



UNIVERSIDAD
POLITECNICA
DE VALENCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

UNIVERSIDAD POLITÉCNICA DE VALENCIA
ESCUELA TÉCNICA SUPERIOR DE INFORMÁTICA APLICADA

2009/2010

*Implantación en una empresa de un sistema
Business Intelligence SaaS / On Demand a través
de la plataforma LITEBI*

PROYECTO FIN DE CARRERA

Autor: Rafael Matamoros Zapata

Director: Antonio Hervás Jorge

Tutor Empresa: Javier Giménez Aznar

Empresa: LITE INTERNET SOLUTIONS "LITEBI" S.L



UNIVERSIDAD
POLITECNICA
DE VALENCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica



AGRADECIMIENTOS

A mis compañeros en LITEBI, en especial a Javier Giménez Aznar y Jorge López Mateo, fundadores de la empresa, quienes me brindaron la oportunidad de formar parte de este proyecto y ampliar mi experiencia profesional, además con sus conocimientos, paciencia, dedicación y apoyo hicieron posible que saliera adelante este trabajo.

A mi director académico Antonio Hervás Jorge, por la ayuda y los consejos prestados en la realización de la presente documentación, siempre sacando lo mejor de mí mismo a través de una mirada crítica y dedicando todo el tiempo posible a responder todas mis cuestiones.

A mis padres Rafael y María del Carmen, mis hermanos Gemma y Juan Pablo, y al resto de mi familia por apoyarme siempre en mis decisiones y esforzarse para ofrecerme la oportunidad de llegar hasta este punto de mi vida, sin ellos no habría llegado hasta aquí.

A todos, gracias.



UNIVERSIDAD
POLITECNICA
DE VALENCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

ÍNDICE DE CAPÍTULOS

| | |
|--|------------|
| 1 INTRODUCCIÓN | 7 |
| 2 LA INTELIGENCIA DE NEGOCIO | 15 |
| 3 CLOUD COMPUTING | 43 |
| 4 LA EMPRESA CLIENTE: CECAV | 46 |
| 5 PROYECTO BI CECAV: MAPA SANITARIO | 49 |
| 6 ANÁLISIS DE REQUERIMIENTOS | 52 |
| 7 DISEÑO DE LA SOLUCIÓN | 67 |
| 8 IMPLEMENTACIÓN DE LA SOLUCIÓN | 82 |
| 9 RESULTADOS DE LA SOLUCIÓN | 108 |
| 10 CONCLUSIONES | 114 |
| 11 BIBLIOGRAFÍA | 115 |
| ANEXO 1: PLATAFORMA BI SAAS, LITEBI | 118 |
| ANEXO 2: USO DE IMAGEN CORPORATIVA | 173 |



LISTA DE ABREVIATURAS

| | |
|--------------|---|
| BI | Business Intelligence |
| SaaS | Software as a Service |
| PaaS | Platform as a Service |
| IaaS | Infrastructure as a Service |
| ETL | Extract, Transform and Load |
| OLTP | On-line Transaction Processing |
| OLAP | On-Line Analytical Processing |
| ERP | Enterprise Resource Planning |
| CRM | Customer Relationship Management |
| MDX | MultiDimensional eXpressions |
| SQL | Structured Query Language |
| DW | Data Warehouse |
| DSS | Decision Support System |
| EIS | Executive Information System |
| SIG | Sistema de Información Geográfica |
| CMI | Cuadro de Mando Integral |
| BPM | Business Performance Management |
| CPM | Corporate Performance Management |
| CIF | Corporate Information Factory |
| GIF | Government Information Factory |
| CECAV | CEntro de Calidad Avícola y alimentación animal de la comunidad Valenciana |

1 INTRODUCCIÓN

1.1 *Introducción y motivaciones*

Actualmente la mayoría de las organizaciones y empresas poseen y generan diariamente una enorme cantidad de datos imposibles de analizar a simple vista. La mayor parte de estos datos generados no aportan la información necesaria a la toma de decisiones empresarial, pues para poder usarlos es necesario que se transformen en conocimiento útil para quienes dispongan de ellos.

Estos datos se transforman en información cuando se analizan para estructurarlos de forma inteligente. En la actualidad, poseer un conocimiento proveniente de información comprensible, detallada, relevante y útil es vital para lograr y sostener una ventaja competitiva en el mundo empresarial. Para transformar los datos y convertirlos en información, y ésta a su vez, ser aprovechada como conocimiento, se necesitan distintas técnicas y procesos. A todos estos procesos de tratamiento de datos se les atribuye el término de Business Intelligence (BI, en adelante) o Inteligencia de Negocio.

En el mercado actual podemos encontrar muchas herramientas de BI que ofrecen al usuario la posibilidad de analizar sus datos realizando diferentes tratamientos sobre éstos, como pueden ser el análisis y la realización de informes.

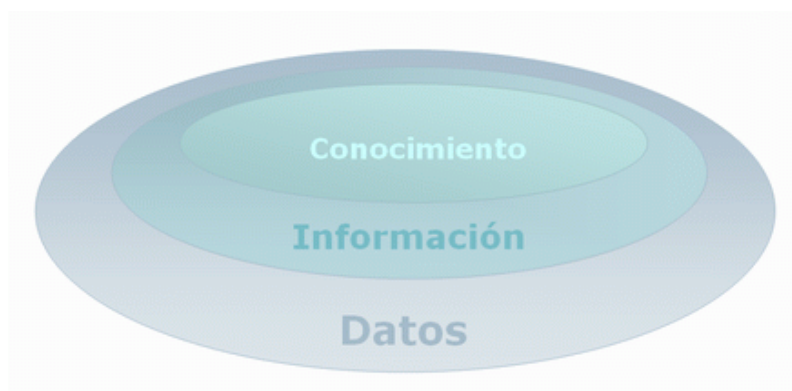


Figura 1: Extracción del conocimiento a partir de los datos

Desde la creación del Data Warehouse o Almacén de Datos en los años 80 el volumen de datos y el nivel de detalle almacenado han ido creciendo a pasos agigantados. Dos son al menos los factores que han provocado este aumento, por un lado el desarrollo de la tecnología y con éste, la automatización de los procesos en las organizaciones. A partir de este punto, surge la necesidad de controlar la información de cualquier movimiento de una organización evitando los inconvenientes de los sistemas de gestión de datos tradicionales y obsoletos, ya que día a día, aumentan los datos operacionales asociados a dichos sistemas y consecuentemente los datos que se pueden analizar y almacenar de forma organizada en estas inmensas bases de datos.

Relacionando los diferentes conceptos expuestos sobre el BI y la situación actual en los sistemas de gestión de datos, se puede decir que, cualquier organización a día de hoy necesita disponer de estrategias y herramientas de inteligencia empresarial, y en definitiva, de la potencia de las tecnologías de la información para poder obtener la mayor cantidad de información útil en el menor tiempo posible a partir de todos los datos que se generan, y transformarlos de esta forma en un activo intelectual que preste beneficios y se pueda compartir, facilitando así la toma y la corrección de las decisiones del negocio. Además, esto supone un ahorro de tiempo y dinero en el análisis y el estudio de cualquiera de las actividades de la entidad, evitando de esta manera el costoso acceso a datos de diferentes procedencias o departamentos, la generación de informes a partir de complicadas herramientas o de forma manual, así como reducir el riesgo empresarial.

Las plataformas tradicionales de inteligencia de negocio en el mercado actual aportan un sinfín de ventajas a la organización empresarial, pero suelen ser costosas, tanto a la hora de la implantación en el sistema informático de la empresa como en el mantenimiento de éste, haciendo que muchas organizaciones tengan que adaptar sus sistemas para poder obtener los beneficios de estas herramientas, por ello, muchos usuarios son reacios cuando se plantean la posibilidad de implantar estos sistemas en sus empresas. Con Litebi estos problemas desaparecen, ya que al tratarse de una plataforma de BI que utiliza el modelo de distribución de Software como Servicio (Software as a Service, SaaS) el usuario puede consultar los movimientos de su empresa, realizar informes, cuadros de mando...etc. desde cualquier lugar sólo disponiendo de un ordenador con conexión a la red, haciendo que la plataforma se

adapte perfectamente a su sistema y no al contrario, olvidándose además de esta manera de las tareas de mantenimiento.

Esta potente herramienta se caracteriza por aportar a los desarrolladores de soluciones de BI la facilidad de implementar soluciones personalizadas y adaptadas a cada caso en particular, independientemente de las diferentes fuentes de datos, en un tiempo récord, reduciendo extensos proyectos que tradicionalmente durarían varios meses a unas pocas semanas. Una vez diseñada la solución, Litebi presenta al usuario la integración en una misma plataforma de un conjunto de potentes herramientas de Business Intelligence para que el análisis de los indicadores del negocio sea lo más detallado y sencillo posible.

Todas estas cuestiones han motivado la realización del presente PFC en el que se pretende realizar una solución de Business Intelligence a la medida de las necesidades de un cliente concreto e implantarla en su sistema a través de la plataforma de Business Intelligence SaaS / On Demand Litebi , garantizando al cliente la toma de mejores decisiones para su negocio a un coste significativamente menor que otras alternativas de BI tradicionales y sin necesidad de hardware ni software especializado, únicamente accediendo a través de interfaces web.

1.2 Objetivos

Este proyecto se realizó con el fin de dar soporte y solución en la gestión y análisis de datos a una determinada empresa apoyándose sobre todo en el conocimiento de las tecnologías de la información y en concreto en las técnicas y herramientas que aporta el Business Intelligence. Para ello, se propuso realizar el análisis, diseño e implementación de una solución de BI sobre la plataforma Business Intelligence SaaS / On Demand Litebi.

Para alcanzar este objetivo se planteó la consecución de los siguientes objetivos parciales:

- Estudio de las diferentes técnicas, herramientas y conceptos sobre el Business Intelligence para su posterior aplicación en el planteamiento y diseño de la solución.
- Estudio de la plataforma de Business Intelligence SaaS / On Demand Litebi a partir de la realización de otros pequeños proyectos, ya que no existía en su día documentación asociada a la plataforma ni manual de usuario, al tratarse de una aplicación en desarrollo constante y con poco tiempo en el mercado. Debido a esto, se planteó estudiar la aplicación de forma práctica y adjuntar en este proyecto la documentación pertinente con sus principales características.
- Desarrollo de una solución de Business Intelligence destinada a cubrir las necesidades demandadas por el cliente, implementando, a partir de una arquitectura previamente diseñada, los procesos ETL que alimenten de forma actualizada la información contenida en el Data Warehouse, así como el diseño de una capa de metadatos capaz de albergar las estructuras que permitan la comunicación entre el usuario y la información necesaria con el fin de poder analizarla de forma rápida y sencilla.

1.3 Contenido del proyecto

El presente documento se puede dividir a grandes rasgos en estas secciones:

- Descripción de la información requerida para la comprensión del proyecto: situación, conocimientos, técnicas y herramientas sobre Business Intelligence y la gestión de la información para convertirla en conocimiento.
- Estudio de la funcionalidad general de la aplicación de Business Intelligence Litebi, sus principales características y las ventajas que aporta sobre otras plataformas tradicionales de BI.
- Exposición de la empresa cliente del proyecto, sus actividades y el sector al que pertenecen.

- Análisis, diseño e implementación de la solución escogida y otra serie de tareas relacionadas para lograr satisfacer los requisitos especificados por el cliente, siendo esta la base principal del proyecto.
- Generación de informes, presentación de resultados, beneficios y conclusiones obtenidas con la implantación de la solución propuesta.

1.4 Actividad de la empresa, metodología de trabajo y recursos a utilizar

1.4.1 Actividad de la empresa

El presente proyecto se desarrollará en la empresa Lite Internet Solutions “Litebi” S.L ubicada en C\Pintor López 3-5 dedicada a la implantación y comercialización de soluciones de Business Intelligence a través de su plataforma SaaS/On Demand Litebi. Dicho proyecto será respaldado y dirigido por D. Javier Giménez Aznar, Responsable de Desarrollo de Negocio.



Figura 2: Logo de Litebi

Se trata de una empresa Start-Up, con sede en Valencia (España) instaurada por Javier Giménez Aznar y Jorge López Mateo a comienzos de 2008 con el objetivo de poder aportar el Business Intelligence al alcance de todos, evitando manejar y analizar información por medio de otras herramientas como Excel y Access y así proporcionar la recuperación de la información de manera oportuna.

La Inteligencia de Negocio o Business Intelligence, que engloba conceptos como el control de gestión, los cuadros de mando o los sistemas de soporte a la toma de decisiones, tiene como objetivo mejorar el control de gestión las empresas, mediante un uso avanzado de la información disponible.

Los productos actualmente existentes se caracterizan por una enorme complejidad y un coste muy elevado. Esto deja desatendidas a la inmensa mayoría de las empresas de tamaño medio y medio-alto, que se beneficiarían enormemente del uso de estas herramientas. Existe una gran oportunidad en la oferta de soluciones de Inteligencia de Negocios bajo un modelo que permita reducir la complejidad y el coste de adopción.

En este escenario, la empresa desarrolló la plataforma Litebi con el objetivo de democratizar y extender (también para grandes empresas) los beneficios de la Inteligencia de Negocios, eliminando la complejidad y coste de adopción mediante el modelo conocido como Software como Servicio, del cual se prevé un gasto en software de este tipo de un 25% para 2012 (Gartner).

1.4.2 Metodología de trabajo y recursos a utilizar

La metodología seguida para la realización del proyecto será la siguiente:

- Como primer punto se llevará a cabo un estudio exhaustivo de qué es el Business Intelligence, los conocimientos, las herramientas y técnicas que lo componen para obtener así una mayor comprensión de cuál es el sector en el que se mueve la empresa, los beneficios que aporta con su producto y sus soluciones, además de adaptarse al modo de trabajo y obtener la metodología necesaria para concluir con éxito la realización de este proyecto.
- Una vez estudiadas las técnicas necesarias, además de la herramienta de la que se dispone, la siguiente fase a seguir será tratar directamente con el cliente, conocer su sector, las actividades que realizan, su gestión de los datos y presentarle los beneficios que la solución y la plataforma le pueden aportar para su negocio.



Gran parte de la solución se desarrollará a través de una conexión a Escritorio Remoto directamente con los servidores centrales de la empresa cliente (concretamente a través de un sistema operativo Microsoft® Windows® Server 2003 for Small Business Server) , ya que de esta manera el acceso a los datos de origen es mucho más seguro para ambas partes, teniendo únicamente como desventaja para desarrollar la solución la dependencia clara del número de accesos al servidor corporativo y la velocidad de conexión a la red, que afectará claramente al rendimiento y al tiempo de realización de los procesos que conforman la solución.

Para la realización de esta serie de tareas se dispondrá gracias a la empresa Lite Internet Solutions “Litebi” S.L de la siguiente máquina DELL Vostro 1520 con las siguientes especificaciones técnicas:

- Procesador Intel® Core™ 2 Duo T6670 2.20 GHz.
- 3GB Memoria RAM.
- Sistema Operativo Microsoft® Windows® 7 Professional 32 bits.

El software instalado para realizar los procesos y otra serie de tareas será el siguiente:

- Microsoft® Office
- MySQL Browser & MySQL Administrator (Open Source).
- Kettle™ Pentaho Data Integration 3.2.0 (Open Source).
- Navegador Google™ Chrome (Elección más potente y liviano que otros navegadores).

1.4.3 Planificación temporal de las actividades

El proyecto tiene cuatro fases principales con sus tareas asociadas:

FASE 1: Estudio del Business Intelligence (Enero)

- Investigación sobre Business Intelligence.

FASE 2: Estudio de la plataforma Litebi (Enero – Marzo)

- Estudio de la funcionalidad de la plataforma y redacción de la documentación.
- Fases de implantación de proyectos de BI.
- Realización de pequeños proyectos de BI.
- Pruebas a partir de estos proyectos.

FASE 3: Realización de la solución de BI (Marzo – Mayo)

- Toma de contacto.
- Análisis.
- Diseño.
- Implementación.
- Pruebas y presentación de resultados.

Hay que tener en cuenta que este proceso es cíclico hasta dar con una solución que obtenga unos resultados favorables en la fase de pruebas.

FASE 4: Redacción de la documentación (Mayo-Julio)

- Realización del presente proyecto a partir de los resultados obtenidos a lo largo de la estancia en la empresa y la implantación exitosa de la solución.

2 LA INTELIGENCIA DE NEGOCIO

2.1 *Introducción a la Inteligencia de Negocio*

La competencia empresarial es vital para la economía de cualquier organización que nos podamos encontrar en el mundo corporativo. Partiendo de esta base, debemos preguntarnos la manera de cómo llegar a conseguir que nuestro negocio pueda ser competitivo y más en la época tan perjudicial que rodea a nuestra economía a día de hoy. La solución: La Inteligencia de Negocio o Business Intelligence (BI).

La mayor parte de las empresas existentes generan, almacenan y modifican una enorme cantidad de datos de cualquier actividad que se registre en la empresa a través de aplicaciones de gestión de datos cada vez más complicadas de utilizar y cada vez más obsoletas (p.e sistemas de planificación de recursos empresariales, o ERP's). Es posible llegar a almacenar una gran cantidad de datos, tantos que es prácticamente imposible que una sola persona pueda obtener alguna información factible, útil y beneficiosa para llegar a tomar buenas decisiones respecto a la organización corporativa. A causa de esta necesidad, sobre los años 80 comenzaron a aparecer sistemas que ofrecían soluciones a la causa y siguen desarrollándose en el siglo XXI, unas soluciones que permiten que un usuario pueda obtener una visión global de esos datos, y por tanto un activo intelectual sin necesidad de altos conocimientos técnicos a muy corto plazo para la toma de mejores decisiones.

A lo largo de los años fueron apareciendo una serie de técnicas, herramientas y metodologías que actualmente de forma conjunta conforman lo que se conoce como el término "Business Intelligence" acuñado por Howard Dresner del grupo Gartner en 1989. Este término pretende ser la base para reunir a todo tipo de tecnologías capaces de extraer los datos corporativos almacenados por un sistema de gestión y tratarlos de manera que, al presentárselos a cualquier persona pueda obtener un conocimiento intelectual para así llevar a cabo las tareas necesarias para la consecución exitosa de las metas propuestas en su negocio.



Este término ya se vislumbró brevemente al comienzo de este proyecto pero en este capítulo trataremos de ofrecer una visión más específica de qué es el BI y ahondaremos en las técnicas y las herramientas que integra.

2.1.1 ¿Qué es la Inteligencia de Negocio?

En este capítulo trataremos de exponer o ilustrar mediante una serie de definiciones de qué trata la Inteligencia de negocio:

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

(CherryTree & Co., 2000)

“Llamamos Business Intelligence (BI) al conjunto de estrategias y herramientas enfocadas a la administración y creación de conocimiento mediante el análisis de datos existentes en una organización o empresa.”

(Wikipedia)

“Conjunto de tecnologías, métricas, procesos y sistemas que una organización usa para controlar y gestionar su rendimiento empresarial.”

(Bani Brandolini, Presidente Internacional de Tagetik)

“Business Intelligence (BI) es un conjunto de conceptos y metodologías para mejorar la toma de decisiones a través del uso de hechos y sistemas basados en hechos.”

(Gartner Group)



“La inteligencia de negocio es un proceso sistemático de recolección, análisis y gestión de información interna y externa y de conocimiento para mejorar el proceso de toma de decisiones de una empresa.”

(Jay Liebowitz, *Strategic Intelligence: Business Intelligence, Competitive Intelligence, and Knowledge Management*, 2006)

“Los sistemas de BI convierten los datos en bruto de una compañía en información usable que pueda ayudar a la dirección a identificar tendencias importantes, analizar el comportamiento de clientes y tomar decisiones de negocio inteligentes rápidamente”

(Sun Microsystems 2005)

“La inteligencia de negocio es una amplia categoría de aplicaciones y tecnologías para recoger, almacenar, analizar y proveer acceso a datos para ayudar a los usuarios de la empresas a tomar mejores decisiones de negocio. Las aplicaciones de BI incluyen las actividades de los sistema de soporte a las decisiones (DSS), consultas e informes, tecnologías OLAP, análisis estadístico y data mining.”

(SearchCRM.com)

En definitiva podemos afirmar que la inteligencia de negocio puede tener dos proyecciones diferentes según se mire desde el punto de vista del negocio o desde el punto de vista técnico, donde se tienen más en cuenta las herramientas y las tecnologías por encima de la metodología de uso de la información en la toma de decisiones del negocio. En la figura 3 podemos observar el esquema clásico de una solución de BI.

