SISTEMA DE INTELIGENCIA DE NEGOCIO PARA LA MEJORA DE UN SERVICE DESK

Marcos Pereiro Conde
Grado de Ingeniería Informática
Trabajo Final de Grado – Área de Business Intelligence

Humberto Andrés Sanz
Atanasi Daradoumis Haralabus

Junio de 2019

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Sistema de Inteligencia de Negocio para la mejora de un Service Desk*
Nombre del autor:	*Marcos Pereiro Conde*
Nombre del consultor/a:	*Humberto Andrés Sanz*
Nombre del PRA:	*Atanasi Daradoumis Haralabus*
Fecha de entrega:	Junio/2019
Titulación:	*Grado de Ingeniería Informática*
Área del Trabajo Final:	*Business Intelligence*
Idioma del trabajo:	*Español*
Palabras clave	*business intelligence, service desk, machine learning*

Resumen del Trabajo

En la actualidad, los servicios de TI constituyen una parte esencial de las organizaciones y garantizar su correcto funcionamiento es una condición necesaria para asegurar el éxito corporativo. En este sentido, la correcta gestión de incidentes y solicitudes a través del Service Desk contribuye a mejorar la satisfacción del cliente asegurando que se cumplen los niveles de servicio acordados. Una incorrecta gestión del Service Desk puede resultar en unos niveles de eficiencia y productividad bajos, con las consiguientes pérdidas económicas que ello supone, y en una pérdida de confianza del usuario en el servicio y en los sistemas. Esto es especialmente importante en los proveedores de servicios de TI, donde la función del Service Desk está íntimamente ligada a su cadena de valor y el éxito o fracaso de esta gestión impacta directamente en su cuenta de resultados.

Este trabajo tiene como finalidad el diseño y construcción de un sistema de Inteligencia de Negocio aplicado a la gestión de un Service Desk en el ámbito de un proveedor de TI. Se verá cómo la construcción de un almacén de datos y la utilización de herramientas de presentación que facilitan el análisis y la exploración de los datos permite mejorar la toma de decisiones y contribuye a optimizar el uso de los recursos, aumentar la calidad del servicio y mejorar la experiencia del cliente.

Abstract:

IT services have become an essential part of organizations and ensuring their proper operation is required to achieve corporate success. In that sense, the correct handling of incidents and requests through Service Desk contributes to improve customer satisfaction by ensuring service level agreements are met. Improper management of the Service Desk can result in low levels of efficiency and productivity, with consequent economic losses, and the customer's satisfaction loss in service as well as systems. That is especially important in IT service providers, where the Service Desk function is strongly related to their value chain and the success or failure of this management impacts directly on their incomes.

The purpose of this work is the design and construction of a Business Intelligence system applied to the management of a Service Desk in the scope of an IT provider. It will show how the construction of a data warehouse and the use of presentation tools eases data analysis and exploration, bringing better decisions and contributing to optimize the use of resources, increase the quality service and improve customer experience.

A mis padres, a quienes se lo debo todo

A Marcela, que me ha acompañado estoicamente durante estos duros meses

Índice

Lista de tablas

Lista de figuras

1. Introducción

1.1 Contexto y justificación del Trabajo

En la actualidad, los servicios de soporte de TI (en adelante se usará el término inglés, IT Service Desk) forman parte integral del ecosistema de las organizaciones. En los últimos años, las empresas y las organizaciones han adoptado de forma masiva diferentes marcos de trabajo para la gestión de servicios IT (ITSM), basadas principalmente en ITIL, ISO/IEC 20000 y Agile. Son varias las razones que motivan la adopción de un ITSM, aunque en general se pueden resumir en las siguientes: mejorar la eficiencia operacional y reducir el gasto de TI, alinear las funciones de TI con los objetivos de negocio y aumentar la calidad del servicio y la satisfacción de los usuarios [1].

Entre los distintos procesos que contemplan estas metodologías, el de gestión de incidentes es el que tiene mayor adopción y el que está más desarrollado. La gestión de incidentes es una función crítica dentro de los departamentos de TI y tiene como objeto restaurar la operativa normal del servicio y minimizar el impacto que puedan tener las incidencias. En la resolución de incidencias el Service Desk actúa como punto único de contacto con el cliente-usuario, a través del cual estos hacen sus consultas e informan de las incidencias. El Service Desk registra la incidencia y la clasifica para que después sea resuelta por los especialistas que suelen organizarse en distintos niveles.

Una incorrecta gestión de los incidentes puede resultar en unos niveles de eficiencia y productividad bajos, con las consiguientes pérdidas económicas que ello supone, y en una pérdida de confianza del usuario en el servicio y en los sistemas. Si esto es importante dentro de cualquier empresa en general, aún lo es más en los proveedores de servicios de TI que prestan este servicio a otras empresas. Durante las últimas dos décadas, en línea con la tendencia del outsourcing, muchas organizaciones han externalizado esta función delegándola en empresas especializadas. En estas empresas, la gestión de incidentes está íntimamente ligada a su cadena de valor, y el éxito o fracaso de esta gestión impacta directamente en su cuenta de resultados.

El autor ha trabajado durante los últimos quince años como gerente de un Service Desk, en los que ha conocido las herramientas más habituales, ha diseñado los procesos y se ha encontrado con las problemáticas más habituales: incumplimiento de los SLAs, servicios sobrepasados por un mal dimensionamiento de los recursos, errores en la clasificación de incidencias y el efecto "ping pong" (incidencias que repetidamente saltan de un equipo de soporte a otro que las devuelve), etc.

El autor considera que la aplicación de la inteligencia de negocio a la gestión de incidentes tiene un enorme potencial para la mejora de este proceso. En el mercado existen multitud de soluciones de Service Desk, muchas de las cuales incorporan funcionalidades de Reporting avanzadas. Sin embargo, la utilización de la inteligencia de negocio y la analítica avanzada permite llevar más lejos ese análisis. Por ejemplo, es posible integrar los datos de otras fuentes como un sistema CRM, para determinar cómo las ventas actuales pueden impactar en el Service Desk en un futuro cercano y ayudar así a dimensionar o planificar los equipos de soporte. Por otro lado, la analítica avanzada puede ayudar a clasificar y segmentar las incidencias, dando una visión difícil de obtener con las herramientas tradicionales.
Este trabajo se enfoca como un ejercicio práctico para mostrar las posibilidades del uso de estas tecnologías en la mejora del proceso de gestión de incidentes en el ámbito de un proveedor de servicios IT ficticio.

1.2 Objetivos del Trabajo

1.2.1 Objetivos generales

El objetivo de este trabajo es el diseño e implementación de una solución de BI que facilite la adquisición, el almacenamiento y el análisis de datos sobre la gestión de incidentes de un proveedor de servicios de TI.

Se persigue dar una visión general que ayude a entender el funcionamiento del servicio y a mejorar la toma de decisiones relativas al mismo, con un enfoque principalmente táctico; es decir, estará enfocado a los gerentes del servicio.

El sistema debe ayudar a responder preguntas del tipo:

- ¿Se cumplen los acuerdos de nivel de servicio (SLA)?
- ¿Existen diferencias significativas en el volumen de incidencias según su categoría?
- ¿Hay recursos infrautilizados?
- ¿Se corresponde el volumen de incidencias por distintas variables (categoría, fecha/hora) con lo planificado para dimensionar los equipos?
- ¿Se puede predecir la demanda para dimensionar los equipos?
- ¿Se encuentran patrones en las incidencias que permitan adelantar problemas?
- ¿Los criterios para la clasificación, priorización y escalado de las incidencias están siendo correctos?

1.2.2 Objetivos específicos

- Diseñar un almacén de datos (Data Warehouse) para el almacenamiento de la información proveniente de una solución típica de Service Desk. Este trabajo no utilizará ninguna en concreto, por lo que se diseñará un modelo general con los campos más frecuentes y que se consideren necesarios para el desarrollo del resto de elementos.
- Evaluar las distintas soluciones disponibles en el mercado (principalmente Open Source o de uso gratuito) y seleccionar una.
- Implementar el almacén de datos y diseñar los procesos ETL necesarios
- Definir las métricas y los KPIs más relevantes para la gestión del servicio
- Implementar un dashboard
- Enriquecer la información de las incidencias con algoritmos de analítica avanzada, como los de clasificación, que permitan detectar patrones y tendencias que puedan utilizarse para uso predictivo.

1.3 Enfoque y método seguido

El trabajo se dividirá en cuatro fases, que se corresponden con los distintos entregables que se generarán:

- En una primera fase se definirá el modelo de datos y se construirá el Data Warehouse. Aquí se incluye la elaboración de ETL y unos informes de análisis multidimensional que respondan a las cuestiones analíticas antes citadas. En este punto se seguirá la metodología típica de implantación de este tipo de proyectos: se hará un análisis de los requerimientos para concretar el alcance, se diseñará el modelo de datos, se realizará el diseño físico y se implementará con la solución elegida.

- En una segunda fase se construirá un cuadro de mando que aglutine la información generada en las fases anteriores. En este caso también se seguirá la metodología habitual para la construcción de dashboards: identificar los requisitos del negocio, determinar los indicadores clave y realizar el diseño del cuadro de mando.

- En la tercera fase se abordará el análisis de los datos mediante algoritmos analíticos como la segmentación y la clasificación, para explorar las posibilidades de estos algoritmos como uso predictivo. Esta fase tiene un carácter más exploratorio y se incluye porque el autor considera muy interesante enriquecer el análisis "tradicional" con técnicas analíticas avanzadas. En este caso los resultados esperados presentan más incertidumbre, aunque se trabajará en la línea de caracterizar aquellas incidencias con más probabilidades de sufrir retrasos.

- La última fase consistirá en la composición de la memoria final (que se habrá ido realizando a lo largo del proyecto) y la presentación del trabajo.

1.4 Planificación del Trabajo

Para la planificación de este trabajo se ha calculado un esfuerzo de 300 horas, que resulta de multiplicar el número de créditos (12 ECTS) por las horas estimadas por crédito (25 h/ECTS). La unidad de programación es la jornada, que corresponde a una dedicación de entre 3,5 y 4 horas diarias. No se han incluido los fines de semana ni los

días festivos (Semana Santa y primero de mayo), aunque estos días pueden actuar como "colchón" de tiempo en caso de que sea necesario.

La estimación de la duración de cada fase se ha hecho en base a los conocimientos previos del autor y al estudio comparativo de otros trabajos finales de grado de temática similar. En cualquier caso, queda abierto a posibles ajustes a medida que avance el proyecto.

La descomposición del trabajo en fases y actividades, con su correspondiente temporización queda de la siguiente forma:

Tabla 1. Cronograma del proyecto

Nombre	Fecha de inicio	Fecha de fin	Duración (días)
1. Plan de proyecto	**27/02/19**	**11/03/19**	**9**
1.1 Introducción	27/02/19	27/02/19	1
1.2 Justificación del proyecto	28/02/19	01/03/19	2
1.3 Objetivos	04/03/19	05/03/19	2
1.4 Planificación	06/03/19	11/03/19	4
Entrega PAC1 (hito)	11/03/19	11/03/19	0
2. Diseño y construcción del DW	**12/03/19**	**12/04/19**	**24**
2.1 Análisis y selección de la tecnología y las herramientas	12/03/19	18/03/19	5
2.2 Diseño del modelo	19/03/19	22/03/19	4
2.3 Construcción del modelo	25/03/19	29/03/19	5
2.4 Programación de ETL	01/04/19	04/04/19	4

2.5 Elaboración de informes	05/04/19	09/04/19	3
2.6 Documentación memoria	10/04/19	12/04/19	3
3. Cuadro de mando	**15/04/19**	**07/05/19**	**14**
3.1 Análisis y selección de herramienta	15/04/19	17/04/19	3
Entrega PAC2 (hito)	22/04/19	22/04/19	0
3.2 Definición de KPIs	18/04/19	24/04/19	3
3.3 Construcción de dashboard	25/04/19	02/05/19	5
3.4 Documentación memoria	03/05/19	07/05/19	3
4. Analítica de datos	**08/05/19**	**29/05/19**	**16**
4.1 Análisis y selección de la tecnología	08/05/19	14/05/19	3
Entrega PAC3 (hito)	20/05/19	20/05/19	0
4.2 Enriquecimiento con algoritmos de clasificación	15/05/19	24/05/19	8
4.3 Documentación memoria	27/05/19	29/05/19	3
5. Entrega final	**30/05/19**	**14/06/19**	**12**
5.1 Elaboración memoria final	30/05/19	07/06/19	7
5.2 Preparación presentación virtual	10/06/19	13/06/19	4
5.3 Autoinforme	14/06/19	14/06/19	1
Entrega final (hito)	17/06/19	17/06/19	0

1.5 Breve sumario de productos obtenidos

Además de esta memoria, como resultado de este trabajo se entregan los siguientes productos:

- Modelo de datos del Data Warehouse realizado con MySQL Workbench

- Trabajos de transformación de datos (ETL) realizados con Pentaho Data Integration

- Dataset: Conjunto de datos utilizado para la alimentación del Data Warehouse y la elaboración de los informes

- Fichero de Power BI Desktop con los informes y el Cuadro de Mando. Se incluye una versión estática de los informes en Power Point.

- Fichero en R Markdown con la elaboración del modelo predictivo de violación de SLA

1.6 Breve descripción de los otros capítulos de la memoria

En el Capítulo 2 se hará una introducción al Business Intelligence, describiendo brevemente su cometido y los elementos esenciales que lo conforman.

En el Capítulo 3 se lleva a cabo el análisis de herramientas para conformar el entorno tecnológico. Se describen los criterios de evaluación, se comparan las herramientas y se elige una para cada capa del sistema.

En el Capítulo 4 se define el modelo de datos con el que se construirá el almacén de datos. Se diseña el modelo conceptual, el modelo lógico y finalmente el modelo físico con MySQL.

En el Capítulo 5 se describe el conjunto de datos sobre el que trabajará este proyecto. Ese conjunto proviene en parte de una fuente abierta y ha sido completado artificialmente en algunos campos mediante la generación de series aleatorias que son explicadas en este capítulo.

En el Capítulo 6 se lleva a cabo la construcción de los procesos de transformación de los datos (ETL) para integrar los datos en el modelo dimensional del almacén de datos

En el Capítulo 7 se presentan los informes diseñados para resolver las cuestiones analíticas de partida y se ejemplifica su uso mediante el análisis de las dimensiones más relevantes de un Service Desk.

En el Capítulo 8 se diseña el Cuadro de Mando, detallando los KPI que lo conformarán en base a los requerimientos de este trabajo.

En el Capítulo 9 se hace una introducción al aprendizaje automático (machine learning) situándolo en el contexto de un Service Desk. Se lleva a cabo un ejercicio práctico mediante la creación de un modelo que permita predecir los tickets que incumplirán el SLA.

En el Capítulo 10 se presentan las conclusiones de este trabajo, las lecciones aprendidas y la reflexión sobre la consecución de los objetivos establecidos, un análisis crítico sobre el seguimiento de la planificación y líneas de trabajo futuras.

2. Introducción a los sistemas de BI

2.1 La Inteligencia de Negocio

La gestión de una organización requiere tomar decisiones continuamente y en todos los niveles: qué productos vender y cuáles no, en qué mercados se debe estar presente, con qué socios se quiere trabajar, qué descuentos se pueden aplicar, cuándo aumentar o reducir la plantilla, etc. En un mercado altamente competitivo, las empresas que toman mejores decisiones que sus competidores tienen más posibilidades de sobrevivir y de tener éxito. No es de extrañar, por tanto, que las empresas hayan buscado, desde hace muchos años, maneras para mejorar sus procesos de toma de decisiones.

Un factor clave en este objetivo es disponer de más y mejor información. Cuando a las personas de negocio se les ofrece información relevante, fiable y oportuna, las posibilidades de que las decisiones tomadas sean correctas aumentan considerablemente [2]. Las empresas obtienen esa información a través de múltiples fuentes, tanto internas a la organización como externas, y de muchas maneras distintas – con procesos formales e informales, con datos cuantitativos y cualitativos, etc.

Una de las fuentes de información más valiosa para una organización se encuentra en los sistemas de gestión que ésta utilizan para dar soporte a su actividad diaria. Estos sistemas de información, como los ERP y los CRM, están enfocados al procesamiento operativo, suelen trabajar a nivel de transacción (servir pedidos, emitir facturas, registrar clientes, realizar pagos, etc.) y son muy eficaces para esta labor.

Sin embargo, las necesidades de las personas de negocio que han de tomar decisiones, normalmente los mandos intermedios y la dirección, son muy distintas. Estos “usuarios analíticos” rara vez trabajan al nivel de una transacción y hacen preguntas que estos sistemas no suelen responder: variaciones en el tiempo, tendencias, agrupaciones, …

El Business Intelligence o Inteligencia de Negocio permite poner en valor esos datos, llevándolos a sistemas que facilitan su interpretación y análisis por parte de los usuarios

de negocio, haciendo así que esa información se convierta en conocimiento útil para tomar mejores y más rápidas decisiones.

2.2 Arquitectura de un sistema de BI

Desde un punto de vista técnico, un sistema de BI está formado por una estructura de componentes que se implementan mediante diversas herramientas y aplicaciones.

Esta estructura parte de las fuentes donde se encuentran los datos que se quieren analizar, normalmente bases de datos relacionales, hojas de cálculo y ficheros de texto. Estos datos se extraen de los sistemas originales y se llevan a un almacén de datos específico donde se integran y se unifican, normalmente tras aplicarles un proceso de depuración y transformación para adecuarlos al proceso analítico. Por último, se encuentran los componentes que permiten explotar esta información mediante visualización y el análisis (informes, cuadros de mando, …)

A continuación, se describirán estos componentes con más detalle.

2.2.1 Data Warehouse

El data warehouse o almacén de datos es un elemento esencial dentro la estrategia de Inteligencia de Negocio, dado que actúa como un repositorio central e integrado de la información corporativa, que servirá para alimentar los procesos de análisis y reporting que permitirán tomar mejores decisiones.

El almacén de datos se nutre principalmente de los datos provenientes de sistemas operacionales, como los ERP y los CRM, aunque también puede incorporar datos contenidos en hojas de cálculo, ficheros de texto, logs, etc. La información obtenida de estas fuentes se integra y se consolida para constituirse en el punto central de información de la compañía, dando una visión única y fiel del negocio.

A diferencia de las bases de datos transaccionales, que se usan para la operativa del día a día y solo mantienen la versión más actual de la información, los almacenes de datos contienen datos históricos que permite hacer comparaciones y ver tendencias. Además, estos datos son no volátiles, es decir, una vez los datos se incorporan al almacén, ya no se modifican ni se eliminan.

En la práctica es habitual construir subconjuntos más pequeños del Data Warehouse, llamados Data Marts, que contienen información orientada a un área específica (Ventas, Finanzas, …), por su mayor facilidad a la hora de implementarlos.

2.2.2 ETL

Las fuentes que alimentan un sistema de inteligencia de negocio pueden ser muchas y muy heterogéneas: utilizan modelos de datos distintos, tecnologías distintas y tienen diferentes puntos de vista sobre el negocio. La función de las ETL es lidiar con esa diversidad, extrayendo los datos de las fuentes de origen e integrándolos en el almacén de datos para proporcionar una visión única útil para los analistas.

La integración de los datos requiere abordar problemas como la unificación de tipos de datos, de las codificaciones, del nivel de agregación, de la semántica, corregir valores inconsistentes, eliminar duplicados, etc. En esta fase se llevan a cabo procesos de transformación y depuración de los datos para homogeneizar su representación y garantizar su calidad al ser integrados en el almacén de datos, desde donde serán utilizados en los procesos de análisis y reporting.

2.2.3 Reporting

Una de las formas más habituales de extraer información de los almacenes de datos es a través de los informes. Un informe es un documento que presenta los resultados de un proceso de negocio, habitualmente con texto y cifras mostrados en forma tabular, al que también se pueden añadir gráficos para facilitar la comprensión de los datos. En un

sistema de BI es habitual encontrar un componente de reporting, que incluye las herramientas para diseñar, generar y distribuir los informes a lo largo de la organización.

Los informes pueden ser estáticos, parametrizables o ad hoc. Los informes estáticos tienen siempre la misma estructura y el mismo criterio para obtener los datos; los informes parametrizables incorporan la posibilidad de especificar unos parámetros de entrada que permiten variar los resultados, y los informes ad-hoc son creados por el usuario final para responder a preguntas concretas.

2.2.4 Análisis OLAP

Los sistemas OLAP (Online Analytical Process) permiten al usuario analizar rápidamente la información que se ha resumido en vistas y jerarquías multidimensionales.

Estos sistemas constan de un motor y un visor. El motor se encarga de almacenar los datos en una estructura en forma de cubo, a menudo calculando de antemano todas las posibles consultas, para ofrecer un acceso más rápido a la información. El visor es la herramienta que se utiliza para explorar los datos de los cubos, facilitando operaciones que permiten navegar a través de los distintos niveles de detalle de la información (drill down y drill up), ver los datos diferentes perspectivas (pivot) y hacer selecciones de dimensiones particulares para obtener nuevos subconjuntos de datos. Todas estas funciones facilitan el análisis de causas raíz o, dicho de otra manera, el responder a preguntas de por qué pasó.

2.2.5 Cuadros de mando

Los cuadros de mando o dashboards son herramientas de presentación que se utilizan para realizar un seguimiento de los indicadores clave de rendimiento, las métricas y otros puntos críticos relevantes para una organización, departamento o proceso específico. Las visualizaciones de datos simplifican la presentación de datos complejos

y proporcionan al usuario información del rendimiento actual mediante un simple vistazo. Los cuadros de mando se basan en el uso extensivo de elementos gráficos, pensados para ser vistos en una pantalla de ordenador, y donde se suelen mostrar indicares que presentan tendencias para poder comparar.

3. Selección del entorno tecnológico

3.1 Selección de herramientas candidatas

El mercado de herramientas de BI es un mercado maduro y en continua evolución, que ofrece infinidad de opciones para llevar a cabo los procesos descritos anteriormente. En él se pueden encontrar soluciones comerciales de grandes fabricantes de software como Microsoft, Oracle o SAP que cubren todo el espectro de un ecosistema de BI, y otras más especializadas como las que ofrecen Informatica o SAS. También se observa una marcada tendencia hacia herramientas cada vez más intuitivas y fáciles de usar, que permiten a los usuarios de negocio sin conocimientos técnicos realizar análisis de datos por su cuenta, como Microsoft Power BI, Tableau, Sisense o Qlik.

No está dentro del alcance de este trabajo hacer un análisis comparativo de tan vasta oferta. Los requisitos de este proyecto pueden ser fácilmente asumibles con herramientas gratuitas o de código abierto que, por otro lado, están demostrando ser punta de lanza en el mercado, facilitando la adopción del BI a las organizaciones que no podían afrontar los costes de soluciones comerciales.

Las herramientas de código abierto no suelen ser analizadas por las grandes consultoras tecnológicas, como Gartner o Forrester, que proporcionan informes comparativos que sirven de referencia para la selección de estas tecnologías. Por lo tanto, para la búsqueda de candidatos se ha usado fundamentalmente la exploración en internet. Se han revisado rankings de herramientas gratuitas y de código abierto elaborados en los últimos tres años, basados en las opiniones de los usuarios, como los que ofrecen Capterra, Software Advice o TrustRadius y se han seleccionado las soluciones que aparecían recurrentemente en los primeros puestos de estos rankings.

El análisis de herramientas se dividirá en tres grupos:

a) **Herramientas de integración**: Aquí se valorarán las herramientas ETL encargadas de extraer los datos de las fuentes origen, transformarlos y limpiarlos

e integrarlos en el almacén de datos. Las opciones para analizar son: Jaspersoft ETL, Talend Open Studio y Pentaho Data Integration.

b) **Soluciones para el almacén de datos**: En este punto se analizarán sistemas de gestión de bases de datos que permitan implementar el almacén de datos. Las opciones elegidas son: MySQL, MariaDB y PostgreSQL.

c) **Herramientas de presentación**: Aquí se examinarán las herramientas que permiten explorar los datos, visualizarlos y analizarlos para extraer conclusiones. Las opciones elegidas son: Microsoft PowerBI, Qlik Sense y Tableau Public.

3.2 Metodología de evaluación

Para facilitar el análisis y la comparación de las opciones escogidas se establecerán una serie de criterios de evaluación que cubren los aspectos más relevantes de estas herramientas. A cada criterio se le asignará un peso porcentual en función de la importancia que se le concede, totalizando todos los criterios el 100%. En la evaluación de cada herramienta, se calificará cada criterio con una puntuación que indique el grado de cumplimiento de esa herramienta en el respectivo criterio. Estos valores serán: 0 – No cumple, 1 – Muy pobre, 2 - Aceptable, 3 – Bueno, 4 – Excelente. Tras ponderar estas calificaciones se obtendrá una puntuación total que servirá para determinar la solución elegida.

Los criterios de evaluación miden aspectos funcionales, técnicos y organizativos de las soluciones.

- Los **criterios funcionales** son los que cobran más importancia en la evaluación y se refieren a las características de la aplicación, es decir, a lo que la aplicación hace o se supone que ha de hacer.

