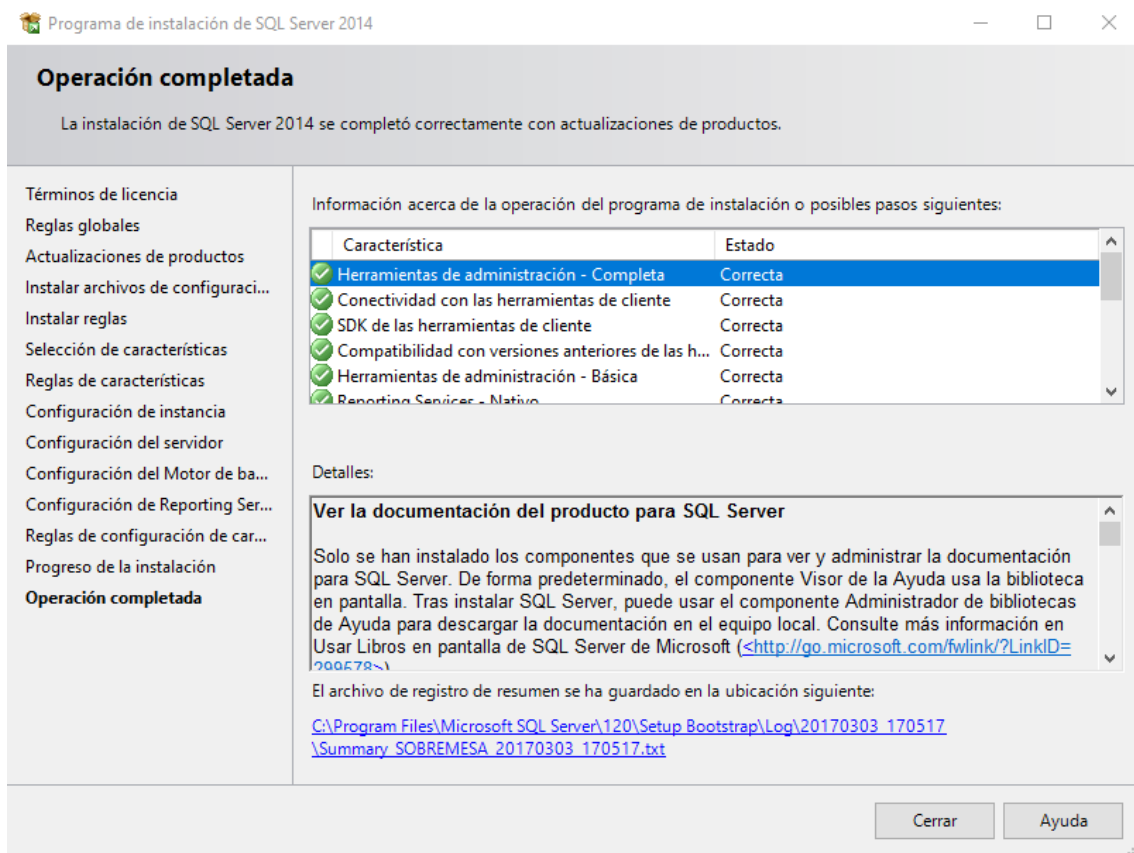
- Se selecciona el modo mixto, ya que así lo requiere Pentaho y se introduce una contraseña segura:



- Se instala *reporting services*:



- Se comprueba la correcta instalación

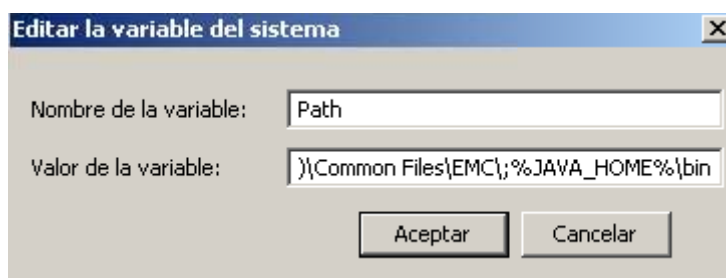


Descargar y desempaquetar Pentaho Community

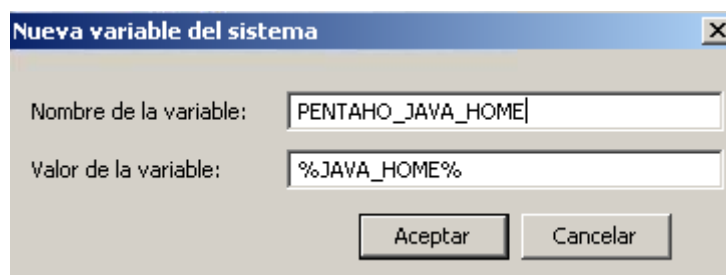
Se debe descargar el archivo de la web de Pentaho Community y descomprimirlo en una carpeta. En este caso, la carpeta escogida es la que se recomendada por Pentaho y la que se decidió al crear la estructura de directorios.

Configurar las variables de entorno

- Se configuran las variables necesarias para el funcionamiento de Java:



- Se configura la variable de entorno necesaria para Pentaho:



[Volver](#) al documento principal.

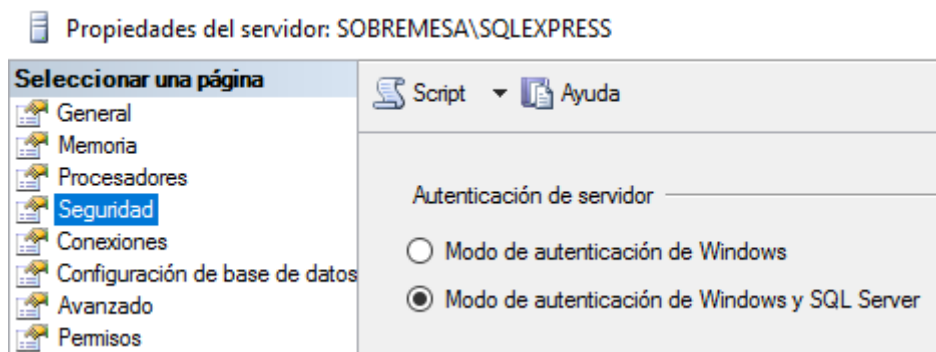
Anexo 2. Configurar e inicializar MS SQL Server como la Base de Datos del repositorio de Pentaho

El repositorio de Pentaho residirá en la Base de Datos de Microsoft SQL Server, que constará de tres bases de datos:

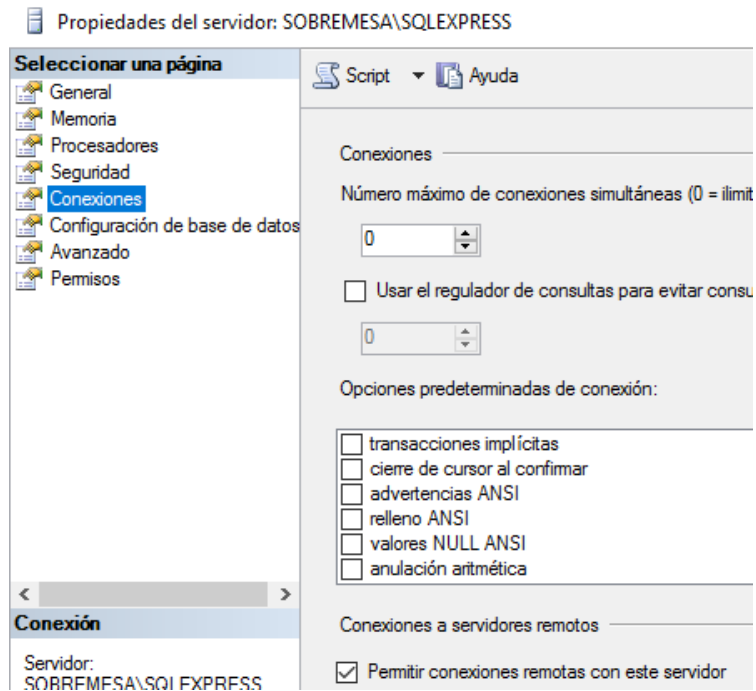
- *Jackrabbit*, que contiene la solución del repositorio. Por ejemplo, los datos a utilizar en los *reports*.
- *Quartz*, el cual contiene los datos que están relacionados con la programación de trabajos.
- *Hibernate*, que contiene datos relacionados con auditorías.
- *Operations Mart* (Opcional), el cual emite informes sobre el uso del sistema y el rendimiento del mismo. Esta base de datos no se utilizará en el presente trabajo.