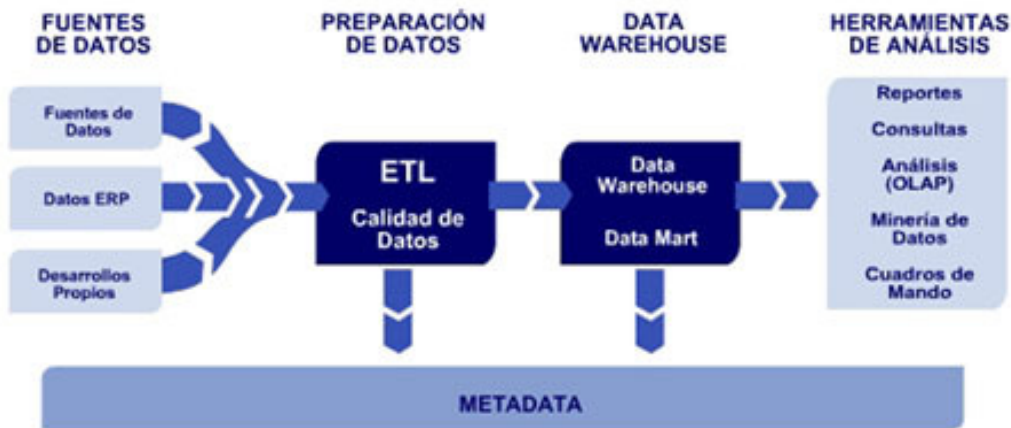


Figura 3: Esquema de una solución de BI

2.1.2 La Inteligencia de Negocio: Cifras y Beneficios

El Business Intelligence ha evolucionado a lo largo de los años para dotar, sobre todo a los altos ejecutivos encargados de analizar la información estratégica, de la capacidad de garantizar mejores decisiones. Anteriormente las personas encargadas del análisis competitivo a nivel corporativo solamente mostraban superficialmente el potencial de la inteligencia de negocios dentro de la empresa involucrando quizá el 5% de los usuarios y el 10% de los datos disponibles (*Information Builders, 2005*).

Con la llegada del concepto de BI se observaron una serie de ventajas destacables en la implantación de estas soluciones para el análisis de información en una organización:

- **Información fácil, potente y asequible:** Sin necesidad de conocimientos técnicos cualquier usuario, como un director comercial, puede acceder a información útil, organizada e integrada (gracias al concepto de Data Warehouse) en una misma aplicación en pocos minutos.
- **Información segura:** La información perteneciente a cada usuario queda distribuida y controlada por medio de un sistema securizado vía web, aunque esta información queda centralizada en un sólo lugar y siempre quedará allí (no desaparecerá jamás).

- **Análisis más sencillo y fiable:** Toma de mejores decisiones por parte de la directiva a partir de cómodas herramientas (Cubos OLAP, reporting systems, consultas ad-hoc).
- **Control de estrategia empresarial:** Análisis de un cierto campo desde diferentes puntos de vista (Dashboards o Balanced ScoreCards).
- **Datos útiles, actualizados y flexibles:** En el diseño del Data Warehouse sólo se seleccionan aquellos datos que vayan a ser relevantes en el análisis y se les dota de una presentación o un formato adecuado que puede cambiar según los requerimientos de la empresa.
- **Eliminación del error humano:** Todos los informes se realizan a través de la plataforma evitando la generación de costosos documentos en Excel de forma manual.
- **Automatización de circulación de información:** Planificación automática de envío de informes dentro de una periodicidad a las personas necesarias, esto permite una mejora en la comunicación (Sistemas de alarmas, reporting automático).
- **Previsión sobre hipotéticos escenarios:** Análisis de ciertos indicadores clave a través de ciertas herramientas que, junto con alertas, indiquen el impacto a futuro de éstos (Análisis what-if).
- **Organización de gran cantidad de información:** Gracias a herramientas como el Data Mining y la utilización de Data marts.
- **Sistemas de estructura escalable:** Esto hace posible que los sistemas crezcan de forma regulada.
- **Ahorro de tiempo y dinero:** Estas soluciones permiten que cada sujeto cumpla su función y dedique tiempo para tareas más importantes, por tanto permite un aumento de la productividad. El tiempo es dinero.

Una encuesta realizada por Gartner en 2005, situó la Inteligencia de Negocio en el número 2 en la lista de prioridades tecnológicas de los CIO, ya que el mercado de herramientas software de BI creció un 7.7 % en 2004. Este crecimiento se produjo a través de compañías como Cognos y Microsoft, los cuales fueron los máximos beneficiados de aquella época.

Año tras año Gartner realiza previsiones de este tipo sobre la evolución del mercado de Business Intelligence ofreciendo cifras y predicciones a los interesados en este sector. La última encuesta reseñable es la de 2009, donde destaca como la crisis económica está afectando a la estrategia empresarial así como al mundo de la Inteligencia de Negocios:

- **Predicción 1:** *“En el 2012 las unidades de negocio (no los departamentos de sistemas o tecnología) serán responsables del más del 40% de presupuesto de los proyectos de BI.”*

En la actualidad la decisión de implantar una solución de BI suele delegarse sobre el departamento técnico cuando deberían ser los responsables de analizar la información, y por tanto de tomar decisiones corporativas, quienes deberían optar por estas opciones apostando por una visión de negocio más que por una herramienta.

- **Predicción 2:** *“Aún en el 2012, más del 35% de las principales 5,000 empresas mundiales (TOP 5,000) tomarán decisiones de manera desinformada debido a la insuficiente inversión en infraestructura de la información y herramientas de negocio para los usuarios.”*

Esto es debido a causa de la crisis económica, las empresas deberían invertir en información para superar el bache donde nos encontramos pero desgraciadamente no es así, cuando lo más aconsejable sería utilizar este recurso para garantizarse estrategias de mercado fiables.

- **Predicción 3:** *Para el 2010, el 20% de las empresas usará una aplicación de análisis relativa a su industria a través del esquema de SaaS (Software as a Service).*

Esta filosofía se ha impuesto en los últimos años como compañera inseparable de la Inteligencia de Negocio por su simplicidad, bajo coste y eficacia en la implantación de estas herramientas, aunque todavía se encuentra en perfeccionamiento.

- **Predicción 4.** *En el 2009, la toma de decisiones en un medio colaborativo creará un nuevo título en el BI que combinará el software social con la plataforma de BI.*

Se ha comprobado que la toma de decisiones suele ser mucho más fructífera cuando se cuenta con una serie de opiniones para ello. Las redes sociales hoy en día han abierto un abanico de posibilidades de estrategias de negocio y por tanto en un futuro se prevé que se puedan utilizar conjuntamente con las herramientas de BI.

- **Predicción 5.** *En el 2012, una tercera parte de las aplicaciones analíticas aplicadas a los procesos de negocio serán entregadas a través de mashups.*

Los mashups son un tipo de aplicaciones web formadas por diferentes contenidos de otros sitios web. Se apuesta que este modelo sirva para completar con mayor información las aplicaciones de BI.

2.1.3 Tipos de usuarios BI

La gran mayoría de las organizaciones se encuentra organizada jerárquicamente en una forma piramidal. En cada escalón de esta pirámide podemos situar a los diferentes tipos de usuario a la hora de tratar las herramientas que ofrece la Inteligencia de Negocio, así como determinar las decisiones de las que se harán cargo como podemos ver en la figura 4.

Dirección General: Aquí se sitúan los altos cargos de la empresa, quienes cargan con la mayor responsabilidad en la organización y, por tanto, no disponen de tiempo suficiente para dedicarlo al análisis de información. Es por esto que utilizan herramientas como scorecards o dashboards, donde obtienen una visión rápida y global de los movimientos del negocio, que además puede contener enlaces a informes más concretos.

Cargos Medios: A esta altura encontramos el perfil típico de la rama Administrativa de la empresa. Estos cargos disponen de herramientas para el análisis de ciertos indicadores del negocio a mayor detalle que sus jefes inmediatos, ya que deben responder con resultados e informes ante ellos. Las herramientas de OLAP y consultas ad-hoc permiten que este tipo de usuarios pueda representar de forma visual diferentes tipos de análisis tanto en informes de tablas como en gráficos, partiendo de una gran cantidad de datos distribuida en cubos multidimensionales para realizar consultas.

Operarios: Son los cargos más bajos de la empresa, pero aún así las herramientas de Inteligencia de Negocio les aportan cantidad de beneficios en su labor. Son usuarios de lo que se conoce como Informes Predefinidos, es un tipo estándar de informe con posibilidad de compartirlos con un formato típico corporativo, donde se incluyen gráficas y tablas de información obtenidas directamente de los procesos ETL.

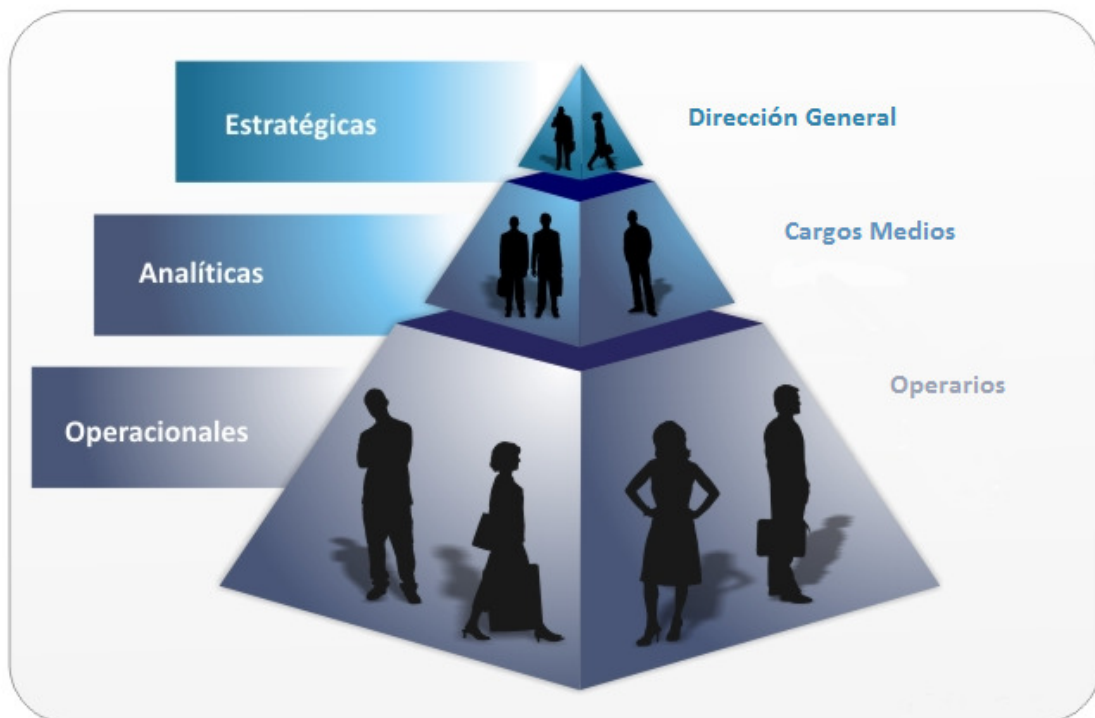


Figura 4: Toma de decisiones de acuerdo a un tipo de usuario

Todas estas herramientas se describirán detalladamente en el siguiente apartado.

2.2 Herramientas y técnicas

Como se ha mostrado a lo largo del presente capítulo, la Inteligencia de Negocio integra una serie de herramientas, tecnologías, metodologías y técnicas orientadas a aportar beneficios frente al tratamiento de la información en el negocio. En este capítulo se enumeran y se detallan los diferentes conceptos que forman la inteligencia de negocios, tanto técnicas referidas más al negocio como herramientas obtenidas con la informática.

2.2.1 OLTP (On-line Transaction Processing)

Tecnología que se utiliza para administrar aplicaciones que utilizan operaciones transaccionales, es decir, sistemas donde se realizan una gran cantidad de modificaciones y entradas de datos y pocas lecturas masivas de los mismos. En estos sistemas es necesario tener un tiempo de respuesta aceptable a la hora de realizar las modificaciones de los datos.

2.2.2 OLAP (On-line Analytical Processing)

Estas herramientas manejan una serie de consultas de forma interactiva sobre estructuras multidimensionales (Cubos OLAP) cargadas previamente con los datos almacenados en las bases de datos corporativas tradicionales. Permiten realizar informes y obtener grandes cantidades de información a partir de lo que resultaría ser a modo rutinario una serie de complejas consultas sobre una base de datos de forma sencilla. Al estar los datos precompilados sobre una estructura intermedia, el tiempo de respuesta de las consultas es menor, posee una enorme potencia de cálculo y técnicas de indexación especializadas. Esta tecnología es favorable en un sistema OLTP pero suele ser lenta si se realizan complejas consultas. Con estos sistemas es posible analizar la información almacenada en un data warehouse, pero no es estrictamente necesario, ya que la información puede provenir de diferentes bases de datos. El objetivo de estas herramientas es obtener una mejor comprensión de lo almacenado en las bases de datos.

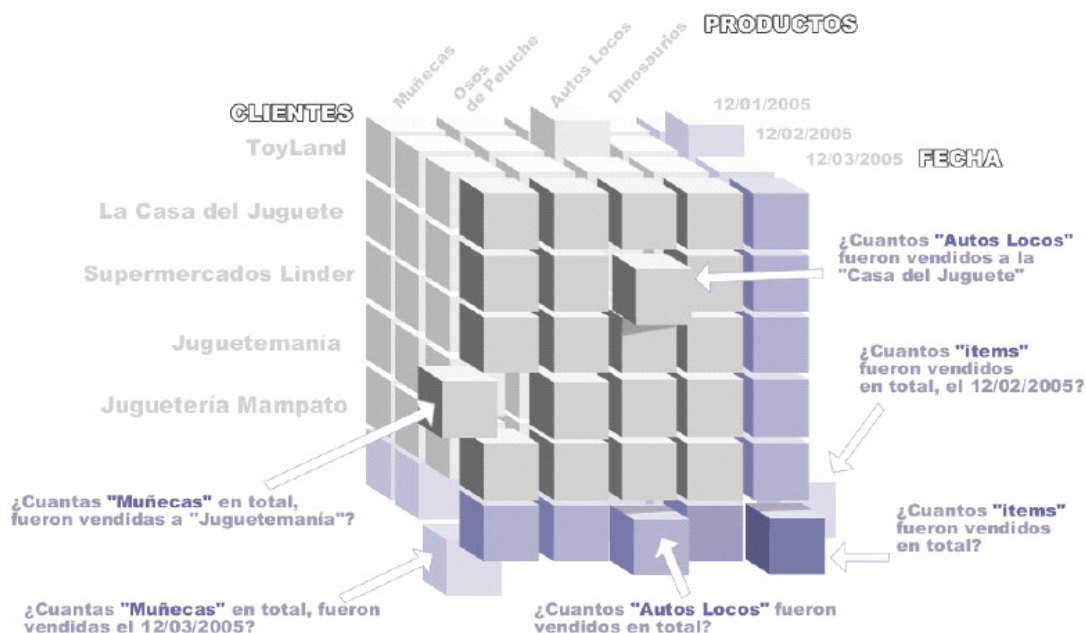


Figura 5: Ejemplo de Cubo OLAP

Existe una categorización para estas herramientas según su arquitectura:

- M-OLAP (Multidimensional OLAP): Sistema OLAP que posee los datos almacenados en una base de datos multidimensional. Esta implementación mejora los tiempos de acceso a los datos ya que están precalculados a costa de necesitar mayor espacio de almacenamiento, aunque algunos sistemas utilizan la compresión. Es un sistema OLAP compuesto por Cubos.

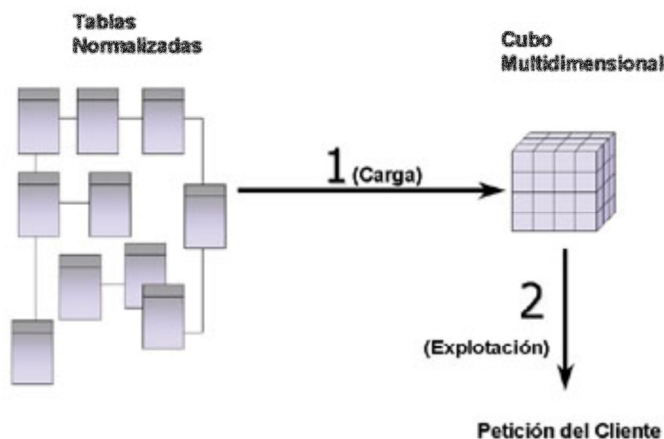


Figura 6: Sistema M-OLAP

- R-OLAP (Relational OLAP): Sistema OLAP que mantiene los datos almacenados en una base de datos relacional. Para esta implementación se realiza un Cubo virtual o tablas en forma de estrella con lo que se consigue una mayor capacidad de almacenamiento sacrificando tiempo de respuesta.

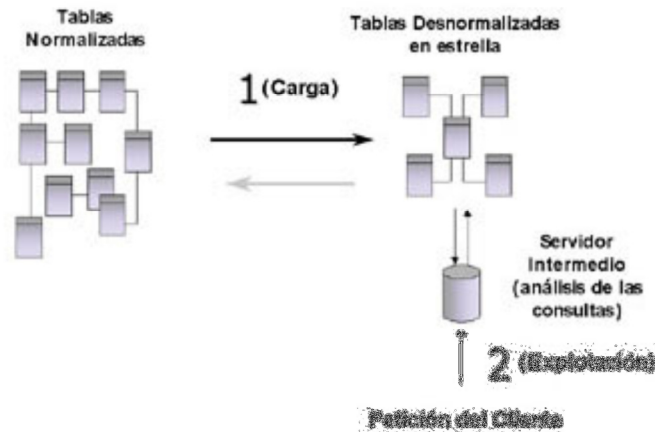


Figura 7: Sistema R-OLAP

- H-OLAP (Hybrid OLAP): Combinación de los dos sistemas anteriores donde los datos se almacenan repartidos en implementaciones M-OLAP y R-OLAP. Esta combinación permite obtener ventajas de ambas implementaciones según donde se almacene el dato y las operaciones que se vayan a realizar sobre él.

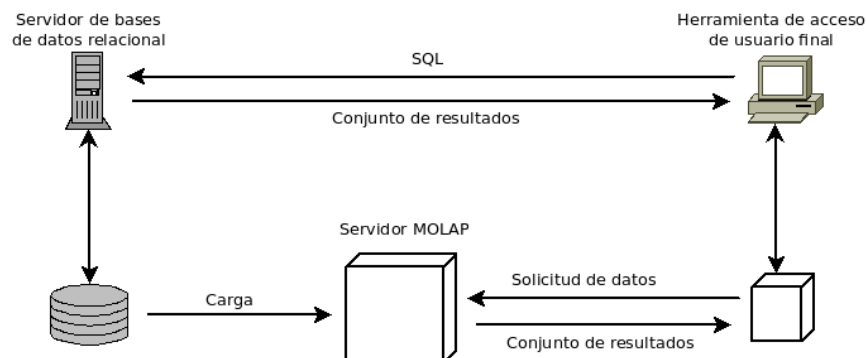


Figura 8: Sistema H-OLAP

- D-OLAP (Desktop OLAP): Esta implementación se basa en almacenar los datos en un servidor intermedio para mejorar el rendimiento, ya que muchos de los datos de los sistemas M-OLAP y R-OLAP son descargados en máquinas individuales. Este sistema se utiliza cuando la solución BI está diseñada sobre una aplicación web.

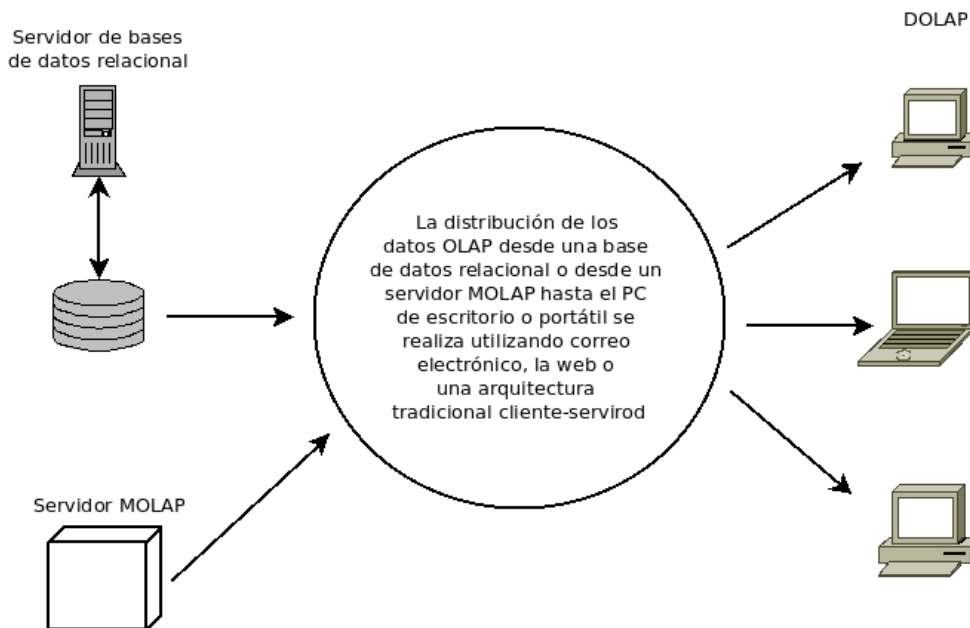


Figura 9: Sistema D-OLAP

En cualquier herramienta OLAP encontramos dos tipos de variables características, independientemente del modo en que estén almacenados los datos:

- Dimensiones: Estas variables nos indican los diferentes puntos de vista con los que podemos analizar la información, representa una perspectiva de los datos. Forman parte de la Tabla de Dimensiones. Las dimensiones son usadas para seleccionar y agregar datos a un cierto nivel deseado de detalle. Las dimensiones se relacionan en jerarquías o niveles, esto es, un conjunto de niveles cada uno expresando un nivel de profundidad en la información. Permiten analizar la información de forma agrupada (ventas por año) o al detalle (ventas diarias). *Ejemplo: La dimensión tiempo está formada por la jerarquía "Por Trimestre", que tiene los niveles "Año > Trimestre > Mes > Día".*

- Indicadores o Métricas: Es el dato que está siendo analizado, aquello que es cuantificable en lo que se desea analizar, suelen ser valores numéricos. *Ejemplo: Número de productos vendidos en el mes de Mayo.* Están contenidos en la Fact Table o Tabla de Hechos, base del modelo dimensional, la cual se ubica en el centro y alrededor tendrá las Tablas de Dimensiones, formando un esquema en estrella. Todo objeto de análisis es un hecho y por tanto, éstos contienen las métricas, y las dimensiones contienen los diferentes niveles que componen las jerarquías.