- Los **criterios técnicos** se refieren a cómo lo hace y en qué entorno: ¿qué arquitectura tiene?,¿en qué plataformas funciona?, ¿es seguro y fiable?

- Por último, los **criterios organizativos** valoran aspectos relativos a la comunidad y a la adopción: ¿existe una comunidad importante alrededor de esta solución – desarrolladores, formadores, consultores, etc.? ¿el producto evoluciona? ¿es fácil obtener soporte? ¿existe experiencia previa en la organización?

3.3 Análisis de herramientas de integración

A continuación, se analizan las herramientas que permiten llevar a cabo los procesos de extracción, transformación y carga (ETL).

Talend Open Studio

Talend Open Studio es una de las soluciones ETL más reconocidas del mercado. Talend Data Integration, su hermano "mayor" en versión comercial, está reconocido como uno de los líderes en el cuadrante mágico de Gartner.

Talend Open está desarrollado en Java y utiliza un planteamiento orientado a la generación de codigo, lo que permite que los trabajos se puedan exportar y ejecutar en entornos run-time. Posee un entorno gráfico para el modelado de los trabajos, que lo hacen muy fácil de usar. Dispone de potentes funciones de ETL y ELT (mapeo, agregación, ordenación, enriquecimiento, …) y proporciona un entorno unificado para todo el ciclo de vida de los datos: se complementa con otros módulos como Open Studio for Data Quality u Open Studio for MDM para la gestión de datos maestros y calidad de datos. Dispone de un gran abanico de conectores y componentes para bases de datos, aplicaciones y servicios en la nube.

La comunidad alrededor de Talend es muy amplia y no es difícil obtener soporte técnico, personal cualificado o formación, aunque la mayoría de los servicios son de pago.

Pentaho Data Integration

Al igual que Talend, Pentaho Data Integration también dispone de una versión comercial y una gratuita con funciones algo más limitadas.

Desarrollado en Java, este sistema tiene un innovador enfoque basado en metadatos y una arquitectura asentada sobre la base de estándares. Dispone de una interfaz muy intuitiva y una rápida curva de aprendizaje. Entre sus funcionalidades se encuentran la migración de datos entre aplicaciones y bases de datos, la exportación de ficheros planos de la base de datos, el soporte de consultas SQL personalizadas y de JavaScript, y un fácil modelado de datos. La versión gratuita carece del programador de trabajos y de opciones de monitorización más allá de los típicos logs.

Pentaho es uno de los sistemas con más tradición en el mercado de código abierto, aunque se mantiene en continua evolución y presenta un roadmap que garantiza su futuro, cada vez más orientado al big data y a la nube. La base de usuarios es muy amplia y es fácil encontrar colaboradores y técnicos preparados para dar soporte o participar en desarrollos e implantaciones.

Jaspersoft ETL

Jaspersoft ETL surge como resultado de la cooperación entre las compañías Jaspersoft y Talend. En lo esencial, Jaspersoft ETL se basa en Talend Open Studio y se pueden encontrar muchas similitudes entre ellos, como las funciones avanzadas de ETL y ELT, un entorno gráfico para el modelado de trabajos, etc. Jaspersoft ETL puede ser más recomendable cuando es utilizado con otros productos de la suite de BI, como JasperReports.

Al igual que los anteriores, Jaspersoft también cuenta con una amplia comunidad de desarrollo y soporte alrededor, aunque en menor medida que los anteriores.

A continuación, se muestran los resultados de la evaluación de las herramientas de integración, que dan como mejor alternativa a Pentaho Data Integration, seguido muy de cerca por Talend Open.

Tabla 2. Evaluación de herramientas ETL

Criterio de evaluación	Peso	Puntuación			Ponderación		
		Talend Open	Pentaho Data Integration	Jaspersoft ETL	Talend Open	Pentaho Data Integration	Jaspersoft ETL
GUI y facilidad de uso	20%	3	4	3	0,6	0,8	0,6
Conectores y adaptadores	15%	3	4	2	0,45	0,6	0,3
Funciones de transformación	30%	4	3	4	1,2	0,9	1,2
Metadatos	5%	4	2	2	0,2	0,1	0,1
Arquitectura	10%	3	3	3	0,3	0,3	0,3
Comunidad y soporte	15%	3	4	2	0,45	0,6	0,3
Experiencia previa	5%	2	4	0	0,1	0,2	0
Puntuación total					**3,3**	**3,5**	**2,9**

3.4 Análisis de soluciones para el almacén de datos

Como se ha comentado anteriormente, el almacén de datos es el repositorio centralizado donde se consolidan los datos de las distintas fuentes de origen y constituye el elemento que alimentará a los procesos de análisis y reporting.

Técnicamente los almacenes de datos se implementan con sistemas de bases de datos, frecuentemente de tipo relacional, aunque estructurados y configurados para soportar las consultas analíticas a las que serán sometidos.

A continuación, se analizan los tres sistemas seleccionados.

MySQL

MySQL es un producto propiedad de Oracle, distribuido bajo la licencia GNU. Es un sistema fácil de usar a través de las consolas de administración y modelado de datos (MySQL Workbench y otras de terceros) y su rendimiento es alto, aunque sus

limitaciones en el almacenamiento no lo aconsejan para grandes proyectos de BI. Aun así, en los proyectos que pueden abordar normalmente las pequeñas y medianas empresas es una solución excelente por un coste de adquisición cero.

MySQL es el motor de base de datos de código abierto más popular, de hecho, es casi un estándar de facto en el mundo del desarrollo, lo que facilita encontrar documentación y soporte de este, así como múltiples aplicaciones con conectores específicos para este sistema y soporte de muchas plataformas.

MariaDB

MariaDB es una bifurcación de MySQL y ambos comparten la misma estructura de base de datos y de índices. Esto significa que son compatibles en la definición de tablas, los protocolos cliente y las API y, además, los conectores de uno funcionarán sin cambios con el otro.

MariaDB es un proyecto muy vivo y tecnológicamente más avanzado que MySQL, aunque el soporte de la comunidad aún debe crecer.

PostgreSQL

PostgreSQL es un sistema de base de datos relacional de código abierto desarrollado por el PostgreSQL Global Development Group, un grupo formado por varias compañías y colaboradores individuales. Es un motor de alto rendimiento con un abanico de utilización muy amplio, que va desde desarrollos locales a grandes almacenes de datos, y donde destaca en su capacidad de concurrencia.

Cuenta con una interfaz gráfica (pgAdmin) que lo hace fácil de administrar, aunque en menor medida que MySQL Workbench.

A continuación, se muestran los resultados de la evaluación de los sistemas de bases de datos para el almacén datos, que dan como mejor alternativa a MySQL, aunque casi en el mismo nivel que PostgreSQL:

Tabla 3. Evaluación de bases de datos

Criterio de evaluación	Peso	Puntuación			Ponderación		
		MySQL	PostgreSQL	MariaDB	MySQL	PostgreSQL	MariaDB
Rendimiento	30%	3	4	3	0,9	1,2	0,9
Escalabilidad	15%	2	4	2	0,3	0,6	0,3
Facilidad de uso	20%	5	3	4	1	0,6	0,8
Arquitectura	10%	3	4	3	0,3	0,4	0,3
Comunidad y soporte	20%	5	4	2	1	0,8	0,4
Experiencia previa	5%	4	1	0	0,2	0,05	0
Puntuación total					**3,7**	**3,65**	**2,7**

3.5 Análisis de herramientas de presentación

Lo que se espera de este tipo de herramientas es que proporcionen autonomía al usuario para poder realizar sus análisis con la mínima intervención por parte de TI, que faciliten la exploración de los datos mediante filtros, jerarquías y navegación drill-down y que puedan representar visualmente la información. Y últimamente, además, que el acceso sea ubicuo, desde cualquier lugar y utilizando dispositivos móviles. En este sentido, las tres herramientas son muy similares y cumplen sobradamente con esas necesidades, proporcionando capacidades de análisis básicas a casi cualquier usuario.

Microsoft Power BI

La solución de Microsoft está avanzando rápidamente gracias a la enorme base de usuarios de su solución de Office 365, en la que se integra, y a una atractiva política de planes y precios. Dispone de una versión gratuita que limita las posibilidades de colaboración y compartición de los paneles.

Es quizás la solución con una mejor curva de aprendizaje y una mayor facilidad de uso gracias a sus similitudes con Microsoft Excel, con el que comparte las mismas funciones, que permiten crear informes visualmente atractivos sin ninguna complicación.

Dispone de un extenso catálogo de conectores para acceder a fuentes de datos, como bases de datos, servicios en la nube, big data, … que se pueden ampliar con adaptadores de terceros a través de su propio marketplace.

Power BI puede escalarse para albergar grandes proyectos, garantizando la inversión realizada en etapas tempranas.

Tableau Public

Tableau es el líder en visualización de datos, con una interfaz muy fácil de usar que permite a los usuarios con menos conocimientos técnicos crear visualizaciones interactivas e informes de forma rápida y sencilla. También es muy eficiente para crear mapas multidimensionales con sus capacidades integradas de geo codificación.

Tableau es una solución fácil de usar y de rápido aprendizaje, siendo pionera en funciones como arrastrar y soltar y en la detección automática del tipo de gráfico. Esto es especialmente importante si se considera el tipo de usuarios al que se dirigen.

Qlik Sense

Las posibilidades de visualización de Qlik también son muy destacables, y dispone de tipos de gráficos que no están disponibles en Tableau o PowerBI, como las Gauge o los gráficos 3D, aunque su entorno, sobre todo en lo que respecta a la carga de archivos, puede ser el más complejo de los tres.

Donde destaca Qlik Sense es en sus capacidades de análisis, gracias a su Associative Engine, un motor in-memory que permite descubrir las relaciones entre los datos, y a su función de análisis guiado que permite orientar al usuario a medida que éste interactúa

con los datos. Incorpora además funciones específicas para analizar redes sociales como Twiter.

Tabla 4. Evaluación de herramientas de presentación

Criterio de evaluación	Peso	Puntuación			Ponderación		
		PowerBI	Qlik Sense	Tableau Public	PowerBI	Qlik Sense	Tableau Public
Visualización	25%	4	4	5	1	1	1,25
Descubrimiento de datos	20%	3	4	3	0,6	0,8	0,6
Experiencia de usuario	20%	5	4	4	1	0,8	0,8
Conectividad	15%	4	4	4	0,6	0,6	0,6
Comunidad y soporte	15%	4	3	4	0,6	0,45	0,6
Experiencia previa	5%	3	4	1	0,15	0,2	0,05
Puntuación total					**3,95**	**3,85**	**3,9**

3.6 Herramientas seleccionadas

Tras esta pequeña revisión de las herramientas y después de someter cada grupo a los criterios de evaluación, las aplicaciones que conformarán el entorno tecnológico de este proyecto serán:

- Pentaho Data Integration para la capa de integración
- MySQL para el almacén de datos
- Microsoft PowerBI para la visualización de datos, los informes y el cuadro de mando.

4. Modelo de datos

El primer paso para construir un almacén de datos consiste en definir el modelo de datos. Este modelo parte de la descripción de una determinada realidad, de la identificación de las entidades de datos que definen el ámbito de un problema y sus asociaciones, y se concreta en una especificación que describe cómo se organizará la base de datos. Es necesario, por tanto, conocer el problema que se quiere resolver y comprender los datos con los que se va a trabajar, cuestión que se abordará en el siguiente apartado.

Este proceso se hará siguiendo el enfoque del modelado dimensional propuesto por Ralph Kimball [3]. El modelo dimensional se caracteriza por presentar una estructura no normalizada, donde los datos se organizan de una forma más cercana a como los usuarios de negocio la perciben y optimizados para ser consultados, resumidos y analizados.

Este modelo se compone de un elemento central, la tabla de hechos, que representa un evento medible de negocio, básicamente lo que se quiere analizar – una venta, por ejemplo – y las tablas de dimensiones, que definen el contexto del evento medido – cuándo se vendió, dónde se vendió, a quién se vendió, etc.

Normalmente estas tablas se representan en un diagrama con la tabla de hechos en el centro y las tablas de dimensiones a su alrededor, dando lugar a lo que se conoce como esquema de estrella.

4.1 Descripción del caso

En la actualidad, los servicios de tecnología de información constituyen una parte esencial de las organizaciones. En consecuencia, la operación de estos servicios representa un componente crucial del éxito corporativo. La función del Service Desk se enmarca en este proceso y su objetivo principal es asegurar la satisfacción del cliente, encargándose de:

- Proporcionar un punto de comunicación único para los usuarios del servicio
- Coordinar grupos de trabajo y procesos del servicio para asegurar que se cumplen los niveles de servicio acordados.

Las tareas que lleva a cabo un Service Desk se pueden resumir en:

1. Registrar, categorizar y priorizar las peticiones de los clientes.
2. Realizar un diagnóstico de las peticiones y actuar como primer nivel de resolución
3. Escalar las peticiones que no puede resolver
4. Monitorizar la resolución de las peticiones, vigilando que se encuentre siempre dentro de los parámetros marcados por el acuerdo de nivel de servicio.
5. Cerrar las peticiones resueltas.
6. Medir la satisfacción de los clientes.

A partir de esta definición, se tratará de desgranar y describir las principales entidades de datos que se utilizarán.

Cliente: El cliente es el usuario del servicio. Cada cliente se identifica con un número único. Con el objeto de poder hacer análisis por zonas geográficas, también se almacenará la ubicación física del cliente.

SLA: El nivel de servicio acordado es el contrato bajo el cual se regulan las condiciones en las que se prestará el servicio. Este acuerdo incluye, entre otras cosas, los tiempos previstos para atender al cliente y resolver una incidencia, basados en el impacto y la prioridad que se le asigna a cada solicitud. A partir del SLA se puede calcular la **fecha de vencimiento** que nos indica cuál es el momento tope para resolver la incidencia.

Tickets: Un ticket representa el hecho de que un cliente contacte con el Service Desk, bien porque ha tenido alguna incidencia (p.ej. un problema de hardware o de software) o bien para requerir un servicio (hacer una consulta, solicitar un cambio de datos, pedir una instalación, etc.). Los tickets se categorizan y se priorizan en función del SLA correspondiente, que estará definido en base al servicio y al cliente.

Para cada solicitud se registran varios datos que se han considerado necesarios con fines analíticos. Estos datos son el:

7. **Modo** de comunicación de la solicitud (por mail, chat, teléfono, …)
8. **Impacto** de la incidencia en el servicio (alto, medio, bajo, …)
9. **Categoría** o área en la que se enmarca la solicitud (redes, software, internet, …)

Para medir el desempeño del servicio se registran por cada solicitud los siguientes datos:

10. **Tiempo de resolución en días**. Nos da la cantidad de días transcurridos desde el momento en que se abre la solicitud hasta que está resuelta.
11. **SLA violado**: Es un indicador booleano (Verdadero, Falso) que indica si se han cumplido los plazos marcados por el SLA o no.
12. **Reabierta**: Es un booleano que indica si ha sido necesario reabrir la solicitud.
13. **Escalada**: Es un booleano que indica si ha sido necesario escalar la solicitud.

Técnicos: Los técnicos son los agentes encargados de dar solución y respuesta a las incidencias y peticiones planteadas por el cliente. Cada técnico pertenece a un grupo de soporte.

4.2 Modelo conceptual

Tras la introducción realizada en el punto anterior ya se puede empezar a crear el modelo que dará respuesta a las preguntas analíticas. En primer lugar, se define el modelo conceptual, donde se identifica la tabla de hechos, que en este caso son los Tickets, y las dimensiones que ayudan a contextualizar esos tickets.

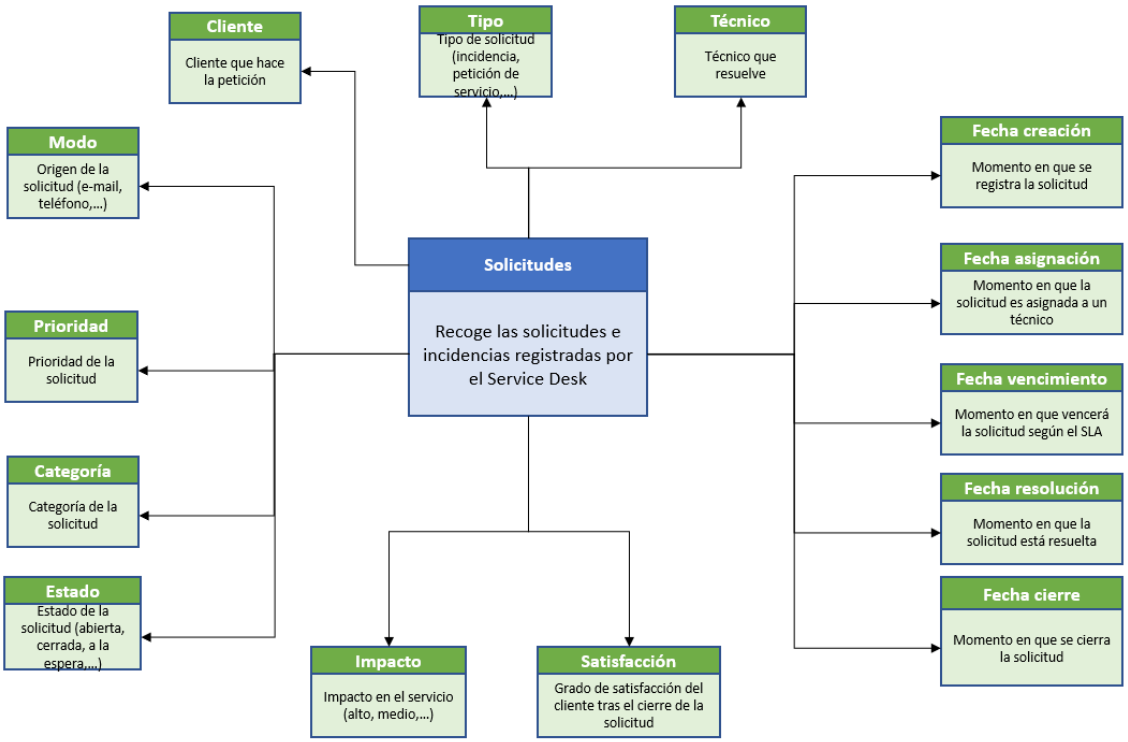


Figura 1. Modelo conceptual

4.3 Modelo lógico

Una vez se dispone del modelo conceptual se puede pasar a diseñar el modelo lógico, donde se identifican la tabla de hechos, con sus atributos y métricas, y las tablas de dimensiones, también con sus atributos.

Tabla	**h_solicitudes**
Tipo de tabla	Hechos
Descripción	Contiene una fila por cada ticket registrado en el sistema.
Clave primaria	id_ticket
Claves foráneas	id_fecha_creacion id_tecnico id_impacto id_modo id_urgencia id_prioridad id_categoria id_estado id_satisfaccion id_sla id_fecha_asignacion id_fecha_vencimiento

	id_fecha_resolucion id_fecha_cierre id_localizacion
Métricas	reabierta escalada sla_violado tiempo_resolucion_horas

Tabla	**d_estado**
Tipo de tabla	Dimensión
Descripción	Estado de la solicitud (abierta, cerrada, a la espera, …)
Clave primaria	id_estado
Atributos	desc_estado

Tabla	**d_categoria**
Tipo de tabla	Dimensión
Descripción	Categoría de la solicitud (redes, software, internet, …)
Clave primaria	id_categoria
Atributos	desc_categoria

Tabla	**d_prioridad**
Tipo de tabla	Dimensión
Descripción	Prioridad de la solicitud (alta, baja, media).
Clave primaria	id_prioridad
Atributos	desc_prioridad

Tabla	**d_modo**
Tipo de tabla	Dimensión
Descripción	Modo de apertura (teléfono, chat, email, formulario, …)
Clave primaria	id_modo
Atributos	desc_modo

Tabla	**d_cliente**
Tipo de tabla	Dimensión
Descripción	Cliente que hace la solicitud.
Clave primaria	id_cliente
Atributos	desc_cliente

Tabla	**d_tipo**
Tipo de tabla	Dimensión
Descripción	Tipo de solicitud (incidencia, petición de servicio).
Clave primaria	id_tipo
Atributos	desc_tipo

Tabla	**d_tecnico**
Tipo de tabla	Dimensión
Descripción	Técnico que atiende la incidencia y grupo al que pertenece
Clave primaria	id_tecnico
Atributos	desc_tecnico grupo

Tabla	**d_fecha_creacion**
Tipo de tabla	Dimensión
Descripción	Fecha de creación de la solicitud.
Clave primaria	id_fecha_creacion
Atributos	fecha día semana mes trimestre

Tabla	**d_fecha_vencimiento**
Tipo de tabla	Dimensión
Descripción	Fecha de vencimiento según el SLA
Clave primaria	id_estado
Atributos	fecha día semana mes trimestre

Tabla	**d_fecha_asignacion**
Tipo de tabla	Dimensión
Descripción	Fecha en que se empieza a atender una solicitud.
Clave primaria	id_fecha_asignacion
Atributos	fecha día semana

	mes trimestre

Tabla	**d_fecha_resolucion**
Tipo de tabla	Dimensión
Descripción	Fecha en que se resuelve la solicitud.
Clave primaria	id_resolucion
Atributos	fecha día semana mes trimestre

Tabla	**d_fecha_cierre**
Tipo de tabla	Dimensión
Descripción	Fecha en que se cierra la solicitud. No tiene por qué coincidir con la fecha de resolución, ya que es necesario esperar el ok del cliente.
Clave primaria	id_fecha_cierre
Atributos	fecha día semana mes trimestre

Tabla	**d_impacto**
Tipo de tabla	Dimensión
Descripción	Impacto de la solicitud
Clave primaria	id_impacto
Atributos	desc_impacto

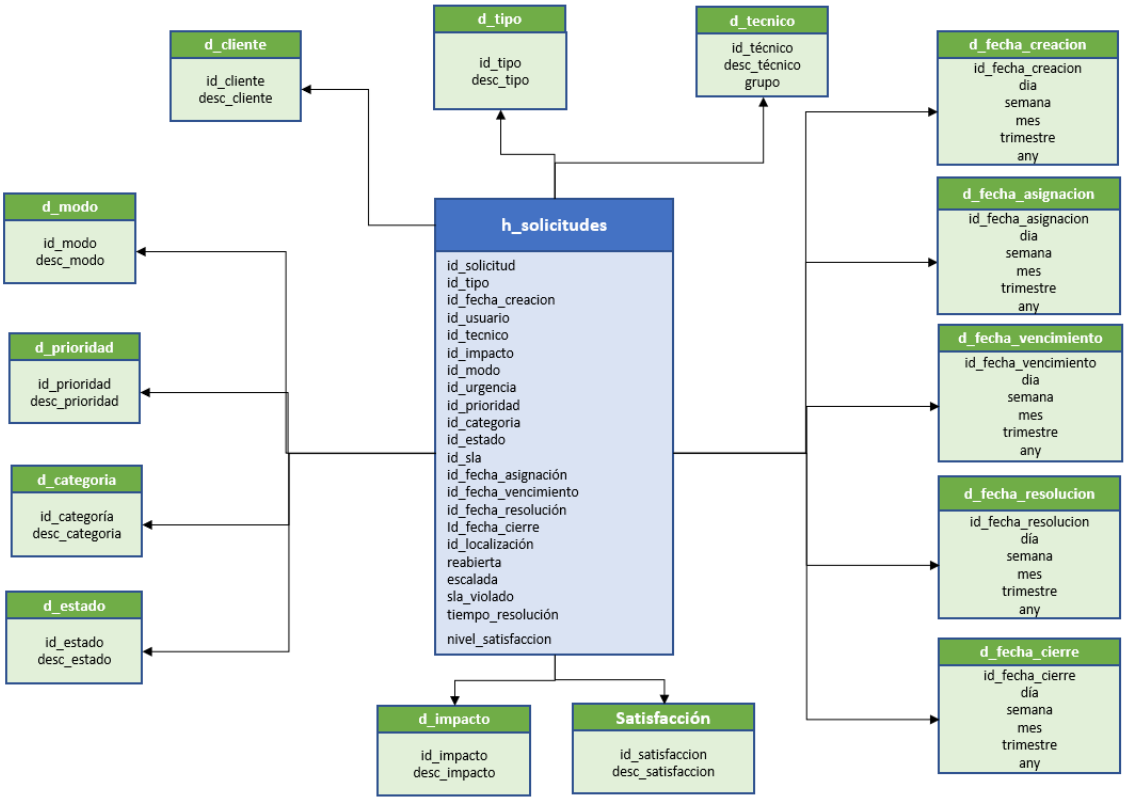


Figura 2. Modelo lógico de datos

4.4 Modelo físico

Finalmente se procede a crear el modelo físico, donde se especifican todas las tablas y columnas, el tipo de datos de cada columna, y las claves foráneas que se usan para identificar las relaciones entre las tablas. El modelo físico es dependiente del sistema de base de datos que se utilice; en este proyecto se utilizará MySQL, como se vio en la selección de herramientas previas. Para diseñar este modelo se ha utilizado la herramienta MySQL Workbench que acompaña a este sistema gestor. Se muestra el diagrama a continuación:

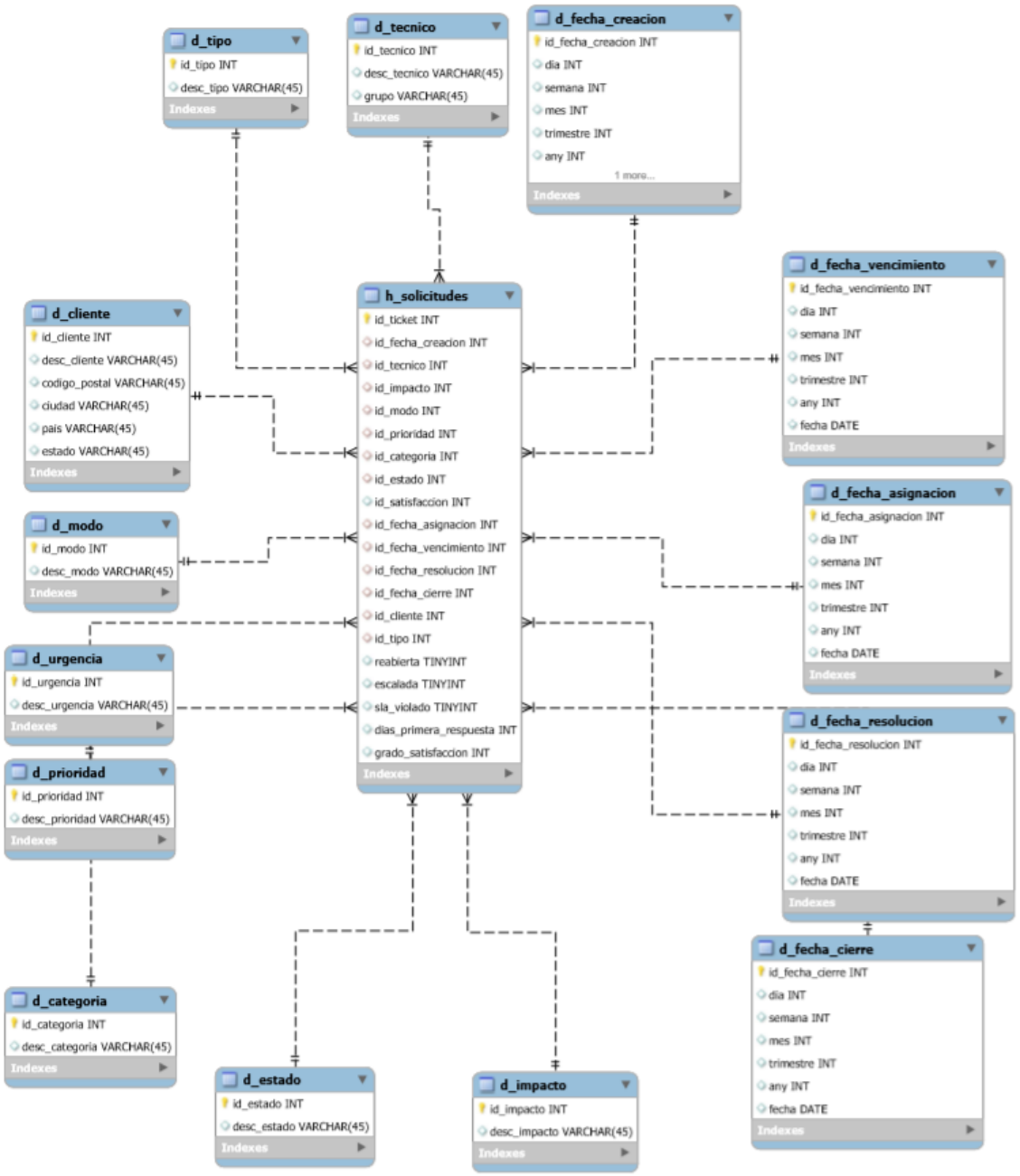


Figura 3. Modelo físico de datos

5. Generación del conjunto de datos fuente

Para dotar de contenido al almacén de datos se utilizará como base un conjunto de datos reales de un departamento de TI que se complementará artificialmente con los campos que requiere este proyecto y que no están presentes en ese fichero, tal como se verá a continuación.

El fichero original es un conjunto de datos abiertos proporcionado por IBM como muestra para explorar el uso de su solución Watson Analytics. [4]. Estos datos provienen de un departamento de tecnología de la información (TI) interesado en examinar con qué rapidez se resuelven los problemas y mejorar la satisfacción de los clientes. El fichero incluye 100.000 registros sobre solicitudes tratadas por su Service Desk.

Las primeras líneas del contenido del archivo se pueden ver en la siguiente imagen:

	A
1	**ticket,requestor,RequestorSeniority,ITOwner,FiledAgainst,TicketType,Severity,Priority,daysOpen,Satisfaction**
2	1,1929,1 - Junior,50,Systems,Issue,2 - Normal,0 - Unassigned,3,1 - Unsatisfied
3	2,1587,2 - Regular,15,Software,Request,1 - Minor,1 - Low,5,1 - Unsatisfied
4	3,925,2 - Regular,15,Access/Login,Request,2 - Normal,0 - Unassigned,0,0 - Unknown
5	4,413,4 - Management,22,Systems,Request,2 - Normal,0 - Unassigned,20,0 - Unknown
6	5,318,1 - Junior,22,Access/Login,Request,2 - Normal,1 - Low,1,1 - Unsatisfied
7	6,858,4 - Management,38,Access/Login,Request,2 - Normal,3 - High,0,0 - Unknown
8	7,1978,3 - Senior,10,Systems,Request,2 - Normal,3 - High,9,0 - Unknown

Figura 4. Primeras líneas del fichero de datos original

De este archivo se utilizarán los siguientes campos para rellenar el modelo creado en el capítulo anterior:

- **Ticket**: Número de ticket
- **Requestor**: Será utilizado como número de cliente que hace la solicitud
- **ITOwner**: Código del técnico que atiende la solicitud
- **FiledAgainst**: Categoría asignada a la solicitud (“sistemas, software, hardware,…)
- **TicketType**: Tipo de solicitud (incidencia o petición)
- **Severity**: Indicará el impacto de la solicitud

- **Prioridad**: Equivale a la prioridad asignada a la
- **daysOpen**: Días que ha estado abierta
- **Satisfaction**: Nivel de satisfacción del cliente en relación al tratamiento de la solicitud.

Se ha considerado interesante utilizar datos reales para evitar en la medida de lo posible el sesgo que se introduce al utilizar conjuntos de datos generados artificialmente, sobre todo de cara a obtener un resultado más realista en la parte final de este trabajo, donde se utilizará técnicas de aprendizaje automático para extraer valor de estos datos.

En cualquier caso, el proyecto requiere otros campos que no están presentes en ese fichero y que deberán ser generados de manera artificial. Estos datos se han generado mediante series de números aleatorios.

Para romper la uniformidad en los datos que vendría dada por una generación puramente aleatoria, donde todos los valores tienen la misma probabilidad de aparecer, y hacer más realista y variado el conjunto de datos, se han utilizado distintas distribuciones estadísticas en los valores generados.

En concreto:

- **Fecha de creación**: Este campo indica la fecha en que se genera la solicitud. Para su creación se ha generado una serie de números aleatorios que sigue una distribución beta con parámetros alfa=1,4 y beta=1, lo que da lugar a una distribución que se incrementa paulatinamente para descender un poco al final. Esto representará el volumen de creación de solicitudes durante el periodo fijado (desde enero de 2018 hasta mayo de 2019), similar a este gráfico:

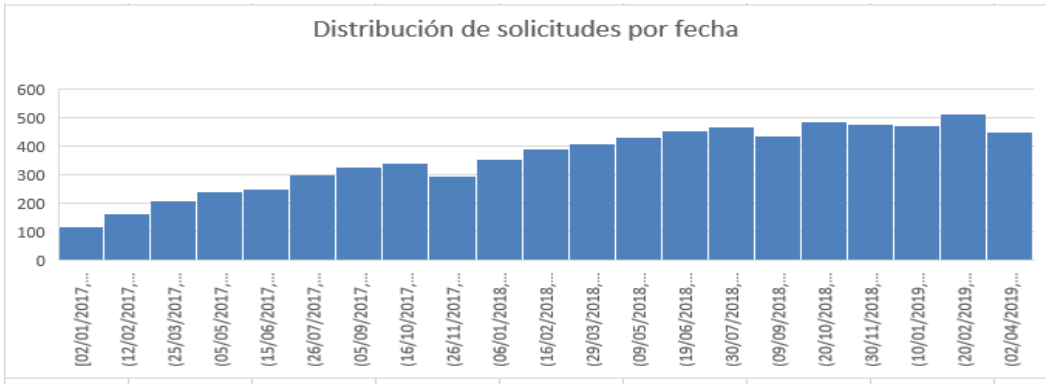


Figura 5. Distribución beta de las fechas de creación generadas aleatoriamente

En concreto, la fórmula de Excel utilizada ha sido:

=ENTERO(INV.BETA.N(ALEATORIO(); 1,4; 1; 42736; 43595))

Donde 1,4 es el parámetro alfa, 1 es el parámetro beta, 42736 es el valor correspondiente a la fecha 01/01/2017 y 43595 es el valor correspondiente a 10/05/2019.

- **Modo**: El campo Modo representa el medio a través del cual se abre la solicitud: a través de una llamada telefónica, vía mail, con un formulario web, etc. Este campo también se ha construido aleatoriamente, en este caso con unas probabilidades prefijadas que siguen esta distribución:

Tabla 5 Probabilidades usadas para la generación aleatorio de los modos de contacto

Modo	Probabilidad	Prob. Acumulada
Llamada	55%	55%
E-mail	15%	70%
Chat	5%	75%
Formulario web	25%	100%

- **Fecha de asignación**: Este campo representa la fecha en que la solicitud es asignada a un técnico para comenzar su resolución, es decir, cuando se empieza a trabajar sobre ella. Este campo se ha calculado en base a la fecha de creación de la solicitud y al número de días que se ha tardado en completar su resolución, introduciendo también cierta aleatoriedad, de la siguiente forma:

=SI([Días de resolución]>0;ALEATORIO.ENTRE([Fecha creación];[Fecha creación]+[Días de resolución]-1)+1; [Fecha creación])

- **Fecha de vencimiento**: La fecha de vencimiento indica en qué fecha debería estar resuelta la incidencia, y se basa en el SLA asignado a la solicitud. Este campo se ha calculado en función del SLA, haciendo equivaler a cada SLA un número prefijado de días máximo de resolución, según esta tabla:

SLA	Días para resolver
0	4
1	6
2	7
3	5
4	6
5	7
6	5
7	8
Otros	6

- **SLA violado**: El campo SLA violado es un indicador que puede tomar dos valores (0 – falso, -1 – verdadero) y que indica si el tiempo de resolución ha sido mayor que el marcado por el SLA. Este campo se ha calculado en base al campo Fecha de creación y al campo Tiempo de resolución.

- **Reabierta:** El campo Reabierta es un indicador que puede tomar dos valores (0 – falso, -1 – verdadero) y que indica si la avería ha necesitado ser reabierta tras su primer cierre. Este campo se ha rellenado aleatoriamente de tal forma que el 10% de las solicitudes lo tengan marcado a verdadero. Fecha de creación y el campo Tiempo de resolución.

6. Diseño de los procesos ETL

Una vez se ha diseñado la estructura del almacén de datos se lleva a cabo el diseño de los procesos de integración, que serán los encargados de extraer los datos de la fuente original, y transformarlos para darles la estructura adecuada que permita cargarlos en el almacén.

En este proyecto se utilizará como fuente de datos un fichero con formato Excel, con una estructura plana. El objetivo de esta ETL es convertir los datos de este fichero en una estructura basada en el modelo dimensional creado en el capítulo anterior.

La construcción de un almacén de datos en un entorno real suele ser un proceso largo y complejo para el que existen diferentes acercamientos sobre cómo realizar la carga y actualización de los datos. Es habitual, por ejemplo, almacenar información auxiliar que permiten tener una trazabilidad de los datos: en qué fecha se cargaron en el almacén, de qué fuente provenían, qué versión constituyen, … También es habitual realizar cargas sucesivas basadas en procesos batch que se ejecutan a intervalos definidos, (normalmente diarios) que agregan al almacén la información más reciente para ir construyendo el histórico. La gestión de las dimensiones y cómo lidiar con los valores que van cambiando en ellas también constituye objeto de un profundo análisis y preparación.

No es el objeto de este proyecto llevar a cabo la construcción de un almacén de datos completo, sino mostrar los elementos esenciales que intervienen en un proyecto de Business Intelligence. Por ello, en esta fase de construcción de la ETL se procederá a realizar una carga “limpia” en cada ejecución. Es decir, no se mantendrá ningún histórico y el proceso ETL se encargará de vaciar el almacén de datos antes de poblarlo de nuevo con los datos de los hechos y las dimensiones.

La herramienta que se utilizará es, como se vio en el capítulo dedicado al análisis y selección de las herramientas del entorno tecnológico, Pentaho Data Integration.

6.1 Creación de las dimensiones

El trabajo de ETL para la creación de las dimensiones se ha implementado en un único fichero. Los pasos que se siguen para la creación de cada dimensión son similares en todas ellas, así que se describirán de forma genérica los pasos que se siguen y luego se verán algunos ejemplos concretos.

Los pasos que constituyen este trabajo son:

1) En primer lugar, se vacía la tabla de hechos, para evitar que el proceso falle al recrear las dimensiones debido a la integridad referencial.

2) En segundo lugar, se procede a la lectura del fichero con los datos originales, que como se ha comentado se encuentra en un fichero con formato Excel contenido en una única hoja. Las columnas del fichero original se mapean a campos de Pentaho que luego serán arrastrados a los sucesivos pasos.

3) A continuación, para cada categoría, se ordenan los datos por el campo correspondiente. Este paso es necesario para poder eliminar los duplicados, acción que se lleva a cabo en el siguiente paso.

4) Se eliminan los valores duplicados para cada campo de dimensión. Para crear la tabla de dimensión en el almacén, sólo se necesita una lista "maestra" de los posibles valores en cada dimensión. Este paso y el anterior llevan a cabo una función típica que en el lenguaje SQL se ejecutaría con la sentencia SELECT campo FROM tabla GROUP BY campo.

5) A continuación, para cada valor se debe crear una clave que lo identifique de forma unívoca, distinta de la clave natural. Esta clave se conoce como clave sustituta (surrogate key). Para ello añadimos un paso al proceso que genera un contador que empieza por el valor 1, y se incrementa de uno en uno.

6) Por último, cuando ya se dispone de los valores únicos y las claves sustitutas creadas, se procede a insertar los datos en la tabla de dimensión correspondiente

dentro del almacén de datos. Este paso vacía la tabla antes de insertar los nuevos registros.

A continuación, se muestra una vista general del proceso:

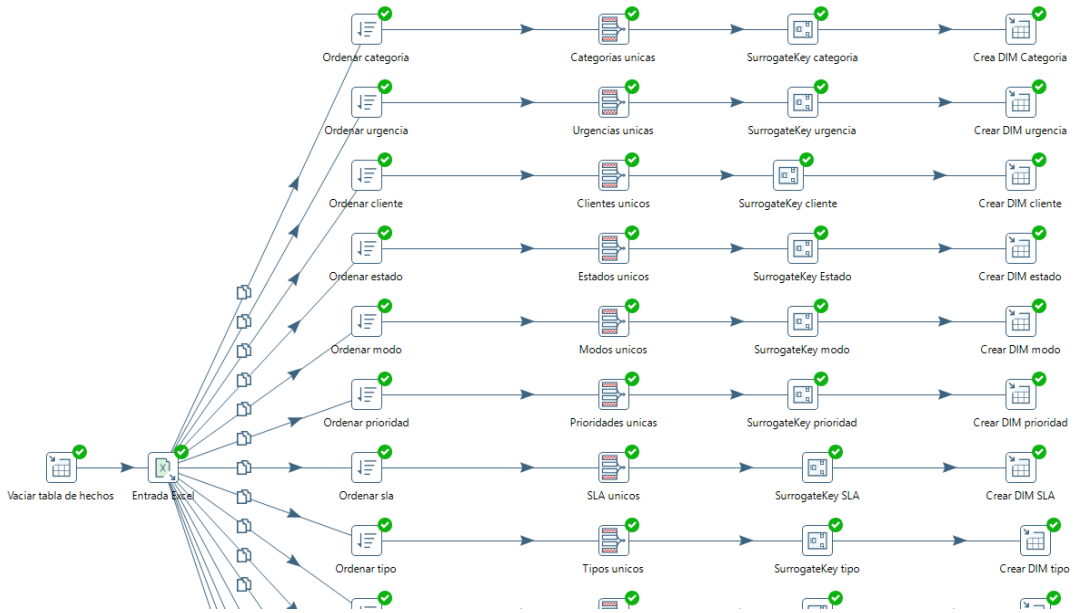


Figura 6. Vista general de la ETL de creación de dimensiones

6.1.1 Creación detallada de una dimensión

A continuación, se verá en detalle el proceso para la creación de la dimensión Cliente. Como se ha comentado, el resto de las dimensiones, excepto las de tipo fecha, se crean de forma similar, por lo que este ejemplo servirá para explicar el resto.

Figura 7. Vista de la ETL de creación de la dimensión cliente

1. **Vaciar tabla de hechos**: Este es un paso de tipo “Salida de Tabla” de Pentaho, que sirve para volcar datos en una tabla. Dispone de un parámetro para indicar que previamente se quiere vaciar la tabla. Si se activa esta opción y no se le pasa ningún dato de salida, el efecto que tiene es simplemente el de vaciar la tabla.

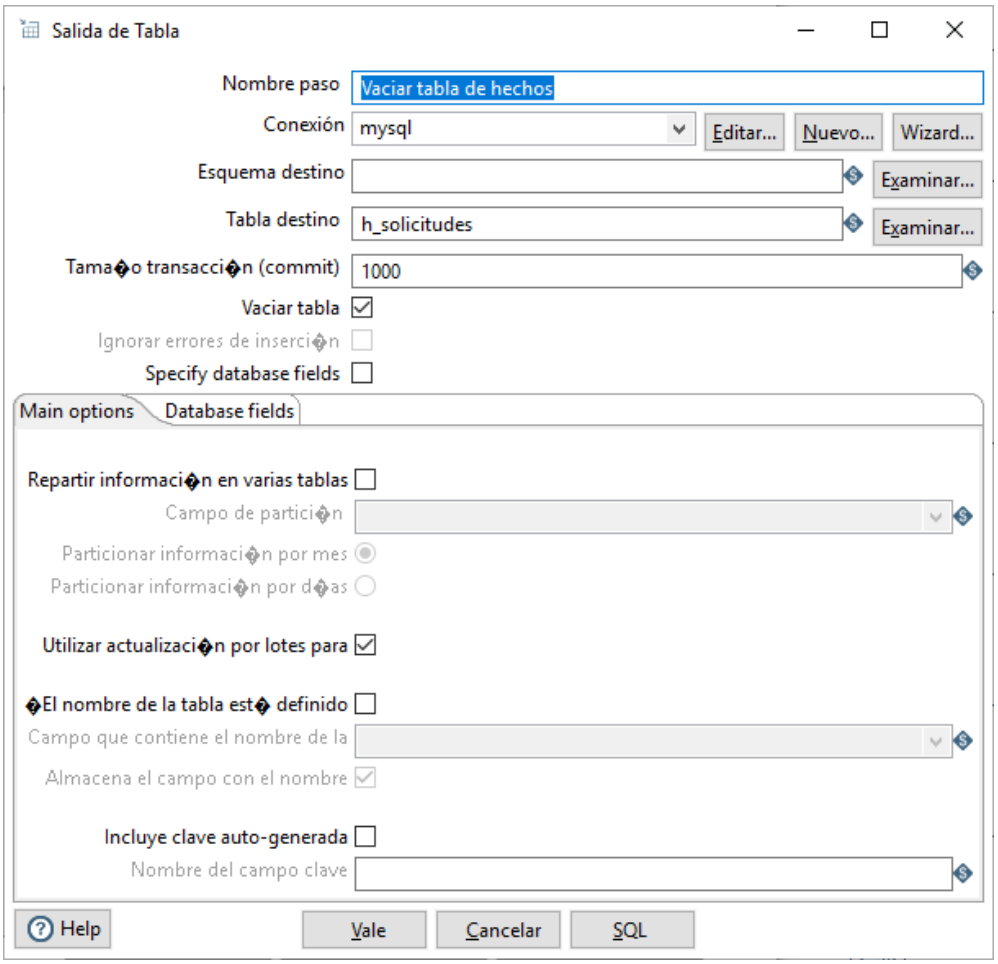


Figura 8. Paso para vaciar tabla de hechos

2. **Entrada Excel**: Este paso se encarga de leer el fichero original (Dataset.xlsx) y hacer el mapeo de los campos.

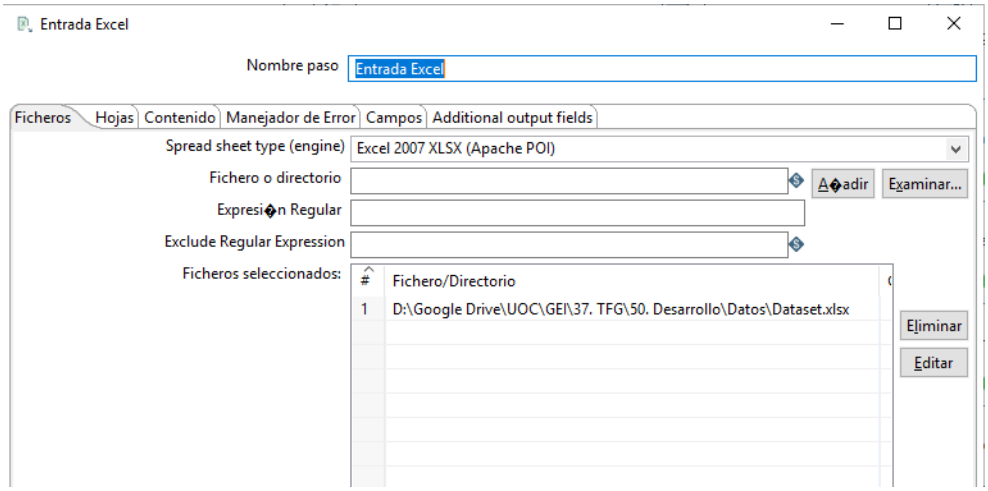
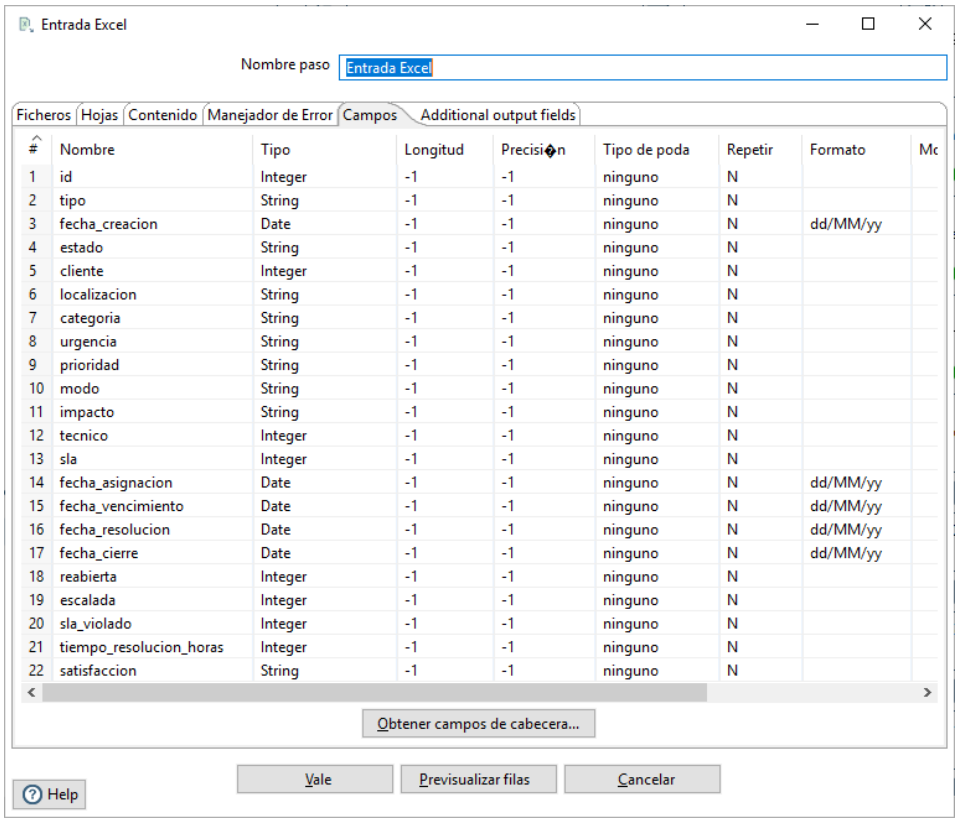


Figura 9 Lectura de fichero origen (1)



#	Nombre	Tipo	Longitud	Precisi&#65533;n	Tipo de poda	Repetir	Formato	Mc
1	id	Integer	-1	-1	ninguno	N		
2	tipo	String	-1	-1	ninguno	N		
3	fecha_creacion	Date	-1	-1	ninguno	N	dd/MM/yy	
4	estado	String	-1	-1	ninguno	N		
5	cliente	Integer	-1	-1	ninguno	N		
6	localizacion	String	-1	-1	ninguno	N		
7	categoria	String	-1	-1	ninguno	N		
8	urgencia	String	-1	-1	ninguno	N		
9	prioridad	String	-1	-1	ninguno	N		
10	modo	String	-1	-1	ninguno	N		
11	impacto	String	-1	-1	ninguno	N		
12	tecnico	Integer	-1	-1	ninguno	N		
13	sla	Integer	-1	-1	ninguno	N		
14	fecha_asignacion	Date	-1	-1	ninguno	N	dd/MM/yy	
15	fecha_vencimiento	Date	-1	-1	ninguno	N	dd/MM/yy	
16	fecha_resolucion	Date	-1	-1	ninguno	N	dd/MM/yy	
17	fecha_cierre	Date	-1	-1	ninguno	N	dd/MM/yy	
18	reabierta	Integer	-1	-1	ninguno	N		
19	escalada	Integer	-1	-1	ninguno	N		
20	sla_violado	Integer	-1	-1	ninguno	N		
21	tiempo_resolucion_horas	Integer	-1	-1	ninguno	N		
22	satisfaccion	String	-1	-1	ninguno	N		

Figura 10 Lectura de fichero origen (2)

3. **Ordenar cliente**: Este es un paso de tipo “Ordenar filas” que lleva a cabo una ordenación alfanumérica de los valores del campo; destacar el hecho de que en este punto ya nos deshacemos del resto de campos y sólo se continua con el que es objeto de la dimensión que queremos crear.