Ajustar la configuración de MS SQL Server

- Configurar en MS SQL Management Studio la autenticación mixta de SQL Server y Windows:



Comprobar que la base de datos admite conexiones remotas

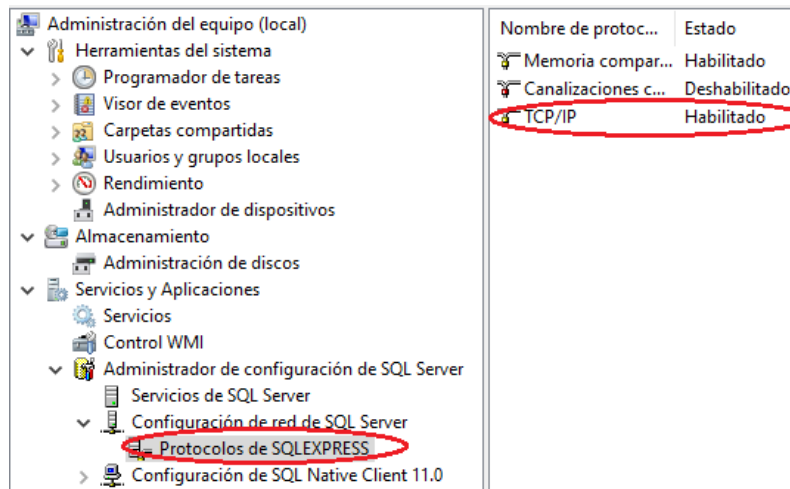


Comprobar que SQL está correctamente configurado para recibir conexiones remotas

- Se comprueba que están en ejecución los servicios SQL Server y SQL Server Browser. Este último no viene iniciado por defecto, por lo que habrá que establecer el modo de inicio en automático:

Nombre	Estado	Modo de inicio
SQL Server (SQLE...)	En ejecución	Automático
SQL Full-text Filte...	En ejecución	Manual
SQL Server Repor...	En ejecución	Automático
Agente SQL Serv...	Detenido	Otro (arranque, sist...
SQL Server Browser	En ejecución	Automático

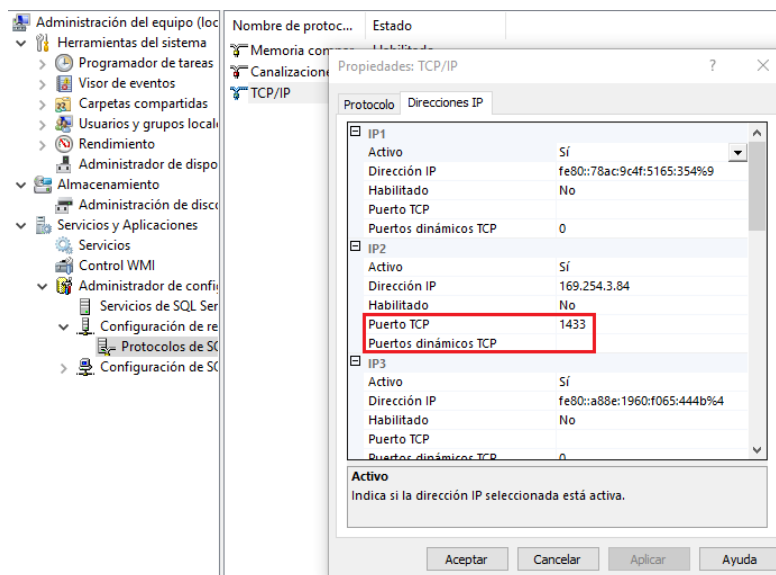
- Se comprueba que el protocolo TCP/IP está habilitado para SQL:



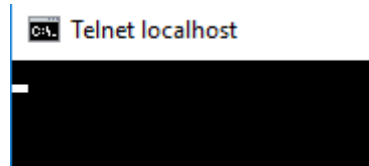
- Comprobar que esté abierto el puerto 1433:

```
C:\Users\Chiqui>telnet localhost 1433
Conectándose a localhost...No se puede abrir la conexión al host, en puerto 1433: Error en la conexión
```

- Se comprueba que el puerto no está abierto, por lo que se hace necesario modificar la configuración de TCP/IP para SQL:



- Como se ha tenido que iniciar algún servicio y modificar el protocolo TCP/IP, será necesario reiniciar el servicio SQLEXPRESS para comprobar que telnet ya conecta con el puerto 1433:



Cambiar en Pentaho las contraseñas que vienen por defecto

Con la finalidad de hacer el entorno más seguro, se puede cambiar la password en los siguientes archivos con la ayuda de un editor de textos:

- C:\pentaho\server\pentaho-server\data\mysql5\create_jcr_mysql.sql
- C:\pentaho\server\pentaho-server\data\mysql5\create_quartz_mysql.sql
- C:\pentaho\server\pentaho-server\data\mysql5\create_repository_mysql.sql

Sin embargo, se mantendrá la contraseña por defecto, ya que se trata de un trabajo con el objetivo de aprender, lo cual nos ahorrará tiempo en la configuración. No obstante, hay que tener en cuenta que, para un proyecto en producción, será necesario establecer diferentes contraseñas en cada uno de los repositorios, ya que de este factor dependerá la seguridad y confidencialidad de los datos que contenga.

Ejecutar los scripts de SQL para crear los repositorios de Pentaho

Ejecutar en MS SQL Server Management Studio los scripts existentes en la carpeta de Pentaho C:\pentaho\server\pentaho-server\data\sqlserver. Dichos scripts crearán los repositorios de Pentaho:

- Se ejecuta el script para crear Quartz:

```
SQLQuery1.sql - POR...rtatil\chiqui (54))* X
--Begin--
USE master
IF EXISTS(select * from sys.databases where name = N'quartz')
  DROP DATABASE quartz
GO
CREATE DATABASE quartz
GO
IF NOT EXISTS
  (SELECT name
   FROM master.sys.server_principals
   WHERE name = N'pentaho_user')
  CREATE LOGIN pentaho_user WITH PASSWORD = N'password', CHECK_POLICY = OFF
GO
```

100 % Mensajes

(1 filas afectadas)

(1 filas afectadas)

(1 filas afectadas)

(1 filas afectadas)

(1 filas afectadas)

- Se ejecuta el script para crear el repositorio Hibernate:

```
SQLQuery1.sql - POR...rtatil\chiqui (54))* X
--Begin--
USE master
IF EXISTS(select * from sys.databases where name = N'hibernate')
  DROP DATABASE hibernate
GO
CREATE DATABASE hibernate
GO
IF NOT EXISTS
  (SELECT name
   FROM master.sys.server_principals
   WHERE name = N'hibuser')
  CREATE LOGIN hibuser WITH PASSWORD = N'password', CHECK_POLICY = OFF
GO
USE hibernate;
CREATE USER hibuser FOR LOGIN hibuser
EXEC sp_addrolemember N'db owner', N'hibuser'
```

100 % Mensajes

Comandos completados correctamente.