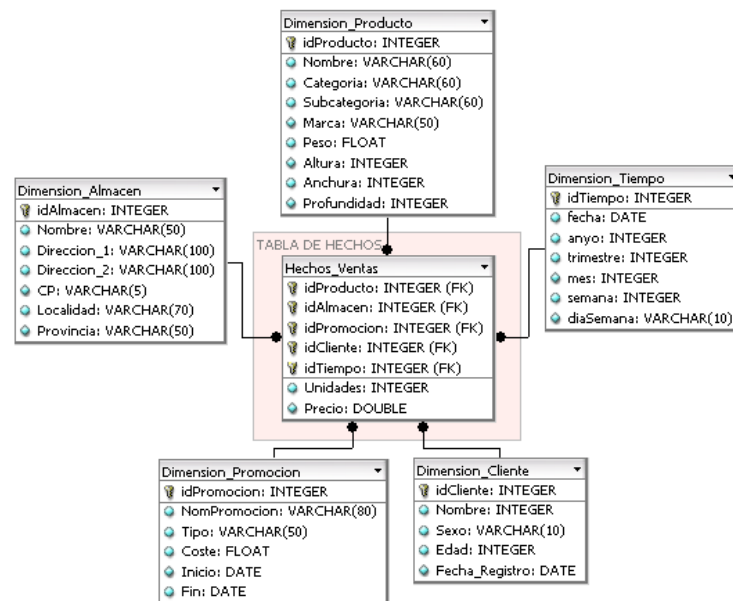


Figura 10: Ejemplo de esquema en estrella

La forma de exploración de los datos en análisis OLAP suele ser en forma de matriz, donde sobre cada uno de los ejes se sitúa una dimensión y sobre las celdas se sitúan las métricas, conteniendo el valor en función de las dimensiones escogidas. A partir de estas matrices de datos es posible generar distintos tipos de gráficas asociadas a las mismas.

| | | Measures | | |
|------------------|----------------|-------------|-------------|-------------|
| Products | Region | Measures[0] | Measures[1] | Measures[2] |
| -All Products[0] | -All Region[0] | 902.31 ↗ | 1,069.87 ↗ | 1,195.14 ↘ |
| | +Region[0] | 848.53 ↘ | 1,031.69 | 884.32 |
| | -Region[1] | 975.27 ↗ | 805.13 ↗ | 948.12 ↘ |
| | +City[0] | 893.16 ↘ | 1,034.77 | 834.51 |
| | +City[1] | 1,012.79 ↗ | 884.11 ↗ | 1,101.19 ↘ |
| | +City[2] | 1,013.88 ↘ | 1,054.62 | 972.94 |
| | +City[3] | 881.08 ↗ | 1,175.80 ↗ | 967.78 ↘ |
| | +City[4] | 1,094.37 ↘ | 1,032.40 | 857.15 |
| | +City[5] | 1,002.53 ↗ | 997.01 ↗ | 1,073.30 ↘ |
| | +City[6] | 918.05 ↘ | 1,085.70 | 953.94 |
| | +City[7] | 1,099.73 ↗ | 944.40 ↗ | 1,190.58 ↘ |
| | +Region[2] | 936.99 ↘ | 1,178.08 | 1,032.93 |
| | +Region[3] | 903.77 ↗ | 956.61 ↗ | 1,133.11 ↘ |
| | +Region[4] | 1,140.93 ↘ | 1,052.94 | 978.66 |
| +Category[0] | -All Region[0] | 1,017.41 ↗ | 787.35 ↗ | 1,035.24 ↘ |
| | +Region[0] | 1,019.25 ↘ | 1,123.19 | 923.65 |
| | +Region[1] | 877.00 ↗ | 1,148.39 ↗ | 1,085.34 ↘ |
| | +Region[2] | 752.89 ↘ | 1,069.97 | 943.07 |
| | +Region[3] | 829.57 ↗ | 916.46 ↗ | 1,010.03 ↘ |
| | +Region[4] | 920.09 ↘ | 1,017.18 | 1,041.96 |
| +Category[1] | +All Region[0] | 1,032.02 ↗ | 898.40 ↗ | 1,022.07 ↘ |
| +Category[2] | +All Region[0] | 963.97 ↘ | 1,016.30 | 935.30 |

Figura 11: Ejemplo de análisis OLAP

A partir de esta disposición inicial existen diferentes operaciones a realizar sobre los datos para obtener diferentes tipos de vistas o diferentes tipos de información:

- Drill Down: Descomponer de manera visual algún dato en detalle según una cierta jerarquía de una dimensión.
- Drill Up: Añadir un dato concreto según una cierta jerarquía de una dimensión.
- Drill Through: Obtener una vista detallada de un elemento concreto. Por ejemplo, se desea observar un listado de los pedidos de cierto país al máximo detalle.
- Rotación: Intercambiar las dimensiones de ejes.
- Filtro: Seleccionar cierta información según un criterio de filtrado. Por ejemplo, obtener únicamente los pedidos de 4 países concretos en un trimestre determinado.



2.2.3 Query & Reporting

Herramientas para elaborar informes y listados no demasiado complejos con los datos más usuales y más sencillos de analizar, tanto de manera agregada como detallada de la información. Este tipo de herramientas trabajan de forma óptima a través de sistemas basados en almacenes de datos ya que su tecnología y su forma de estructurar la información favorecen que con estas herramientas se obtengan tiempos de respuesta menores en las consultas, no siendo así en los sistemas tradicionales.

Estas herramientas ofrecen diferentes formas de presentación y diseño (Excel, PDF,...), según el tipo de información analizada y dependiendo de para qué causa esté destinada. Algunas de estas herramientas ofrecen al usuario la posibilidad de refrescar la información de informes almacenados y que poseen información estática por medio de alarmas previamente programadas, de esta manera el usuario siempre dispondrá de la información actualizada. Los informes se pueden compartir con otros usuarios haciéndolos públicos o gestionando la suscripción a ciertos informes para dotar a los usuarios de diferentes permisos sobre los informes.

2.2.4 EIS y DSS

El concepto de Sistema de Apoyo a las Decisiones (Decision Support System, DSS) es muy amplio, ya que existen diversos puntos de vista dependiendo del ámbito en el que se deben tomar las decisiones. Se podría decir que se trata un sistema informático sobre el que se apoya la dirección en la toma de decisiones empresarial. Lo que a finales de los 80 era conocido como DSS ha progresado y evolucionado hasta el punto de convertirse en lo que hoy conocemos como Inteligencia de Negocio, aunque se sigue adoptando este concepto cuando se habla de reporting.

Un sistema de información ejecutiva (Executive Information System, EIS) es un tipo de DSS. Estas herramientas, también denominadas Cuadro de Mando Analítico, ayudaban a los altos ejecutivos a tener acceso al estado de los indicadores de su negocio, estudiarlos con detalle y tomar las medidas necesarias para obtener un rendimiento óptimo. Una de sus principales características era una atractiva y sencilla

interfaz, la cual hacía posible que usuarios con un perfil no técnico pudiesen disfrutar de las posibilidades de la herramienta. Los EIS se apoyan sobre un almacén de datos para elaborar presentaciones con los indicadores clave de gestión del negocio (KPI's) para la dirección, a diferencia de la generación de informes y listados. En la actualidad se ha adaptado la estructura de los EIS para obtener lo que se conoce como Cuadro de Mando Integral o Balanced Scorecard.

2.2.5 Cuadro de mando integral

El Cuadro de Mando Integral (CMI) o Balanced Scorecard fue presentado por los consultores e investigadores de negocios Robert Kaplan y David Norton en 1992 como base de un trabajo realizado para una empresa de semiconductores. Este modelo parte de que la estrategia empresarial es el punto más importante a tener en cuenta para controlar la evolución de la organización, así que se podría decir que está fuertemente ligado a los conceptos de “Dirección Estratégica” y “Business Performance Management”.

El cuadro de mando toma como núcleo principal la visión estratégica y la organización para aplicarla en diferentes perspectivas a toda la empresa y así mejorar su rendimiento por medio de indicadores de acción, objetivos y estándares. Estas perspectivas son:

- **Perspectiva financiera:** Esta perspectiva aborda los objetivos financieros de la empresa. Se trata de ofrecer una estrategia sobre la contabilidad por medio de indicadores financieros para obtener un estado financiero favorable para la empresa.
- **Perspectiva del cliente:** Para obtener el estado financiero deseado se necesitan estrategias para obtener clientes y satisfacer sus necesidades. Desde esta perspectiva se miden los indicadores y se tratan los objetivos que ayudan a la empresa en la relación con el cliente.

- Perspectiva de procesos: Desde esta perspectiva se pretende controlar las actividades internas de la empresa (compras, ventas, producción...etc.) para satisfacer los objetivos propuestos con el resto de categorías: satisfacer las necesidades del cliente y conseguir un estado financiero en crecimiento por medio de indicadores de gestión de procesos, objetivos e iniciativas.
- Perspectiva de formación y crecimiento: Estos indicadores forman el conjunto de recursos humanos que dotan a la organización de habilidades para mejorar y aprender. Esta perspectiva suele ser la menos desarrollada hoy en día y esto se traduce en que algunas empresas experimenten fracasos en el resto de perspectivas debido a que no poseen unos objetivos e iniciativas de formación y aprendizaje correctamente fijados.

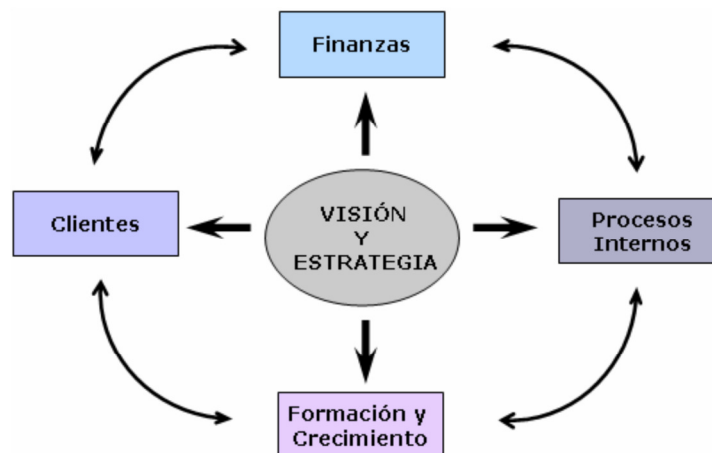


Figura 12: Perspectivas de un cuadro de mando integral

A partir de esta serie de categorías en las que aplicar la estrategia de negocio, se puede deducir que los cuadros de mando integrales son una potente herramienta de inteligencia de negocios para la dirección donde poder apoyarse y tomar las decisiones estratégicas necesarias para obtener el máximo rendimiento y evolución posible de su empresa. Un cuadro de mando ideal sería aquel que proporcionara una



visión global instantánea del estado actual de la empresa para definir posteriormente las estrategias necesarias. Se podría decir que estaríamos ante la cumbre de los sistemas de inteligencia de negocios.

2.2.6 Data Mining

La minería de datos consiste en extraer conocimiento útil a partir de los datos en bruto de una organización. Las empresas almacenan grandes cantidades de información oculta en sus datos y gracias a estas técnicas y herramientas informáticas y estadísticas es posible que esta información vea la luz aportando conocimiento beneficioso para el usuario por medio de clasificaciones y predicciones.

El objetivo de estas técnicas no es otro que encontrar patrones ocultos de comportamiento, tendencias y correlaciones entre los datos para disponer de suficiente información como para realizar modelos estadísticos que pueden servir para prever ciertas situaciones de la organización. Esta serie de patrones y tendencias se suelen agrupar en lo que se denomina como “Modelo de minería”, los cuales se pueden utilizar posteriormente en diferentes escenarios hipotéticos o simulados de negocio. Data mining se apoya en una serie de técnicas de adquisición de conocimiento y aprendizaje basadas en:

- Redes neuronales: modelos de aprendizaje de conocimiento semejantes a una red neuronal biológica.
- Árboles de decisión: Modelos en forma de árbol los cuales representan caminos de decisiones a tomar. Gracias a estas decisiones se obtienen clasificaciones para conjuntos de datos.
- Algoritmos genéticos: Diseño basado en conceptos de evolución tales como mutaciones y combinaciones genéticas para la optimización de procesos.
- Método del vecino más cercano: Técnica utilizada para la clasificación de registros en una serie de conjuntos basándose en similitudes con una serie de datos históricos.
- Reglas de inducción: Partiendo de una premisa verdadera es posible llegar a una conclusión que quizás también lo sea. Se basa en técnicas estadísticas y lógicas.

- Análisis de series temporales: Técnicas de análisis de las relaciones subyacentes de un conjunto de datos ordenados cronológicamente con el fin de predecir su comportamiento futuro.

Cabe destacar que, a diferencia de otras herramientas de BI, con la minería de datos no sólo se traslada y se presenta la información sino que se presentan datos al usuario que antes no eran visibles a simple vista.

2.2.7 CRM

Customer Relationship Management en inglés. Es un concepto que trata de romper con el marketing tradicional, más centrado en el producto, y centrarse en el cliente, en establecer buenas relaciones con él y satisfacer sus necesidades. Hay que tratar al cliente de tal manera que se tenga como objetivo a cumplir el obtener su lealtad y confianza en el servicio o el producto que le estemos ofreciendo. Esta manera de gestionar la empresa orientándola hacia el cliente no sólo se puede conseguir por medio de una herramienta informática sino que cabe inculcar una cultura en la organización y cambiar algunos aspectos tradicionales.

Estas herramientas cubren en la actualidad aspectos como: automatización de la fuerza de ventas, seguimiento de oportunidades, control de agenda y contactos, control de campañas de marketing, análisis de la información obtenida de los clientes...etc. En definitiva son herramientas que potencian la generación y la obtención de información sobre los diferentes perfiles de clientes para anticiparse a su demanda. En la actualidad podemos encontrar soluciones CRM On Demand (On-line), que parecen ser las que mejor aceptación están teniendo entre los usuarios y los departamentos de RRHH que manejan estas aplicaciones. Algunos ejemplos que podemos encontrar son Salesforce o como alternativa open source, SugarCRM.

2.2.8 BPM y CPM

BPM y CPM son conceptos que se consideran sinónimos, Business Performance Management y Corporate Performance Management. Este concepto viene a considerar el análisis y el control de los indicadores clave del rendimiento del negocio con el objetivo de mejorar la eficiencia de la organización, tal como se vio en los Cuadros de Mando Integral pero de un modo más amplio.

Se podría decir que el BPM, también coincidente con Business Process Management, va más allá del Business Intelligence, ya que se podría controlar y mejorar la empresa a través de una serie de aplicaciones informáticas que gestionen los procesos del negocio, apoyándose en las herramientas anteriormente expuestas. Por tanto, se puede observar que el BI y el BPM están claramente ligados en cuanto nos referimos a técnicas de mejora empresarial y a la monitorización de los indicadores clave del negocio, aunque el BPM se podría considerar una progresión del BI.

2.2.9 CIF y GIF

Las siglas CIF y GIF vienen de Corporate Information Factory y Government Information Factory respectivamente, presentados por el iniciador de la teoría del data warehousing Bill Inmon. Estos términos se refieren en si a una solución integral de inteligencia de negocios teniendo en cuenta además el data warehouse propuesto por Inmon y, añadiendo para completar el concepto, cualquier herramienta o técnicas fuera del apartado informático que aporten beneficios a la compañía. Se podría decir que estos términos agrupan cualquier sistema informático y de información presente en la compañía para el análisis y la explotación de los datos.

2.2.10 Gestión del conocimiento

El término gestión del conocimiento agrupa una serie de técnicas para gestionar, controlar y transmitir toda la información acumulada en la compañía a lo largo de su historia para que este conocimiento no quede restringido solo en ciertos sectores o a ciertos empleados de la empresa o que caiga en el olvido sin darle el uso apropiado. Este concepto ha tomado más forma en cuanto se empezaron a incorporar herramientas informáticas en las organizaciones, basadas en tecnologías de la información con el fin de almacenarla, gestionarla y compartirla (Intranets, Sites, Data warehouses, Wikis...).

Hay autores que afirman que no es posible la implantación de un sistema de BI sin la existencia previa de un sistema o una serie de técnicas para la gestión del conocimiento. Esta afirmación no es una ley, pero de esta manera se puede observar la relación entre los sistemas de gestión del conocimiento y los sistemas de BI a la hora del manejo y el transporte de información en una compañía, en la cual la combinación y el apoyo mutuo de ambas facilitaría enormemente la transformación de los datos en conocimiento y la utilización de éste en la toma de mejores decisiones.

2.3 El Data warehouse

2.3.1 Introducción

El término Data warehouse o Almacén de datos fue acuñado por Bill Inmon en 1990.

Esta idea surgió debido a las necesidades informacionales de las empresas, entendiéndose como la necesidad de tener acceso a la información necesaria que sirva de base para la toma de decisiones tanto a escala estratégica como táctica. Años atrás se pusieron en práctica una serie de medidas para satisfacer estas necesidades pero no fueron suficientes debido a innumerables problemas como: consultas masivas y complejas de información que interferían en el rendimiento de otros procesos, limitada flexibilidad de navegación a través de la información, tiempo de respuesta elevado debido al acceso a múltiples lugares al mismo tiempo...etc. Inmon fue capaz de solventar esta serie de problemas gracias a la teoría del Data warehouse. Este sistema supone la centralización y almacenamiento de todas las fuentes de datos de la



empresa en un solo lugar y de manera organizada e integrada. El objetivo era disponer de toda la información histórica de la empresa centralizada facilitando de esta manera el análisis, además de mantener la información con una estructura organizada unívoca y acorde a las necesidades de consulta, y alimentando el almacén con la información contenida en las aplicaciones de manera periódica para no dejar escapar ningún tipo de información. Utilizando las herramientas expuestas en el capítulo anterior sobre el contenido de un data warehouse es cuando obtenemos una solución completa de inteligencia de negocios. Por tanto, se puede afirmar que el data warehouse es imprescindible en cualquier solución de BI.

2.3.2 Definiciones

La teoría del data warehouse (DW) fue un gran impacto para el mundo corporativo. Este concepto fue introducido además de por Bill Inmon, por Ralph Kimball. Ambos autores coincidían en la base del concepto, pero discrepaban en una serie de diferencias según el punto de vista de cada uno en relación al concepto y el uso del data mart (subconjunto de información del data warehouse que hace referencia a un departamento o sector concreto de la empresa). A partir de estas diferencias cada uno de los autores definió el data warehouse según su punto de vista:

“Un data warehouse es una colección de datos orientada a un dominio, integrado, no volátil y variable en el tiempo que ayuda en la toma de decisiones en una organización”

(Bill Inmon, 1992)

Para este autor el data warehouse es una parte del sistema de la inteligencia de negocios. Los data marts se crean después de diseñar el DW y obtienen la información de éste, no siendo posible la consulta directa de información (se dice que la información no está almacenada de forma dimensional). Esta aproximación es conocida como “Top-down”.



“Un data warehouse es una copia de los datos de transaccionales especialmente estructurada para la consulta y el análisis”

(Ralph Kimball, 1992)

Para Kimball el data warehouse es el conjunto de todos los data marts existentes en la empresa, aunque normalmente se almacenen de forma separada. Bajo esta concepción la información está almacenada en un modelo dimensional y por tanto está lista para ser consultada. Esta aproximación se conoce como “Bottom-up”, una metodología ascendente a la hora de diseñar un almacén de datos.

En la actualidad estas diferencias carecen de importancia aunque se han dado casos en grandes empresas que la concepción de Inmon es mucho más efectiva que la de Kimball, ya que tener un data warehouse centralizado resuelve muchos problemas.