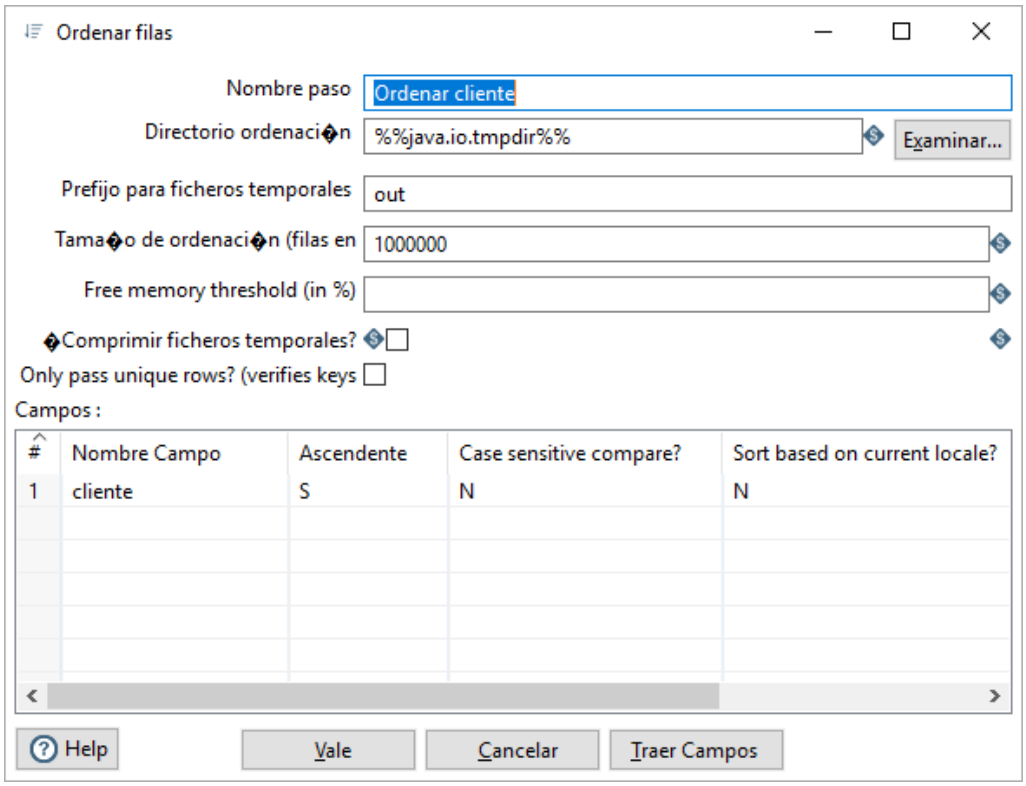


Figura 11 Paso para ordenar campo Cliente

4. **Surrogate Key Cliente**: Este es un paso de tipo “Obtener valor de la secuencia de base de datos” que sirve para recuperar un valor de una secuencia implementada en la base de datos. Sin embargo, también puede ser utilizado para generar un contador desde el propio entorno de Pentaho, que es como se utiliza en esta ocasión, generando un contador que tiene como valor inicial el número 1 y se incrementa a intervalos de 1. Este contador se convertirá en el identificador de las filas de las tablas de dimensiones en el almacén de datos.

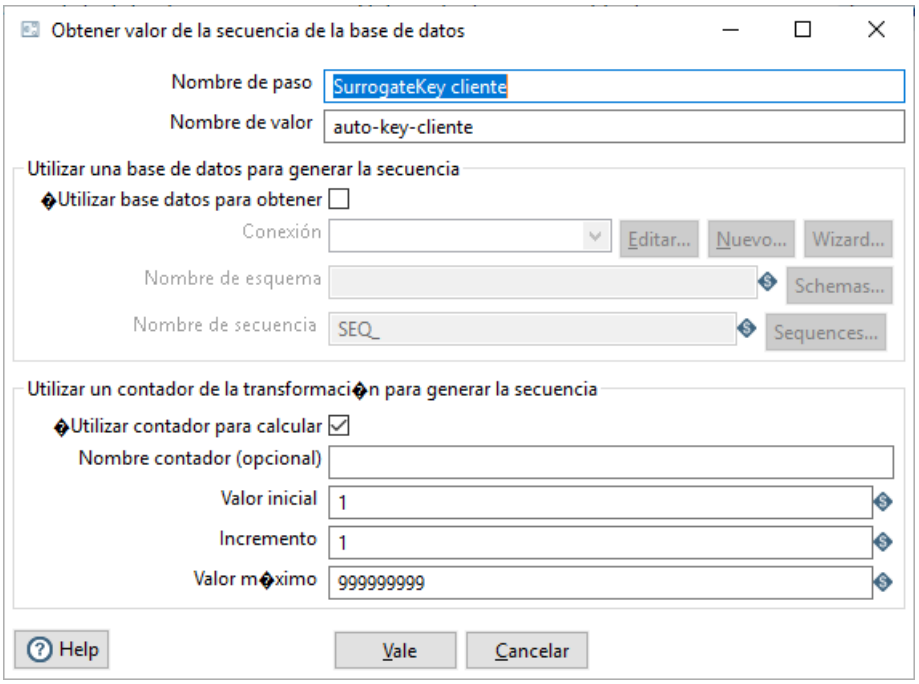


Figura 12. Paso para generar la clave sustituta de Cliente

5. **Crear DIM Cliente**: Finalmente, se procede a volcar los datos en la tabla de la base de datos de la dimensión correspondiente, mapeando los campos que tenemos en el flujo de Pentaho con los datos de la tabla.

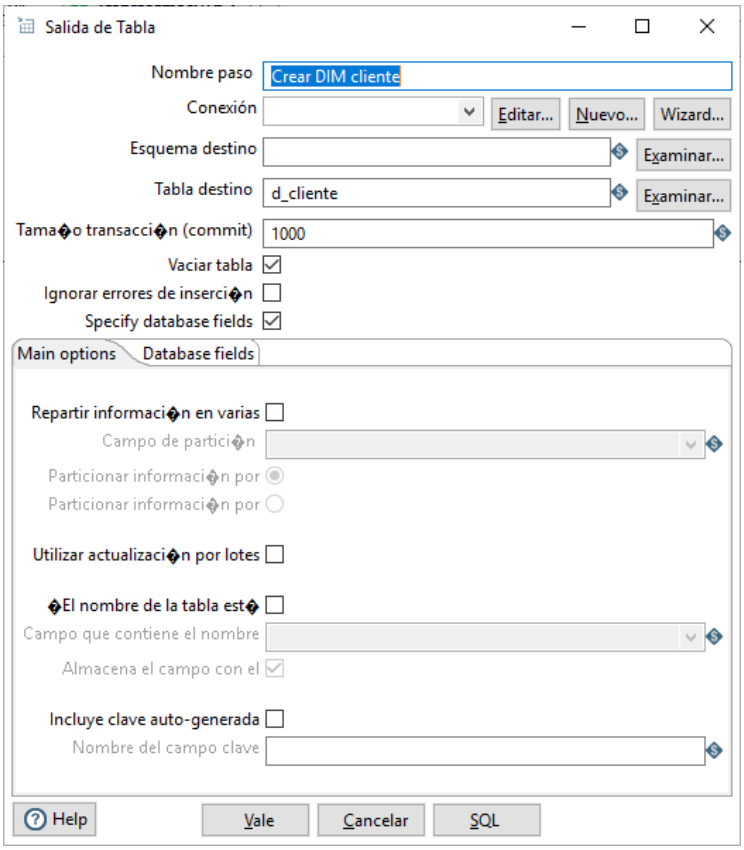


Figura 13. Paso para volcar datos en tabla de dimensión (1)

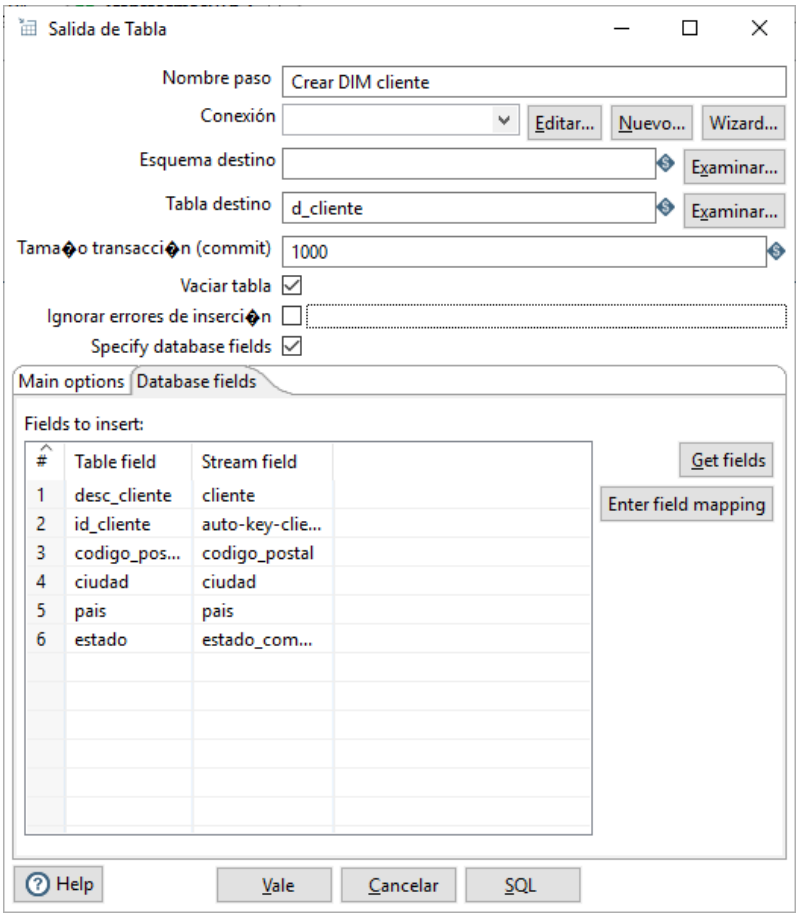


Figura 14. Paso para volcar datos en tabla de dimensión (2)

6.1.2 Creación de una dimensión de fecha

Las dimensiones de fecha se crean de forma similar al resto, pero incorporan un paso adicional para crear la jerarquía de datos en la dimensión. Las jerarquías son estructuras lógicas que utilizan niveles ordenados como un medio para organizar los datos. Se puede usar una jerarquía para definir la agregación de datos. Por ejemplo, en una dimensión temporal, como es el caso, una jerarquía puede agregar datos desde el nivel del mes al nivel del trimestre hasta el nivel del año.

El proceso de general para crear una dimensión de fecha es como el siguiente:

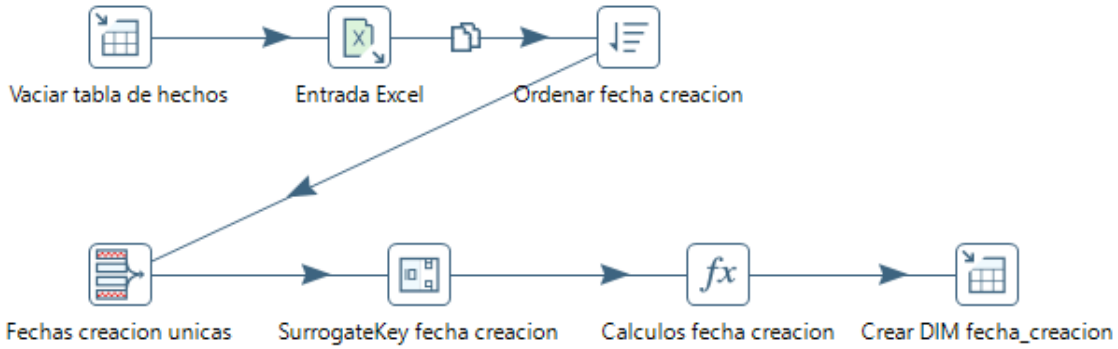


Figura 15. Vista de la ETL para generar una dimensión de tipo fecha

Como se puede apreciar, los pasos son idénticos al del resto de dimensiones, pero incorpora un paso adicional llamado “Cálculos fecha creación”.

Este paso se encarga de calcular los distintos niveles de la jerarquía de fecha, añadiendo un campo para el año, el trimestre, la semana el mes y el día que son calculados mediante funciones del propio Pentaho.

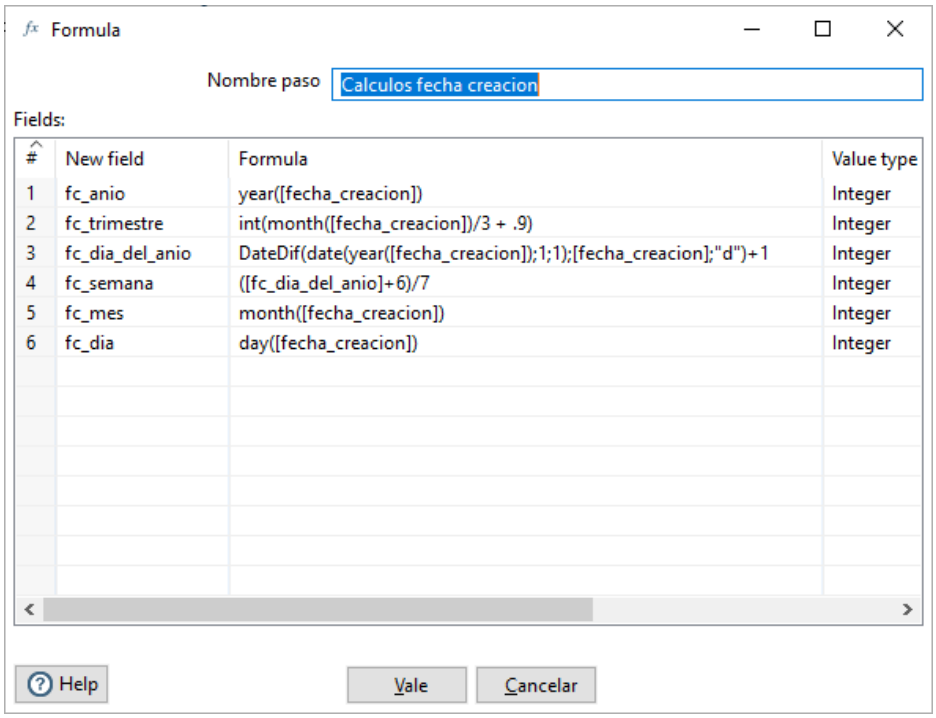


Figura 16. Cálculos para generar jerarquía de fecha

6.2 Creación de la tabla de hechos

El proceso para la creación de la tabla de hechos se compone básicamente de dos elementos:

1. En primer lugar, se buscan las claves sustitutas (surrogate keys) en las tablas de dimensiones. Como se recordará del modelo dimensional detallado en capítulos anteriores, la tabla de hechos se compone esencialmente de las claves foráneas que enlazan con las dimensiones y de las métricas que definen el hecho. Cuando se lee un campo de dimensión del fichero origen, se debe buscar su correspondiente clave en la tabla de dimensiones, puesto que ésta es la que almacenaremos en la tabla de hechos.

2. Una vez hemos recopilado todas las claves, procedemos a insertar el registro en la tabla de hechos del almacén de datos.

En una vista general, el proceso es como el siguiente:

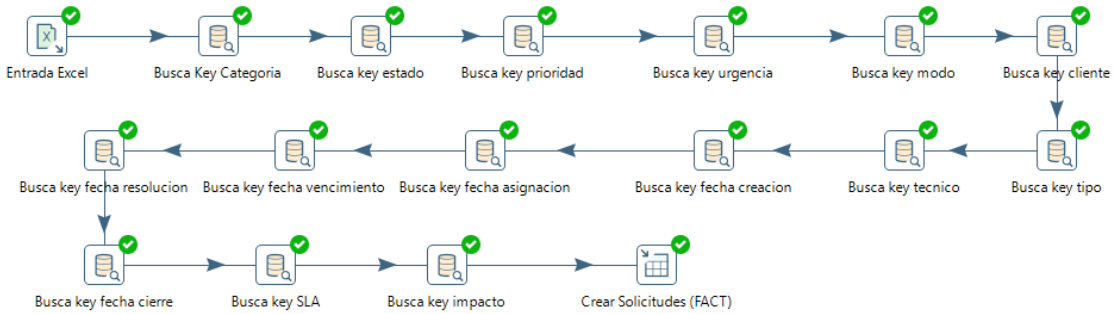


Figura 17. Vista general de la ETL para generar la tabla de hechos

Salvo el primer y último paso, el resto realizan una acción similar con cada una de las dimensiones. Por tanto, se verá en detalle sólo uno de ellos a modo de ejemplo.

1. **Entrada Excel**: Este es paso es idéntico al usado en la creación de las dimensiones. Se encarga de leer el fichero origen en formato Excel y hacer un mapeo de las columnas a campos de Pentaho.

2. **Busca Key <dimensión>:** Como se ha comentado, todos estos pasos realizan la misma acción: buscan en la tabla de dimensión correspondiente la clave primaria que se corresponde con el dato leído del fichero y la devuelven al flujo para que sea utilizada en pasos posteriores.

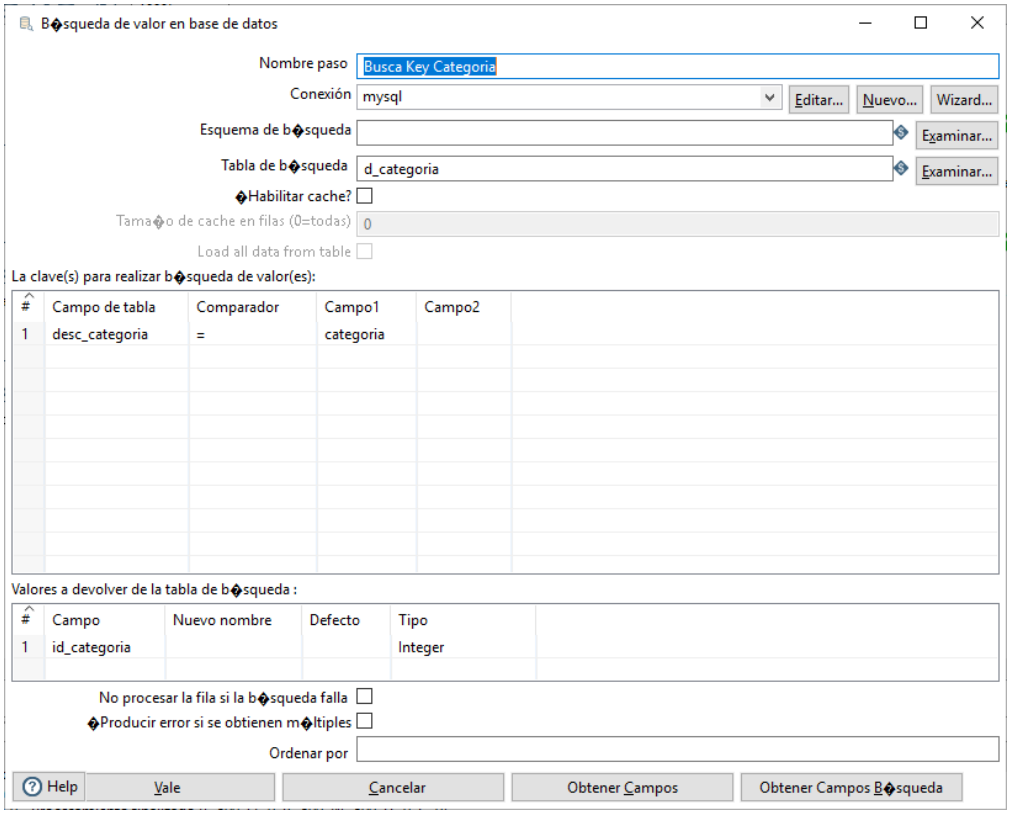
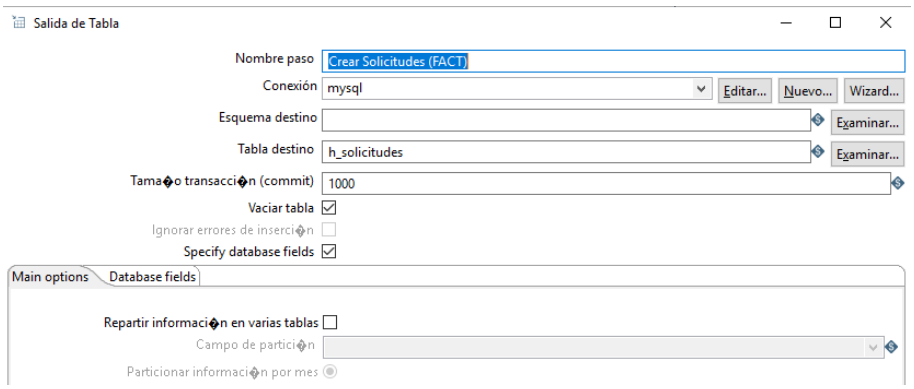


Figura 18. Paso para localizar la clave sustituta en la tabla de dimensión

3. **Crear solicitudes (FACT)**: El último paso, recoge los datos del flujo, al que se han ido incorporando las claves sustitutas junto con los valores originales leídos del fichero y los vuelca en la tabla de hechos del almacén de datos, previo vaciado de ésta.



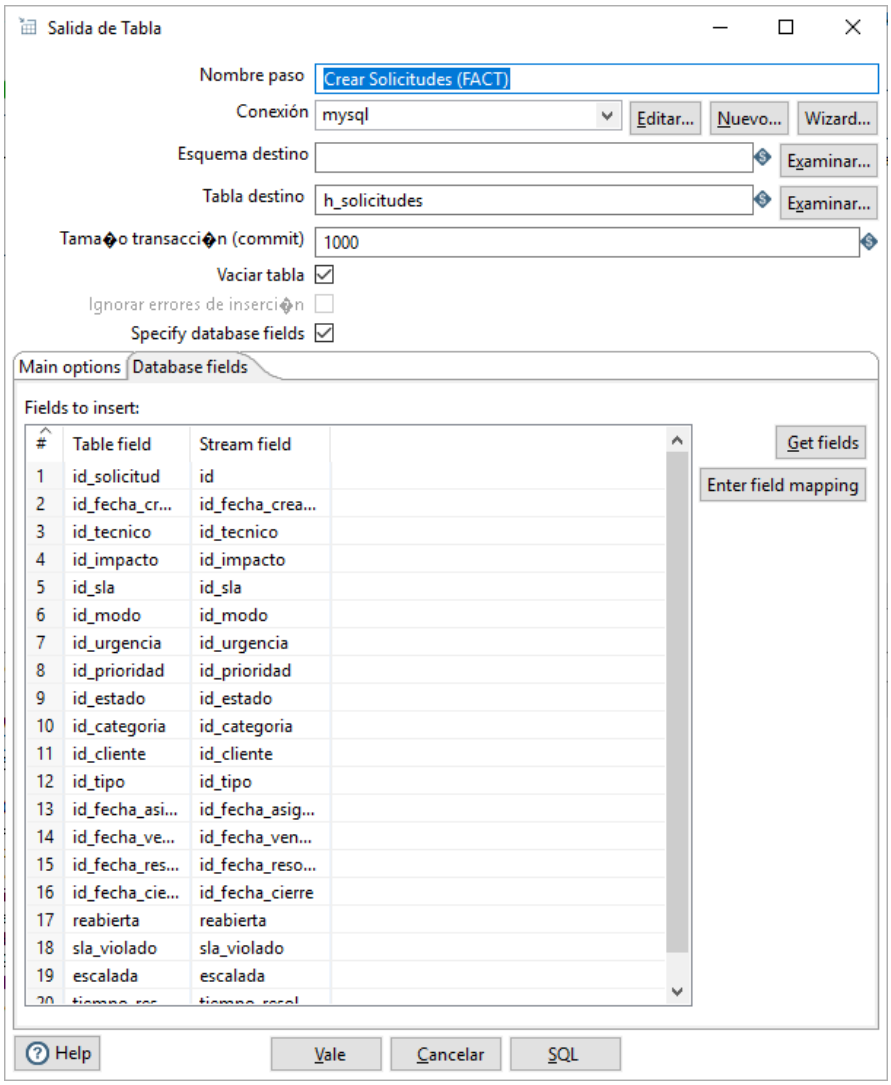


Figura 19. Paso para el volcado de datos en la tabla de hechos

7. Informes y visualizaciones

Éste es un capítulo de especial relevancia, porque en él se acaban encajando todas las piezas. Aquí es donde se hace aflorar la información que estaba oculta en los datos, convirtiéndola en elemento fundamental para la toma de decisiones, que es el objetivo de cualquier sistema de Inteligencia de Negocio.