- Se ejecuta el script para crear el repositorio Jackrabbit:

```
SQLQuery1.sql - POR...rtati\chiqui (54)* X
USE master
IF EXISTS(select * from sys.databases where name = N'jackrabbit')
  DROP DATABASE jackrabbit
GO
CREATE DATABASE jackrabbit
GO

IF NOT EXISTS
  (SELECT name
   FROM master.sys.server_principals
   WHERE name = N'jcr_user')
  CREATE LOGIN [jcr_user] WITH PASSWORD = N'password', CHECK_POLICY = OFF
GO

USE jackrabbit;
CREATE USER jcr_user FOR LOGIN jcr_user
EXEC sp addrolemember N'db owner', N'jcr user'
```

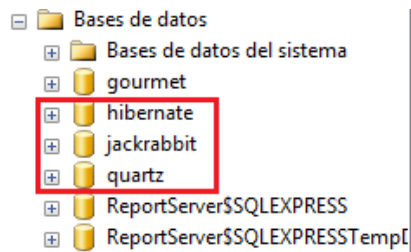
100 %

Mensajes

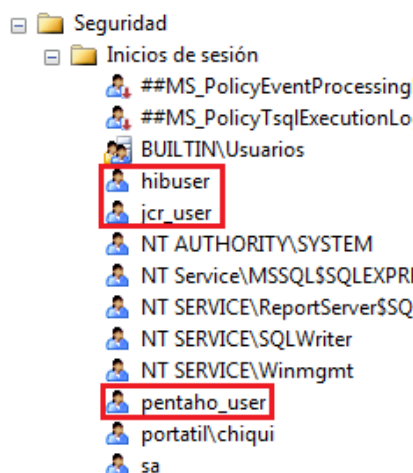
Comandos completados correctamente.

Verificar la correcta inicialización de MS SQL SERVER

- Se comprueba que se han creado las bases de datos que actuarán como repositorios de Pentaho:



- Se comprueba que se han creado los usuarios necesarios para Pentaho:



Configurar los repositorios de Pentaho

Una vez inicializada la base de datos del repositorio, será necesario configurar Quartz, Hibernate y Jackrabbit para MS SQL Server.

Configurar Quartz

Se especifica dónde se almacenan los trabajos. Para eso, será necesario abrir el siguiente archivo con un editor de textos y realizar una serie de cambios:

pentaho/server/pentaho-server/pentaho-solutions/system/quartz/quartz.properties

- Localizar la sección “#_replace_jobstore_properties” y establecer el siguiente valor:

```
org.quartz.jobStore.driverDelegateClass = org.quartz.impl.jdbcjobstore.MSSQLDelegate
```

```
#_replace_jobstore_properties
org.quartz.jobStore.misfireThreshold = 60000
#org.quartz.jobStore.driverDelegateClass = org.quartz.impl.jdbcjobstore.PostgreSQLDelegate
org.quartz.jobStore.driverDelegateClass = org.quartz.impl.jdbcjobstore.MSSQLDelegate
org.quartz.jobStore.useProperties = false
org.quartz.jobStore.dataSource = myDS
org.quartz.jobStore.tablePrefix = QRTZ_
org.quartz.jobStore.isClustered = false
```

- Localizar la sección “# Configure Datasources” y establecer el siguiente valor (valor configurado por defecto):

```
org.quartz.dataSource.myDS.jndiURL = Quartz
```

Configurar Hibernate

Se configura dónde se encuentra el archivo de configuración del repositorio Hibernate. Para eso, habrá que acceder al siguiente directorio:

pentaho/server/pentaho-server/pentaho-solutions/system/hibernate

- Abrir con un editor de textos el archivo “hibernate-settings.xml”. Buscar la etiqueta <config-file> y realizar el siguiente cambio:

Desde:

```
<config-file>system/hibernate/hsqldb.hibernate.cfg.xml</config-file>
```

Hasta:

```
<config-file>system/hibernate/sqlserver.hibernate.cfg.xml</config-file>
```

```
<!-- <config-file>system/hibernate/hsql.hibernate.cfg.xml</config-file> -->
<config-file>system/hibernate/sqlserver.hibernate.cfg.xml</config-file|
```

- Abrir con un editor de textos el archivo “sqlserver.hibernate.cfg.xml”. Comprobar que la contraseña y puerto coinciden con la configuración establecida:

```
<!-- SQL Server Configuration -->
<property name="connection.driver_class">com.microsoft.sqlserver.jdbc.SQLServerDriver</property>
<property name="connection.url">jdbc:sqlserver://localhost:1433;DatabaseName=hibernate</property>
<property name="dialect">org.hibernate.dialect.SQLServerDialect</property>
<property name="connection.username">hibuser</property>
<property name="connection.password">password</property>
```

Reemplazar el archivo de auditoría por defecto por el de MS SQL Server

Habrá que reemplazar el archivo de auditoría por uno que esté configurado para MS SQL Server.

Para eso, hay que copiar el archivo:

“pentaho-solutions/system/dialects/sqlserver/audit_sql.xml”

En el directorio

“pentaho-solutions/system”.

Configurar Jackrabbit

Es necesario cambiar algunas partes del código para cambiar el repositorio de Jackrabbit por defecto al de MS SQL Server.

- Abrir con un editor de textos el archivo “pentaho/server/pentaho-server/pentaho-solutions/system/jackrababbit/repository.xml” y realizar los cambios indicados en la siguiente tabla, quitando los comentarios de las líneas que hacen referencia a MS SQL Server y comentando las líneas que hacen referencia a otras bases de datos:

Elemento	Código a modificar
Repository	<pre> 1 <FileSystem class="org.apache.jackrabbit.core.fs.db.MSSqlFileSystem"> 2 <param name="driver" value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/> 3 <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/> 4 ... 5 <param name="schema" value="mssql"/> 6 </FileSystem> </pre>
DataStore	<pre> 1 <DataStore class="org.apache.jackrabbit.core.data.db.DbDataStore"> 2 <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/> 3 ... 4 <param name="schema" value="mssql"/> 5 </DataStore> </pre>
WorkSpaces	<pre> 1 <FileSystem class="org.apache.jackrabbit.core.fs.db.MSSqlFileSystem"> 2 <param name="driver" value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/> 3 <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/> 4 ... 5 <param name="schema" value="mssql"/> 6 </FileSystem> </pre>
PersistenceManager	<pre> 1 <PersistenceManager class="org.apache.jackrabbit.core.persistence.bundle.MSSqlPersistenc eManager"> 2 <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/> 3 ... 4 <param name="schema" value="mssql"/> 5 </PersistenceManager> </pre>
Versioning	<pre> 1 <FileSystem class="org.apache.jackrabbit.core.fs.db.MSSqlFileSystem"> 2 <param name="driver" value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/> 3 <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/> 4 ... 5 <param name="schema" value="mssql"/> 6 </FileSystem> </pre>
PersistenceManager (Parte 2)	<pre> 1 <PersistenceManager class="org.apache.jackrabbit.core.persistence.bundle.MSSqlPersistenc eManager"> 2 <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/> 3 ... 4 <param name="schema" value="mssql"/> 5 </PersistenceManager> </pre>

Se quita el comentario del apartado Repository:

```
<FileSystem class="org.apache.jackrabbit.core.fs.db.MSSqlFileSystem">
  <param name="driver" value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/>
  <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/>
  <param name="user" value="jcr_user"/>
  <param name="password" value="password"/>
  <param name="schema" value="mssql"/>
  <param name="schemaObjectPrefix" value="fs_repos_"/>
</FileSystem>
```

- Se quita el comentario del apartado DataStore:

```
<DataStore class="org.apache.jackrabbit.core.data.db.DbDataStore">
  <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/>
  <param name="driver" value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/>
  <param name="user" value="jcr_user"/>
  <param name="password" value="password"/>
  <param name="databaseType" value="mssql"/>
  <param name="minRecordLength" value="1024"/>
  <param name="maxConnections" value="3"/>
  <param name="copyWhenReading" value="true"/>
  <param name="tablePrefix" value="" />
  <param name="schemaObjectPrefix" value="ds_repos_"/>
</DataStore>
```

- Se quita el comentario del apartado Workspaces:

```
<FileSystem class="org.apache.jackrabbit.core.fs.db.MSSqlFileSystem">
  <param name="driver" value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/>
  <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/>
  <param name="user" value="jcr_user"/>
  <param name="password" value="password"/>
  <param name="schema" value="mssql"/>
  <param name="schemaObjectPrefix" value="fs_ws_"/>
</FileSystem>
```