Por el contrario, la aproximación multidimensional de Kimball ha terminado imponiéndose como estándar a la hora del diseño de data warehouses.

Para un concepto tan extenso y revolucionario, las definiciones dadas por Inmon y Kimball en su día parecen escasas e incompletas. A lo largo de los años se han ido obteniendo definiciones más completas y detalladas:

“Un data warehouse es un conjunto integrado de bases de datos que se diseña y utiliza para apoyar en la toma de decisiones y en él cada unidad de datos es relevante en algún instante de tiempo, además, contiene información no sólo de bases de datos relacionales, sino de otras fuentes relacionadas con la actividad de la organización y cuya finalidad no sólo se centra en el almacenamiento de esos datos, sino en su análisis y procesamiento mediante los procesos encargados de su gestión para la obtención de información estructurada y en definitiva útil para la toma de decisiones”

(Delgado, 1999)

Como se puede apreciar el concepto ha evolucionado y ha tomado un rumbo más orientado hacia su uso que hacia las metodologías de diseño.

2.3.3 Modelos de tablas

A la hora de diseñar un data warehouse, uno de los elementos esenciales para su diseño es el modelo de tablas a escoger. Actualmente el modelo de tablas que normalmente se sigue en el diseño de un data warehouse suele ser el modelo multidimensional propuesto por Kimball, ya que se ha impuesto con el tiempo al modelo relacional de Inmon, salvo algunas excepciones. El modelo a seguir se denomina esquema en estrella.

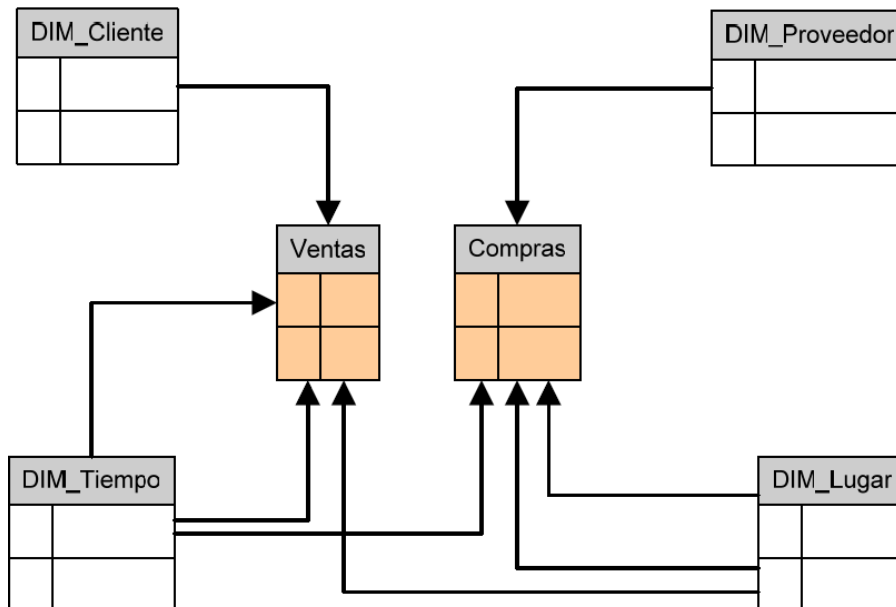


Figura 13: Ejemplo de modelo en estrella

Este modelo permite el análisis de información a partir de una base de datos relacional, estructurando los datos y organizándolos de forma multidimensional.

Este modelo está compuesto de una serie de tablas que se clasifican en:

- Tablas de hechos: Es la tabla central o nuclear de un esquema multidimensional. Esta tabla contiene los datos a medir, aquello que es cuantificable, datos que suelen ser las mediciones numéricas del negocio. Todo dato a medir es un hecho donde la

granularidad de éste hecho viene condicionada según el nivel de detalle con el que se almacene en la tabla, ya que cada medida es tomada según la intersección de las dimensiones que la definen. Se distinguen por tanto dos tipos de columnas en una tabla de hechos: columnas hecho y columnas clave, donde las columnas de hechos son aquellas donde se almacenan las medidas y las columnas clave donde se almacena la clave primaria referente a las dimensiones.

- Tablas de dimensiones: Estas tablas se sitúan alrededor de la tabla de hechos, relacionándose con ella. Contienen información dimensional que permite filtrar, organizar y alimentar la información almacenada en la tabla de hechos. Cada una de las tablas de dimensiones relacionadas con la tabla de hechos contiene una única dimensión, la cual se reparte en jerarquías compuesta por una serie de niveles, las cuales se encuentran de forma denormalizada. Al producirse una serie de hechos estos varían según la relación obtenida con el resto de tablas dimensionales que envuelven a los hechos. A partir de un hecho “Cantidad Compra” y una relación con la dimensión “Producto” se podría filtrar la cantidad por algún producto en concreto y además se podría observar de manera jerárquica a qué grupo de productos pertenece, de que compañía...etc.

Como se puede observar, esta manera de organizar la información en un data warehouse favorece a técnicas como el análisis OLAP, las consultas de información y la generación de informes.

Existen otros esquemas como el del modelo de copo de nieve, donde la principal diferencia es que las jerarquías se encuentran normalizadas, esto es, los niveles de las jerarquías en las dimensiones se encuentran en tablas separadas y se estructuran y se relacionan entre sí de forma jerarquizada.

2.3.4 Procesos ETL

Extract, Transform and Load (Extraer, Transformar y Cargar). Los procesos ETL son los encargados de extraer datos de múltiples fuentes, darles formato y presentación, convertirlos en información útil y organizada, y cargarlos y almacenarlos en un almacén de datos o data mart para su posterior análisis a través de las herramientas expuestas anteriormente.

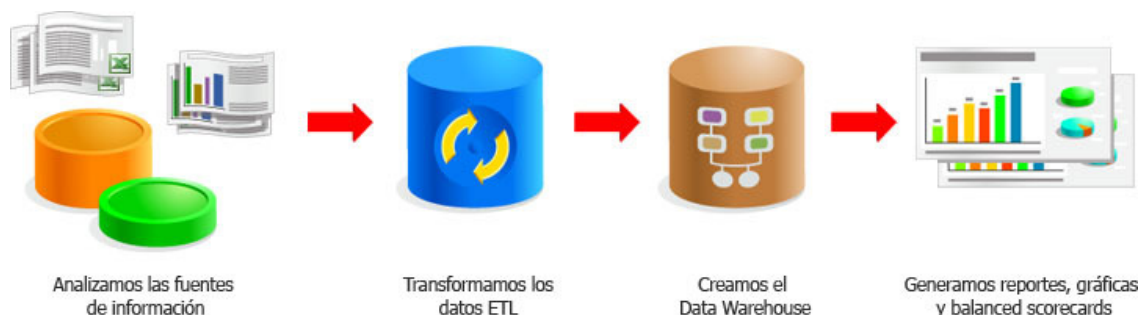


Figura 14: Etapas de una solución de BI

- 1- Extraer: Este paso se basa en extraer e integrar la información de diferentes fuentes (ERP, CRM, Excel) en el data warehouse. En este proceso de extracción se estandariza un formato para todos los datos que poblarán el data warehouse, ya que provienen de diferentes fuentes y cada una originalmente poseerá un formato propio.
- 2- Transformar: En esta fase se ponen en práctica una serie de reglas de negocio para seleccionar únicamente la información necesaria para el data warehouse. Utilizando técnicas de filtrado, manipulación de datos y cálculos evitaremos almacenar información no necesaria, redundante o errónea.
- 3- Cargar: En esta última fase los datos ya formateados, integrados y seleccionados se almacenan en el data warehouse. Cabe destacar que esta carga no siempre se realiza de la misma manera, ya que hay organizaciones que optan por borrar todo el contenido del data warehouse y cargarlo de nuevo, y otras que optan por actualizar el data warehouse únicamente con la información que ha llegado nueva.

Los procesos ETL adaptarán la información original en función del perfil del usuario final en distintos formatos de presentación, como pueden ser aplicaciones de análisis, informes, scorecards o cuadros de mando.

Un mal diseño de procesos ETL puede ocasionar graves problemas operativos a la compañía. Actualmente el procesamiento de grandes cantidades de datos está siendo mucho más ligero gracias al procesamiento en paralelo. Una potente herramienta ETL de procesamiento en paralelo, y además open source, es Kettle™ Pentaho Data Integration, la cual se utilizó a lo largo de la realización del presente proyecto.

2.4 El Sistema informacional

Un sistema informacional está formado por aquellas herramientas y elementos orientados al tratamiento de datos e información y que son capaces de acercarla al usuario mientras esté almacenada en un data warehouse. El sistema informacional viene a englobar todas las herramientas de consulta y análisis de datos anteriormente expuestas y que además forman el concepto de Business Intelligence.

En un sistema informacional puede darse el caso que encontremos estas herramientas separadas o por el contrario que interactúen entre sí, dando mayor versatilidad al usuario (con Litebi esto es posible además de otras plataformas de BI como Cognos o Business Objects). Además de esta interacción entre las herramientas, los sistemas se apoyan en herramientas extras para obtener una facilidad al usuario en el acceso y el uso compartido de éstas (seguridad, portales...).

Podemos decir que el sistema informacional es la capa de interacción entre el usuario y el data warehouse (lo que el usuario percibe del sistema es únicamente la capa de aplicación). A este tipo de sistemas también se les suele conocer como aplicaciones o plataformas de inteligencia de negocios.

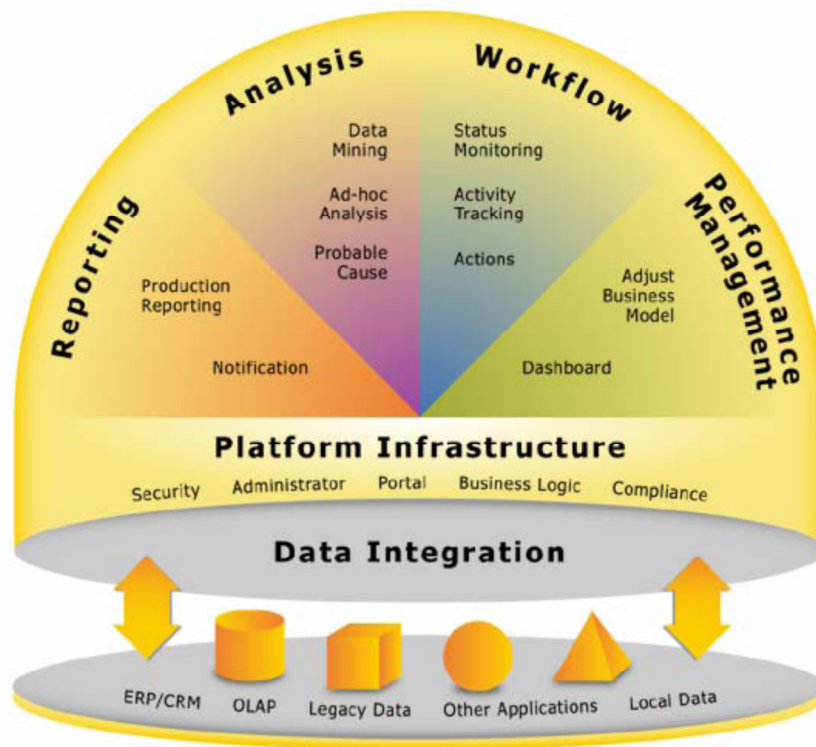


Figura 15: Estructura de una aplicación de BI según Pentaho

3 CLOUD COMPUTING

El cloud computing se basa en la teoría de ofrecer servicios informáticos a través de Internet. En los últimos años se ha utilizado la metáfora de la nube para referirse a la red de redes, por encima de otros conceptos como ASP u On Demand.

La idea principal de este concepto es evitar al cliente la necesidad de albergar en su propio entorno los sistemas de información, dejando esta tarea al proveedor de software que ofrece su uso a través de internet. Hoy en día esta idea está en plena expansión, teniendo como referentes algunas empresas como Salesforce o Netsuite.

Las ventajas de obtener servicios de Internet frente a la forma tradicional de almacenarla en nuestros equipos son claras:

- El cliente se despreocupa del despliegue inicial de cualquier sistema de información, la necesidad de adquirir hardware especializado o preocuparse del mantenimiento.
- El cliente puede acceder al servicio desde cualquier lugar, únicamente con una máquina conectada a la red.
- El proveedor tiene la posibilidad de ofrecer precios más competitivos (ya que no tiene que mantener cada implementación individual por cliente, sino una única).
- El pago del software es por uso, evitando el pago por licencia.
- La escalabilidad no tiene límites.
- La actualización del software es más fácil y más frecuente, normalmente esta fase suele ser transparente para el cliente.

Una de las pocas desventajas a considerar es la velocidad del servicio a causa del estado actual de las infraestructuras de las comunicaciones en este país y en algunos otros.

Dentro del concepto comodín del cloud computing hay consenso respecto a las tres clases fundamentales que existen:



- Software como Servicio (SaaS): Este modelo de distribución de software consiste en que una empresa proveedora ofrece el mantenimiento, soporte y operación durante el tiempo que haya contratado el cliente el servicio. La empresa proveedora alojara el sistema del cliente manteniendo la información y ofreciendo los recursos necesarios para explotarla. Este modelo se sitúa en la capa superior del cloud computing.
- Plataforma como Servicio (PaaS): En principio este modelo puede parecer la evolución del SaaS, pero este modelo ofrece al cliente los recursos necesarios para soportar el ciclo de vida completo de construcción y puesta a punto de aplicaciones y servicios web disponibles a través de la red (Se podría decir que es un entorno de desarrollo de aplicaciones alojado en la nube). No existe descarga de software y los servicios que se ofrecen suelen ser una solución integral en la web. Un ejemplo es Google App Engine, que permite ejecutar e implementar aplicaciones web sobre la estructura de Google. Esta es la capa intermedia del cloud computing.
- Infraestructura como Servicio (IaaS): Se basa en distribuir infraestructura de computación como servicio, normalmente a través de una plataforma de virtualización. Los clientes encargan al proveedor los recursos necesarios como servidores, espacio de datos o equipamiento de redes y estos servicios se le dan al usuario de manera integral a través de la web. Esta clase se sitúa en la capa inferior del cloud computing.

El esquema de cloud computing va creciendo notablemente, en un futuro cercano será una opción en la que muchas empresas apostarán por las ventajas de costo que trae consigo, adicionalmente el modelo de SaaS permitirá la clusterización de la información por industria que facilitará el compartir información relevante entre empresas del mismo sector industrial.

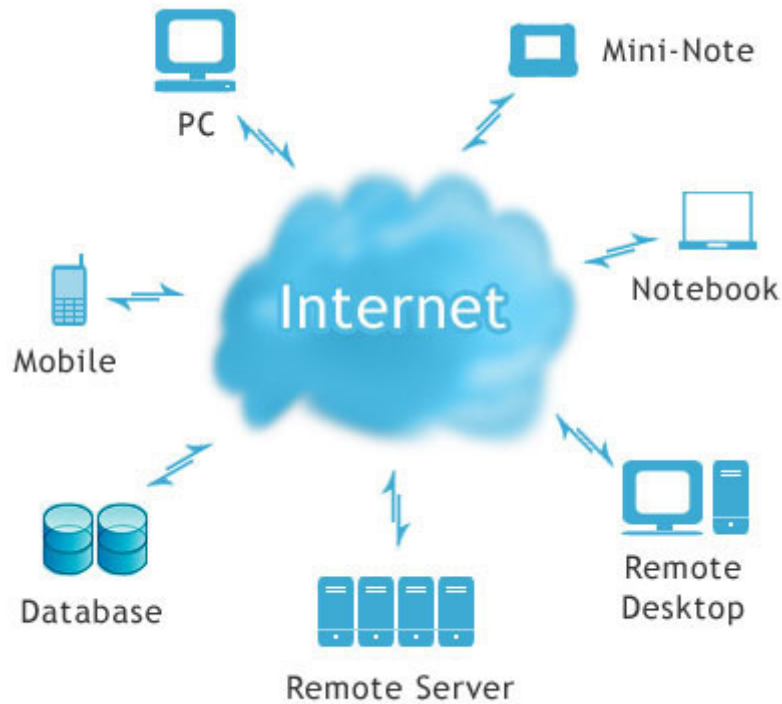


Figura 16: Esquema de computación en la nube

En la actualidad el modelo de SaaS para Business Intelligence enfrenta grandes retos como el volumen de información, el confiar la información confidencial de la empresa a un intermediario, las capacidades analíticas de las herramientas actuales, entre otros. Por estas razones es posible que el modelo de SaaS para BI evolucione a medio plazo con estrategias complementarias que permitan cubrir en mayor medida las deficiencias actuales del modelo.

4 LA EMPRESA CLIENTE: CECAV

4.1 *Introducción*



Figura 17: Logo del CECAV

CECAV es el Centro de Calidad Avícola y alimentación animal de la Comunidad Valenciana. Situado en Alquerías del Niño Perdido en Castellón, fue inaugurado en abril de 2007 por el Honorable Conseller d'Agricultura, Pesca i Alimentació D. Juan G. Cotino Ferrer. Se trata de un centro que agrupa todas las ramas existentes dentro de la cadena de producción avícola: fabricación de piensos, producción primaria y mataderos. Actualmente CECAV es un centro de referencia nacional e internacional dentro del sector agroalimentario.

4.2 *Historia*

La idea principal dentro del sector agroalimentario en la Comunidad Valenciana es la de garantizar unos productos de calidad que generen riqueza para la Comunidad. Hace unos años la avicultura valenciana comenzó a ser un modelo a seguir para otros sectores y para otras Comunidades, y por ello, se consideró como condición necesaria seguir evolucionando dentro del sector, ofreciendo más programas de control y mayor oferta de servicios, lo que conllevaba aumentar el espacio de trabajo, ya que las posibilidades de crecimiento del Laboratorio Pecuario de Valencia eran limitadas.

Apostando por este proyecto, la Generalitat Valenciana se puso manos a la obra realizando estudios de financiación. En la Asamblea de la Asociación Avícola Valenciana, celebrada en noviembre de 2001, la entonces Consellera de Agricultura, Pesca y Alimentación, Dña. Angeles Ramón Llin, hacía pública esta iniciativa.

Tras esto, se iniciaron conversaciones con otras dos Asociaciones relacionadas con el sector para integrarlas dentro del proyecto, de esta manera estarían presentes todos

los eslabones de la cadena de producción avícola: fabricación de piensos, producción en granja y mataderos avícolas.

En enero de 2002 se constituye la Federación Avícola Valenciana y de Alimentación Animal, que integraba los tres eslabones anteriormente citados. Esta Federación nació con el objetivo prioritario de promover la construcción del Centro de Calidad Avícola y Alimentación Animal de la Comunidad Valenciana (CECAV).

4.3 Estructura de la empresa

El centro se asienta en una superficie de 3.441,5 m²; 1.340 m² edificados. El terreno es una concesión del Ayuntamiento de Alquerías del Niño Perdido.

Las actividades de CECAV se desarrollan en la planta baja, donde se realizan actividades de perfil investigador a través de análisis, detecciones y diagnósticos relacionados con la sanidad y calidad avícola, y fabricación de productos para la alimentación animal.

Existen tres áreas diferenciadas dentro del centro donde se reparten una serie de departamentos y dependencias:

- En la primera área se sitúa la Zona de Espera, Recepción de Muestras, Departamento de Necropsias, Departamento de Anatomía Patológica, Limpieza de Material y Almacén de Material Limpio.
- Un área donde se sitúan los Despachos y Departamentos de Serología, Microbiología, Cámara refrigerada y PCR.
- Y por último los departamentos de Química y Virología, que se encuentra de momento inoperativo.

Además de los Departamentos previstos a tales fines, existen otras dependencias complementarias tales como, Vestuarios, Administración, Despachos, Comedor, Sala de Reuniones, Biblioteca, Almacenes y un Gran Salón de Actos dividido en dos áreas, una destinada para seminarios y otra para grandes eventos y conferencias.

4.4 Actividades de la empresa

El fin de este **Centro de Calidad Avícola y Alimentación Animal (CECAV)** es el de realizar actividades que contribuyan a la mejora del sector avícola, y de la alimentación, potenciando la adecuada aplicación de la legislación en materia sanitaria, tanto de engorde y puesta, como en mataderos avícolas y fábricas de piensos, incrementando de esta forma la calidad de la producción, fomentando además la exportación y realizando todas aquellas actividades que de una forma directa e indirecta contribuyan al progreso y a la mejora del sector.

La actividad a desarrollar por el **CECAV** es la de analizar, detectar y diagnosticar anomalías y enfermedades avícolas, de aquellas muestras que traen entidades privadas o públicas, además de formar a personal que esté trabajando o esté directamente relacionado con el sector avícola, para conseguir una mayor profesionalización y un incremento de la calidad en el sector.