Se verá cómo, sobre la base construida en fases anteriores, se puede utilizar una herramienta de análisis y visualización para explorar los datos y extraer conclusiones. A partir de una serie de preguntas analíticas que cualquier responsable de un Service Desk podría hacerse, se verá como los informes diseñados permiten responden a esas preguntas y cómo esas respuestas podrían ligarse a acciones de mejora del servicio.

7.1 El entorno de Power BI

Como se vio en el apartado de análisis y selección de herramientas, la aplicación escogida para la visualización de los datos será Microsoft Power BI.

Esta familia de productos se compone de varias aplicaciones que se complementan entre sí para ofrecer un abanico completo de funcionalidades para la elaboración y distribución de informes y cuadros de mando. Entre otros productos, incluye:

a) La aplicación **Power BI Desktop**, una herramienta local orientada a la creación de informes; está enfocada a los analistas y profesionales de TI que crean informes y paneles para los usuarios de negocio. Esta herramienta es gratuita, aunque existen dos versiones adicionales de pago (Pro y Premium) que proporcionan funcionalidades extendidas.

b) El **servicio Power BI** es un servicio en la nube que facilita la compartición de informes y paneles, mediante funciones de publicación y de trabajo colaborativo.

c) La aplicación **Power BI Mobile** es una aplicación de móvil pensada en la consulta de los datos exclusivamente.

El entorno de Power BI Desktop se estructura en tres apartados:

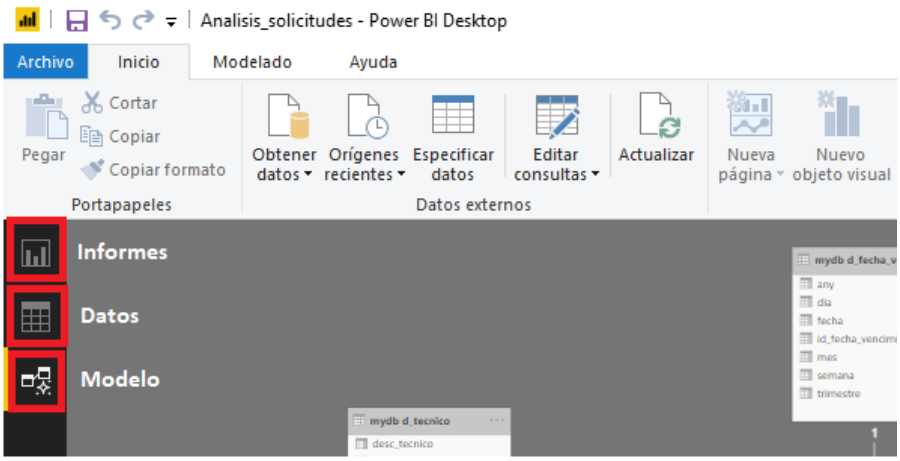


Figura 20. Vista parcial del entorno de Power BI Desktop

1) **Datos**

Desde aquí se proveen las funciones para conectar a los orígenes de datos y llevar a cabo la limpieza y transformación de los mismos. A través del Editor de Consultas se pueden realizar operaciones típicas de una herramienta ETL, como la combinación o separación de columnas, la extracción de valores, la validación de datos o la aplicación de cálculos y funciones.

2) **Modelo**

En este punto se establecen las relaciones entre las tablas para crear el modelo de datos y se asignan los roles para administrar la seguridad de los mismos.

3) **Informes**

En este apartado se proporcionan las herramientas para crear visualizaciones y paneles, que es el objetivo último de esta apliación.

En el caso concreto de este trabajo, se ha realizado la conexión a la base de datos MySQL donde se ha creado el almacén de datos y se ha replicado el modelo dimensional, tal como se puede apreciar en la siguiente figura:

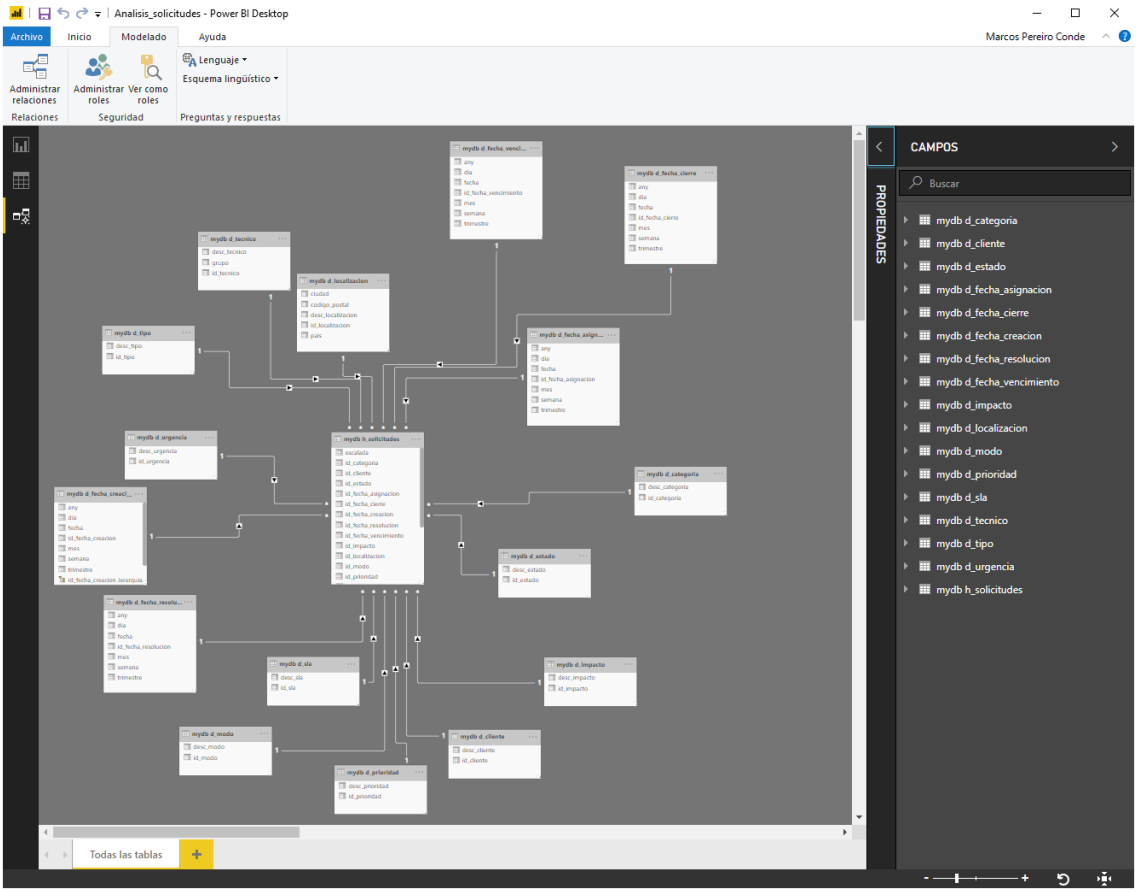


Figura 21. Modelo de datos dimensional creado en Power BI

Una vez que se ha establecido la conexión con el almacén de datos y se ha creado el modelo de datos, ya se puede pasar a crear los informes y las visualizaciones de los datos que permitan responder a las cuestiones analíticas para explicar comportamientos y extraer conclusiones.

7.2 Preguntas analíticas

- ¿Se cumplen los acuerdos de nivel de servicio (SLA)?
- ¿Existen diferencias significativas en el volumen de incidencias según su categoría?
- ¿Hay recursos infrautilizados?
- ¿Se corresponde el volumen de incidencias por distintas variables (categoría, fecha/hora) con lo planificado para dimensionar los equipos?
- ¿Se puede predecir la demanda para dimensionar los equipos?
- ¿Se encuentran patrones en las incidencias que permitan adelantar problemas?
- ¿Los criterios para la clasificación, priorización y escalado de las incidencias están siendo correctos?

7.3 Análisis del volumen y distribución de tickets

Este informe proporciona algunos datos básicos que ayudan a fijar un contexto y a conocer las magnitudes de las dimensiones más importantes, lo que facilitará la interpretación del resto de informes.

Aquí se encuentra el volumen total de tickets tratados, el número de clientes, número de técnicos y periodo al que corresponde el análisis. También se muestra la distribución de los tickets en función de varias dimensiones: por tipo de ticket, por categoría asignada, por fecha de apertura y por día de la semana.

Figura 22. Informe de volumen y distribución de tickets

No por ser básicos, estos datos dejan de aportar información muy útil de cara a la gestión del Service Desk. En el caso concreto de estudio se aprecia, por ejemplo, que la apertura de tickets durante la semana no es uniforme, sino que sigue una tendencia decreciente: el volumen de tickets abiertos los lunes o los martes es mucho mayor que los abiertos los jueves o los viernes. También se puede apreciar que los fines de semana sigue habiendo una actividad relativamente importante.

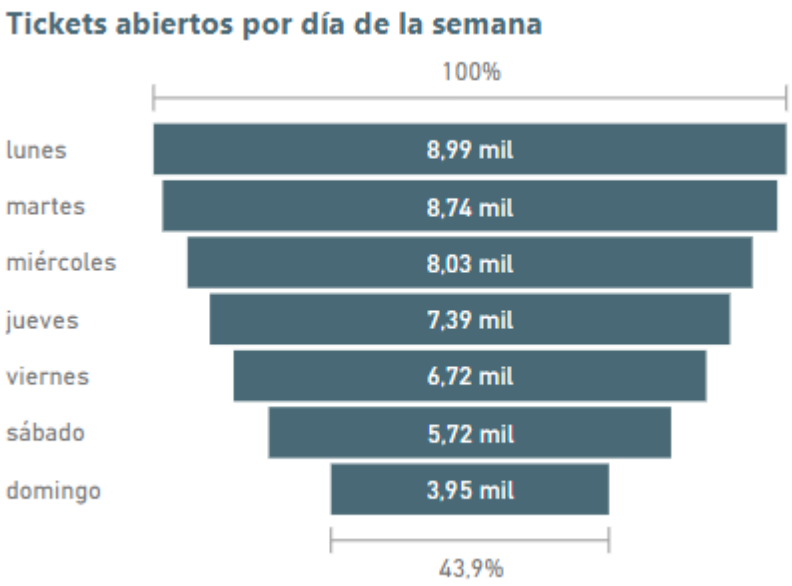


Figura 23. Gráfico de tickets abiertos por día de la semana

Este dato es muy importante de cara a planificar las agendas de los técnicos y los turnos, ya que el personal necesario para atender el servicio de forma efectiva a principios de semana es mayor que en otros días.

De la misma forma, en el gráfico de volumen de tickets por mes, se puede apreciar que la tendencia en el volumen de tickets es ligeramente creciente, sin grandes saltos entre un mes y otro.

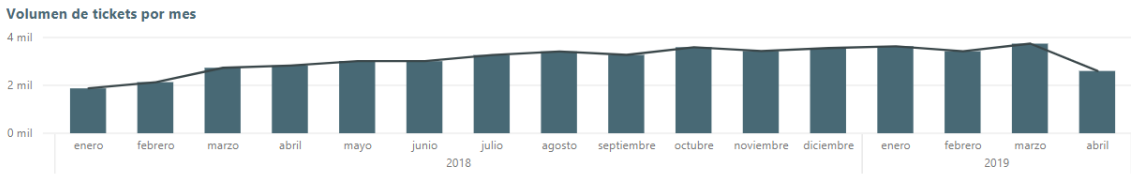


Figura 24. Gráfico de tickets abiertos por mes

Desde la perspectiva de negocio, este dato es importante porque el volumen de tickets tratados crece y ello significa que el negocio también crece (no hay que olvidar que este trabajo se sitúa en el ámbito de un proveedor de TI cuya actividad y fuente de ingresos consiste en tratar incidencias y consultas de sus clientes). Sin embargo, desde una perspectiva más operativa, también se puede ver que la actividad no decrece durante el mes de julio y agosto, ni tampoco en diciembre, meses en los que generalmente hay menos movimiento debido a las vacaciones. Este es otro dato importante a tener en cuenta de cara a esa planificación y dimensionamiento de los equipos a la que se aludía anteriormente.

En la misma línea se sitúan los otros dos gráficos que muestran la distribución por tipo de ticket y por categoría o área de actuación. En cuanto el tipo de ticket se observa que el volumen de solicitudes (consultas o peticiones de los clientes) es casi el triple que el de incidencias.

Por otro lado, se comprueba que los tickets más frecuentes se corresponden con incidencias y solicitudes relacionadas con Sistemas, seguidas de aquellas que se refieren a problemas o consultas con al Acceso y el Login (permisos, cambios de contraseñas, problemas de autenticación, …).

En definitiva, los hechos que revela este informe son de vital importancia a la hora de conformar los equipos: decidir cuántos perfiles se tienen de cada especialidad, planificar los turnos y las vacaciones, etc.

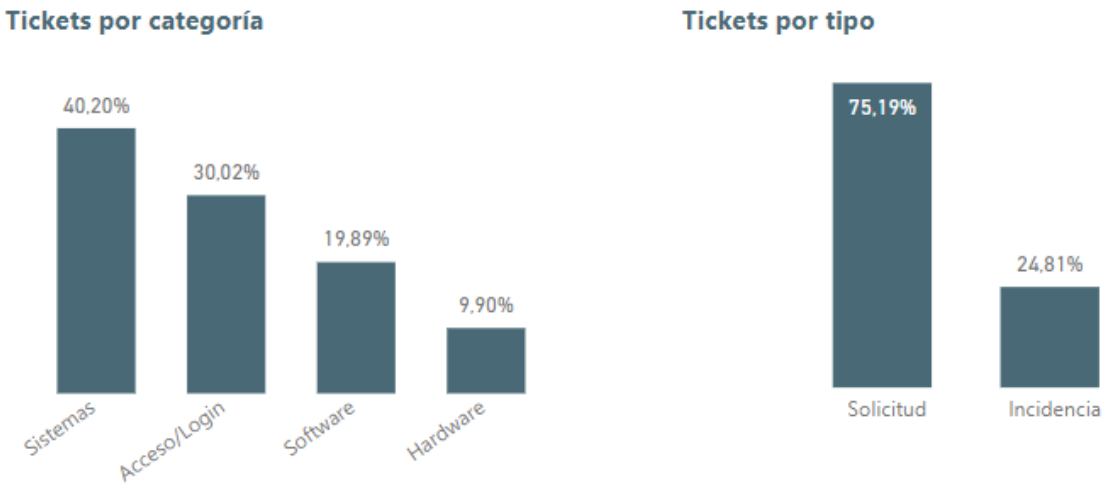


Figura 25. Gráfico de tickets por categoría y tipo

Toda esta información se podría obtener a través de una hoja de cálculo o mediante informes más tradicionales. Sin embargo, una de las ventajas de usar sistemas de presentación como Power BI es que facilitan la exploración y el análisis de los datos a personas sin conocimientos técnicos avanzados, permitiendo que estas se centren en lo realmente importante para sus objetivos. Gracias a las posibilidades de filtrado que proporciona esta herramienta, es posible ir refinando sucesivamente la consulta para obtener una visión de los datos más concreta.

Por ejemplo, si se quisiera comprobar si los patrones que se han descubierto anteriormente se cumplen también para el subconjunto de tickets de la categoría de Software, bastaría con seleccionar ese elemento en el gráfico y automáticamente todos los datos del informe se actualizarían con esa restricción, como se puede ver en la siguiente figura.

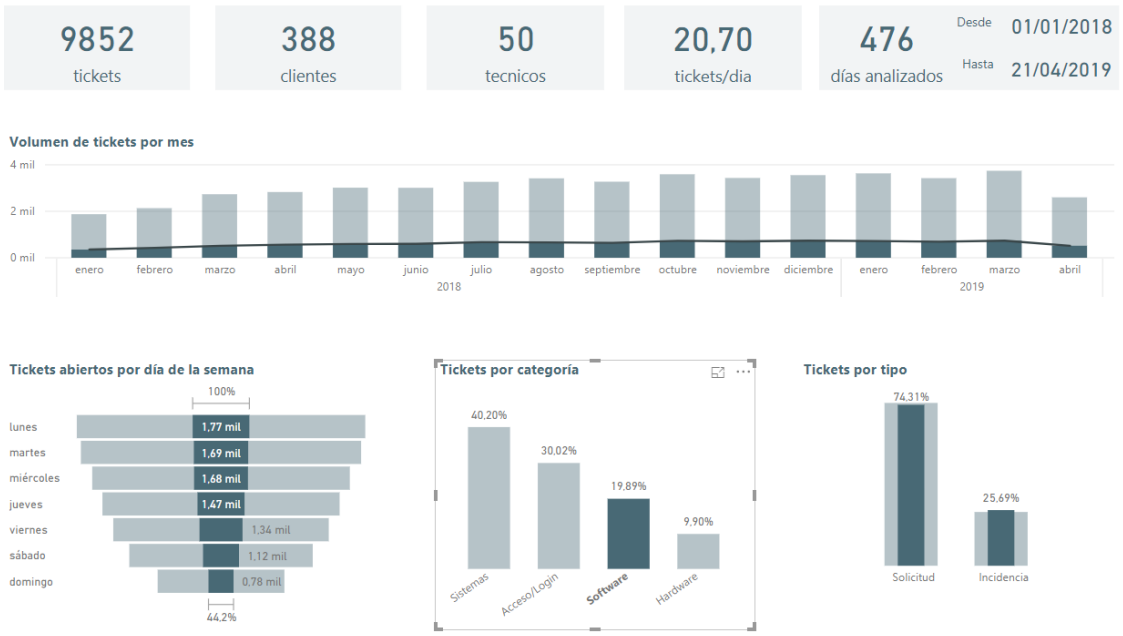


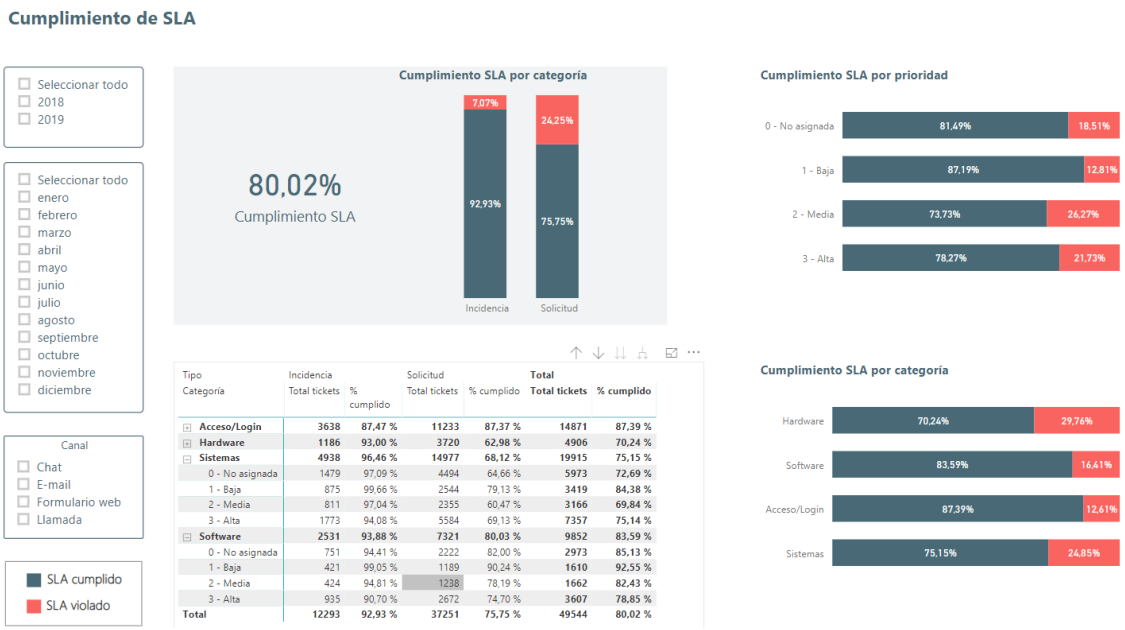
Figura 26. Ejemplo de filtrado de datos por selección

7.4 Análisis de cumplimiento de SLA

Una de las preguntas que cualquier encargado de gestionar un Service Desk se hará en primer lugar es si se están cumpliendo los SLA. Los SLA (Service Level Agreements o Acuerdos de Nivel de Servicio en español) constituyen un elemento crítico en el funcionamiento de un Service Desk, ya que representan un compromiso entre el cliente y el proveedor sobre el nivel esperado de desempeño del servicio y ayudan a medir su calidad.

Este nivel de desempeño se puede determinar en función de múltiples variables, aunque es habitual encontrar entre ellas las que hacen referencia al tiempo de primera respuesta y al tiempo de resolución, es decir, cuánto tardará el cliente en ser atendido cuando tiene un problema y cuándo puede esperar que el problema esté resuelto.

El siguiente informe trata de responder a esas preguntas y presenta los aspectos más relevantes en cuanto el cumplimiento del SLA.



Tipo / Categoría	Incidencia Total tickets	Incidencia % cumplido	Solicitud Total tickets	Solicitud % cumplido	Total Total tickets	Total % cumplido
Acceso/Login	**3638**	**87,47 %**	**11233**	**87,37 %**	**14871**	**87,39 %**
Hardware	**1186**	**93,00 %**	**3720**	**62,98 %**	**4906**	**70,24 %**
Sistemas	**4938**	**96,46 %**	**14977**	**68,12 %**	**19915**	**75,15 %**
0 - No asignada	1479	97,09 %	4494	64,66 %	5973	72,69 %
1 - Baja	875	99,66 %	2544	79,13 %	3419	84,38 %
2 - Media	811	97,04 %	2355	60,47 %	3166	69,84 %
3 - Alta	1773	94,08 %	5584	69,13 %	7357	75,14 %
Software	**2531**	**93,88 %**	**7321**	**80,03 %**	**9852**	**83,59 %**
0 - No asignada	751	94,41 %	2222	82,00 %	2973	85,13 %
1 - Baja	421	99,05 %	1189	90,24 %	1610	92,55 %
2 - Media	424	94,81 %	1238	78,19 %	1662	82,43 %
3 - Alta	935	90,70 %	2672	74,70 %	3607	78,85 %
Total	**12293**	**92,93 %**	**37251**	**75,75 %**	**49544**	**80,02 %**

Figura 27. Informe de cumplimiento de SLA

En primer lugar se destaca el porcentaje global de cumplimiento de SLA. Como se verá más adelante, este dato será uno de los indicadores que conformarán el cuadro de mando, debido a su importancia para evaluar rápidamente la calidad del servicio. Sin embargo, al analizarlo más en detalle se pueden encontrar importantes diferencias en el grado de cumplimiento alcanzado en los distintos grupos de tickets.

El más llamativo es el que se observa al comparar el cumplimiento de SLA entre las incidencias y las solicitudes, mucho mayor en las primeras. Preguntarse el por qué de este dato puede llevarnos a detectar puntos de mejora en la organización del Service Desk:

- Si solicitudes e incidencias son gestionadas por equipos distintos, ¿puede estar infradimensionado el equipo que atiende las solicitudes?
- Si las atiende un único equipo, ¿se están priorizando las incidencias por encima de las solicitudes?
- Se atienden los tickets en el orden asignado por el sistema o se está pervirtiendo este orden en función del cliente que “más grita” (normalmente los clientes que tienen una incidencia en el servicio – algo ha dejado de funcionar - ejercen más presión que los esperan algún servicio o una respuesta a una consulta).

Si se observa esta variable por la prioridad asignada a cada ticket, se descubre que el porcentaje de cumplimiento más alto se da en incidencias con una prioridad baja, siendo menor en las de prioridad media y alta. Esto lleva a pensar nuevamente en si la asignación de técnicos en función de las prioridades se está realizando correctamente o necesita un ajuste.

Se estudia a continuación el porcentaje de cumplimiento de SLA en función del técnico. Llama la atención las importantes diferencias que hay entre unos y otros, donde los mejores tienen cerca del 100% de cumplimiento y los peores sólo cumplen el SLA en el 40% de los tickets tratados.