- Se quita el comentario del apartado PersistenceManager (Parte 1):

```
<PersistenceManager class="org.apache.jackrabbit.core.persistence.bundle.MSSqlPersistenceManager">
  <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/>
  <param name="driver" value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/>
  <param name="user" value="jcr_user"/>
  <param name="password" value="password"/>
  <param name="schema" value="mssql"/>
  <param name="schemaObjectPrefix" value="{wsp.name}_pm_ws_"/>
</PersistenceManager>
```

- Se quita el comentario del apartado Versioning:

```
<FileSystem class="org.apache.jackrabbit.core.fs.db.MSSqlFileSystem">
  <param name="driver" value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/>
  <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/>
  <param name="user" value="jcr_user"/>
  <param name="password" value="password"/>
  <param name="schema" value="mssql"/>
  <param name="schemaObjectPrefix" value="fs_ver_"/>
</FileSystem>
```

- Se quita el comentario del apartado PersistenceManager (Parte 2):

```
<PersistenceManager class="org.apache.jackrabbit.core.persistence.bundle.MSSqlPersistenceManager">
  <param name="url" value="jdbc:sqlserver://localhost:1433;DatabaseName=jackrabbit"/>
  <param name="driver" value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/>
  <param name="user" value="jcr_user"/>
  <param name="password" value="password"/>
  <param name="schema" value="mssql"/>
  <param name="schemaObjectPrefix" value="pm_ver_"/>
</PersistenceManager>
```

Ejecutar las tareas de conexión para Tomcat

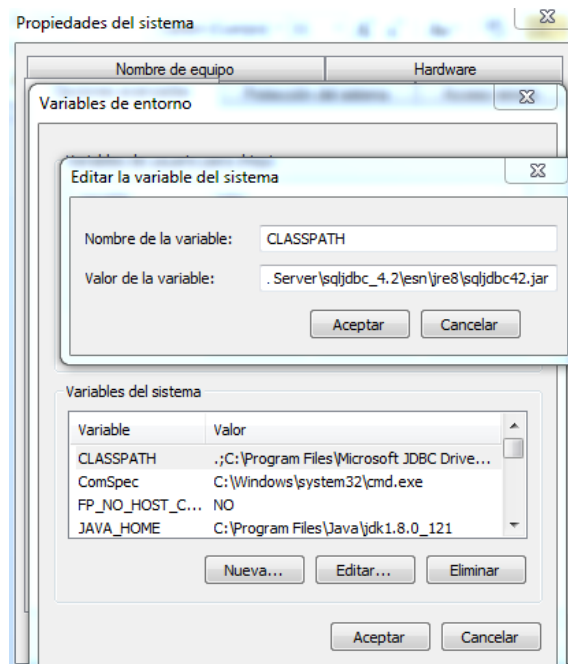
Una vez configurado el repositorio es necesario configurar el servidor Web para que se pueda conectar al repositorio de Pentaho. Para eso, se instalará Java DataBase Connectivity (JDBC), que se trata de la API que permite ejecutar operaciones sobre bases de datos desde el lenguaje java, independientemente del sistema operativo o de la base de datos que se utilice.

Configurar los drivers para Pentaho

- Descargar JDBC versión 4.26, ya que es el driver recomendado para JDK 8 y JRE8. A continuación, descomprimir el archivo en la ruta recomendada por Microsoft:

```
%Program Files%\Microsoft JDBC Driver 6.0 for SQL Server"
```

- Añadir JDBC en la variable de entorno CLASSPATH con la ruta del archivo sqljdbc43.jar:



- Copiar el archivo “sqljdbc42.jar” en el directorio “pentaho/server/pentaho-server/tomcat/lib”.
- Asegurarse de que existe el archivo “hsqldb-2.3.2.jar” en el directorio “pentaho-server/tomcat/lib”.

Modificar la información de la conexión JDBC en Tomcat

En el fichero “pentaho-server/tomcat/webapps/pentaho/META-INF/content.xml” de Tomcat se encuentra la información de la conexión a la base de datos y a la red (usuario, contraseña, dirección IP, dominio, puertos, etc.), por lo que será necesario comprobar que todo cuenta con los mismos valores que se han configurado en el sistema:

```
<Resource validationQuery="select 1" url="jdbc:sqlserver://localhost:1433;DatabaseName=hibernate"
driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver"
password="password" username="hibuser" maxWaitMillis="10000" maxIdle="5" maxTotal="20"
factory="org.apache.commons.dbcp.BasicDataSourceFactory" type="javax.sql.DataSource" auth="Container" name="jdbc/Hibernate"/>
```

- Incluir el siguiente código (modificando los parámetros si es necesario) y comentar las referencias a otras bases de datos:

```
<Resource validationQuery="select 1"
url="jdbc:sqlserver://localhost:1433;DatabaseName=hibernate"
driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver" password="password"
username="hibuser" maxWaitMillis="10000" maxIdle="5" maxTotal="20"
factory="org.apache.commons.dbcp.BasicDataSourceFactory" type="javax.sql.DataSource"
auth="Container" name="jdbc/Hibernate"/>
```

```
<Resource validationQuery="select 1"
url="jdbc:sqlserver://localhost:1433;DatabaseName=hibernate"
driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver" password="password"
username="hibuser" maxWaitMillis="10000" maxIdle="5" maxTotal="20"
factory="org.apache.commons.dbcp.BasicDataSourceFactory" type="javax.sql.DataSource"
auth="Container" name="jdbc/Audit"/>
```

```
<Resource validationQuery="select 1"
url="jdbc:sqlserver://localhost:1433;DatabaseName=quartz"
driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver" password="password"
username="pentaho_user" maxWaitMillis="10000" maxIdle="5" maxTotal="20"
factory="org.apache.commons.dbcp.BasicDataSourceFactory" type="javax.sql.DataSource"
auth="Container" name="jdbc/Quartz"/>
```

```
<Resource validationQuery="select 1"
url="jdbc:sqlserver://localhost:1433;DatabaseName=hibernate"
driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver" password="password"
username="hibuser" maxWaitMillis="10000" maxIdle="5" maxTotal="20"
factory="org.apache.commons.dbcp.BasicDataSourceFactory" type="javax.sql.DataSource"
auth="Container" name="jdbc/pentaho_operations_mart"/>
```

```
<Resource validationQuery="select 1"
url="jdbc:sqlserver://localhost:1433;DatabaseName=hibernate"
driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver" password="password"
username="hibuser" maxWaitMillis="10000" maxIdle="5" maxTotal="20"
factory="org.apache.commons.dbcp.BasicDataSourceFactory" type="javax.sql.DataSource"
auth="Container" name="jdbc/PDI_Operations_Mart"/>
```

Iniciar el servidor Pentaho

Ejecutar el script start-pentaho.bat para iniciar el servidor.

Desde el propio servidor se puede abrir un navegador web e introducir la URL del servidor:

<http://localhost:8080/pentaho>

También se puede hacer este paso desde otro ordenador de la misma red:

<http://portatil:8080/pentaho>

[Volver](#) al documento principal.

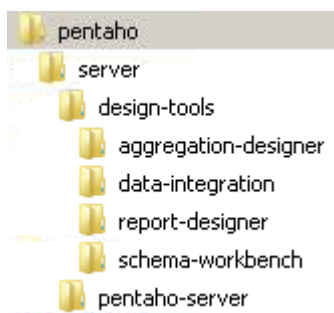
Anexo 3. Instalar otras aplicaciones principales y de desarrollo accesorias a Pentaho

Sobre la plataforma Pentaho Business Analytics se pueden instalar otras aplicaciones que mejorarán la inteligencia de negocio:

- *Business Analytics Platform*: Se trata de la plataforma principal de Pentaho que ya se ha instalado. Es una plataforma que permitirá a los usuarios de la organización poder mezclar datos y utilizar herramientas de análisis sin necesidad de tener que recurrir al departamento de tecnologías de la información.
- *Data Integration*: Aplicación utilizada para la extracción, transformación y carga de datos (ETL, de sus siglas en inglés Extract, Transform and Load), conocida en Pentaho como Kettle.
- *Report Designer*: Es una herramienta utilizada para generar informes interactivos, los cuales podrán ser exportados a diferentes formatos como PDF, Excel, etc.
- *Schema Workbench*: Interfaz que permite crear cubos OLAP y cruzar datos para presentarlos de forma multidimensional.