5 PROYECTO BI CECAV: MAPA SANITARIO

5.1 *Introducción*

Esta sección tiene como objeto presentar la propuesta de desarrollo por parte de Litebi de una solución de Business Intelligence para CECAV (Centro de Calidad Avícola y alimentación animal de la Comunidad Valenciana). El objetivo del proyecto fue proporcionar una herramienta informática (Business Intelligence) que posibilite la **explotación on-line**, a través de Internet, de la información de **resultados analíticos** realizados por el CECAV en la Comunidad Valenciana.

Para ello se planteó el uso de la aplicación de **Business Intelligence SaaS Litebi**.

5.2 *Situación previa a la solución*

Cómo parte de su actividad, el CECAV, realiza análisis continuos del estado de determinadas enfermedades avícolas en la Comunidad Valenciana.

Anteriormente se disponía de diferentes sistemas y herramientas informáticas que dan soporte a esta actividad, sin embargo no se disponía de ninguna herramienta que permitiera **integrar la información** de las diferentes aplicaciones y ofreciera un **análisis sencillo y potente** de los resultados.

Además, anteriormente a la implantación de este sistema se estaban realizando de forma **manual y anual mapas sanitarios dinámicos** del sector avícola de la Comunidad Valenciana donde se muestran los resultados analíticos por comarca y por municipio para toda la comunidad.

5.3 *Objetivos de la solución*

Los objetivos a cumplir con el desarrollo de esta solución fueron los siguientes:

- **Integrar la información** analítica existente para su análisis.
- **Reducir el tiempo** necesario para **preparar informes** sobre los resultados de análisis (Mapas dinámicos o de otro tipo).

- Disponer de una **herramienta de exploración y análisis OLAP** de la información potente para uso interno del CECAV.
- Disponer de una herramienta de **visualización geográfica** de los resultados por comarcas y municipios (Esta herramienta se desarrolló expresamente para este proyecto en colaboración con el departamento de sistemas).

5.4 Solución propuesta

Existen en CECAV tres aplicaciones que se utilizan para diversos tipos de análisis (estos sistemas se detallarán más adelante). Algunas de estas aplicaciones cuentan con herramientas de reporting básico y exportación, pero no se disponía de la posibilidad de analizar la información de los tres orígenes sobre una misma plataforma ni se disponía de una herramienta para la realización de mapas a partir de los datos contenidos en las aplicaciones.

Por tanto, se propuso construir utilizando la aplicación de Business Intelligence Litebi una solución analítica que integrara la información de los tres sistemas existentes y posibilitara su análisis temporal, geográfico y desde múltiples puntos de vista (ej. enfermedad) de la forma más sencilla posible.

Se planteó para el presente proyecto integrar únicamente la información de resultados analíticos, quedando fuera del alcance otros conjuntos de información (gestión, contable, recursos, etc...).

El proyecto incluía por lo tanto:

- Desarrollo de los procesos de **Integración de la Información (ETL)** para las tres aplicaciones.
- Configuración del **espacio privado securizado** de Litebi para CECAV, con un cubo analítico por cada aplicación.
- Desarrollo de la **funcionalidad de análisis geográfico**.
- Formación en el manejo de **Litebi** a los usuarios finales.
- **12 meses de cuota** de suscripción de Litebi para un proyecto con el alcance mencionado.

Todo el proyecto se enfocó además teniendo en cuenta las posibles ampliaciones del sistema a otras áreas de información como por ejemplo contabilidad.

5.5 Planificación

Se estimó una duración del proyecto de desarrollo e implantación de **3 meses**, incluyendo tareas de análisis, diseño e implementación de la solución.

Las tareas asociadas al proyecto dieron comienzo a partir de la firma del contrato, además, en dicho contrato, se agregó a la valoración el **coste de la subscripción** a la plataforma Litebi **durante un año (12 meses)**, comenzando en Mayo del 2010, tarifa que incluye hardware, software y mantenimiento básico.

Se aplicó para la **subscripción mensual** la **tarifa BASIC (95 € / mes)**, en caso de ser necesario se dispone de la posibilidad de contratar espacio, usuarios o cubos adicionales.

A lo largo de las siguientes secciones se presentara de forma detallada la realización de esta solución una vez obtenidos los suficientes conocimientos relacionados con el BI y habiendo adquirido la suficiente experiencia en proyectos más pequeños como para afrontar uno de semejante magnitud.

6 ANÁLISIS DE REQUERIMIENTOS

Cuando la empresa cliente decidió formalizar su proyecto de Business Intelligence con Litebi, se inició el procedimiento rudimentario de toma de requerimientos de las áreas o departamentos implicados en el proyecto. Esta fase es la más importante dentro de la implantación de una solución de BI, ya que, si se falla en este proceso, las posteriores fases no tendrán sentido y obtendremos una solución errónea que supondrá un fracaso para la empresa que lo implanta así como para el cliente. Esta fase es la que establece el rumbo que tomará dicha solución en la empresa.

Se trata de una tarea meticulosa donde hay que actuar como consultor de estrategia de negocio además de como desarrollador, ya que hay que atender al detalle, a través de sucesivas reuniones con el cliente y/o los departamentos involucrados, sobre las necesidades del cliente, así como analizar los sistemas y métodos de obtención de información que se están utilizando previamente a la implantación de la solución. A partir de esta serie de análisis se diseñará y se implantará una solución acorde a las necesidades del cliente, que muestre toda la información necesaria de manera rápida y sencilla, y en definitiva, que la solución suponga un beneficio en tiempo y dinero para la empresa solicitante.

Para la toma de requerimientos se determinan diferentes formas de hacerlo. Aunque el objetivo parece similar, el resultado de cada escenario puede ser totalmente diferente. Por ello, nos basaremos en establecer los escenarios comunes a cualquier solución de BI para dicha toma de requerimientos, y tras esto, centrarnos en los puntos específicos del caso concreto que se plantea.

Cuando se formalizó el proyecto con CECAV, a través de la firma de un contrato, se incluyó una cláusula de confidencialidad, la cual se cita a continuación. Por esta razón, no se presentará ningún dato numérico real que comprometa a la empresa cliente o a Litebi a lo largo del presente documento. La presentación a lo largo del proyecto del resto de datos asociados tanto a las empresas como a la información parcial presentada en la plataforma (se puede considerar la plataforma como tal) queda autorizada tanto por la empresa donde se desarrolló el proyecto como por la empresa cliente (Ver Anexo 2).

COMPROMISO DE CONFIDENCIALIDAD

“**LITEBI**” se compromete a mantener la información introducida en la Plataforma Litebi reservadamente, otorgándole a la misma el carácter de estrictamente confidencial, y mantenerla protegida del acceso de terceros no autorizados.

“**LITEBI**” se compromete a no permitir la copia o reproducción total o parcial de la información que se encuentre en la Plataforma Litebi.

“**LITEBI**” se compromete a guardar estricta confidencialidad, discreción y cuidado respecto a la información que se encuentran en la Plataforma Litebi.

“**LITEBI**” se compromete a entregar una copia de seguridad de los datos que se encuentran en la Plataforma Litebi mediante pedido expreso de la empresa contratante.

“**LITEBI**” se compromete a la destrucción total de los datos introducidos en la plataforma Litebi transcurridos 3 meses de la baja del contrato de servicio “**EL CLIENTE**” cede en este acto información y secretos comerciales a “**LITEBI**”, con el fin de que ésta pueda desarrollar el proyecto encomendado.

Asimismo, “**EL CLIENTE**” autoriza a “**LITEBI**” para que pueda ceder a su vez la información y secretos facilitados a empresas con quienes tenga acuerdos comerciales, exclusivamente para la ejecución y desarrollo del proyecto contratado.



6.1 Toma de requerimientos

A la hora de iniciar la fase de la toma de requerimientos es necesario plantearse como punto de partida los objetivos a conseguir con cualquier solución de BI y tener en cuenta los diferentes fallos que se han presentado a lo largo del tiempo a la hora de llevar a cabo un proyecto de BI para no caer en ellos.

6.1.1 Requerimientos genéricos

En cualquier solución de BI existen una serie de requerimientos genéricos, donde deberíamos plantearnos esta serie de puntos:

- Proveer de un sistema intuitivo y orientado hacia un usuario final con pocos conocimientos técnicos donde éste sea capaz de generar sus propios reportes y análisis: Litebi es una plataforma que cumple estos requisitos, ya que se ha desarrollado siempre pensando en el uso por un usuario (Interfaz Windows) que sea capaz de generar sus análisis de forma sencilla con unos simples clicks de ratón.
- Tener una sola versión de la información: A través de los procesos ETL, la información queda formateada a gusto del usuario final y se presenta siempre actualizada de la misma manera en cualquier ocasión.
- Proveer información de toda la compañía en un solo sistema: En el punto siguiente se presenta el objetivo de integrar tres fuentes diferentes de información dentro de la misma plataforma.
- Que los usuarios puedan acceder a la información desde cualquier lugar y en cualquier momento: Gracias al modelo SaaS de la plataforma esto es posible accediendo a través de Internet.

Cumpliendo dichos objetivos base de cualquier solución podemos plantearnos otra serie de requerimientos a cumplir.



6.1.2 Requerimientos funcionales

Los requerimientos funcionales son aquellos que marcan la base para tomar decisiones en las posteriores etapas de algunos proyectos de BI. Es muy común que muchas empresas decidan su proyecto de BI en base a las herramientas disponibles. Algunos requerimientos funcionales que deben cumplir las plataformas de BI son los siguientes:

- El sistema deberá proveer la facilidad de drill down, slice and dice, fórmulas avanzadas...etc.: Litebi presenta estas funcionalidades a través de su interfaz y el lenguaje MDX (Ver Anexo 1).
- El sistema deberá tener los mecanismos para controlar la seguridad de los datos por departamento, área o gerencia así como una distribución organizacional jerárquica de la información: Litebi es una plataforma securizada y por medio de los roles se tiene la posibilidad de distribuir la información contenida en la plataforma de forma jerárquica según el cargo y departamento del usuario.
- El sistema proveerá un mecanismo de notificaciones y alertas, con criterios y reglas configurables: Este sistema está por desarrollar (Ver Anexo 1).
- El sistema permitirá la integración de diferentes fuentes de datos: Como se explicará más adelante este es uno de los mayores problemas que arrastra la empresa cliente y con la solución fue posible integrar tres fuentes diferentes de datos en un solo lugar.

Analizando de forma objetiva estos requerimientos podemos observar que la mayoría de plataformas de Business Intelligence en el mercado actual cumplen esta serie de objetivos, aunque esto no es suficiente para que la solución desarrollada cumpla con el objetivo principal que es la de aportar valor al negocio.

6.1.3 Requerimientos de Datos

Otra vía de toma de requerimientos es la de solicitar al cliente un listado de los datos que requieren sus reportes, o reportes que están sacando actualmente con sus herramientas para conseguir incluir todo lo necesario en la solución para que los usuarios finales puedan realizar los reportes sin problemas.

Los usuarios finales son los que conocen de primera mano su departamento y su área de trabajo, así como los reportes que se andan realizando, por lo tanto, es muy común y entendible que dichos usuarios marquen la pauta de los datos que se requieren y en el formato que los quieren.

Estos datos concretos y ejemplos de informes sobre la empresa cliente se presentan en el apartado 6.3.

Aunque este escenario es importante para que los usuarios dispongan de toda la información necesaria, por sí sola no es una estrategia de toma de requerimientos completa, debido a que el resultado final no sería el aportar un valor de negocio sino una manera de automatizar las tareas de reporting.

6.1.4 Requerimientos de Negocio

Una manera de poder dar valor al negocio es hacer requerimientos que dependan del mismo. Para ello es necesario definir requerimientos donde se establezcan los objetivos de la solución que sean medibles, controlables y confiables.

La toma de requerimientos sobre este escenario puede resultar más compleja que en los casos anteriores, ya que cada decisión debe aportar valor al negocio de manera directa o indirecta según una serie de objetivos:

- Definir la estrategia de negocio en procesos que sean medibles y controlables.
- Identificar la información necesaria para cada proceso de negocio, detallando su forma de análisis, seguimiento, técnicas de análisis, así como requerimientos tecnológicos para su cumplimiento.

- Priorizar los objetivos de la solución en función de su impacto en las metas del negocio.
- Diseñar una administración flexible que permita la adaptación de las estrategias del negocio a corto y medio plazo basado en el aprendizaje continuo de la operación y seguimiento del negocio.

Esta serie de objetivos se cumplieron finalmente estudiando la actividad de la empresa y teniendo como objetivo cubrir una serie de necesidades demandadas, dejando las cuestiones tecnológicas a un lado, ya que gracias al modelo SaaS no fue necesario considerarlas.

6.1.5 Fallos comunes en las soluciones de Business Intelligence

Además de tener en cuenta los diferentes requerimientos que se deben tomar a la hora de comenzar la implantación de una solución de inteligencia de negocios, debemos tener presente una serie de fallos en los que se suele caer a la hora de llevar a cabo un proyecto de BI. Aquí presentamos algunos de estos fallos según la experiencia en otros proyectos:

- Los data warehouses crecen en gran tamaño porque los técnicos acceden a cualquier exigencia que los usuarios demandan.
- Se confía demasiado en la gente de la empresa cliente para realizar el proyecto, cuando estos no tienen ni tiempo ni los conocimientos necesarios para apoyarnos en la toma de requisitos.
- Se suelen producir retrasos en la entrada a producción del sistema debido a una mala planificación.
- Presupuesto erróneo, el precio presupuestado es escaso en comparación con la complejidad de lo que se quiere desarrollar.
- Mala elección de software y hardware sin tener en cuenta los criterios técnicos.
- No se realizan pruebas de concepto para determinar si el proyecto es viable.

- Los datos de origen no están limpios. Esto implica procesos ETL más costoso, mayor tamaño de la base de datos y peor rendimiento.
- Mala elección de los consultores y excesiva rotación entre ellos.
- Escasa involucración de los usuarios finales, la cual les lleva a sentir cierta frustración cuando se les presentan unos resultados que no son los esperados.
- Creer que en informática todo es posible y plantearse cosas prácticamente inalcanzables.
- No alinear el proyecto dentro de una estrategia de negocio.

Existen más factores que pueden hacer fallar la implantación de un proyecto de BI, pero si caemos en estos, la oportunidad que nos brindó el cliente se volverá contra nosotros, y solo conseguiremos una mala imagen de los consultores, del producto y, en definitiva, de la empresa.

6.2 Estado previo de los sistemas de información

Una vez analizados los requerimientos básicos de cualquier solución de BI, la siguiente fase es analizar los requerimientos propios del caso que nos ocupa.

En el CECAV se utilizaban 3 aplicaciones diferentes según los tipos de análisis a realizar. Algunas de ellas, como hemos comentado, poseen herramientas integradas de reporting básico y exportación de información. Esto fue de gran ayuda a la hora de comparar resultados obtenidos con la solución y los resultados extraídos de cada una de las aplicaciones.

La intención era la de integrar la información de las 3 aplicaciones en una misma solución y proporcionar la posibilidad al usuario de realizar algo más allá de los reportes básicos, como el análisis de los datos desde un punto de vista geográfico, el cual se estaba realizando de forma manual.



Para entender a qué nos enfrentábamos, se tuvo que estudiar con detalle cada uno de estos orígenes de datos, ya que iban a ser la fuente de información desde donde partiríamos para alimentar nuestros data marts.

Como se comentó en el apartado 1.4.2 gran parte de la solución se desarrolló a través de una conexión a Escritorio Remoto directamente con los servidores centrales de CECAV (concretamente a través de un sistema operativo Microsoft® Windows® Server 2003 for Small Business Server) , ya que de esta manera el acceso a los datos de origen fue mucho más seguro y cómodo para ambas partes, no teniéndose que realizar backups de bases de datos ni utilizar bases de datos de prueba, sino que se accedían directamente a los orígenes de datos con los que la empresa trabajaba a diario. Esta forma de acceder conllevaba una ralentización en el tiempo de desarrollo del proyecto a causa de la dependencia de los accesos al servidor y el ancho de banda disponible.

6.2.1 Origen de datos 1

Se trata de una aplicación de gestión destinada para centros de investigación, desarrollo e innovación, y laboratorios de control de calidad y pruebas. Esta aplicación posee módulos para conectar equipos de laboratorio, integrarse con ERP's y para ofrecer servicios en línea.

Este sistema está basado en el concepto LIMS, una solución de tecnología de la información que permite a los usuarios modificar el formato de los datos sin tener conocimientos de programación y adaptarse de forma rápida y flexible a las nuevas necesidades laborales. Se trata de una solución modular estandarizada que permite a los clientes incorporar módulos y procedimientos nuevos para adaptar el sistema sus propias necesidades. Este concepto permite ofrecer una solución accesible, fácil de instalar con un mínimo esfuerzo necesario para validación.

Para CECAV, esta aplicación es utilizada para realizar un cierto tipo de análisis, por ejemplo para analizar muestras que puedan estar infectadas con salmonella. Para este origen de datos se conoció que la empresa realizaba estos análisis de salmonella para todas las muestras que entraban al laboratorio, y por otra parte, se realizaba otros

tipos de análisis sobre unas determinadas muestras (no todas) las cuales se encontraban repartidas en forma de placa o matriz de 8x8 (64 casillas, no todas utilizadas, cada una con una muestra a analizar pero toda la placa se contaba como un mismo registro con el mismo código). Este sistema se apoya en una base de datos Oracle.

6.2.2 Origen de datos 2

Es una aplicación orientada a los análisis de muestras de avicultura y ganadería. Se trata de una aplicación altamente editable por medio de plantillas y con gran capacidad de realizar exportaciones (Excel), informes y gráficas de los resultados analizados y con múltiples formatos de presentación.

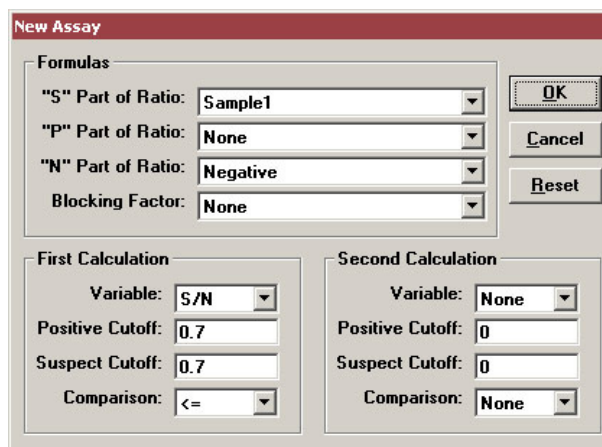


Figura 18: Aplicación de análisis de muestras

Por medio de esta aplicación, CECAV analiza una serie de muestras en busca de enfermedades como IBD o IBV.

Gracias a la capacidad de exportación a Excel de las plantillas utilizadas para los análisis, se llegó al acuerdo con CECAV de que en vez de leer directamente los resultados de la base de datos de la aplicación (un fichero Access que no era accesible), se generaran mensualmente ficheros de Excel con los resultados del mes de todos los análisis realizados, tarea bastante sencilla para el personal del laboratorio y que ahorra bastantes complicaciones a la hora de acceder a este origen de datos.

Este acuerdo se empezó a llevar a cabo a partir de finales de Enero de 2010, donde se ubicó un fichero con nombre "20100101.xls", que contenía los datos

correspondientes a ese mes, en el directorio acordado que sería el directorio base para este origen de datos. Antes de esto, se ubicaron en el mismo directorio otros ficheros con los datos correspondientes a los años 2007, 2008 y 2009, como podemos ver en la Figura 19.







| | | |
|--|----------|------------------------|
|  2007.XLS | 4.271 KB | Hoja de cálculo de ... |
|  2008.XLS | 2.230 KB | Hoja de cálculo de ... |
|  2009.1.XLS | 3.053 KB | Hoja de cálculo de ... |
|  2009.2.XLS | 3.526 KB | Hoja de cálculo de ... |
|  20100101.XLS | 559 KB | Hoja de cálculo de ... |
|  20100201.XLS | 626 KB | Hoja de cálculo de ... |

Figura 19: Exportación a Excel de datos de análisis de muestras

6.2.3 Origen de datos 3

Esta aplicación es un programa informático destinado a la captura de las densidades ópticas (DO) desde un lector ELISA (lectores para identificar enfermedades) e interpretación gráfica de los resultados de las muestras. Se puede utilizar como una herramienta para interpretar fácil y rápidamente los resultados o como una herramienta más completa en estudios epidemiológicos.

CECAV utiliza este software para otro tipo diferente de análisis, como para detectar enfermedades como TRT. Aunque esta aplicación también tiene capacidad de reporting, resulta mucho más costosa que en otros casos. Los resultados se almacenan en una base de datos en formato Access la cual resultó de fácil acceso a la hora de construir el proceso ETL asociado a este origen de datos. Aquí podemos observar la estructura de tablas que posee la base de datos asociada a la aplicación. Esto resultó de gran ayuda a la hora de identificar los datos necesarios e imprescindibles para el cliente presentes en esta fuente de datos.