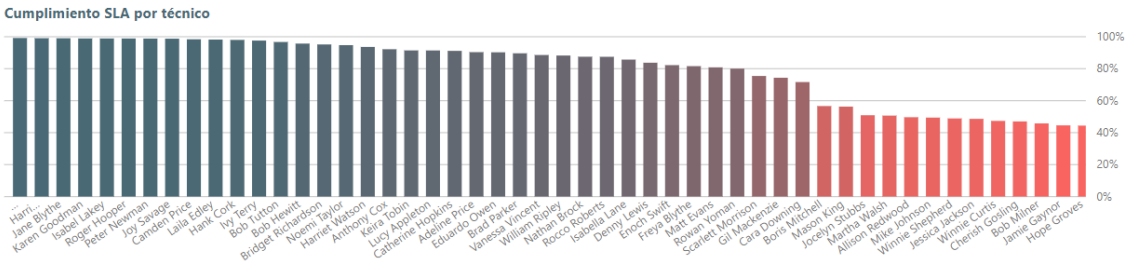


Figura 28. Informe de cumplimiento de SLA por técnico

A la vista de esa desigualdad cabe preguntarse si las diferencias se deben a la categoría o prioridad de los tickets que les han sido asignados. Como se ha visto anteriormente, los tickets de prioridad baja presentan un grado de cumplimiento mayor que los de prioridad alta; de la misma forma, los tickets y solicitudes del área de Sistemas incumplen en más casos que el resto de áreas. En definitiva: ¿los técnicos que cumplen menos los SLA lo hacen porque se les asignan tickets donde es más dificil cumplir los plazos?

Gracias a las posibilidades de selección y filtrado que ofrece Power BI se puede seleccionar el grupo de técnicos de más a la derecha – los que tienen un cumplimiento de SLA más bajo – y ver cómo se refleja esto en la distribución por tickets.

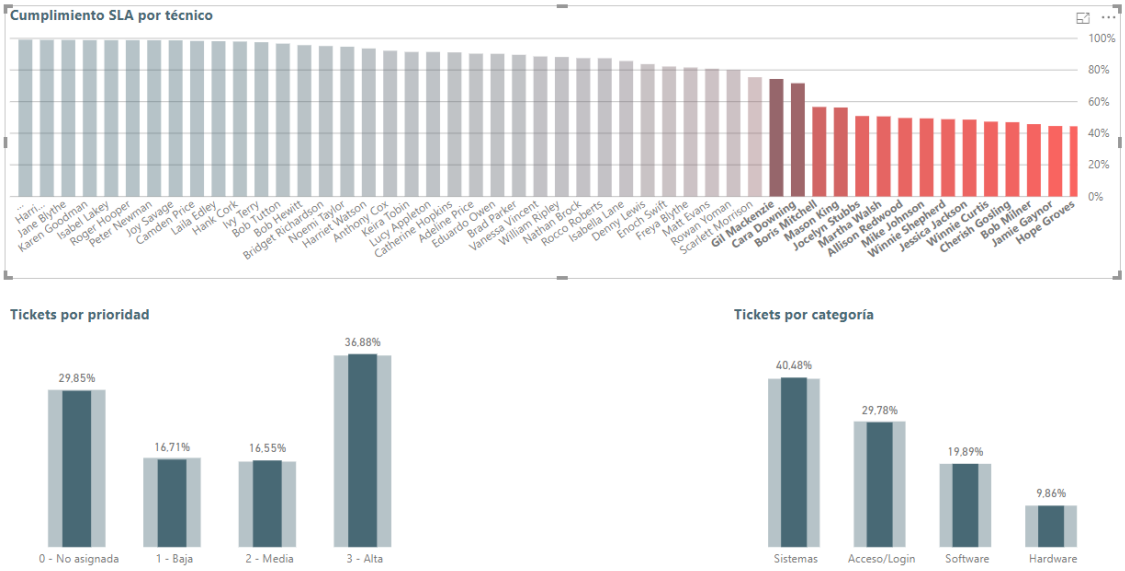


Figura 29. Comparación de cumplimiento seleccionando un grupo

Como se puede apreciar, la proporción de tickets por prioridad para este grupo es prácticamente idéntica al total. Ocurre lo mismo con la distribución de tickets por categoría. Por tanto, se puede concluir que el grado de incumplimiento de SLA de estos técnicos no tiene relación con la asignación de tickets y habrá que buscar las explicaciones en otros puntos.

7.5 Análisis de la satisfacción del cliente

El siguiente informe está diseñado para evaluar la satisfacción del cliente desde diferentes perspectivas. El dato de satisfacción se obtiene de las respuestas que dan los propios clientes a una pequeña encuesta que se realiza tras finalizar cada actuación. Los clientes pueden valorar el servicio recibido en una escala de 1 a 3 (1 – Insatisfecho, 2 – Satisfecho, 3 – Muy satisfecho). El valor 0 queda para aquellos que no responden a la preguntan o no dan una valoración.

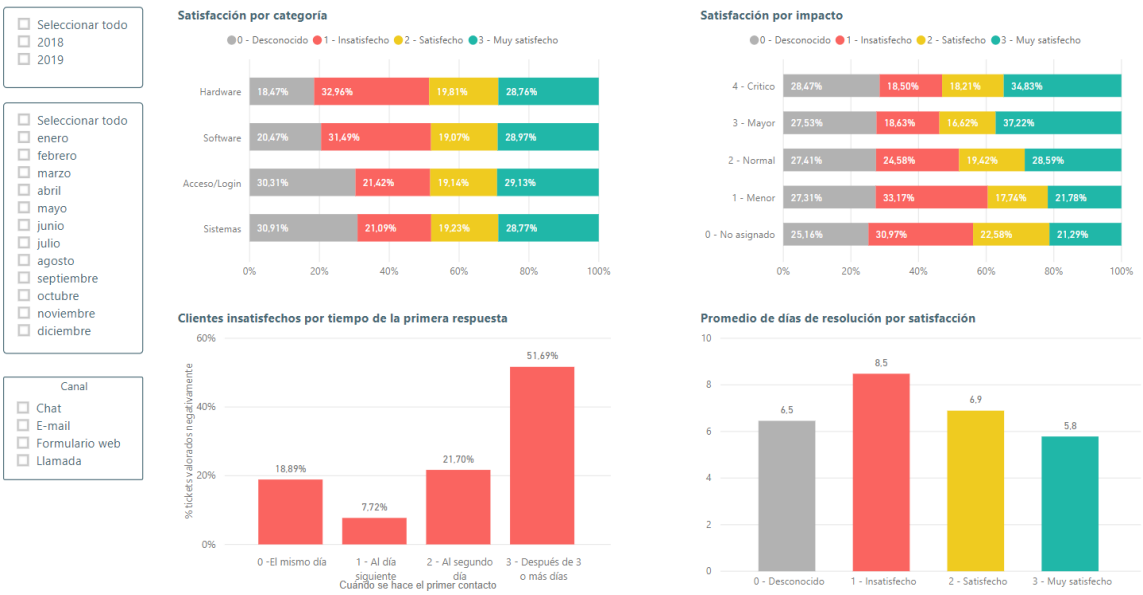


Figura 30. Informe de satisfacción de cliente

A modo de ejemplo, se puede observar que el porcentaje de clientes insatisfechos en los tickets que han sido cualificados con un impacto Menor o donde el impacto no ha sido especificado es claramente superior al resto. Este hecho podría apuntar a problemas en el proceso de cualificación y priorización de los tickets, ya que la calificación del impacto tiene influencia en la prioridad del ticket y, en consencuencia, en los plazos que se deben cumplir.

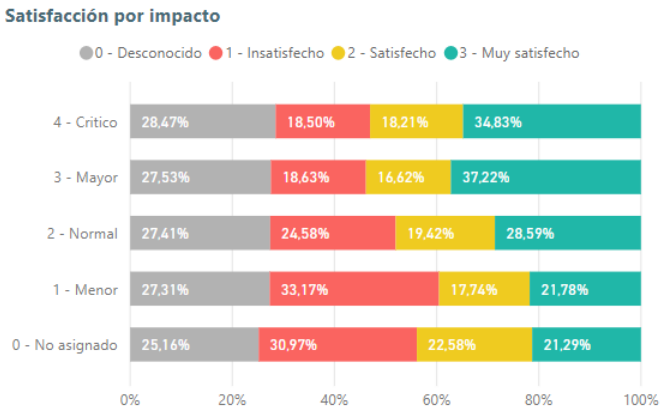


Figura 31. Satisfacción por impacto

Se advierten también diferencias importantes en la satisfacción del cliente según el área a la que corresponda el ticket tratado. Más de un 30% de los tickets asociados a incidencias o consultas de Hardware y Software tienen una valoración negativa y eso obligaría a hacer un análisis más profundo para encontrar la causa de esa diferencia.

Analizando la relación entre la satisfacción y el tiempo de primera respuesta (el tiempo que transcurre desde que un cliente abre un ticket hasta que éste es asignado a un técnico para su resolución) se observa que el porcentaje de clientes insatisfechos aumenta a medida que aumenta este tiempo, llegando al punto de que más de la mitad de los tickets que son contestados después de 3 días acaban teniendo una valoración negativa. Esto no hace más que confirmar la importancia de esta variable en la calidad percibida por el cliente, y señala un punto crítico para mejorar la satisfacción del cliente de forma sustancial: aumentar el tiempo de primera respuesta.

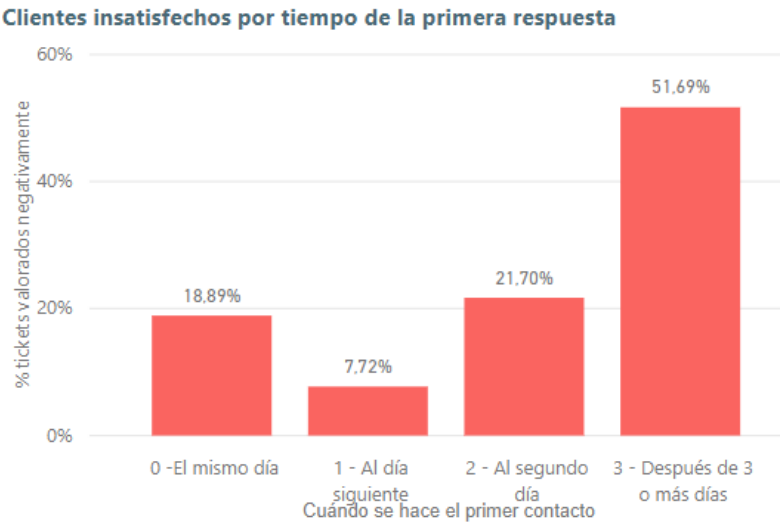
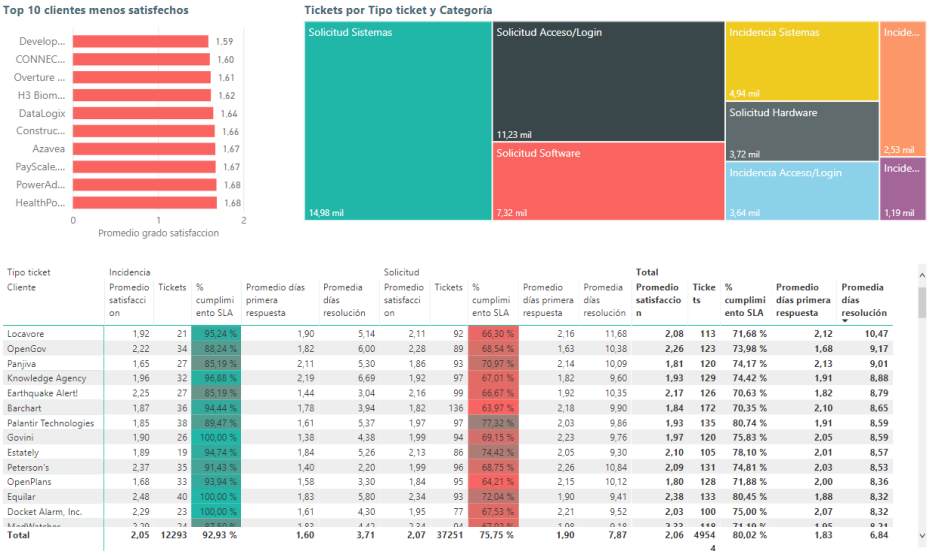


Figura 32. Gráfico clientes insatisfechos por tiempo

Por último, se ofrece también un informe para detectar los clientes más insatisfechos y poder analizar individualmente datos sobre el servicio recibido, como el promedio de tiempo de primera respuesta, tiempo de resolución, grado de cumplimiento de SLA, etc. Este grupo de clientes requiere una atención especial para tratar de revertir su sentimiento y además, al analizar las causas de su insatisfacción, se pueden alumbrar problemas que estaban ocultos.



Tipo ticket / Cliente	Incidencia: Promedio satisfacción	Tickets	% cumplimiento SLA	Promedio días primera respuesta	Promedia días resolución	Solicitud: Promedio satisfacción	Tickets	% cumplimiento SLA	Promedio días primera respuesta	Promedia días resolución	Total: Promedio satisfacción	Tickets	% cumplimiento SLA	Promedio días primera respuesta	Promedia días resolución
Locavore	1,92	21	95,24 %	1,90	5,14	2,11	92	66,30 %	2,16	11,68	**2,08**	**113**	**71,68 %**	**2,12**	**10,47**
OpenGov	2,22	34	88,24 %	1,82	6,00	2,28	89	68,54 %	1,63	10,38	**2,26**	**123**	**73,98 %**	**1,68**	**9,17**
Panjiva	1,65	27	85,19 %	2,11	5,30	1,86	93	70,97 %	2,14	10,09	**1,81**	**120**	**74,17 %**	**2,13**	**9,01**
Knowledge Agency	1,96	32	96,88 %	2,19	6,69	1,92	97	67,01 %	1,82	9,60	**1,93**	**129**	**74,42 %**	**1,91**	**8,88**
Earthquake Alert!	2,25	27	85,19 %	1,44	3,04	2,16	99	66,67 %	1,92	10,35	**2,17**	**126**	**70,63 %**	**1,82**	**8,79**
Barchart	1,87	36	94,44 %	1,78	3,94	1,82	136	63,97 %	2,18	9,90	**1,84**	**172**	**70,35 %**	**2,10**	**8,65**
Palantir Technologies	1,85	38	89,47 %	1,61	5,37	1,97	97	77,32 %	2,03	9,86	**1,93**	**135**	**80,74 %**	**1,91**	**8,59**
Govini	1,90	26	100,00 %	1,38	4,38	1,99	94	69,15 %	2,23	9,76	**1,97**	**120**	**75,83 %**	**2,05**	**8,59**
Estately	1,89	19	94,74 %	1,84	5,26	2,13	86	74,42 %	2,05	9,30	**2,10**	**105**	**78,10 %**	**2,01**	**8,57**
Peterson's	2,37	35	91,43 %	1,40	2,20	1,99	96	68,75 %	2,26	10,84	**2,09**	**131**	**74,81 %**	**2,03**	**8,53**
OpenPlans	1,68	33	93,94 %	1,58	3,30	1,84	95	64,21 %	2,15	10,12	**1,80**	**128**	**71,88 %**	**2,00**	**8,36**
Equilar	2,48	40	100,00 %	1,83	5,80	2,34	93	72,04 %	1,90	9,41	**2,38**	**133**	**80,45 %**	**1,88**	**8,32**
Docket Alarm, Inc.	2,29	23	100,00 %	1,61	4,30	1,95	77	67,53 %	2,21	9,52	**2,03**	**100**	**75,00 %**	**2,07**	**8,32**
Total	**2,05**	**12293**	**92,93 %**	**1,60**	**3,71**	**2,07**	**37251**	**75,75 %**	**1,90**	**7,87**	**2,06**	**49544**	**80,02 %**	**1,83**	**6,84**

Figura 33. Informe para análisis de satisfacción por cliente

7.6 Análisis del desempeño individual

Este informe permite analizar el trabajo individual de los técnicos desde dos perspectivas que se considera que resumen bien su desempeño: el número de tickets tratados y el porcentaje de éstos que han sido valorados positivamente por el cliente.

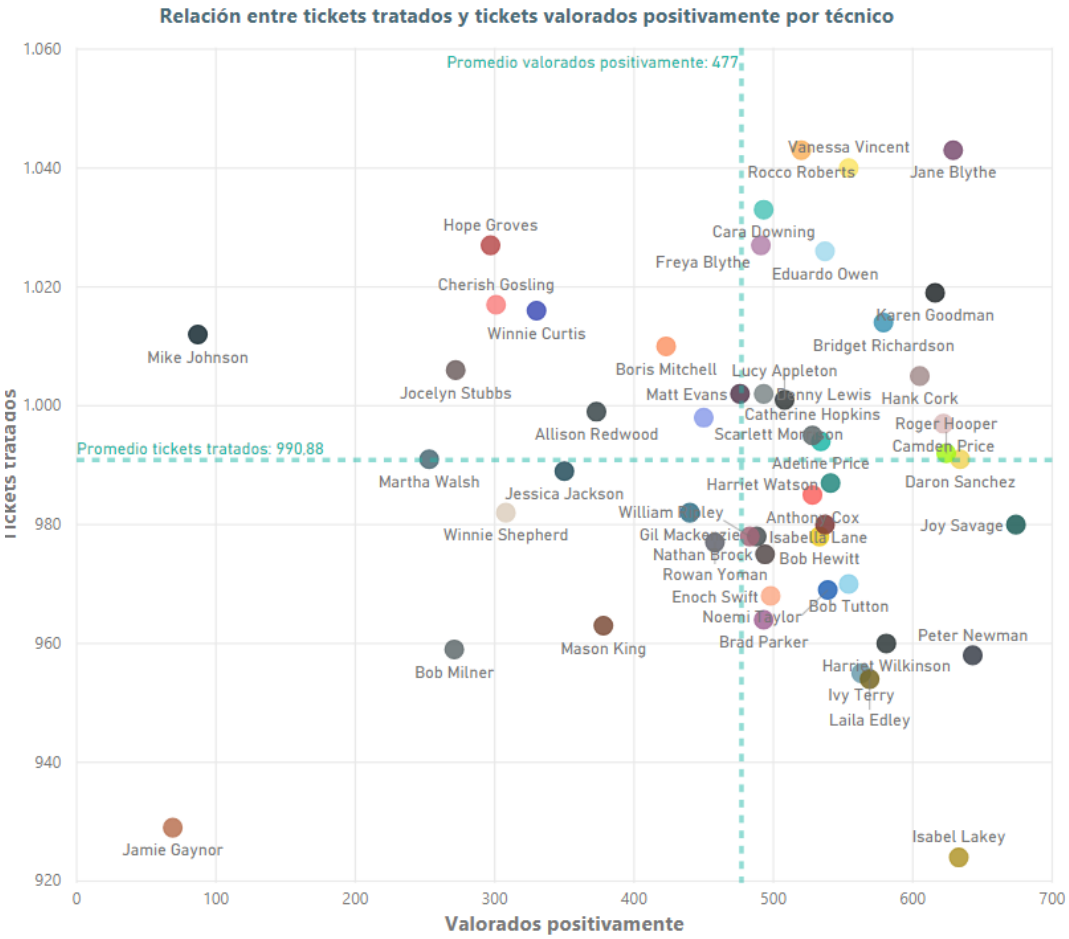


Figura 34. Gráfico de relación entre tickets tratados y valoraciones positivas

Se puede observar que el gráfico queda dividido en cuatro cuadrantes tras representar sobre él las líneas que marcan el valor promedio de cada eje (tickets tratados y tickets valorados positivamente). Los puntos situados más arriba y más a la derecha corresponden a los técnicos que han tratado más tickets y que tienen más valoraciones positivas; en el otro extremo, cuanto más abajo y más hacia la izquierda, se encuentran aquellos técnicos que han tratado menos tickets y que han recibido menos valoraciones positivas. Intuitivamente, este gráfico permite localizar a los técnicos con mejor y peor desempeño, a la vez que expone rápidamente los valores más extremos. En el caso mostrado en la figura, por ejemplo, se puede ver que los técnicos Jamie Gaynor o Bob Milner son los que tienen un menor número de valoraciones positivas, claramente menor que el resto, lo que implica que serán también los que tengan un mayor porcentaje de valoraciones negativas en los tickets tratados. Además, el rendimiento de Jamie Gaynor, en lo que se refiere al número de tickets tratados (que es el elemento generador de negocio) es de los más bajos.

Determinar las causas de este pobre rendimiento puede ser el paso para llevar a cabo acciones de mejora en el departamento, que pueden pasar por ofrecer una formación específica a estos técnicos, cambiar el modelo de asignación de tickets para encargarles tareas más acordes a sus conocimientos, etc.

7.7 Análisis de los tiempos de respuesta y resolución

Este grupo de informes permite estudiar con mayor profundidad los tiempos de respuesta y de resolución, dos datos muy ligados a la eficiencia de un Service Desk.

Tipo ticket	Incidencia		Solicitud		Total	
Categoría	Tickets	Promedia días resolución	Tickets	Promedia días resolución	Tickets	Promedia días resolución
Hardware	1186	11,05	3720	19,40	4906	17,38
1 - Baja	231	13,32	651	23,91	882	21,13
Chat	11	14,00	33	24,85	44	22,14
Formulario web	47	14,26	153	23,93	200	21,66
E-mail	34	13,53	97	23,74	131	21,09
Llamada	139	12,91	368	23,85	507	20,85
2 - Media	192	11,89	612	20,97	804	18,80
E-mail	29	13,79	102	20,96	131	19,37
Total	12293	3,71	37251	7,87	49544	6,84

Figura 35. Informe de tiempos de respuesta y resolución

En cuanto a los tiempos medios de resolución, se aprecian importantes diferencias entre un área y otra, tanto en los tickets de incidencias como en los de solicitudes. Como se puede ver en la figura, los tiempos para resolver incidencias son aproximadamente la mitad que para resolver solicitudes. Esto tiene sentido si en el diseño del proceso de gestión del Service Desk se ha dado prioridad a la resolución de incidencias frente a las solicitudes; en caso contrario, sería un claro indicador de alguna disfunción.

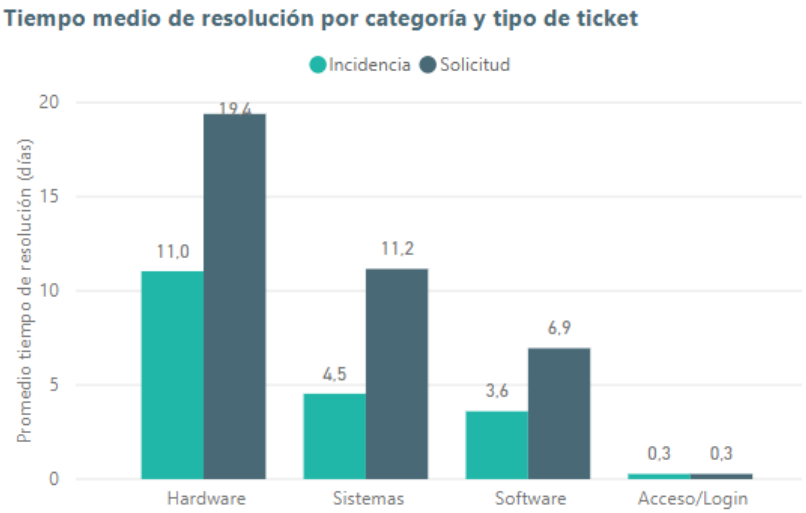


Figura 36. Gráfico tiempo medio de resolución por tipo y categoría

Por otro lado, la disparidad de tiempos en la resolución de tickets de distintas áreas también puede corresponder a un patrón normal, ya que las incidencias de hardware, por ejemplo, que suelen exigir desplazamientos y envíos físicos de material acostumbran a tener un periodo de resolución más largo. En cualquier caso, se trata aquí de identificar si estos valores se corresponden con lo esperado o pueden apuntar a problemas que requieran una corrección.

A veces los datos revelan hechos inesperados. En el siguiente gráfico se muestra el grado de satisfacción del cliente en función del tiempo de resolución del ticket. Parece lógico pensar que a medida que el tiempo de resolución sea mayor, la satisfacción será menor; y eso ocurre aquí, excepto en un caso: las incidencias que duran más de un mes tienen un grado de satisfacción mayor que cuando se resuelven en menos tiempo. Como se ha comentado en otros ejemplos, el descubrimiento de estos datos ayuda a conocer mejor el funcionamiento del servicio y ofrece oportunidades para mejorarlo. En este caso, averiguar las causas por las que el grado de satisfacción en incidencias tan largas es alto puede ayudar a descubrir otros factores en la satisfacción del cliente que no se habían tenido en cuenta.