Crear la estructura de directorios e instalar las aplicaciones

Se creará la estructura de directorios recomendada en Pentaho para albergar las herramientas:



Se descomprimirán los archivos en un directorio temporal y se copiarán a su ubicación definitiva.

Así, las aplicaciones que se instalarán son:

- *Pentaho aggregation designer.*
- *Pentaho Data Integration.*
- *Pentaho report designer.*
- *Pentaho Schema Workbench.*

Configurar *Data Integration* para conexión con MS SQL Server

Habrá que copiar dos archivos para que funcione la conexión JDBC entre Pentaho Data Integration y MS SQL Server:

- Copiar sqljdbc_auth.dll

Desde

%Program Files%\Microsoft JDBC Driver 4.2 for SQL Server\sqljdbc_4.2\esn\auth\x64\

Hasta

%Program Files%\java\jre1.8.0_121\bin

- Copiar el archivo sqljdbc42.jar

Desde

%Program Files%\Microsoft JDBC Driver 4.2 for SQL Server\sqljdbc_4.2\esn\jre8

Hasta

Pentaho\design-tools\data-integration\lib

Para ejecutar *Data Integration* se hará doble clic en el archivo spoon.bat

Configurar *Schema Workbench* para conexión con MS SQL Server

- Copiar el archivo sqljdbc42.jar

Desde

%Program Files%\Microsoft JDBC Driver 4.2 for SQL Server\sqljdbc_4.2\esn\jre8

Hasta

Pentaho\design-tools\schema-workbench\drivers

Para ejecutar *Workbench* se hará doble clic en el archivo workbench.bat

Configurar *Report designer* para conexión con MS SQL Server

- Copiar el archivo sqljdbc42.jar

Desde

%Program Files%\Microsoft JDBC Driver 4.2 for SQL Server\sqljdbc_4.2\esn\jre8

Hasta

Pentaho\design-tools\report-designer\lib\jdbc

Para ejecutar *Report Designer* se hará doble clic en el archivo report-designer.bat

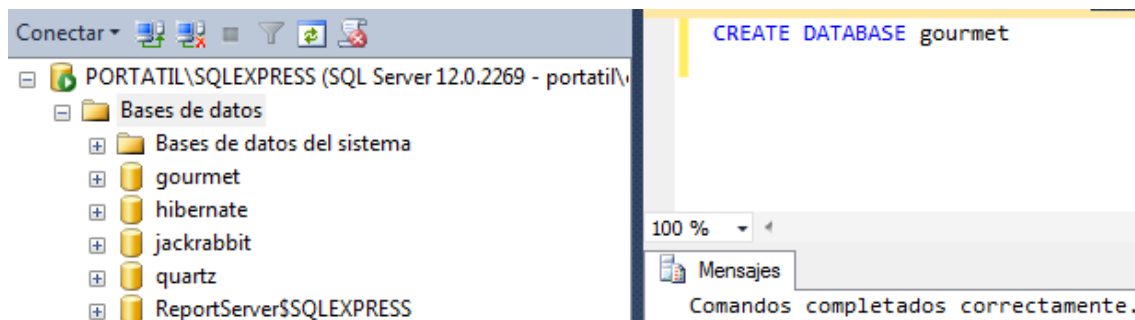
[Volver](#) al documento principal.

Anexo 4. Procesos ETL con Kettle

A continuación, se detallan los pasos realizados en la creación de la base de datos que se utilizará como *data warehouse* y que servirá para alimentar a los modelos de análisis que se vayan realizando:

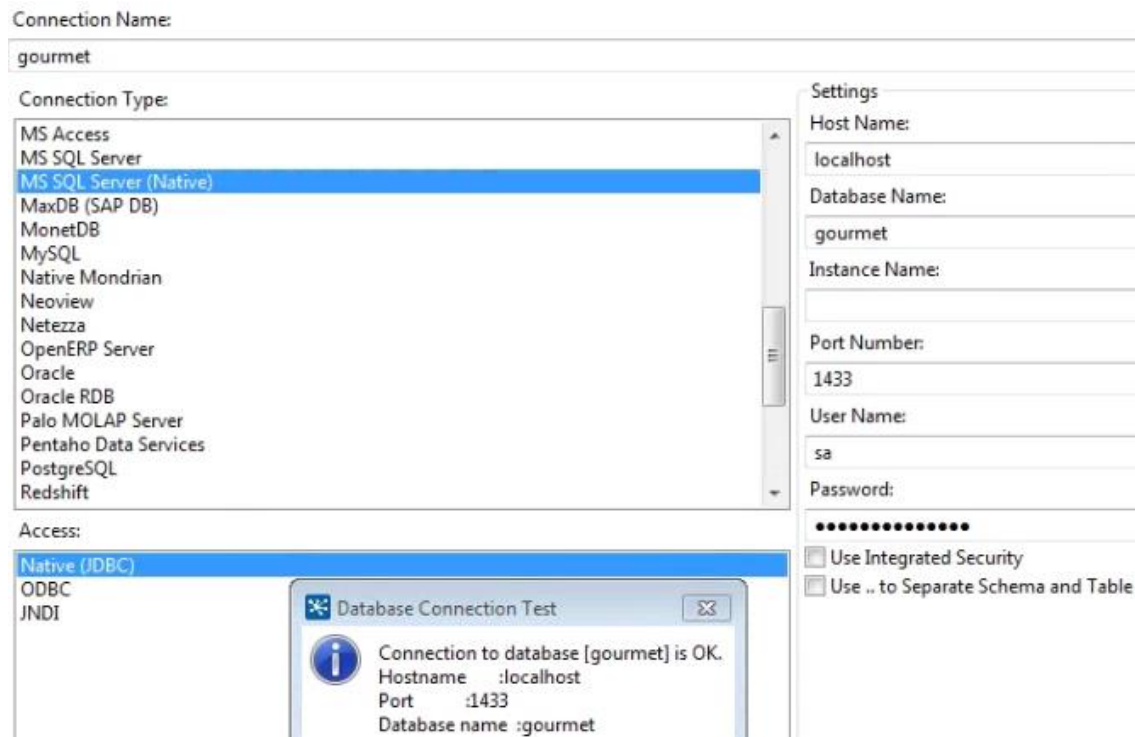
Creación de la base de datos desde MS SQL Server

En primer lugar, se creará la base de datos sobre la que se va a trabajar:



1. Crear una conexión con la base de datos

A continuación, es necesario realizar desde *Data Integration*, una conexión contra la base de datos Gourmet, la cual reside en MS SQL Server:

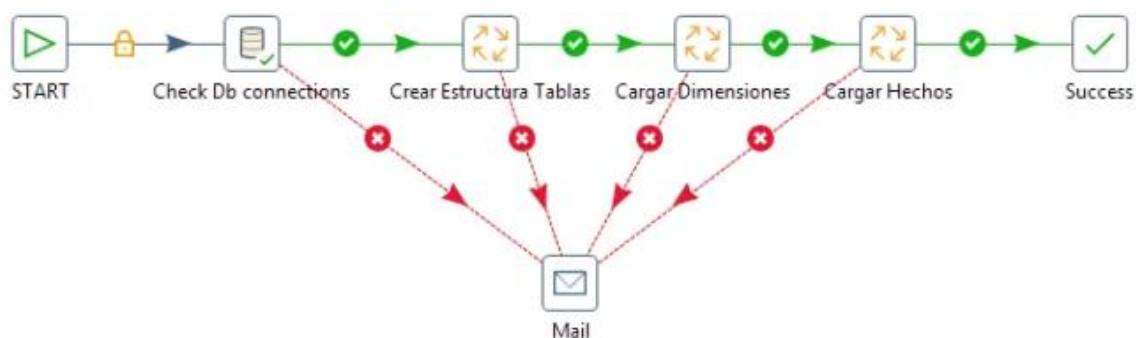


2. Trabajo Principal

Además, para la correcta ejecución del proceso ETL con Kettle, se ha creado un trabajo principal que realizará todo el proceso con las diferentes transformaciones, el cual dará como resultado el almacén de datos del proyecto. Dicho trabajo, realizará los siguientes pasos:

- Comprobar la conexión con la base de datos gourmet
- Llamar a un trabajo que crea la estructura de la base de datos
- Llamar a un trabajo para cargar los datos en las tablas de dimensiones
- Llamar a un trabajo para cargar los datos en las tablas de hechos

En caso de encontrar algún error durante el proceso, enviará un correo electrónico al responsable de ejecución de la tarea.