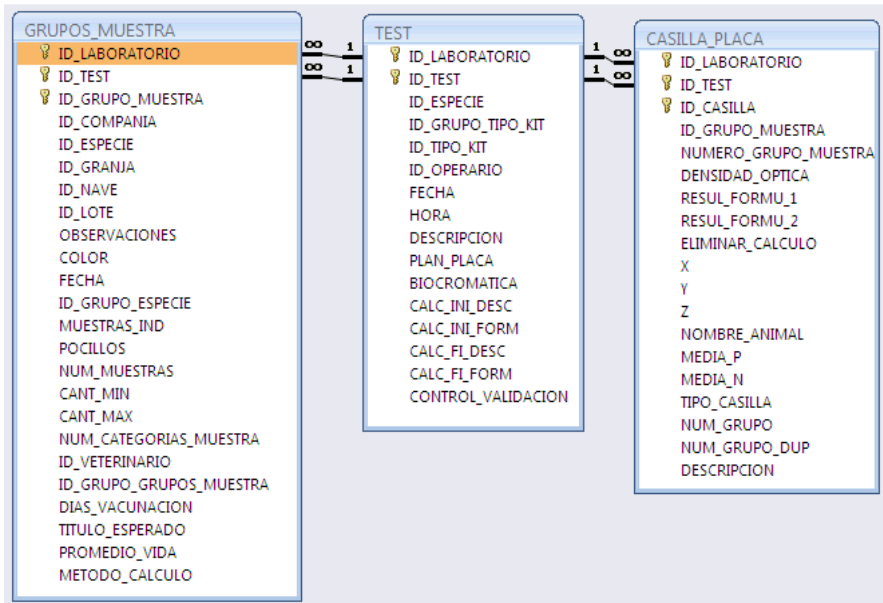


Figura 20: Base de datos Origen 3 Parte 1

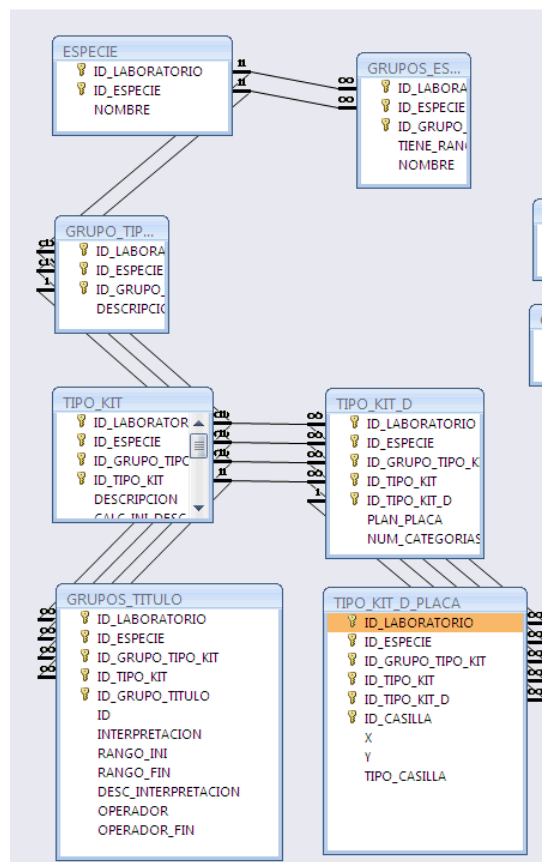


Figura 21: Base de datos Origen 3 Parte 2

6.3 Generación previa de informes y problemas

Como se ha introducido anteriormente, CECAV posee aplicaciones capaces de generar reportes asociados a los diferentes análisis realizados sobre las muestras avícolas que el centro recibe a diario.

Estos reportes se generan en las diferentes aplicaciones dependiendo sobretodo del grado de complejidad del reporte y las capacidades técnicas de los usuarios que los componen. El grado de dificultad de la realización de los reportes y las exportaciones de información depende a su vez de las características de cada software. Aquí se muestran algunos ejemplos de reportes gráficos extraídos de las diversas aplicaciones anteriormente expuestas.

Lo más necesario para CECAV era la realización de Mapas Sanitarios de diferentes enfermedades en la Comunidad Valenciana a nivel de comarca y a nivel de municipio. Estos mapas se componían de manera manual anteriormente a la implementación de la solución de BI. Queda claro que gracias a la solución compuesta, la empresa pudo distribuir el tiempo empleado en la realización de los Mapas Sanitarios en otras actividades, ahorrando de esta manera tiempo y dinero para el centro.



Figura 22: Ejemplo de Mapa Sanitario

Otra de las ventajas que Litebi proporcionaba además de la rapidez y la sencillez a la hora de generar estos informes, era la posibilidad de que estos informes fuesen dinámicos, es decir, fácilmente editables y consultables por el usuario para componer nuevos informes a partir de otros.

6.4 Necesidades de información

A la hora de diseñar la capa de metadatos que compondrá la solución, había que hacer hincapié e identificar aquellos conceptos y elementos utilizados en los informes que previamente se extraían de las diferentes aplicaciones, además de las indicaciones dadas por el centro a lo largo de sucesivas reuniones. Estos factores había que adaptarlos a la estructura de cubo que utiliza Litebi para que el usuario final fuera capaz de, por lo menos, generar los mismos informes que se estaban generando anteriormente. Aquí presentamos los diferentes conceptos de información necesarios para el cliente, su distribución y presentación en estructura de cubo se verá en el apartado de diseño de la solución de manera detallada:

- **Tiempo:** Poder observar los datos desde el punto de vista Año->Trimestre->Mes->Día. También había que contemplar la posibilidad de observar los datos desde la perspectiva de un curso por demanda de CECAV (de septiembre a septiembre). Se identificaron diferentes conceptos que hacían referencia al tiempo: fecha de entrada de la muestra, fecha de toma, fecha oficial, fecha de inicio de análisis, fecha de fin de análisis...etc.
- **Geografía:** Tratar los datos desde un punto de vista geográfico de la siguiente manera: Comunidad->Provincia->Comarca->Municipio->Explotación. Las explotaciones vienen identificadas por un código de explotación alfanumérico asignado por el REGA (registro general de explotaciones ganaderas). Estos códigos se componen de las letras ES más doce dígitos, donde ES identifica el país, en nuestro caso España; los dos primeros dígitos identifican la provincia, tres dígitos que identifican al municipio y 7 dígitos que identifican la explotación dentro del municipio. Los códigos para las provincias y los municipios son impuestos por el INE (Instituto Nacional de Estadística).

- **Enfermedad o Determinación:** Es el factor de análisis de CECAV. Siempre se tienen en cuenta diferentes tipos de resultados según la enfermedad que se esté analizando. En unos casos son resultados numéricos y en otros resultados de positivo o negativo. Nos interesa tener en cuenta cualquier información referente a las enfermedades.

- **Muestra:** Cada muestra se identifica con un código de muestra. Podemos decir que las muestras se clasifican en registros, donde un registro con un mismo código posee una serie de muestras con diferente código (Ejemplo: Si nos llega una placa, la placa completa se trata como un registro, pero cada una de sus celdas es una muestra que se analiza de manera independiente). Las muestras son el objeto de análisis y se pueden ver desde diferentes perspectivas como el producto que hace referencia la muestra (heces, alimento, animal, etc.) o el departamento donde se analiza (necropsias, química, serología, etc.). Está sujeta a diversas características que la identifican.

- **Origen:** Nos indica el origen del análisis para una muestra determinada (Mapa Sanitario, Control, Plan Anual Zoonosanitario, etc.).

- **Tipo de Ave:** Característica indispensable que necesitaba el centro para clasificar sus análisis. Existen diferentes muestras asignadas cada una a un tipo de ave en particular (ponedoras, broilers...etc.).

- **Cliente:** Otro dato a tener en cuenta era saber de qué cliente provenían cada una de las muestras a analizar. Se estableció en identificar al cliente mediante el NIF asociado.

- **Dueño Explotación:** Al centro le interesaba saber no solo de que explotación provenían las muestras, sino que persona llevaba la explotación. En muchos casos el cliente era diferente al dueño de la explotación.

- **Indicadores o métricas:** Tras analizar los diferentes factores a tener en cuenta según los puntos de vista de los diferentes informes, el análisis se centró en las cantidades numéricas que se estaban analizando. Se observó que dependiendo la



enfermedad, los resultados asociados eran diferentes (en unas enfermedades se medía el número de anticuerpos y en otros la densidad óptica) y por tanto no existía un patrón común de medida para todos los casos. Esta cuestión dificultaba el diseño y se planteó seguir adelante e identificar estos factores en concreto una vez el diseño fuese claro. Los únicos indicadores claros y comunes a todas las enfermedades eran: Número de muestras analizadas o número de registros analizados (o ambos) y número de explotaciones analizadas.

7 DISEÑO DE LA SOLUCIÓN

Una vez hecha la toma de requerimientos y teniendo identificada toda la información necesaria para el cliente y acceso a los lugares donde se encontraba, pasamos a la fase de definición y diseño de los data marts en Litebi.

En esta fase el departamento de sistemas definió el espacio de solución para CECAV dentro de la base de datos de Litebi, creando en primer lugar un usuario administrador que pudiera desarrollar la solución sobre el espacio asignado.

Cuando se define un espacio de solución, se define automáticamente la estructura del data warehouse para el cliente, que es común a todas las soluciones y se asigna un espacio de almacenamiento base, pudiéndose cambiar más adelante si el espacio de almacenamiento no es suficiente. Cabe destacar que Litebi utiliza una tecnología R-OLAP para su herramienta de análisis.

A continuación se presenta el diseño y la definición de los cubos OLAP que representan los data marts en los que se divide el data warehouse de esta solución.

7.1 *Objetivos y beneficios*

Como se ha presentado a lo largo del documento, el principal objetivo a cumplir en la arquitectura de la solución era que el CECAV dispusiera de toda la información relativa a cualquier tipo de análisis sobre las muestras de diversos tipos de aves integrada en una misma aplicación y que fuera accesible desde cualquier lugar. Esta información se utilizaría para realizar informes de manera más rápida y sencilla que con las aplicaciones que utilizaban anteriormente; y sobre todo, como objetivo principal a cumplir, la generación de mapas sanitarios de diversas enfermedades codificados por colores, algo muy importante para el centro y que resultaba demasiado costoso.

La inexistencia de un sistema que proporcionara todas estas ventajas fue el factor principal que impulsó el desarrollo de esta solución. Todos estos beneficios tendrían que tenerse en cuenta a la hora de diseñar la solución, ya que eran los principales objetivos a cumplir. Estos objetivos ya fueron marcados y expuestos brevemente en el

apartado 5 donde se explicaban las condiciones del proyecto, pero fue necesario detallarlos a la hora de establecer un diseño que los satisficiera:

- Estandarización: Al utilizarse diferentes sistemas de análisis, la presentación de informes dependía del formato de exportación de cada uno, presentando gran heterogeneidad en el tratamiento de la información. Esto dificultaba claramente el desarrollo de aplicaciones estándar para todos los departamentos de análisis de enfermedades, ya que cada uno trataba los análisis de maneras diferentes. Con la existencia de un data warehouse integrado y una sola aplicación de inteligencia de negocio para todos los departamentos se pretendía homogeneizar estos conceptos de negocio, sustituyéndolos por los definidos en el data warehouse. Las ventajas de esta estandarización son muchas: un informe desarrollado sobre una enfermedad es válido para toda la organización; los diferentes departamentos ven facilitada su comunicación entre sí al usar todos los mismos informes y conceptos de negocio; se facilita el acceso de la información a niveles más elevados dentro de la jerarquía empresarial con necesidad de mayor nivel de consolidación. La estandarización es de una importancia enorme para conseguir el objetivo: Inteligencia.
- Capacidad de análisis: La herramienta de Inteligencia de negocio Litebi posibilita múltiples métodos de análisis de información: Cubos con tecnología OLAP, Drill-Trough, Slice and Dice, y desarrollo de informes simples en un tiempo muy escaso. Todo esto sin necesidad de conocimientos informáticos por parte del usuario. La existencia de estas posibilidades fue muy útil para los usuarios operativos que tienen que tratar con información de cada día al momento y tienen necesidad de analizarla en profundidad para la toma de decisiones operativas. Estas herramientas de análisis también resultaban útiles para el caso en que una persona en un puesto más elevado dentro de la empresa quisiese “investigar” un suceso concreto más detalladamente.
- Generación de Informes: La aplicación por la que se optó posibilita la creación de informes sencillos (por cualquier usuario) o complejos (por un desarrollador especialista). Los informes pueden constar de tablas, gráficos, mapas. Pueden filtrar información dependiendo de quién sea el usuario. Pueden exportarse a otros

formatos además de tenerlo accesible desde el navegador (PDF, Excel,...etc.). Existe la posibilidad de planificar la ejecución de un informe de manera periódica y de ser enviados por mail a un usuario especificado. En definitiva la aplicación cuenta con la posibilidad de generar unos informes potentes y dinámicos que bien podían sustituir la gran mayoría de los informes usados hasta el momento en cada una de las aplicaciones. Lo que se pretendía era la generación de informes estandarizados que fuesen usados desde todos los departamentos.

- Centralización: Uno de los aspectos atractivos de la tecnología es que está totalmente basada en web. Existe un portal desde el que se realizan todos los accesos a la información, donde se encuentran los informes y cubos desde donde se puede consultar la información en el momento. Esto permite guardar todos los desarrollos de la empresa en un mismo lugar, abaratando costes de mantenimiento y de desarrollo.

La conclusión que se obtuvo a través de este análisis en detalle de los objetivos es que era fácil satisfacer estos objetivos gracias a las características de Litebi, donde el más importante y que menos dependía de la plataforma para el apartado de diseño era el de la estandarización, ya que habría que adaptar los tres sistemas a un formato estándar que los agrupara.

7.2 Primer diseño

Una vez realizados todos los pasos de creación del espacio de solución por parte del departamento de sistemas, ya se disponía de un espacio de almacenamiento donde volcar los datos necesarios de la solución. Cuando identificamos la información necesaria y los orígenes de datos, se trabajó en el desarrollo de una capa de metadatos capaz de albergar las estructuras (cubos y dimensiones) que albergaran y permitieran el análisis de la información integrada.

En la fase de diseño las tareas a realizar eran:

- Estructurar la información necesaria en un modelo estrella compuesto de tablas de hechos y tablas de dimensiones que marcarán la pauta para definir los cubos y las dimensiones.
- A partir de los modelos en estrella, crear tantos cubos como modelos haya, crear dimensiones compartidas si existe al menos una dimensión repetida en dos modelos y el resto de dimensiones no comunes definir las como embebidas.

A través de estas reglas básicas se inició la fase de diseño de los data marts de la solución, partiendo sobretodo de que los objetivos a cumplir más importantes en cuanto al almacenamiento de los datos eran los de integración y estandarización. Con esta idea se propuso como primera alternativa el siguiente diseño de modelo en estrella.

Como se observa este diseño contaba con un único cubo OLAP (ya que solamente existe un esquema de modelo en estrella) el cual debería contener datos provenientes de cualquier origen (las 3 aplicaciones mencionadas). Al cubo se le llamo "Resultados" y estaba formado por los siguientes indicadores y dimensiones:

- Indicadores o métricas: Número de registros, Títer (Número de anticuerpos en la muestra), Resultado.
- Dimensiones: Fecha oficial, Fecha toma, Fecha entrada, Fecha inicio análisis, Fecha fin análisis (Todas estas se sacan a partir de una dimensión compartida Tiempo, ya que todas hacen referencias a fechas y la dimensión Tiempo contendrá una serie de fechas tomadas desde una fecha inicial hasta una fecha final, por tanto sólo habría que establecer una asociación), Determinación (dimensión embebida) y Registro (o Muestra, también dimensión compartida).
- Cabe destacar que a partir de una dimensión compartida es posible extraer dimensiones nuevas o separadas a partir de sus diferentes jerarquías (Ver Anexo 1). Por ejemplo, como vemos la dimensión Registro está formada por una serie de niveles, y podría darse el caso que yo quisiera filtrar por tipo de ave y también por origen, y como en este caso, existir los dos niveles en diferentes jerarquías (a la

hora del análisis solo es posible seleccionar una jerarquía de una dimensión y cambiarla por otra, pero nunca dos a la vez), por tanto, a la hora de explorar si no las tratamos como dimensiones separadas no seremos capaces de hacer este filtrado y se descartarían posibilidades infinitas de análisis. Así que, la dimensión Registro se separó en otras dimensiones lo que afectaba beneficiosamente a nivel de creación de informes, pero no afectaba de manera perjudicial a nivel de diseño.

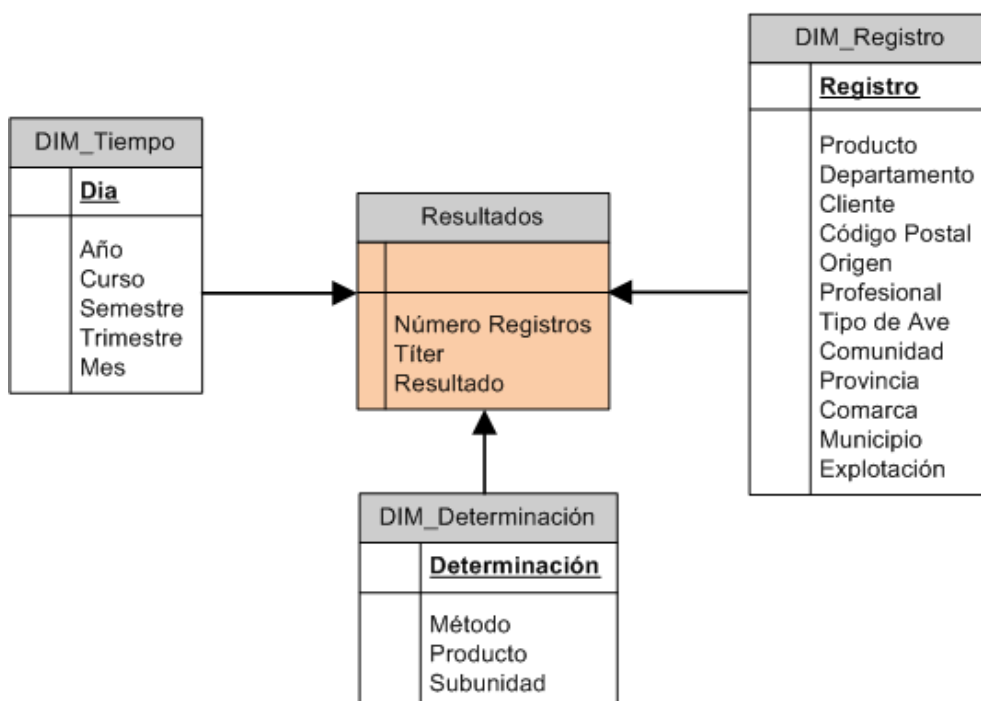


Figura 23: Modelo estrella de la primera solución

Se eligió como primera alternativa este diseño, no por ello el mejor, pero a primera vista conseguía satisfacer los objetivos principales de integración (un cubo albergaba todos los datos de cualquier análisis existente en el centro) y estandarización (los datos provenientes de diferentes fuentes tendrían el mismo formato).

Finalmente este diseño preliminar se desechó una vez nos pusimos a alimentar de datos las estructuras diseñadas (ya se sabe que cualquier proceso que conlleve una serie de fases siempre nos hace retroceder sobre nuestros errores). El caso era que al realizarse diferentes tipos de análisis en cada uno de los orígenes de datos, los resultados a medir eran diferentes factores en cada caso y esto era imposible de

representar únicamente con un indicador “Resultado”. Por otra parte, la dimensión “Determinación” que originalmente se alimentaba del origen de datos 1, no contenía algunas enfermedades presentes en los otros dos sistemas y resultaba bastante complicado unir las enfermedades de los tres lugares sin alterar de arriba abajo la solución.

7.3 Diseño definitivo

Tras detectar los diferentes errores expuestos anteriormente, se planteó la idea de definir un cubo por cada uno de los orígenes de datos y uno global (que se alimentaría de los resultados de las placas analizadas en el origen de datos 1, donde en este proceso, se analizaban una serie de enfermedades determinadas pero no sobre unas muestras o registros determinados sino sobre todos los registros que llegaban al centro, ver punto 6.2.1).

Por tanto el diseño definitivo de la solución quedó de la siguiente manera:

- Cubo Resultados Origen 1
- Cubo Resultados Origen 2
- Cubo Resultados Origen 3
- Cubo Resultados Origen 1 Global
- Además de los cubos se diseñaron las dimensiones compartidas: Tiempo, Determinación y Reg (explicadas en el diseño de los cubos que las utilizan).