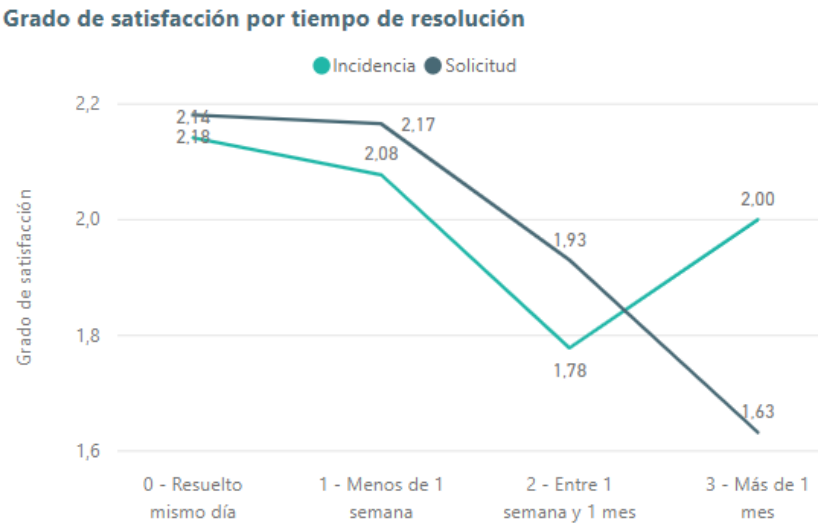
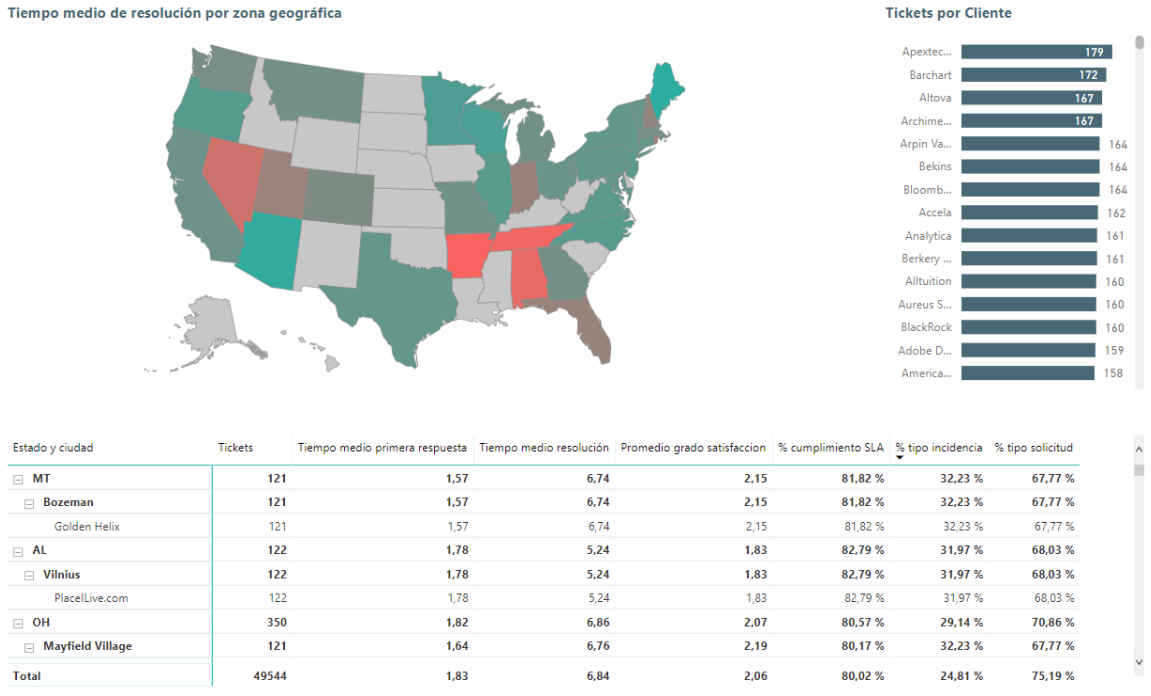


Figura 37. Gráfico de satisfacción por tiempo de resolución

Como conclusión a esta presentación de algunos de los informes que se incluyen en este trabajo, se introduce un informe que permite estudiar los tiempos de resolución por zonas geográficas



Estado y ciudad	Tickets	Tiempo medio primera respuesta	Tiempo medio resolución	Promedio grado satisfaccion	% cumplimiento SLA	% tipo incidencia	% tipo solicitud
MT	121	1,57	6,74	2,15	81,82 %	32,23 %	67,77 %
Bozeman	121	1,57	6,74	2,15	81,82 %	32,23 %	67,77 %
Golden Helix	121	1,57	6,74	2,15	81,82 %	32,23 %	67,77 %
AL	122	1,78	5,24	1,83	82,79 %	31,97 %	68,03 %
Vilnius	122	1,78	5,24	1,83	82,79 %	31,97 %	68,03 %
PlacelLive.com	122	1,78	5,24	1,83	82,79 %	31,97 %	68,03 %
OH	350	1,82	6,86	2,07	80,57 %	29,14 %	70,86 %
Mayfield Village	121	1,64	6,76	2,19	80,17 %	32,23 %	67,77 %
Total	49544	1,83	6,84	2,06	80,02 %	24,81 %	75,19 %

Figura 38. Informe para análisis de tiempos de resolución por zona geográfica

8. Cuadro de mando

En los informes que se han mostrado en el capítulo anterior se han usado repetidamente las métricas, que no son otra cosa que medidas realizadas sobre los distintos procesos. Para la elaboración del cuadro de mando se utilizarán algunas de esas métricas y otras nuevas, que por su especial relevancia se conocen como KPI (Key Performance Indicators o Indicadores Clave de Rendimiento). Los KPI son valores medibles que determinan la eficacia con la que la organización está logrando sus objetivos clave. Hay KPI de alto nivel, que miden el rendimiento general de la empresa, y otros de más bajo nivel que se enfocan a determinados departamentos o procesos, como será este caso.

En el Service Desk, los KPI ofrecen valores medibles basados en el tiempo de respuesta, tiempo de resolución, satisfacción del cliente, cumplimiento de SLA, etc. La monitorización constante de estos indicadores permite conocer el nivel de consecución de los objetivos marcados por el departamento y la Dirección y ayudan a tomar decisiones que permitan corregir las desviaciones para alcanzar los niveles más altos de servicio al cliente.

Para la definición de los KPI es habitual utilizar el criterio SMART que pone el foco en que los indicadores sean:

- (S)pecific – Simples, específicos y fáciles de entender por quién los ha de usar
- (M)easurable – Medibles, con un método de cálculo inequívoco
- (A)ttainable – Asequibles, debe ser posible obtenerlos con un esfuerzo razonable
- (R)epresentative - Representativos, relevantes y relacionados con la función asignada al departamento
- (T)emporal – Temporales, asociados a un periodo de tiempo que permita comparaciones entre periodos.

Los que se reproducen a continuación son indicadores clave que podrían ser relevantes para la mayoría de los proveedores de TI que desean medir su Service Desk. Sin embargo, no hay que olvidar que cada organización es única y sigue una estrategia distinta en cada momento, por lo que no existe un único conjunto de KPI ideal para

todas las organizaciones. La definición de los KPI es un traje a medida para cada empresa, donde lo importante es que estos indicadores sean realmente "claves", estén alineados con sus objetivos, y ayuden a medir el rendimiento de su Service Desk y cómo se están alcanzando esos objetivos.

8.1 Definición de indicadores

a) Satisfacción del cliente

Definición: En cualquier organización de prestación de servicios, la calidad del servicio, medida por la satisfacción del cliente, es de vital importancia. Ninguna organización puede aspirar a tener una larga trayectoria con clientes insatisfechos, así que este indicador debe ser monitorizado continuamente y utilizado como base para valorar otros indicadores (p.ej. un tiempo de respuesta muy rápido en general es bueno, pero un tiempo de respuesta muy rápido con una satisfacción de cliente muy baja es indicativo de problemas).

Medida: Promedio de valoraciones dadas a cada ticket, con un valor entre 1 (Insatisfecho) y 3 (Muy satisfecho).

Valores indicativos:

Óptimo	> 2,5 puntos
Tolerable	2,0 – 2,5 puntos
Deficiente	< 2 puntos

b) Coste por ticket

Definición: El coste por ticket es el gasto promedio por ticket y está basado fundamentalmente en el coste salarial de los agentes y técnicos. Este dato es importante ya que cuanto menor sea el coste de atender un ticket mayor será el beneficio de la compañía. Un coste por ticket alto es indicativo de problemas de eficiencia y pone en peligro la competitividad del proveedor. Sin embargo,

un coste demasiado bajo, si este se logra a costa de sacrificar la calidad del servicio y la satisfacción del cliente, también es problemático. El coste por ticket y la satisfacción de cliente deben ser siempre evaluados en conjunto: el objetivo que se persigue es obtener una alta satisfacción del cliente con unos costes contenidos.

Medida: Medido en cantidad dineraria (€) por ticket. En este trabajo, con un alcance limitado, no se disponen de todos los campos que serían necesarios para calcular de una forma exacta este indicador. Sin embargo, se puede hacer una estimación que ayude a tener una idea bastante aproximada del coste por ticket en base a los salarios medios, el número de técnicos y el número de tickets. Para este cálculo se tomará un coste bruto anual por empleado de 28.000 euros.

La fórmula es:
([Número de técnicos] * [Salario diario por tecnico] * Número de días) / [Número de tickets]

Valores indicativos: Comparado en periodos mensuales, se valorará la reducción del tiempo medio de resolución de esta forma

c) Tiempo medio de primera respuesta

Definición: Se refiere al tiempo promedio que el cliente debe esperar antes de obtener la primera respuesta a su solicitud de soporte. Este dato es importante porque sirve para asegurarse de que los clientes son contactados en un tiempo razonable. En general, cuanto menor sea este tiempo, mejor será el indicador. Un tiempo de respuesta excesivo puede llevar a la pérdida de confianza del cliente en la compañía para resolver sus incidencias eficientemente.

Medida: Medido en promedio de días.

Valores indicativos:

Óptimo	< 1,5 días
Tolerable	1,5 – 2,0 días
Deficiente	> 2 días

d) **Cumplimiento de SLA**

Definición: Este indicador hace referencia al porcentaje de tickets que se resuelven en los tiempos marcados por el SLA. El SLA establece los límites bajo los que se debe desempeñar la función del Service Desk, a menudo mediante un contrato formal que incluye penalizaciones en caso de incumplimiento, y es la base sobre la que el cliente deposita su confianza en el proveedor, de ahí que su cumplimiento sea un objetivo vital.

Medida: Porcentaje de tickets que cumplen el SLA.

Valores indicativos:

Óptimo	> 85%
Tolerable	75% –85%
Deficiente	< 75%

e) **Número de tickets**

Definición: Este indicador mide el volumen de tickets que se abren en un plazo determinado. En general, en las compañías que tienen un Service Desk propio, el crecimiento de este indicador podría ser indicativo de problemas, porque aumenta el número de incidencias. Sin embargo, en el ámbito de este trabajo, el de un proveedor de servicios de TI cuya misión es dar soporte a sus clientes, un volumen mayor representa más clientes y más negocio (más ventas) por lo que se medirá su crecimiento como un elemento positivo.

Medida: Volumen de tickets por periodo.

f) **Tiempo medio de resolución**

Definición: Tiempo medio que tardan en resolverse los tickets. Permite medir la eficiencia del día a día y está directamente relacionado con la satisfacción del cliente. Como se busca la mejora continua, se valorará que este tiempo sea cada vez menor.

Medida: Medido en promedio de días de resolución

Valores indicativos: Comparado en periodos mensuales, se valorará la reducción del tiempo medio de resolución.

g) **Top 10 técnicos**

Definición: Permite conocer a los mejores empleados, que son la base para construir un Service Desk eficiente y con capacidad de respuesta. Conocer a los mejores técnicos puede ser útil para tenerlos en cuenta en posibles ascensos a roles de supervisión, asignarlos a los clientes de mayor valor o establecer su política salarial o de desarrollo de carrera.

Medida: [Total tickets tratados por técnico con valoración del cliente 2-Satisfecho o 3-Muy satisfecho].

Una vez que se han definido los indicadores ya sólo queda incorporar estos indicadores a un panel con elementos gráficos. A continuación, se ve el producto resultante:

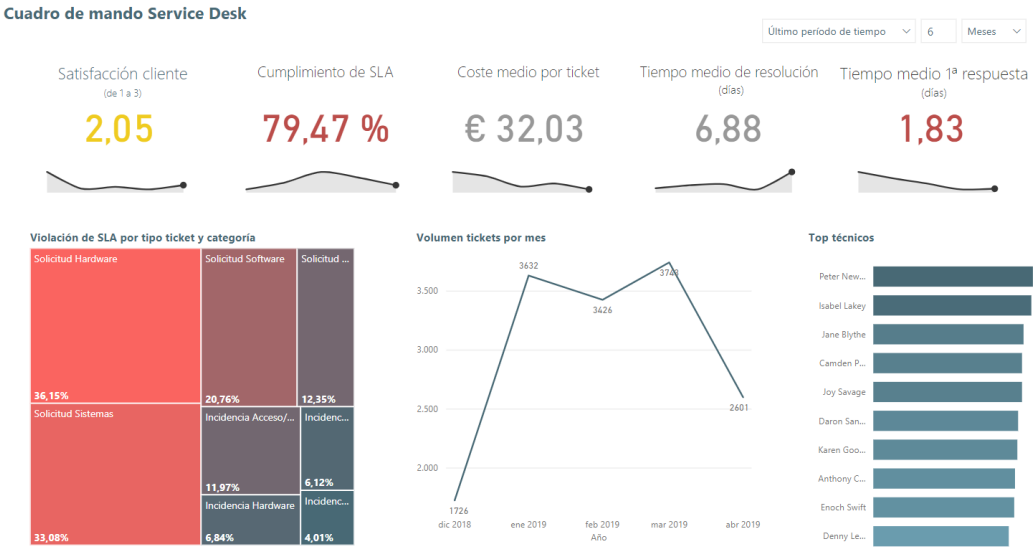


Figura 39. Visión general del Cuadro de Mando

Los distintos indicadores proporcionan el dato acotado al periodo seleccionado (por defecto, los últimos 6 meses). Los indicadores permiten observar la evolución del valor del KPI en los meses precedentes, para tener una visión de la tendencia. Además, es posible situarse sobre los puntos de la gráfica para consultar datos anteriores.

Figura 40. Detalle de un indicador

9. Aprendizaje automático para la mejora del Service Desk

En este capítulo se hará una introducción a los métodos de aprendizaje automático o Machine Learning y su aplicación en la mejora de un Service Desk. Si bien el aprendizaje automático no forma parte de lo que se conoce estrictamente como Business Intelligence, sí que constituye una evolución lógica en el proceso de análisis de datos.

Gartner establece una escala de cuatro capacidades analíticas que permiten extraer valor de los datos: en el primer nivel se encuentra la analítica descriptiva, que permite saber qué paso, y en el segundo nivel se encuentra la analítica diagnóstica, que permite saber por qué paso. Sobre estos dos estadios de la analítica es donde poco el foco el Business Intelligence que, mediante el reporting y los cuadros de mando, permite obtener respuestas sobre hechos pasados, como se ha visto en los capítulos anteriores.

La analítica predictiva va un paso más allá y trata de explicar lo que sucederá en un futuro a partir del estudio de los datos del pasado. Esto supone un importante salto cualitativo y, aunque la complejidad de los métodos es mayor, el valor que se obtiene de los datos también es mucho mayor.

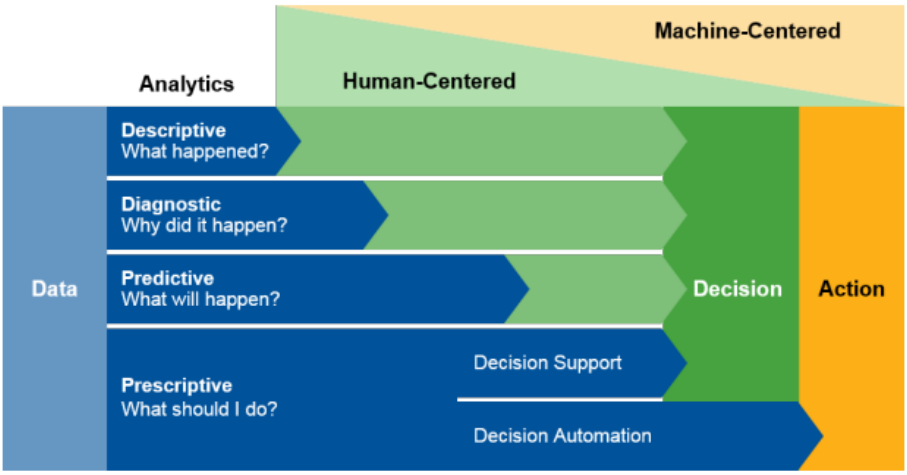


Figura 41. Escala de niveles analíticos. Fuente: Gartner (2006)

Los recientes avances en aprendizaje automático y big data han eliminado alguna de las barreras tecnológicas y económicas que hacían que la analítica avanzada fuera cosa solo

de grandes compañías. Como se verá, su utilización en el ámbito de un Service Desk permite minimizar el tiempo de resolución de los tickets, ahorrar recursos y mejorar la satisfacción del usuario.

9.1 Introducción al aprendizaje automático

El aprendizaje automático es una rama de la Inteligencia Artificial y se basa en la idea de que los computadores pueden aprender de la experiencia sin la necesidad de que el conocimiento sea programado explícitamente. Este aprendizaje se produce por medio de algoritmos que encuentran patrones, relaciones y tendencias en los datos, dando lugar a un modelo que es capaz de predecir comportamientos futuros sin necesidad de disponer de una ecuación predeterminada de partida. El modelo mejora su rendimiento incrementalmente, a medida que se le proporcionan más muestras para el aprendizaje.

Los modelos de aprendizaje automático están ampliamente implantados en todos los ámbitos de la sociedad y se utilizan constantemente para hacer diagnósticos médicos, identificar objetos en una imagen, hacer traducciones, ofrecer recomendaciones, reconocer el lenguaje humano, etc.

El aprendizaje automático emplea fundamentalmente dos tipos de técnicas: el aprendizaje supervisado, donde el modelo es entrenado con datos de entrada y salida conocidos para que pueda predecir salidas futuras, y el aprendizaje no supervisado, que encuentra estructuras y patrones ocultos en los datos de entrada.

9.1.1 Aprendizaje supervisado

El aprendizaje supervisado se utiliza cuando se tiene un conocimiento previo de cuáles deberían ser los valores de salida (las respuestas) para un conjunto de muestras. El algoritmo analiza un conjunto de datos que incluyen la respuesta (etiqueta) y trata de encontrar la función que, dados unos determinados datos de entrada, mejor se aproxime a la salida adecuada, dando lugar a un modelo que es capaz de realizar predicciones.

Por ejemplo, para elaborar un sistema capaz de reconocer figuras humanas, se proporcionaría al algoritmo una gran cantidad de fotografías de todo tipo, y para cada una de ellas se indicaría si tiene presencia de una figura humana o no (la etiqueta). Tras realizar este proceso un número suficiente de veces (entrenamiento) el algoritmo crearía un modelo capaz de reconocer figuras humanas en fotografías que no ha visto anteriormente.

El aprendizaje supervisado se suele usar en problemas de clasificación, como saber si un correo electrónico es spam o no, si una transacción con una tarjeta de crédito es fraudulenta o no, si un determinado símbolo es una letra o un número, etc. En estos casos los modelos se utilizan para predecir variables categóricas (discretas).

También se utiliza para resolver problemas de regresión, que predicen un valor numérico (continuo), como la esperanza de vida, el tiempo antes de que una máquina falle, la predicción meteorológica, etc.

Entre los algoritmos de aprendizaje supervisado más habituales se encuentran las máquinas de soporte vectorial (SVM), los árboles de decisión, la regresión logística, los k-vecinos más cercanos (KNN) y los clasificadores bayesianos (Naïve Bayes).

9.1.2 Aprendizaje no supervisado

En el aprendizaje no supervisado el objetivo no es predecir el valor de una variable, sino hallar o inferir patrones ocultos o estructuras intrínsecas presentes en un conjunto de datos sin respuestas etiquetadas. Tiene, por tanto, un carácter exploratorio.

El clustering es la técnica de aprendizaje no supervisado más habitual. Se trata de encontrar agrupaciones basadas en similitudes en los datos. Entre las aplicaciones más comunes están la segmentación de clientes, la clasificación de documentos por categorías, la detección de fraudes o el reconocimiento de objetos.

Dentro del aprendizaje no supervisado también se encuentran los algoritmos de reducción de la dimensionalidad, como el análisis de componentes principales, que

permiten reducir el tamaño del conjunto de datos eliminando determinadas variables sin afectar a la estructura y a las relaciones inherentes en el conjunto de datos original.

9.1.3 La elección del algoritmo adecuado

Como se ha podido ver en los apartados anteriores existen multitud de algoritmos de aprendizaje automático y la elección del más adecuado a cada caso no es trivial. A menudo se utiliza el ensayo y error para obtener el que da mejores resultados.

Ya se ha podido ver que si se quiere obtener una predicción se utilizará el aprendizaje supervisado, y que en función de si esta predicción es una variable discreta o continua se usará la clasificación o la regresión. Si lo que se quiere es explorar los datos y obtener una representación interna de los mismos se utilizará el aprendizaje no supervisado.

En este punto hay que considerar varios factores que ayudan a determinar el algoritmo más apropiado: de qué tipo de datos se dispone, qué información se quiere obtener, cómo se empleará esa información, …

Aunque no hay una receta universal para realizar este proceso, existen en la comunidad algunas “chuletas” que son ampliamente usadas para obtener una guía que ayude en la selección de los algoritmos. Una de las más conocidas es la que publican los autores de Scikit Learn, una librería de aprendizaje automático de software libre para el lenguaje de programación Python.

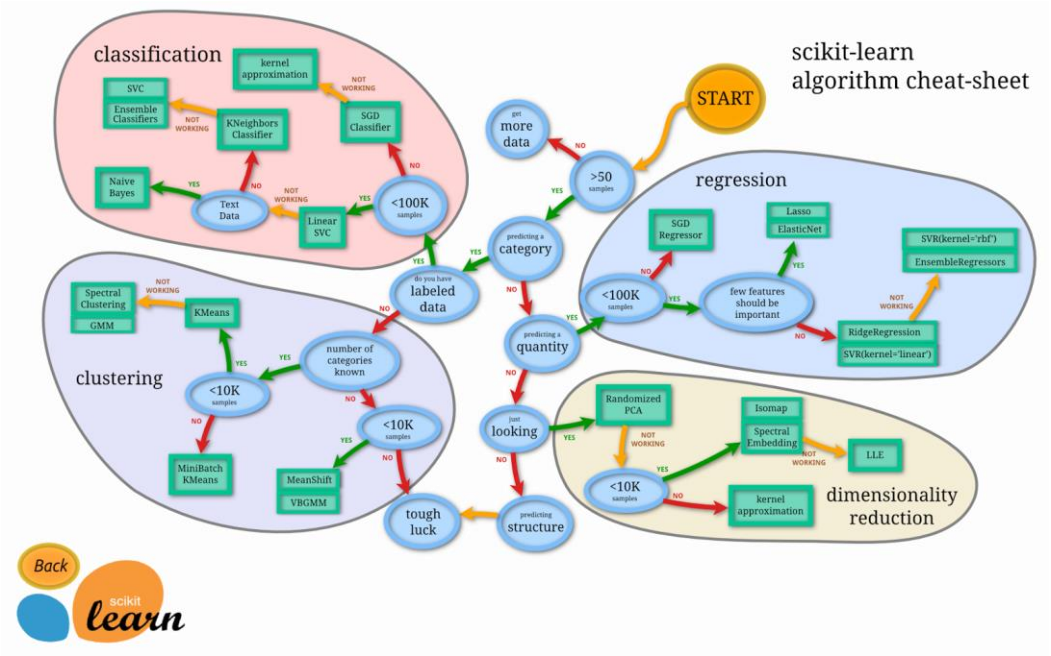


Figura 42. Cheat sheet de Scikit Learn para la elección de algoritmos de machine learning

9.2 Usos del aprendizaje automático en un Service Desk

Los sistemas de ticketing recopilan una gran cantidad de datos sobre lo que los clientes solicitan, junto con información sobre cuándo y por qué, cómo se resuelve, cuánto se tarda en dar una respuesta, etc. Esta es una información muy valiosa que, cuando se utiliza para crear modelos de aprendizaje automático, permite optimizar el uso de los recursos, aumentar la calidad del servicio y mejorar la experiencia del cliente. Un estudio de Gartner estimaba que, para 2019, los Service Desk que utilicen tecnologías mejoradas de aprendizaje automático podrían liberar hasta un 30% de su capacidad de soporte [5].

Como se podrá observar, todos estos casos abren la puerta a la creación de sistemas de Service Desk autónomos, capaces de operar las 24 horas y que cubren todas las tareas de este proceso: desde la apertura y clasificación de tickets, la resolución de incidencias de primer nivel, su asignación al equipo o técnico adecuado, hasta la prevención de problemas.

A continuación, se exponen algunos casos de uso en los que la aplicación del aprendizaje automático permite optimizar este servicio:

a) Categorización automática de tickets

Uno de los problemas más habituales en la gestión de los tickets tiene relación con la categorización. Los errores al determinar la categoría del ticket impiden que éste sea enviado al equipo de soporte adecuado y da lugar a nuevas reasignaciones que acaban introduciendo retrasos en su resolución. La automatización de esta tarea mediante el aprendizaje automático permite eliminar la intervención manual, con la consiguiente reducción de la tasa de errores en la categorización y la mejora de la eficiencia del proceso.