3. Trabajo Crear Estructura Tablas

El presente trabajo, crea la estructura de la base de datos, es decir, las tablas donde residirán los datos. No obstante, las dimensiones Tempo (`dim_fecha` y `dim_hora`) se han creado y rellenado de datos a través de un script en SQL en este paso. Por lo tanto, realiza las siguientes tareas:

- Ejecuta una transformación para crear y cargar la dimensión `dim_hora`
- Ejecuta una transformación para crear y cargar la dimensión `dim_fecha`
- Ejecuta un script para crear las tablas vacías, tanto de hechos como de dimensiones

Al igual que en el trabajo anterior, envía un correo al responsable en caso de ocurrir cualquier error.

3.1 Script para la creación de la dimensión dim_hora:

```

--Base de datos destino
use gourmet

--Crear Dimensión
create table DIM_HORA
(
    Hora smallint not null,
    constraint PKHORA PRIMARY KEY CLUSTERED (Hora asc )
)

--Declaraciones
DECLARE @HoraDesde as smallint, @HoraHasta as smallint
SET @HoraDesde = 1
SET @HoraHasta = 24

BEGIN TRANSACTION

WHILE (@HoraDesde <= @HoraHasta) BEGIN
    INSERT INTO gourmet.dbo.DIM_HORA
    (
        Hora
    ) VALUES

    (
        @HoraDesde
    )

    SELECT @HoraDesde = @HoraDesde + 1
END
COMMIT TRANSACTION

Script para la creación de la dimensión dim_fecha:
--Base de datos destino
use gourmet;

--Crear Dimensión
create table GOURMET.DBO.DIM_FECHA
(
    Fecha date not null,
    Año smallint not null,
    Mes smallint not null,
    Semana smallint not null,
    Dia smallint not null,
    DiaSemana smallint not null,
    NMes char(15) not null,
    NDia char(6) not null,
    NDiaSemana char(10) not null
    constraint PKFECHA PRIMARY KEY CLUSTERED (Fecha asc )
);

--Declaraciones
DECLARE @FechaDesde as DATE, @FechaHasta as DATE;
DECLARE @FechaAAAAMMDD int;

```

```

DECLARE @Año as smallint, @Mes smallint;
DECLARE @Semana smallint, @Dia smallint, @DiaSemana smallint;
DECLARE @NMes char(15), @NDia char(6), @NDiaSemana char(10);
SET DATEFORMAT dmy;
SET DATEFIRST 1;

BEGIN TRANSACTION
  --Rango de fechas a crear: del 01/01/2000 al 31/12/2000
  SELECT @FechaDesde = CAST('20000101' AS DATE);
  SELECT @FechaHasta = CAST('20001231' AS DATE);

  WHILE (@FechaDesde <= @FechaHasta) BEGIN
    SELECT @Año = DATEPART(yy, @FechaDesde);
    SELECT @Mes = DATEPART(m, @FechaDesde);
    SELECT @Semana = DATEPART(wk, @FechaDesde);
    SELECT @Dia = RIGHT('0' + DATEPART(dd, @FechaDesde), 2);
    SELECT @DiaSemana = DATEPART(DW, @FechaDesde);
    SELECT @NMes = DATENAME(mm, @FechaDesde);
    SELECT @NDia = CAST(@Dia as CHAR(2)) + ' ' + RTRIM(@NMes);
    SELECT @NDiaSemana = DATENAME(dw, @FechaDesde);

    INSERT INTO GOURMET.DBO.DIM_FECHA
    (
      Fecha,
      Año,
      Mes,
      Semana,
      Dia,
      DiaSemana,
      NMes,
      NDia,
      NDiaSemana
    ) VALUES

    (
      @FechaDesde,
      @Año,
      @Mes,
      @Semana,
      @Dia,
      @DiaSemana,
      @NMes,
      @NDia,
      @NDiaSemana
    );

    SELECT @FechaDesde = DATEADD(DAY, 1, @FechaDesde);
  END;
  COMMIT TRANSACTION;

```

3.2 Script para la creación de las tablas:

```

use gourmet
-----
--CREAR LA TABLA DIM_TIENDA
-----
CREATE TABLE gourmet.dbo.DIM_TIENDA
(
  NOMBRE VARCHAR(100) not null
, DIRECCION VARCHAR(100)
, SUPERFICIE DECIMAL(5,2)
, FORMATO_TIENDA VARCHAR(100)
, PAIS VARCHAR(100)
, EXTENSIONPAIS BIGINT
, POBLACIONPAIS BIGINT
);

-- Clave Primaria DIM_TIENDA
ALTER TABLE gourmet.dbo.DIM_TIENDA
ADD CONSTRAINT PKTIENDA PRIMARY KEY (NOMBRE);

-----
--CREAR LA TABLA DIM_CLIENTE
-----
CREATE TABLE gourmet.dbo.DIM_CLIENTE
(
  CODCLIENTE VARCHAR(10) not null
, NOMBRECLIENTE VARCHAR(100)
, SEXO VARCHAR(25)
, FECHANACIMIENTO DATE
, ESTADOCIVIL VARCHAR(25)
, PROFESION VARCHAR(100)
, NUMEROHIJOS INT
, NACIONALIDAD VARCHAR(100)
, TOTALCOMPRAS INT
, PUNTOSACUMULADOS INT
);

-- Clave Primaria DIM_CLIENTE
ALTER TABLE gourmet.dbo.DIM_CLIENTE
ADD CONSTRAINT PKCLIENTE PRIMARY KEY (CODCLIENTE);

-----
--CREAR LA TABLA DIM_PRODUCTO
-----
CREATE TABLE gourmet.dbo.DIM_PRODUCTO
(
  CODPRODUCTO VARCHAR(25) not null
, DESCRIPCION VARCHAR(100)
, NOMBREPAIS VARCHAR(100)
, COSTE DECIMAL(5,2)
, PRECIOVENTA DECIMAL(5,2)
, NOMBRESUBFAMILIA VARCHAR(100)
, MARCA VARCHAR(100)
, NOMBREPROVEEDOR VARCHAR(100)
);