De esta manera se seguían conservando los criterios de integración y estandarización, ya que los datos se presentarían en el mismo formato en todos los cubos y seguíamos teniendo presentes todos los datos referentes a los análisis en una misma aplicación. Tras una breve comunicación con el cliente, éste nos resolvió la única desventaja que creíamos que se daba en este diseño: Nunca se comparan enfermedades entre sí. Por tanto esta solución se adaptaba a las necesidades del cliente ya que podrían analizar datos de los análisis pero esta vez además distinguidos por las características que cada una de las aplicaciones ofrecía, como por ejemplo los resultados característicos de cada tipo de análisis. Con esto se consiguió una solución más acorde y más personalizada con cada uno de los sistemas. El diseño de cada uno de los data marts

se expondrá con detalle a continuación (si se desea conocer al detalle la creación de los data marts y cada uno de sus elementos, ver Anexo 1).

7.3.1 Resultados Origen 1

A partir de los informes y datos obtenidos de la base de datos de la primera aplicación, se confeccionó la siguiente estructura de esquema en estrella para elaborar posteriormente este cubo.

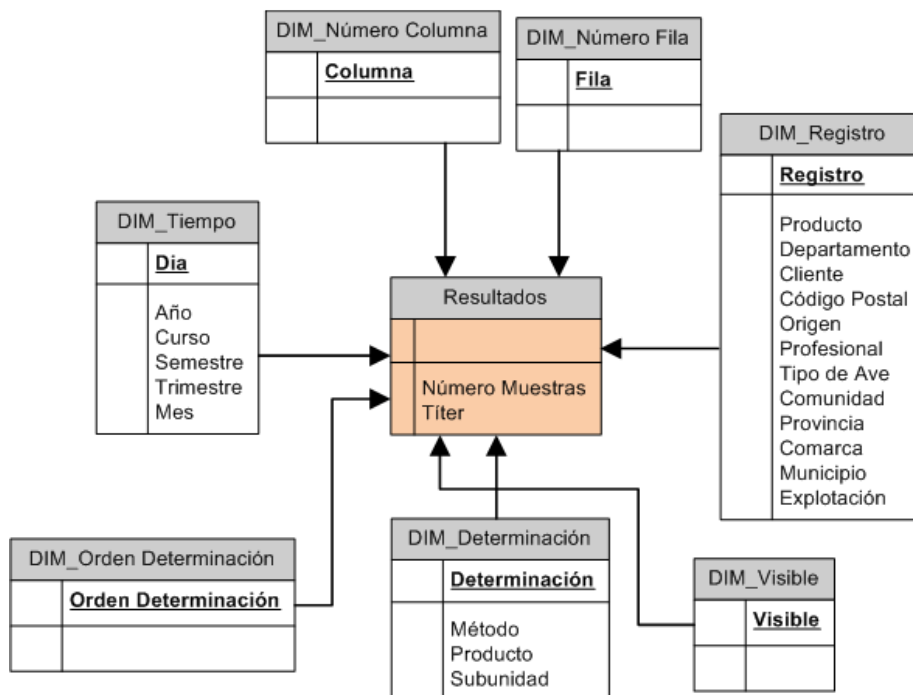


Figura 24: Modelo estrella del diseño del cubo para Resultados Origen 1

A partir de aquí este cubo contaría con los siguientes elementos:

- Indicadores: Títer y Número de muestras de la placa.
- Dimensión Número Columna (Embebida): Al tratarse de unas placas con una serie de muestras nos interesa ver resultados por columna de la placa.
- Dimensión Número Fila (Embebida): Mismo caso que el anterior pero para las filas.
- Dimensión Orden Determinación (Embebida): Número de enfermedades que se analizan para una muestra.

- Dimensión Visible (Embebida): Propiedad Visible del resultado con valor S o N (Sí o No).
- Dimensión Determinación (Compartida): Todas las enfermedades divididas en jerarquías por subunidad, producto, método o simplemente la propia enfermedad como nivel base de la jerarquía. Para este caso es compartida ya que se utilizará en el cubo de Origen 1 y Origen 1 global que comparten el mismo origen de datos.
- Dimensión Fecha Entrada (Compartida): Se alimenta de la dimensión compartida tiempo que se encuentra jerarquizada por curso, por mes, por semestre y por trimestre siendo el último nivel de todas las jerarquías el día. Esta fecha hace referencia a cuando la muestra se recibe en el centro.
- Dimensión Fecha Oficial (Compartida): Mismo caso que el anterior pero esta fecha se refiere a la fecha que oficialmente la muestra se da de alta.
- Dimensión Fecha Toma (Compartida): Fecha de toma de la muestra, sigue la forma de las anteriores fechas.
- Dimensión Fecha Ini Análisis (Compartida): Fecha de inicio del análisis. También se alimenta de la dimensión compartida Tiempo.
- Dimensión Fecha Fin Análisis (Compartida): Fecha de fin del análisis. Asociada a la dimensión compartida Tiempo.
- Dimensión Reg (Compartida): Dimensión más importante de toda la solución. Posee todos los datos asociados a una muestra o registro en concreto, los cuales son válidos para cualquier software aunque estén únicamente presentes en la base de datos del origen de datos 1. Como se había comentado en el diseño anterior esta muestra estaba jerarquizada de diversas maneras a partir de sus características. Estas jerarquías se separaron como dimensiones para poder utilizarlas en filas y en columnas durante la generación de informes (este caso es algo curioso, únicamente ocurre en las soluciones más complejas de BI). Los registros de muestras se pueden clasificar por provincia, municipio, origen, subdepartamento, producto, cliente, código postal, población, operación, profesional o por tipo de ave.
- Dimensión Tiempo Resultado (Compartida): Fecha de obtención del resultado del análisis.

Todos estos puntos de vista fueron necesarios para la perfecta realización de los informes que se necesitaban. Para saber los pasos exactos para definir las dimensiones compartidas y los cubos ver Anexo 1.

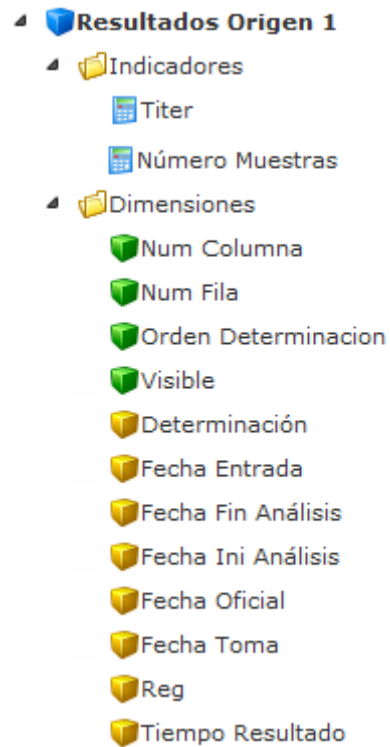


Figura 25: Estructura del cubo para Resultados Origen 1

7.3.2 Resultados Origen 2

Para el caso de la segunda aplicación el esquema en estrella que se diseñó fue el siguiente:

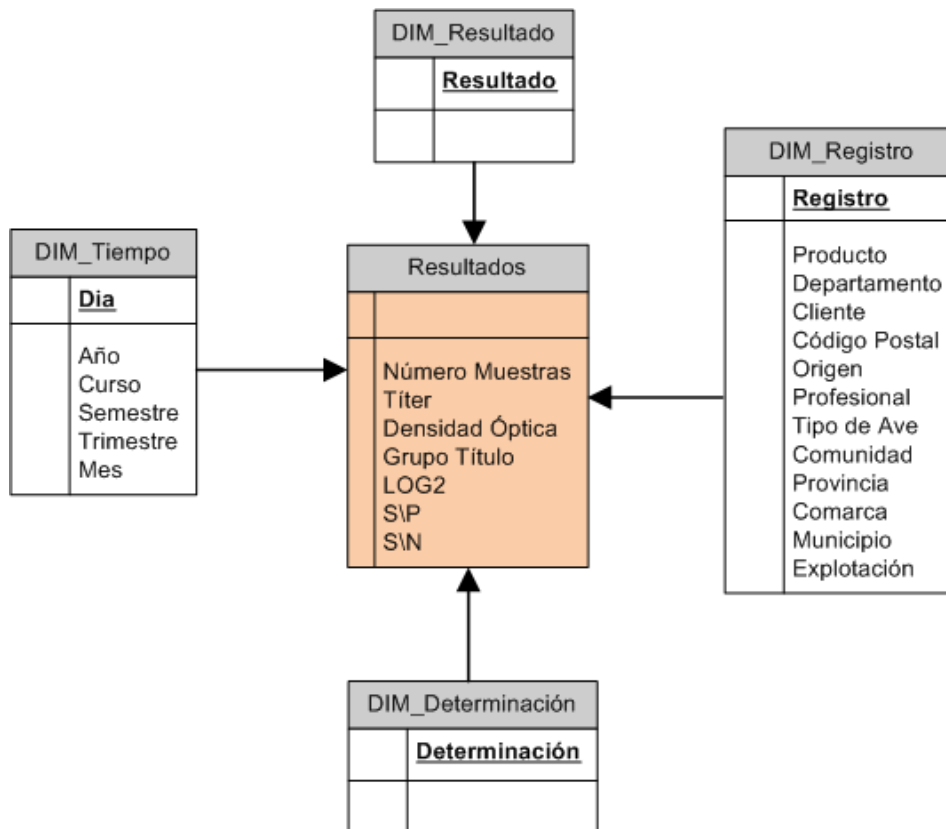


Figura 26: Modelo estrella del diseño del cubo para Resultados Origen 2

Y el cubo se compuso de los siguientes indicadores y dimensiones:

- Indicadores: Densidad óptica, Grupo de Título, Log2, Número Muestras (en este caso cada registro también está formado por x muestras), S/P,S/N, Títer. Cabe destacar que estos indicadores en este caso no tendrán valor siempre, ya que existen enfermedades que miden el Títer y otras que miden la DO.
- Dimensión Determinación (embebida): Enfermedades como en el caso anterior, con la salvedad de que las enfermedades presentes aquí son característica propia de este sistema.
- Dimensión Resultado (embebida): Resultado del análisis de la muestra con valores Positivo, Negativo o Sus (No válido).

- Dimensión Reg, Fecha Oficial, Fecha Toma, Fecha Entrada, Fecha Ini Análisis, Fecha Fin Análisis (compartidas): Las mismas dimensiones compartidas (menos Determinación) que en el caso anterior, pero aplicadas al caso del origen de datos 2.

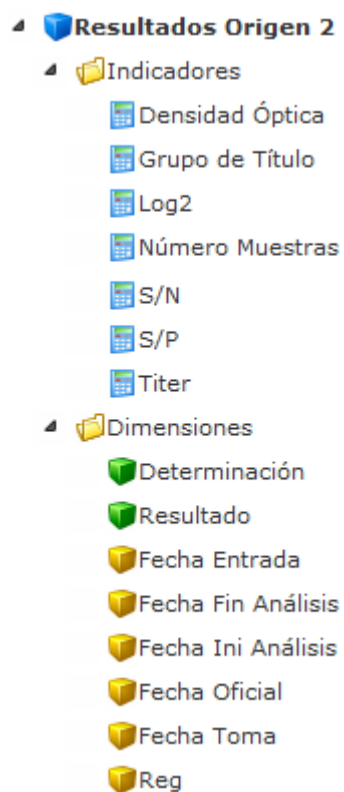


Figura 27: Estructura del cubo para Resultados Origen 2

7.3.3 Resultados Origen 3

De la misma manera que para el caso anterior, el diseño del cubo OLAP de la tercera aplicación se basó en el esquema en estrella que se muestra a continuación:

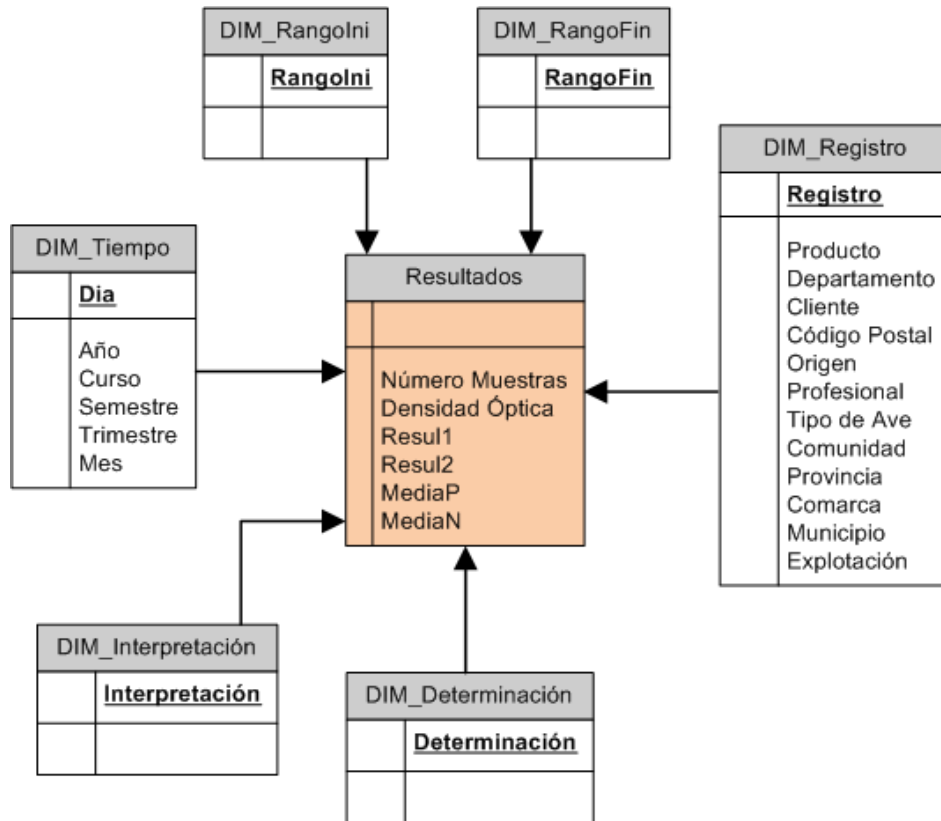


Figura 28: Modelo estrella del diseño del cubo para Resultados Origen 3

Este cubo se compuso de:

- Indicadores: Densidad óptica, MediaN, MediaP, Número de Muestras (cada registro con x muestras), Resul1 y Resul2 (Columnas con resultados para cada una de las muestras)
- Dimensión Determinación (embebida): Para este caso la dimensión de las enfermedades analizadas es embebida, como en el caso anterior, ya que las enfermedades analizadas en este sistema sólo se relacionan con esta aplicación y no están presentes en otras.
- Dimensión Interpretación (embebida): Interpretación de la muestra con valores D,P y N.

- Dimensión Rango Ini (embebida): Rango de inicio desde donde se realiza el análisis.
- Dimensión Rango Fin (embebida): Rango de fin hasta donde se realiza el análisis.
- Dimensión X (embebida): Fila de la muestra en la placa.
- Dimensión Y (embebida): Columna de la muestra en la placa.
- Dimensiones compartidas comunes al software del origen de datos 2.

Como se observa aquí, este tipo de análisis depende de unas características diferentes al del anterior software.

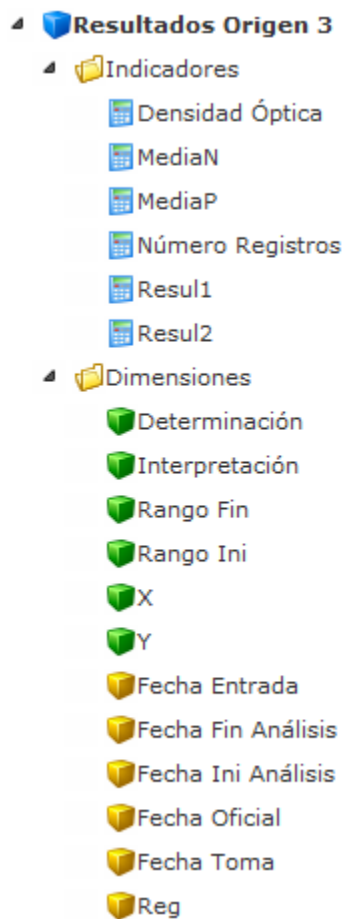


Figura 29: Estructura del cubo para Resultados Origen 3

7.3.4 Resultados Origen 1 Global

En este cubo se muestran datos de pruebas específicas realizadas sobre todas las muestras que llegan al centro, de ahí su nombre “global”. Este cubo se basa sobre el siguiente diseño de esquema en estrella:

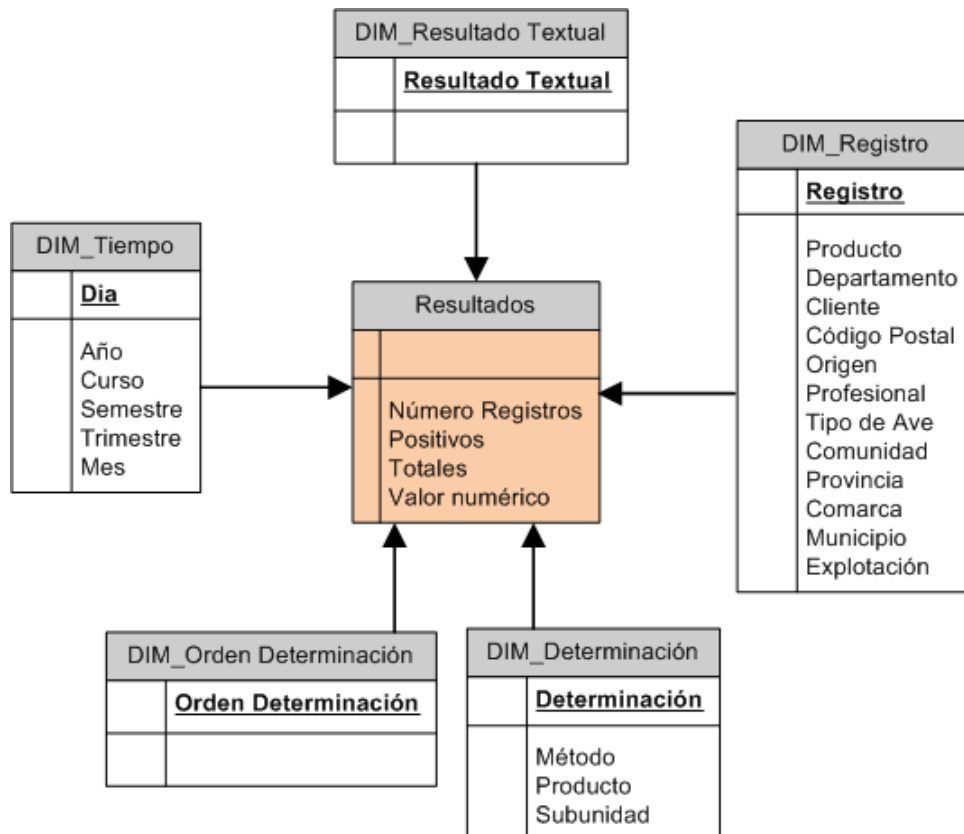


Figura 30: Modelo estrella del diseño del cubo para Resultados Origen 1 Global

El cubo se compone de los siguientes elementos:

- Indicadores: Número de registros, Positivos (cuantas muestras tienen resultado positivo), Totales, Valor Numérico.
- Dimensión Orden Determinación (embebida): Número de enfermedades analizadas sobre una muestra.
- Resultado textual: Un resultado en forma de texto que se añade cuando se realiza el análisis. Estos datos son de la forma: <10, 1'0E1...etc. son unos resultados bastante curiosos pero que son de utilidad para el centro.

- Además se tienen las mismas dimensiones compartidas que para el caso de Origen 1.

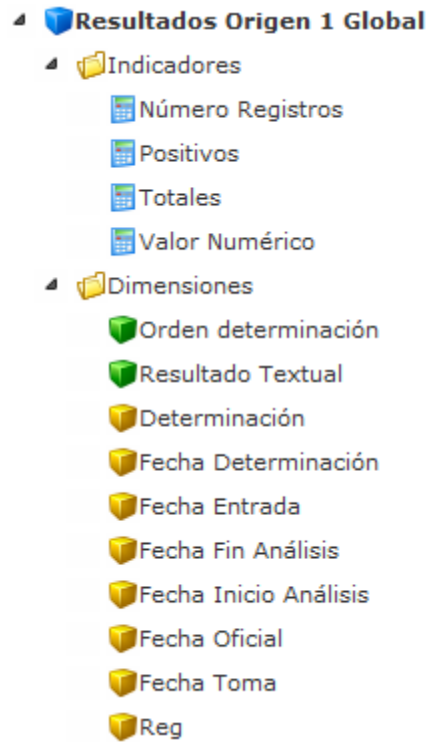


Figura 31: Estructura del cubo para Resultados Origen 1 Global

8 IMPLEMENTACIÓN DE LA SOLUCIÓN

8.1 *Pentaho™ Data Integration (Kettle)*

Kettle es un proyecto belga open source adquirido por Pentaho que incluye un conjunto de herramientas para la realización de ETL. Uno de sus objetivos es que el proyecto ETL sea fácil de generar, mantener y desplegar.

Kettle se compone de 4 herramientas:

SPOON: permite diseñar de forma gráfica transformaciones ETL.

PAN: ejecuta las transformaciones diseñadas con SPOON.