En estos casos es habitual utilizar algoritmos de aprendizaje no supervisado, como el clustering, donde el texto del ticket (p. ej. el cuerpo de un e-mail o un mensaje de chat) es analizado para extraer las características que permiten asignarlo a alguno de los grupos existentes.

b) Enrutado automático de los tickets

El aprendizaje automático también se puede utilizar para la asignación inteligente de los tickets entrantes. En lugar de establecer reglas fijas del tipo “los tickets de categoría X y prioridad Y se asignan al grupo de técnicos Z”, se pueden crear algoritmos que asignen los tickets a los técnicos en base al contexto del ticket, a éxitos pasados en incidencias similares y a la disponibilidad de los distintos recursos tratando de utilizar los equipos y técnicos más adecuados en cada momento.

c) **Detección de tickets susceptibles de violar el SLA**

En este caso, se suelen utilizar algoritmos de clasificación que analizan el cumplimiento del SLA en función de múltiples parámetros en el histórico de tickets para crear modelos que ayudan a predecir cuándo un ticket tiene una alta probabilidad de superar el tiempo de SLA. Esto permite poner en alerta anticipadamente a los responsables del servicio para tomar medidas que puedan paliar ese retraso.

d) Habilitar el autoservicio

Con el aprendizaje automático se pueden proporcionar herramientas de autoservicio, como los chatbots y las autorespuestas inteligentes. Por ejemplo, un cliente puede enviar un e-mail con una incidencia y el sistema es capaz de encontrar en la base de datos de conocimientos la solución más adecuada y enviársela como respuesta.

e) Prevención y predicción proactiva de problemas

Basándose en los patrones de comportamiento de los usuarios, el aprendizaje automático puede anticipar problemas y dar respuestas proactivas que ayuden a reducir el impacto de estos problemas. Por ejemplo, ante un aumento anómalo de tickets sobre un mismo tema en un corto periodo de tiempo, el sistema puede identificar la causa raíz y alertar a los responsables del servicio antes de que ocurra un incidente o enviar mensajes automáticos a los clientes para prevenir sobre un determinado problema.

9.3 Caso práctico

En este apartado se llevará a cabo un ejercicio práctico de aplicación de un algoritmo de aprendizaje automático en el contexto de este trabajo.

A partir de los datos de los que se dispone en el momento en que un nuevo ticket es asignado a un técnico, se tratará de determinar si ese ticket violará o no el SLA. Este es un problema típico de clasificación: se usarán como variables predictoras el tipo de ticket (si es incidencia o solicitud), la categoría asignada (Hardware, Software, …), la prioridad y el impacto (Alto, Medio, Bajo, …), el cliente que abre el ticket, el técnico al que ha sido asignado y el tiempo de primera respuesta. La variable dependiente, aquella que se quiere predecir, es la de SLA violado, que como se vio en el modelo de datos, puede tomar los valores verdadero o falso.

Como se puede apreciar, salvo el tiempo de primera respuesta, todas las variables que se utilizarán son categóricas, lo que limita mucho la elección del algoritmo a utilizar. En este caso se trabajará con el algoritmo de Regresión Logística Binaria, un algoritmo que trata de comprobar relaciones causales entre los datos cuando la variable dependiente es

categórica y tiene solo dos posibles valores (sí o no, verdadero o falso, cumple o incumple, …).

El algoritmo se implementará con R, un lenguaje y entorno de programación muy popular para el análisis estadístico y gráfico.

Como se verá más adelante, este modelo acierta en su predicción en un 87% de los casos. Esto es más de lo que se podría obtener por simple probabilidad. Ya se sabe por lo que se vio en los informes de capítulos anteriores que el 80% de los tickets cumple con el SLA; por tanto, si se hiciese una predicción donde todos los casos fuera positivo se acertaría en el 80% de casos. Se comprueba por tanto que este modelo, con muy poco esfuerzo, proporciona un resultado más ajustado, que podría mejorarse aún más en una situación real incorporando un mayor número de características al conjunto de datos.

A grandes rasgos, los pasos que se llevarán a cabo son:

1) **Creación de los conjuntos de datos de entrenamiento y de test**: Este es un paso fundamental a la hora de programar algoritmos de machine learning. El conjunto de entrenamiento proporciona los datos a través de los cuales el modelo aprende y el conjunto de test es usado para validar los resultados. Es habitual dividir el conjunto de datos original en dos subconjuntos aleatorios, donde el de entrenamiento tiene el 80% de las muestras y el de test el 20% restante.

2) **Preparación de los datos para adaptarlos al proceso analítico**. Normalmente esto implica eliminar o completar valores vacíos, convertir tipos de datos, eliminar valores extremos, etc. En este caso y dado que el conjunto de datos ya viene limpio, simplemente se convertirán las variables de tipo carácter en factores, que es un tipo utilizado por R.

3) **Se creará el modelo con el algoritmo de regresión logística** binaria usando los datos de entrenamiento

4) **Se validará el modelo con los datos de test** y se comprobará el nivel de exactitud que ofrece.

5) Una vez se dispone del modelo adecuado, sólo habría que **implementarlo en el lugar adecuado**. En este ejemplo, lo más lógico sería integrarlo en el sistema de ticketing para que en el momento en que se abren y clasifican los tickets el sistema pudiera alertar de aquellos tickets que se prevé que van a incumplir el SLA.

En primer lugar, se cargará el dataset que está en formato Excel. Para ello se utiliza la librería redxl.

```
library(readxl)

df_completo <- read_excel("DatasetR.xlsx")
```

A continuación, se muestra la estructura del dataset. Se puede comprobar que se trata del mismo fichero que se ha utilizado para la construcción del Data Warehouse y los informes.

```
str(df_completo)

## Classes 'tbl_df', 'tbl' and 'data.frame':    49544 obs. of  23 variables:
##  $ id                    : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ tipo                  : chr  "Incidencia" "Solicitud" "Solicitud" "Solicitud"
...
##  $ fecha_creacion        : POSIXct, format: "2018-10-25" "2018-07-13" ...
##  $ estado...4            : chr  "Cerrado" "Cerrado" "Cerrado" "Cerrado" ...
##  $ cliente               : chr  "Sage Bionetworks" "Azavea" "Ez-XBRL" "Arrive
Labs" ...
##  $ estado...6            : chr  "WA" "PA" "VA" "CA" ...
##  $ ciudad                : chr  "Seattle" "Philadelphia" "Manassas" "San
Francisco" ...
##  $ pais                  : chr  "us" "us" "us" "us" ...
##  $ codigo_postal         : num  98109 19107 22701 94110 20815 ...
##  $ categoria             : chr  "Sistemas" "Software" "Acceso/Login" "Sistemas"
...
##  $ prioridad             : chr  "0 - No asignada" "1 - Baja" "0 - No asignada" "0
- No asignada" ...
##  $ modo                  : chr  "Llamada" "Llamada" "Formulario web" "Formulario
web" ...
##  $ impacto               : chr  "2 - Normal" "1 - Menor" "2 - Normal" "2 -
Normal" ...
##  $ tecnico               : chr  "Mike Johnson" "Jane Blythe" "Jane Blythe"
"Jessica Jackson" ...
##  $ fecha_asignacion      : POSIXct, format: "2018-10-28" "2018-07-18" ...
##  $ fecha_vencimiento     : POSIXct, format: "2018-11-07" "2018-07-28" ...
```

```
##  $ fecha_resolucion       : POSIXct, format: "2018-10-28" "2018-07-18" ...
##  $ fecha_cierre           : POSIXct, format: "2018-10-28" "2018-07-18" ...
##  $ reabierta              : num  0 0 -1 0 0 0 0 0 0 0 ...
##  $ escalada               : num  0 -1 0 0 0 -1 -1 -1 -1 0 ...
##  $ sla_violado            : num  0 0 0 -1 0 0 0 -1 0 0 ...
##  $ tiempo_resolucion_horas: num  3 5 0 20 1 0 9 15 6 1 ...
##  $ satisfaccion           : chr  "1 - Insatisfecho" "1 - Insatisfecho" "0 -
Desconocido" "0 - Desconocido" ...
```

Seguidamente se seleccionará un subconjunto en el que solo se incluirán aquellas variables con las que se va a trabajar y se prepararán los datos. Se crea un campo calculado para conocer el tiempo de primera respuesta.

```
df_completo$dia <- weekdays(df_completo$fecha_creacion)

df_completo$tiempo_primera_respuesta =as.Date(df_completo$fecha_asignacion, "%Y-%m-
%d")-as.Date(df_completo$fecha_creacion, "%Y-%m-%d")

df <- df_completo[,c("tipo","categoria", "prioridad","impacto",
"tecnico","sla_violado", "tiempo_primera_respuesta","satisfaccion")]

categorical_cols <- c("tipo","categoria", "prioridad", "impacto", "tecnico",
"sla_violado","satisfaccion")

df[categorical_cols] <- lapply(df[categorical_cols], factor)

head(df)

## # A tibble: 6 x 8
##   tipo  categoria prioridad impacto tecnico sla_violado tiempo_primera_~
##   <fct> <fct>     <fct>     <fct>   <fct>   <fct>       <drtn>
## 1 Inci~ Sistemas  0 - No a~ 2 - No~ Mike J~ 0           3 days
## 2 Soli~ Software  1 - Baja  1 - Me~ Jane B~ 0           5 days
## 3 Soli~ Acceso/L~ 0 - No a~ 2 - No~ Jane B~ 0           0 days
## 4 Soli~ Sistemas  0 - No a~ 2 - No~ Jessic~ -1          5 days
## 5 Soli~ Acceso/L~ 1 - Baja  2 - No~ Jessic~ 0           1 days
## 6 Soli~ Acceso/L~ 3 - Alta  2 - No~ Keira ~ 0           0 days
## # ... with 1 more variable: satisfaccion <fct>
```

Se crean los conjuntos de entrenamiento y de test en una proporción 80/20. El sistema coge aleatoriamente las muestras para asignarlas a cada grupo. Se muestra también cuántas filas hay que cada conjunto

```
set.seed(123)

ind <- sample(2, nrow(df), replace=TRUE, prob=c(0.8, 0.2))

train <- df[ind==1,]
test  <- df[ind==2,]


test <- test[,c("tipo","categoria", "prioridad", "impacto", "tecnico",
```

```
"sla_violado","tiempo_primera_respuesta", "satisfaccion")]

nrow(df) # Dataset original

## [1] 49544

nrow(train) # Datos entrenamiento

## [1] 39705

nrow(test) # Datos de test

## [1] 9839
```

A continuación, se crea el modelo de regresión logística binomial. Para ello R dispone de la función glm(). Al mostrar los coeficientes del modelo resultante, se puede ver cuáles son aquellos valores que el sistema ha considerado estadísticamente significativos para la predicción del SLA violado (los más significativos aparecen con tres asteriscos al final)

```
model <- glm(sla_violado~., family=binomial(link='logit'), data=train)

summary(model)

##
## Call:
## glm(formula = sla_violado ~ ., family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.9431   0.0657   0.2216   0.5138   2.2707
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                    6.048139   0.420130  14.396  < 2e-16 ***
## tipoSolicitud                 -2.023240   0.049912 -40.536  < 2e-16 ***
## categoriaHardware             -0.667796   0.066665 -10.017  < 2e-16 ***
## categoriaSistemas             -0.307767   0.053629  -5.739 9.53e-09 ***
## categoriaSoftware              0.665951   0.065742  10.130  < 2e-16 ***
## prioridad1 - Baja              0.653539   0.052640  12.415  < 2e-16 ***
## prioridad2 - Media            -0.682635   0.047370 -14.411  < 2e-16 ***
## prioridad3 - Alta             -0.250127   0.039278  -6.368 1.91e-10 ***
## impacto1 - Menor               0.267093   0.416034   0.642 0.520874
## impacto2 - Normal             -0.608012   0.396367  -1.534 0.125039
## impacto3 - Mayor              -1.258022   0.401774  -3.131 0.001741 **
## impacto4 - Critico            -1.411103   0.417663  -3.379 0.000729 ***
## tecnicoAllison Redwood        -2.529264   0.151314 -16.715  < 2e-16 ***
## tecnicoAnthony Cox             0.359100   0.187126   1.919 0.054981 .
## tecnicoBob Hewitt              1.195803   0.228327   5.237 1.63e-07 ***
## tecnicoBob Milner             -2.800803   0.152656 -18.347  < 2e-16 ***
## tecnicoBob Tutton              1.705859   0.278706   6.121 9.32e-10 ***
## tecnicoBoris Mitchell         -2.196135   0.150087 -14.632  < 2e-16 ***
## tecnicoBrad Parker            -0.003715   0.175845  -0.021 0.983147
## tecnicoBridget Richardson      0.955284   0.213664   4.471 7.79e-06 ***
## tecnicoCamden Price            1.982463   0.309419   6.407 1.48e-10 ***
## tecnicoCara Downing           -1.406002   0.152587  -9.214  < 2e-16 ***
## tecnicoCatherine Hopkins       0.078947   0.177024   0.446 0.655620
```

```
## tecnicoCherish Gosling         -2.740165   0.151513 -18.085  < 2e-16 ***
## tecnicoDaron Sanchez            2.765100   0.430643   6.421 1.36e-10 ***
## tecnicoDenny Lewis             -0.567324   0.162577  -3.490 0.000484 ***
## tecnicoEduardo Owen            -0.006168   0.174325  -0.035 0.971777
## tecnicoEnoch Swift             -0.652675   0.161611  -4.039 5.38e-05 ***
## tecnicoFreya Blythe            -0.778302   0.159223  -4.888 1.02e-06 ***
## tecnicoGil Mackenzie           -1.307517   0.155955  -8.384  < 2e-16 ***
## tecnicoHank Cork                1.715404   0.278662   6.156 7.47e-10 ***
## tecnicoHarriet Watson           0.507867   0.195071   2.604 0.009228 **
## tecnicoHarriet Wilkinson        2.560893   0.401946   6.371 1.88e-10 ***
## tecnicoHope Groves             -2.798602   0.151048 -18.528  < 2e-16 ***
## tecnicoIsabel Lakey             2.554589   0.379518   6.731 1.68e-11 ***
## tecnicoIsabella Lane           -0.427142   0.168366  -2.537 0.011181 *
## tecnicoIvy Terry                1.557944   0.262425   5.937 2.91e-09 ***
## tecnicoJamie Gaynor            -2.851198   0.154048 -18.508  < 2e-16 ***
## tecnicoJane Blythe              2.467548   0.360077   6.853 7.24e-12 ***
## tecnicoJessica Jackson         -2.673602   0.151531 -17.644  < 2e-16 ***
## tecnicoJocelyn Stubbs          -2.547393   0.150919 -16.879  < 2e-16 ***
## tecnicoJoy Savage               2.290708   0.344807   6.643 3.06e-11 ***
## tecnicoKaren Goodman            2.533822   0.378673   6.691 2.21e-11 ***
## tecnicoKeira Tobin              0.209837   0.182712   1.148 0.250780
## tecnicoLaila Edley              2.042276   0.309651   6.595 4.24e-11 ***
## tecnicoLucy Appleton            0.171139   0.180223   0.950 0.342317
## tecnicoMartha Walsh            -2.677847   0.151175 -17.714  < 2e-16 ***
## tecnicoMason King              -2.297064   0.151160 -15.196  < 2e-16 ***
## tecnicoMatt Evans              -0.734761   0.159869  -4.596 4.31e-06 ***
## tecnicoMike Johnson            -2.621168   0.151782 -17.269  < 2e-16 ***
## tecnicoNathan Brock            -0.304930   0.170391  -1.790 0.073520 .
## tecnicoNoemi Taylor             0.867423   0.213953   4.054 5.03e-05 ***
## tecnicoPeter Newman             2.653903   0.402358   6.596 4.23e-11 ***
## tecnicoRocco Roberts           -0.301900   0.165037  -1.829 0.067357 .
## tecnicoRoger Hooper             2.372994   0.344844   6.881 5.93e-12 ***
## tecnicoRowan Yoman             -0.778519   0.160683  -4.845 1.27e-06 ***
## tecnicoScarlett Morrison       -1.191427   0.155872  -7.644 2.11e-14 ***
## tecnicoVanessa Vincent         -0.224088   0.170126  -1.317 0.187774
## tecnicoWilliam Ripley          -0.086324   0.171939  -0.502 0.615623
## tecnicoWinnie Curtis           -2.688129   0.151498 -17.744  < 2e-16 ***
## tecnicoWinnie Shepherd         -2.617579   0.151634 -17.262  < 2e-16 ***
## tiempo_primera_respuesta       -0.413626   0.016185 -25.556  < 2e-16 ***
## satisfaccion1 - Insatisfecho    0.040352   0.042790   0.943 0.345676
## satisfaccion2 - Satisfecho     -0.578701   0.053467 -10.824  < 2e-16 ***
## satisfaccion3 - Muy satisfecho -0.768675   0.053824 -14.281  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 39564  on 39704  degrees of freedom
## Residual deviance: 25125  on 39640  degrees of freedom
## AIC: 25255
##
## Number of Fisher Scoring iterations: 7
```

Seguidamente se procede a validar el modelo con los datos de test y se observan los resultados obtenidos. Adicionalmente, se muestra la matriz de confusión, que es un elemento muy útil para describir el rendimiento de un modelo de clasificación contra un conjunto de test cuyos valores verdaderos son conocidos.

En la diagonal principal de esta matriz se muestran los verdaderos positivos y los verdaderos negativos, es decir los casos en los que la predicción ha sido correcta. En la otra diagonal se muestran los falsos positivos (el valor real era No y el sistema ha predicho Sí) y los falsos negativos (el valor real era Sí y el sistema ha predicho No).

```
predictTest = predict(model, type = "response", newdata = test)

table_mat <- table(test$sla_violado, predictTest>0.5)
table_mat

##
##      FALSE TRUE
##   -1  1140  882
##   0    372 7445

accuracy_Test <- sum(diag(table_mat)) / sum(table_mat)
print(paste('Precisión de la predicción ', accuracy_Test))

## [1] "Precisión de la predicción  0.872548023173087"
```

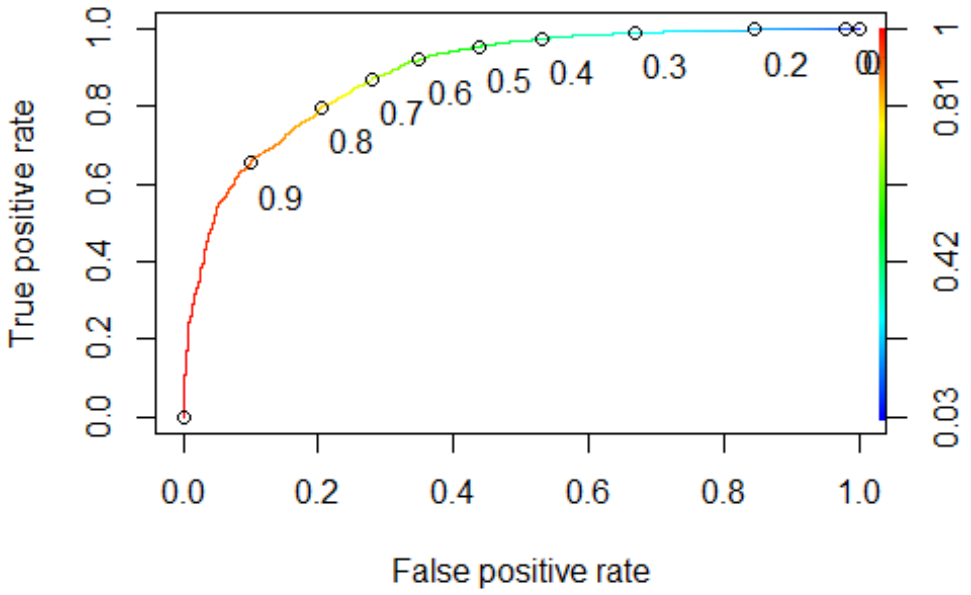
Como se puede observar, el modelo tiene una precisión del 0.87277, es decir, acierta en el 87,2% de los casos. En este caso, este valor es modesto, pero ya es superior al que se obtendría por pura probabilidad (ya se ha visto que el 80% de los tickets cumplen el SLA).

Como último paso se hará un gráfico de la curva ROC y se calculará el valor AUC (area under the curve) que son medidas típicas de rendimiento para los clasificadores binarios.

La curva ROC viene dada por la ratio entre Verdaderos Positivos y Falsos Positivos en distintos umbrales. El valor AUC es el área debajo de la curva ROC. Como regla general, un modelo con buena capacidad predictiva debería tener un AUC más cercano a 1 (1 es ideal) que a 0.5.

```
library(ROCR)

## Loading required package: gplots

##
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':
##
##     lowess
```

```
ROCRpred <- prediction(predictTest, test$sla_violado)
ROCRperf <- performance(ROCRpred, measure = "tpr", x.measure = "fpr")

plot(ROCRperf, colorize = TRUE, text.adj = c(-0.2,1.7), print.cutoffs.at =
seq(0,1,0.1))
```




```
auc <- performance(ROCRpred, measure = "auc")
auc <- auc@y.values[[1]]
auc

## [1] 0.8866771
```

10. Conclusiones

Este trabajo demuestra que la creación de un sistema de BI, con las herramientas disponibles hoy en día, es un proyecto asequible para cualquier organización y tiene mucho que ofrecer. Se ha podido ver cómo un sistema de este tipo permite poner en valor los datos para extraer de ellos conclusiones que impulsen la toma de acciones y que lleven a la mejora del servicio.

En la actualidad se dispone de un amplio abanico de soluciones y herramientas de BI en el mercado, gratuitas y de pago, que cubren cualquier necesidad. Sobre este punto hay que destacar la rápida evolución que se está viviendo en el sector, con la aparición de nuevos actores que ofrecen productos muy novedosos que llevan las capacidades de análisis casi al nivel de commodity. Se observa una tendencia creciente hacia soluciones de análisis predictivo, propiciados por la IA, el machine learning y la computación en la nube, y hacia soluciones muy fáciles de usar, acercando cada vez más la analítica a la gente de negocio.

En cualquier caso, los proyectos de BI siguen siendo complejos y lo son sobre todo por cuestiones relativas al gobierno y la calidad del dato, aspectos que no han sido tratados en este trabajo.

El trabajo da respuesta a los objetivos marcados inicialmente de forma satisfactoria: se ha construido un almacén de datos basado en el modelo dimensional, se han implementado los procesos de integración con una herramienta ETL y se ha puesto el foco en el análisis de datos mediante informes y visualizaciones. Además, ha tratado de hacerlo siempre ligándolo al contexto en que se desarrolla este trabajo: un proveedor de servicios de TI que quiere mejorar su Service Desk.

Se ha abierto también la puerta a las técnicas de aprendizaje automático, un campo con un enorme potencial para la mejora de la función del Service Desk, como se ha podido ver en el último capítulo. Sin duda, este punto podría abrir nuevas líneas de investigación donde se desarrollasen modelos predictivos con un conjunto de datos más amplio.

En cuanto al seguimiento de la planificación de este proyecto se ha trabajado con plazos muy ajustados que han dado lugar a un ligero retraso que se ha ido acumulando y ha obligado a un sprint final ciertamente extenuante. De aquí se pueden extraer dos conclusiones:

- El alcance del trabajo quizás era demasiado ambicioso y ha obligado a investigar y aprender herramientas que cubren todo el espectro de un proyecto de análisis de datos, lo que supone una dedicación mucho mayor de la prevista

- Y, en segundo lugar, recordar la frase de Voltaire: “Lo mejor es enemigo de lo bueno”. Se hace necesario encontrar un equilibrio razonable entre el objetivo buscado y los recursos dedicados para conseguirlo. La gestión del tiempo y este balanceo entre resultados y dedicación seguro que podrán ser mejorados en un futuro proyecto.

11. Glosario

Aprendizaje automático (Machine Learning): Conjunto de técnicas de análisis de datos que buscan patrones y estructuras en los datos para crear modelos con la capacidad de aprender.

Business Intelligence (BI): Conjunto de técnicas, herramientas y métodos orientados a convertir los datos en información útil para la toma de decisiones.

Data Warehouse (DWH): Almacén de datos integrado, no volátil y con información histórica orientado a un uso analítico.

Cuadro de mando: Informe formado por elementos gráficos que representan los KPIs más relevantes.

ETL: Acrónimo de Extract, Transform and Load, procesos que permiten extraer datos de las fuentes originales, transformarlos para adaptarlos a la estructura del almacén de datos e integrarlos en éste.

KPI: Acrónimo de Key Performance Indicator, indicador clave de rendimiento que ayuda, una métrica concreta, medible, comparable en el tiempo y que ayuda a medir el grado de consecución de un objetivo de negocio.

Service Desk: Punto único de contacto entre el proveedor de IT y el cliente o los usuarios para la gestión de incidencias y consultas.

12. Bibliografía

[1] J. Iden y T. Roar Eikebrokk, «Implementing IT Service Management: A systematic literature review,» *International Journal of Information Management*, nº 33, pp. 512-523, 2013.

[2] T. H. Davenport, «Competing on Analytics,» *Harvard Business Review*, vol. Enero, 2006.

[3] R. Kimball, The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, 3rd Edition, Wiley, 2013.

[4] S. Mukherjee, «Watson Analytics Use Case for IT Helpdesk: Minimize resolution times and maximize satisfaction,» IBM Watson Analytics, [En línea]. Available: https://www.ibm.com/communities/analytics/watson-analytics-blog/it-helpdesk-efficiency/. [Último acceso: 8 May 2019].

[5] C. Fletcher y K. Lord, «Apply Machine Learning and Big Data at the IT Service Desk to Support the Digital Workplace,» Gartner, 2016.