```



```

-- Clave Primaria DIM_PRODUCTO
ALTER TABLE gourmet.dbo.DIM_PRODUCTO
ADD CONSTRAINT PKPRODUCTO PRIMARY KEY (CODPRODUCTO);

-----
--CREAR LA TABLA H_TICKET
-----
CREATE TABLE gourmet.dbo.H_TICKET
(
  CODVENTA VARCHAR(25) not null
, NOMBRETIENDA VARCHAR(100) not null
, FECHA DATE not null
, HORA SMALLINT not null
, CODCLIENTE VARCHAR(10) not null
, IMPORTETOTAL INT
, TOTALUNIDADES INT
, PUNTOSTICKET INT
);

-- Clave Primaria H_TICKET
ALTER TABLE gourmet.dbo.H_TICKET
ADD CONSTRAINT PKTICKET PRIMARY KEY (CODVENTA);

--Claves foráneas de H_TICKET
ALTER TABLE gourmet.dbo.H_TICKET
ADD CONSTRAINT FKTICKET_TIENDA FOREIGN KEY (NOMBRETIENDA)
REFERENCES gourmet.dbo.DIM_TIENDA (NOMBRE);

ALTER TABLE gourmet.dbo.H_TICKET
ADD CONSTRAINT FKTICKET_CLIENTE FOREIGN KEY (CODCLIENTE)
REFERENCES gourmet.dbo.DIM_CLIENTE (CODCLIENTE);

ALTER TABLE gourmet.dbo.H_TICKET
ADD CONSTRAINT FKTICKET_FECHA FOREIGN KEY (FECHA)
REFERENCES gourmet.dbo.DIM_FECHA (FECHA);

ALTER TABLE gourmet.dbo.H_TICKET
ADD CONSTRAINT FKTICKET_HORA FOREIGN KEY (HORA)
REFERENCES gourmet.dbo.DIM_HORA (HORA);

-----
--CREAR LA TABLA H_PEDIDO
-----
CREATE TABLE gourmet.dbo.H_PEDIDO
(
  CODPEDIDO VARCHAR(25) not null
, NOMBRETIENDA VARCHAR(100) not null
, CODPRODUCTO VARCHAR(25) not null
, PRECIOCOMPRA DECIMAL(8,4)
, CANTIDAD SOLICITADA INT
, FECHASOLICITUD DATE not null
, CANTIDAD ENTREGADA INT
, FECHA ENTREGA DATE not null
);

```

```

-- Clave Primaria H_PEDIDO
ALTER TABLE gourmet.dbo.H_PEDIDO
ADD CONSTRAINT PKPEDIDO PRIMARY KEY (CODPEDIDO);

--Claves foráneas de H_PEDIDO
ALTER TABLE gourmet.dbo.H_PEDIDO
ADD CONSTRAINT FKPEDIDO_TIENDA FOREIGN KEY (NOMBRETIENDA)
REFERENCES gourmet.dbo.DIM_TIENDA (NOMBRE);

ALTER TABLE gourmet.dbo.H_PEDIDO
ADD CONSTRAINT FKPEDIDO_PRODUCTO FOREIGN KEY (CODPRODUCTO)
REFERENCES gourmet.dbo.DIM_PRODUCTO (CODPRODUCTO);

ALTER TABLE gourmet.dbo.H_PEDIDO
ADD CONSTRAINT FKPEDIDO_FECHASOLICITUD FOREIGN KEY
(FECHASOLICITUD)
REFERENCES gourmet.dbo.DIM_FECHA (FECHA);

ALTER TABLE gourmet.dbo.H_PEDIDO
ADD CONSTRAINT FKPEDIDO_FECHAENTREGA FOREIGN KEY (FECHAENTREGA)
REFERENCES gourmet.dbo.DIM_FECHA (FECHA);

```

4. Cargar Dimensiones

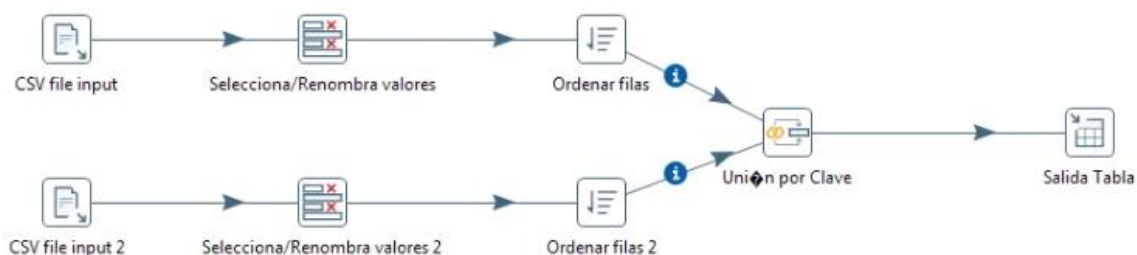
En el presente trabajo, se cargan los datos en las diferentes dimensiones:

- Cargar los datos en la dimensión dim_cliente
- Cargar los datos en la dimensión dim_tienda
- Cargar los datos en la dimensión dim_producto

A continuación se muestran las citadas transformaciones:

4.1 Cargar los datos en la dimensión DIM_TIENDA ([volver](#))

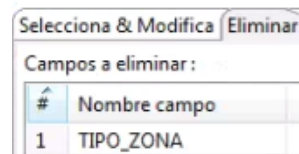
Se crea una transformación a partir del fichero tienda.csv y del fichero país.csv, se modifica el nombre de algunos campos, se ordenan las filas por nombre del país, ya que será la clave de unión JOIN y se guardan los datos en una tabla de MS SQL Server Express:



- Se obtienen los datos desde tienda.csv:

#	Name	Type	Format	Length	Precision	Currency	Decimal
1	NOMBRE	String		100			
2	DIRECCION	String		100			
3	SUPERFICIE	Number	+##,###	5	2		.
4	FORMATO_TIENDA	String		100			
5	PAIS	String		100			
6	TIPO_ZONA	String		100			

- Se elimina el campo tipo_zona:

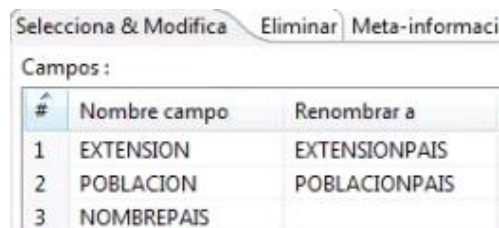


- Se obtienen los datos desde país.csv:

#	Name	Type	Format	Length	Precision	Currer
1	NOMBREPAIS	String		100		
2	EXTENSION	BigNumber		10	0	
3	POBLACION	BigNumber		10	0	
4	NOMBREREGION	String		100		

File encoding UTF-8

- Se modifican los nombres de algunos campos:



- Se unen ambas tablas a través de la clave país:



- Se da salida a la tabla tienda de la base de datos gourmet:

Nombre paso Salida Tabla

Conexión gourmet

Esquema destino

Tabla destino TIENDA

Tamaño transacción (commit) 1000

Vaciar tabla

Ignorar errores de inserción

Specify database fields

Main options Database fields

Fields to insert:

#	Table field	Stream field
1	NOMBRE	NOMBRE
2	DIRECCION	DIRECCION
3	SUPERFICIE	SUPERFICIE
4	FORMATO_...	FORMATO_T...
5	PAIS	PAIS
6	EXTENSION...	EXTENSIONP...
7	POBLACION...	POBLACION...

4.2 Cargar los datos en la dimensión DIM_CLIENTE ([volver](#))

Se crea una transformación que tiene como elemento de entrada un archivo CSV, se rellenan los campos en blanco (valores NULL) y se eliminan los campos que no son relevantes para el estudio:



- Se obtienen los datos desde cliente.csv:

File encoding UTF-8

#	Name	Type	Format	Length
1	CODCLIENTE	String		10
2	NOMBRECLIENTE	String		100
3	SEXO	String		25
4	FECHANACIMIENTO	Date	yyyyMMdd	
5	ESTADOCIVIL	String		25
6	DIRECCION	String		100

- Se rellenan los campos que tienen valores nulos:

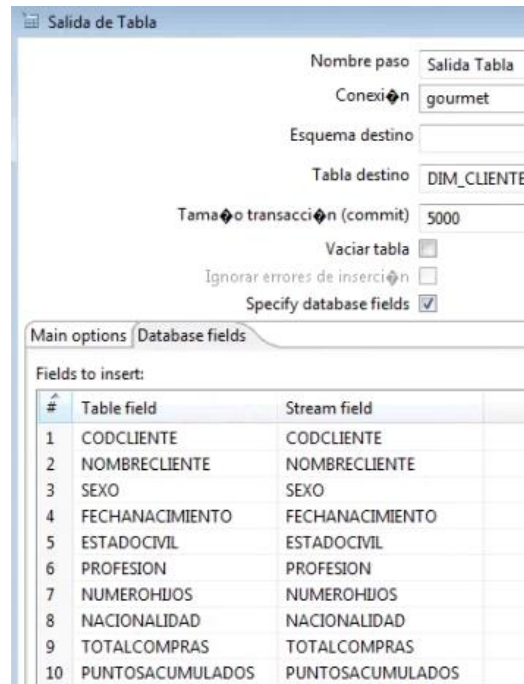
#	Field	Replace by value
1	ESTADOCIVIL	Empresa
2	NUMEROHIJOS	0

- Se eliminan los campos que no son relevantes:

Campos a eliminar :

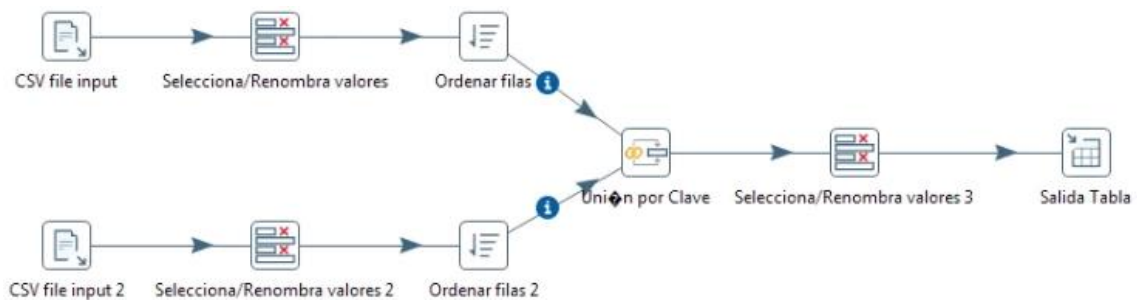
#	Nombre campo
1	DIRECCION
2	REGION

- La salida del proceso crea una tabla en la base de datos gourmet:



4.3 Cargar los datos en la dimensión DIM_PRODUCTO ([volver](#))

Para la creación de esta dimensión se ha utilizado el archivo de los productos y se han añadido algunos campos del proveedor de dicho producto. Se han ordenado las filas y se han unido las tablas a través de una unión por clave (JOIN) y la tabla resultante se ha guardado en la base de datos gourmet:



- Se obtienen los datos desde producto.csv:

File encoding UTF-8							
#	Name	Type	Format	Length	Precision	Currency	Decimal
1	CODPRODUCTO	String		25			
2	DESCRIPCION	String		100			
3	NOMBREPAYS	String		100			
4	COSTE	Number	+#####,##	5	2		.
5	PRECIOVENTA	Number	+#####,##	5	2		.
6	TIPOUNIDAD	String		100			

- Se eliminan los campos que no se necesitan para el estudio:

Campos a eliminar :

#	Nombre campo
1	TIPOUNIDAD

- Se ordenan las filas:

Campos :

#	Nombre Campo	Ascendente
1	CODPROVEEDOR	S

- Se obtienen los datos desde proveedor.csv:

File encoding UTF-8				
#	Name	Type	Format	Length
1	CODPROVEEDOR	String		10
2	NOMBREPROVEEDOR	String		100
3	PERSONACONTACTO	String		100
4	DIRECCION	String		100
5	TELEFONO	String		25
6	PERIODOPAGO	Integer		

- Se eliminan los campos que no son necesarios:

Campos a eliminar :

#	Nombre campo
1	PERSONACONTACTO
2	DIRECCION
3	TELEFONO
4	PERIODOPAGO
5	PAGOPENDIENTE
6	TIPOPROVEEDOR
7	ALCANCE

- Se ordenan las filas:

Campos :

#	Nombre Campo	Ascendente
1	CODPROVEEDOR	S

- Se unen ambas tablas por medio del código de proveedor:



- Se eliminan las antiguas claves de la nueva tabla:

Campos a eliminar :

#	Nombre campo
1	CODPROVEEDOR
2	CODPROVEEDOR_1

- Se inserta una nueva tabla con los datos de ambas tablas:

Fields to insert:

#	Table field	Stream field
1	CODPRODUCTO	CODPRODUCTO
2	DESCRIPCION	DESCRIPCION
3	NOMBREPAIS	NOMBREPAIS
4	COSTE	COSTE
5	PRECIOVENTA	PRECIOVENTA
6	NOMBRESUBFAMILIA	NOMBRESUBFAMILIA
7	MARCA	MARCA
8	NOMBREPROVEEDOR	NOMBREPROVEEDOR

5. Cargar hechos

El presente trabajo se encarga de cargar los datos en las tablas de hechos:

- Cargar tabla de hechos h_ticket
- Cargar tabla de hechos h_pedido

5.1 Carga de datos en la tabla de hechos H_TICKET ([volver](#))

Para la creación de la tabla de hechos H_TICKET se han obtenido los datos del fichero cabeceraticket.csv, se han eliminado las columnas que no eran necesarias y se han rellenado los campos vacíos:



- Se obtienen los datos desde cabeceraticket.csv:

File encoding UTF-8							
#	Name	Type	Format	Length	Precision	Currency	Decimal
1	CODVENTA	String		25			
2	NOMBRETIENDA	String		100			
3	FECHA	Date	yyyyMMdd				
4	HORA	Integer					
5	FORMAPAGO	String		100			
6	CODCLIENTE	String		25			
7	IMPORTETOTAL	Integer	+#####.####	8	4		.
8	TOTALUNIDADES	Integer					
9	PUNTOSTICKET	Integer					

- Se eliminan los campos que no son necesarios:

Campos a eliminar :

#	Nombre campo
1	FORMAPAGO

Los campos vacíos se rellenan con un nuevo código de cliente que habrá que crear en la dimensión dim_cliente:

Fields

#	Field	Replace by value
1	CODCLIENTE	00000000

- Se crea la tabla y se insertan los datos:

Fields to insert:

#	Table field	Stream field
1	CODVENTA	CODVENTA
2	NOMBRETIENDA	NOMBRETIENDA
3	FECHA	FECHA
4	HORA	HORA
5	CODCLIENTE	CODCLIENTE
6	IMPORTETOTAL	IMPORTETOTAL
7	TOTALUNIDADES	TOTALUNIDADES
8	PUNTOSTICKET	PUNTOSTICKET

5.2 Creación de la tabla de hechos H_PEDIDO ([volver](#))

La creación de esta tabla de hechos se realiza a partir del archivo pedido.csv, en el que la única transformación que se hace es la selección y ordenación de filas antes de grabar los datos en la base de datos:



- Se obtienen los datos desde pedido.csv:

#	Name	Type	Format	Length	Precision	Currency	Decimal
1	CODPEDIDO	String		25			
2	NOMBRETIENDA	String		100			
3	COPPRODUCTO	String		25			
4	PRECIOCOMPRA	Number	+#####.###	8	4		.
5	CANTIDAD SOLICITADA	Integer					
6	FECHA SOLICITUD	Date	yyyyMMdd				
7	CANTIDAD ENTREGADA	Integer					
8	FECHA ENTREGA	Date	yyyyMMdd				

- Se seleccionan las columnas que formarán parte de la tabla:

#	Nombre campo
1	CODPEDIDO
2	NOMBRETIENDA
3	COPPRODUCTO
4	PRECIOCOMPRA
5	CANTIDAD SOLICITADA
6	FECHA SOLICITUD
7	CANTIDAD ENTREGADA
8	FECHA ENTREGA

- Se ordenan las filas:

#	Nombre Campo	Ascendente
1	CODPEDIDO	S

- Se crea la tabla y se insertan los datos del origen:

#	Table field	Stream field
1	CODPEDIDO	CODPEDIDO
2	NOMBRETIENDA	NOMBRETIENDA
3	COPPRODUCTO	COPPRODUCTO
4	PRECIOCOMPRA	PRECIOCOMPRA
5	CANTIDAD SOLICITADA	CANTIDAD SOLICITADA
6	FECHA SOLICITUD	FECHA SOLICITUD
7	CANTIDAD ENTREGADA	CANTIDAD ENTREGADA
8	FECHA ENTREGA	FECHA ENTREGA