CHEF: permite, mediante una interfaz gráfica, diseñar la carga de datos incluyendo un control de estado de los trabajos.

KITCHEN: permite ejecutar los trabajos batch diseñados con Chef.

Es utilizado como base para la versión de LiteIntegrator por sus características:

- Producto **sólido** con un largo recorrido.
- Multiplataforma (java).
- **Ampliamente conocido** en el sector del Business Intelligence.
- **Adaptable** a las necesidades de Litebi (Open Source).
- **Ampliamente usado** en software empresarial.
- Basado en estándares (WSDL, SOAP, etc.).

La combinación de Kettle y el paso de Output a Litebi (Ver Anexo 1) conforman el módulo liteIntegrator de Litebi, gracias al cual, la definición y carga de procesos ETL se simplifica de gran manera respecto a otros sistemas de BI.

8.2 *Procesos ETL*

8.2.1 *ETLs Dimensiones Compartidas*

Las dimensiones compartidas son utilizadas en diferentes cubos de esta solución. Estas dimensiones contienen una serie de valores (a priori todos los posibles), los

cuales se asocian a los valores del cubo en el campo que especifiquemos como una dimensión compartida a través de su identificador. Por tanto, es lógico que para producirse esa asociación tengamos que almacenar los valores previamente en algún lugar. De esta manera conseguimos una estandarización de la información, ya que, si se utiliza una variable, por ejemplo Tiempo, en diferentes cubos, cuando los analicemos ambos presentaran esa variable en el mismo formato, independientemente de cómo estuviese representada en el origen de datos. Aquí se muestran los procesos ETL que alimentan las dimensiones compartidas.

- Dimensión Tiempo: Utilizada en todos los cubos para diferentes tipos de fechas. Su ETL consiste, lógicamente, en generar una serie de fechas (tipo Date) y otros campos que se asocien con los diferentes niveles de las jerarquías (Año, Trimestre, Mes...etc.).

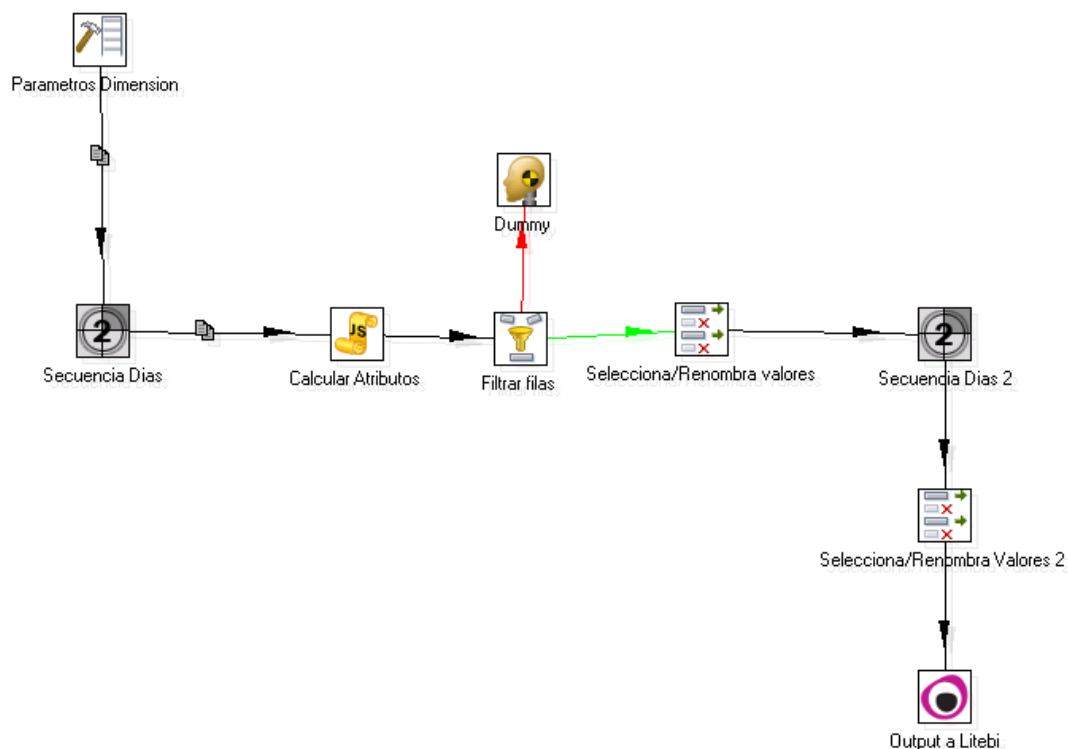


Figura 32: Proceso ETL de la Dimensión Tiempo

A continuación detallaremos la función de cada uno de los pasos que componen su ETL:



- Parámetros Dimensión: Se inician una serie de parámetros identificativos del tiempo a partir de una fecha inicial, en este caso 01-01-2005.
- Secuencia Dias: Añade una columna DaySequence que nos servirá para sumarla a la fecha inicial y así obtener fechas correlativas.
- Calcular Atributos: Proceso Java Script que calcula todos los atributos de una fecha, es decir, los niveles de las jerarquías definidos en la dimensión. Para ver el código ver Anexo 3.
- Filtrar Filas: Este paso hace que solamente pasen aquellas fechas menores que la fecha actual, por tanto, no existirán fechas seleccionables en el análisis que todavía no hayan ocurrido.
- Dummy: A este paso llegan aquellas filas que no cumplen la condición impuesta en el filtro.
- Selecciona/Renombrar Valores: Con este paso deseamos aquellos valores que ya no son importantes en el resultado final. Así rebajamos la lectura de datos en este punto.
- Secuencia Dias 2: Este paso calcula horas para cada día (para este caso no es relevante ya que no miramos a nivel de horas, pero se suele reutilizar el código de la dimensión tiempo en varias soluciones diferentes).
- Selecciona/Renombrar Valores 2: Cambiamos el nombre a los parámetros para que tengan un nombre más identificativo.
- Output a Litebi: En este paso realizamos la asociación de los campos generados para las fechas con los niveles diseñados en Litebi para esta dimensión (tanto identificadores de nivel, como sus descripciones). Al tratarse de la primera carga, elegimos como modo de carga "Standard". A continuación podemos observar la asociación de campos entre Kettle y la dimensión Tiempo.

| #. ^ | Structure | Field type | Type | Input field |
|------|-----------|----------------------|---------|---------------|
| 1 | Mes | LevelId | Integer | MES_ID |
| 2 | Trimestre | LevelId | Integer | TRIMESTRE_COD |
| 3 | Año | LevelId | Integer | PERIODO_ID |
| 4 | Día | BaseLevelId | Date | FECHA_ID |
| 5 | Mes | LevelDescription | String | MES_DES |
| 6 | Trimestre | LevelDescription | String | TRIMESTRE_DES |
| 7 | Día | BaseLevelDescription | String | DIA_DES |
| 8 | Curso | LevelId | String | curso_texto |
| 9 | Semestre | LevelId | Integer | SEMESTRE_COD |
| 10 | Semestre | LevelDescription | String | SEMESTRE_DES |

Figura 33: Carga a Litebi de la Dimensión Tiempo

- Dimensión Determinación: Esta dimensión compartida se utiliza en los cubos de Origen 1 y Origen 1 Global, ya que ambos comparten la misma fuente de origen y por tanto las enfermedades serán las mismas y estarán contenidas en el mismo lugar.

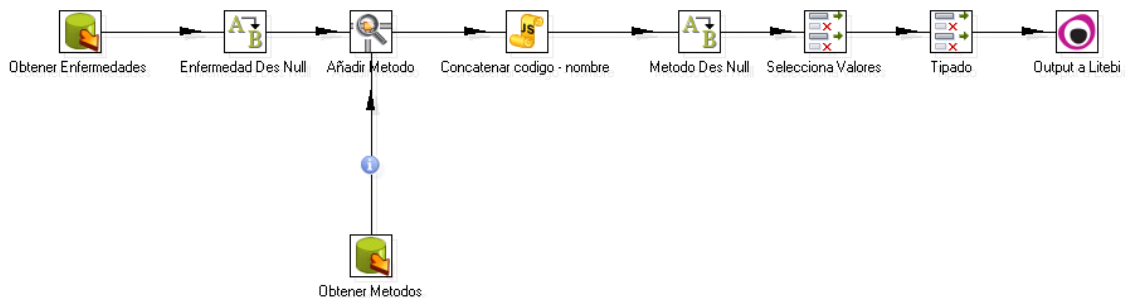


Figura 34: Proceso ETL de la Dimensión Determinación

Mostremos en detalle qué función realiza cada uno de los pasos:

- Obtener enfermedades: En este caso leemos los datos de una tabla a través de una consulta SQL para obtener las enfermedades (los datos no se generan de la nada como en el caso anterior). Para ver el código SQL asociado ver Anexo 3.
- Enfermedad Des Null: Comprobar si la enfermedad posee un nombre y si no lo tiene ponerle un valor por defecto.
- Obtener Métodos: Lectura de todos los métodos que se realizan para obtener las enfermedades.

- Añadir Método: Asociamos a cada una de las enfermedades el método de análisis que le corresponde.
- Concatenar código- nombre: Se crea una nueva variable para obtener la enfermedad como “código-nombre” (demandado este formato por el cliente). Para ver el código ir al Anexo 3.
- Método Des Null: Si no existe un método asociado a una enfermedad le ponemos como método “Desconocido”.
- Selecciona Valores: Seleccionamos los valores que finalmente nos interesan para la dimensión.
- Tipado: Le damos un tipo de datos a los valores o se lo cambiamos, le damos una longitud, un nuevo nombre...etc.
- Output a Litebi: Asociamos los valores obtenidos con los niveles de la dimensión Determinación, teniendo en cuenta que los valores a asociar deben tener el mismo tipo que se le puso a los niveles en el diseño. Aquí podemos observar cómo queda la asociación de los campos.

| #. ▲ | Structure | Field type | Type | Input field |
|------|---------------|----------------------|--------|-------------------|
| 1 | Subunidad | LevelId | String | SUB_UNIDAD |
| 2 | Producto | LevelId | String | PRODU_CODI |
| 3 | Determinación | BaseLevelId | String | DETERMINACION_COD |
| 4 | Determinación | BaseLevelDescription | String | new_NOMBRE |
| 5 | Metodo | LevelId | String | new_METODO |

Figura 35: Carga a Litebi de la Dimensión Determinación

- Dimensión Reg: Esta dimensión compartida se considera la más importante de todas, ya que deberá contener los registros de muestras y todos sus atributos o características. Esta dimensión será utilizada en todos los cubos. Se trata de un proceso ETL complejo que trataremos de exponer de la forma más sencilla posible:

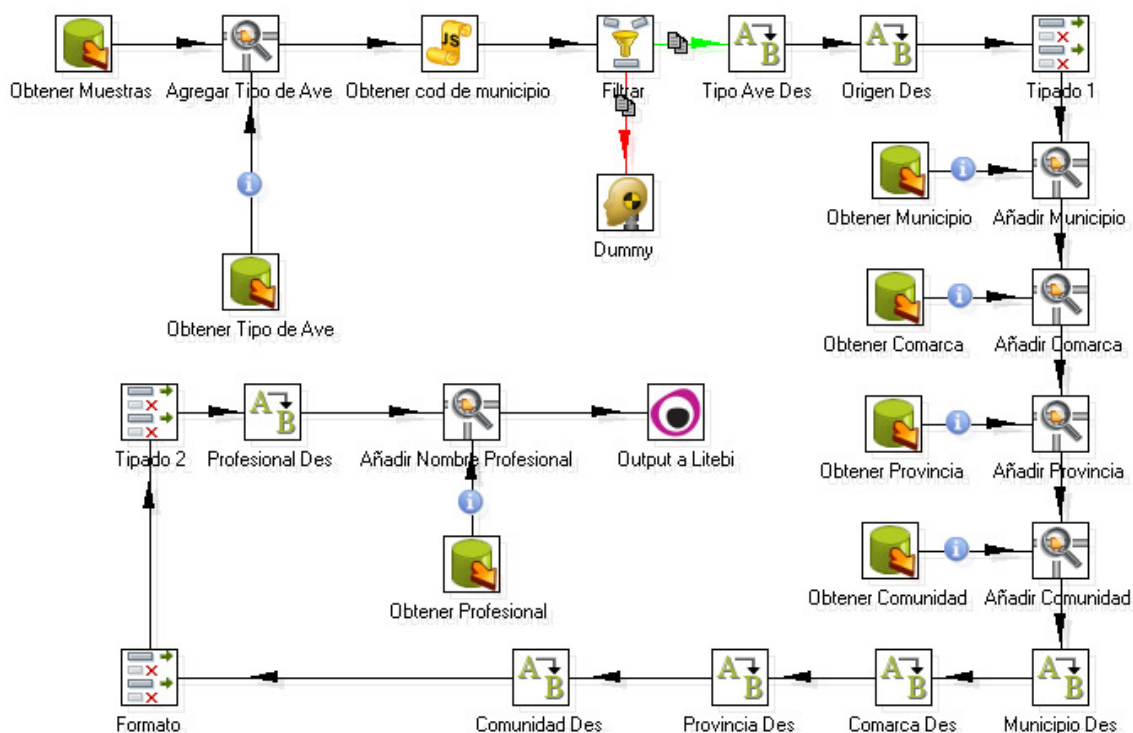


Figura 36: Proceso ETL de la Dimensión Registros

Detallemos sus pasos brevemente:

- Obtener Muestras: Lectura de la tabla “Muestras” de los datos asociados a cada una de las muestras, así como los datos del cliente que solicita el análisis y los datos de la explotación de donde proviene.
- Obtener Tipo de Ave: Lectura de todos los tipos de aves existentes.
- Agregar Tipo de Ave: Asociación del tipo de ave correspondiente a cada muestra.
- Obtener cod municipio: Código Java Script donde se extrae el código del municipio de donde proviene la muestra a partir del código de la explotación según el formato REGA. Además al código de la explotación se le concatena el nombre de su dueño. Código en Anexo 3.
- Filtrar: Dejar pasar sólo aquellos registros que cumplen con el formato REGA del código de explotación.
- Dummy: Desecho de los registros que no cumplen el formato REGA.

- Tipo Ave Des: Si el tipo de ave no se asocia con ninguno de los existentes en la base de datos lo trataremos como “Desconocido”.
- Origen Des: Mismo caso que el anterior pero para el campo Origen.
- Tipado 1: Se formaliza el tipo de dato y la longitud para cada uno de los campos seleccionados.
- Obtener Municipio: Lectura del fichero de Access que contiene la codificación geográfica a través de una conexión ODBC (se intentó primero utilizando el paso de Input Access que viene el Kettle pero ocurrían errores de lectura). En este paso obtenemos los datos necesarios para relacionar los municipios. La consulta la podemos observar en el Anexo 3.
- Obtener Comarca: Mismo caso que el anterior pero obtenemos los datos de las comarcas.
- Obtener Provincia: Obtenemos los datos de las provincias del fichero Access para los mapas.
- Obtener Comunidad: Lectura de los datos de comunidades autónomas.
- Añadir Municipio: Relacionar el código de municipio obtenido a través del código de explotación con el código de municipio que se tiene en los mapas para asignarle a cada muestra su nombre.
- Añadir Comarca: A partir del municipio obtenemos la comarca asociada a él (cabe destacar que solamente se desarrollaron los mapas a nivel de comarca para la Comunidad Valenciana).
- Añadir Provincia: Añadimos la provincia de cada muestra a partir de su municipio.
- Añadir Comunidad: De la misma manera se obtiene la comunidad asociada al municipio.
- Municipio Des, Comarca Des, Provincia Des, Comunidad Des: Si alguno de estos campos no existe lo trataremos como “Desconocido”
- Formato: Renombramos los campos con nombres más significativos.
- Tipado 2: Volvemos a dar formato únicamente a los campos que hemos ido añadiendo nuevos.

- Profesional Des: Si el campo “Profesional” es nulo lo tratamos como “Desconocido”.
- Obtener Profesional: Obtenemos los datos de los profesionales que analizan las muestras.
- Añadir Nombre Profesional: Se relaciona el código de profesional (NIF) para obtener el nombre completo de la persona.
- Output a Litebi: Igual que en los casos anteriores, relacionamos todos los campos obtenidos con los niveles de la Dimensión. Cabe apuntar que los niveles asociados a municipio, comarca, provincia y comunidad tienen lo que se llama un identificador geográfico que es necesario para la generación de los mapas (esto se definió a la hora de definir la dimensión en Litebi).

| # ▲ | Estructura | Tipo de campo | Tipo | Campos de entrada |
|-----|---------------|-------------------|---------|--------------------|
| 1 | Producto | LevelId | String | PRODUCTO_ID |
| 2 | Departamento | LevelId | String | SUBDEPARTAMENTO_ID |
| 3 | Cliente | LevelId | String | CLIENTE_ID |
| 4 | Código Postal | LevelId | Integer | CPOSTAL |
| 5 | Comarca | LevelId | String | COMARCA_ID |
| 6 | Comarca | LevelGeographicId | Integer | COMARCA_GIS |
| 7 | Comunidad | LevelId | String | COMUNIDAD_ID |
| 8 | Comunidad | LevelGeographicId | Integer | COMUNIDAD_GIS |
| 9 | Profesional | LevelId | String | PROFESIONAL |
| 10 | Origen | LevelId | String | ORIGEN |
| 11 | Explotación | LevelId | String | EXPLOTACION_ID |
| 12 | Municipio | LevelId | String | MUNICIPIO_ID |
| 13 | Municipio | LevelGeographicId | Integer | MUNICIPIO_GIS |
| 14 | Provincia | LevelId | String | PROVINCIA_ID |
| 15 | Provincia | LevelGeographicId | Integer | PROVINCIA_GIS |
| 16 | Tipo de Ave | LevelId | String | TIPO_AVE |
| 17 | Muestra | BaseLevelId | Integer | MUESTRA_ID |

Figura 37: Carga a Litebi de la Dimensión Registros

8.2.2 ETL Resultados Origen 1

En este apartado presentaremos el proceso que alimentará de datos al cubo asociado al origen de datos 1.

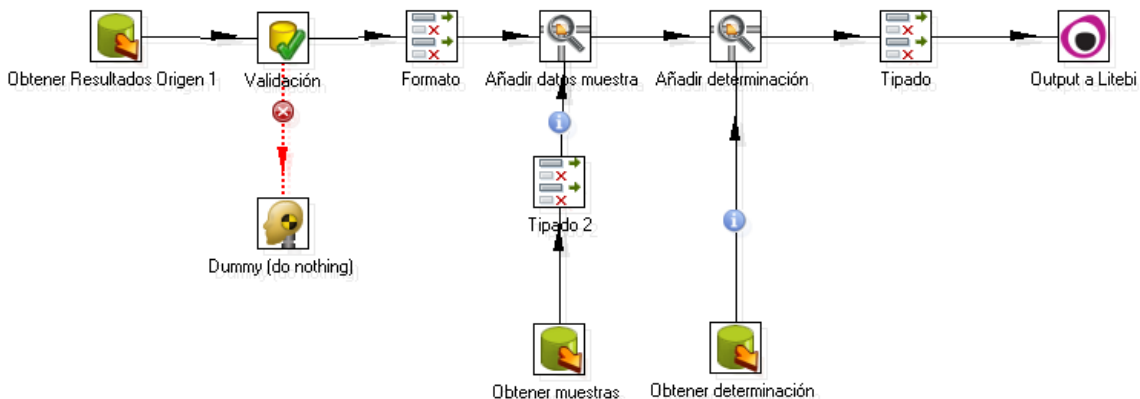


Figura 38: Proceso ETL del Cubo Resultados Origen 1

Este proceso es sencillo y se compone de los siguientes pasos:

- Obtener Resultados Origen 1: Lectura de los datos de las placas analizadas. Ver Anexo 3 para observar el código de la consulta de datos.
- Validación: Validar el campo “Valor” y dejar pasar sólo aquellos que sean numéricos (demandado por el cliente).
- Dummy: Desechar los registros que no cumplen la condición antes expuesta.
- Formato: Damos formato y longitud a los campos.
- Obtener muestras: Se obtienen datos de las muestras para relacionarlos con las dimensiones compartidas (fechas sobretodo)
- Tipado 2: Se da el formato adecuado a los datos de las muestras
- Añadir datos muestra: Se añaden datos asociados a las muestras analizadas en este origen.
- Obtener determinación: Obtenemos la enfermedad que se asocia a cada muestra analizada en todo el centro.
- Añadir determinación: Añadimos la enfermedad que se está analizando para cada muestra.
- Tipado: Se renombran los campos con nombres más identificativos.

- Output a Litebi: Como en las dimensiones, se asocian los campos entre el cubo y los registros leídos. Cabe destacar que para el caso de los cubos se asocian las métricas y los niveles de las dimensiones embebidas, y para el caso de dimensiones compartidas, únicamente se asocia su nivel base, ya que el resto de niveles ya están rellenos en el ETL de la dimensión propia. Este es el momento donde Litebi sabe por medio de esta asociación que registros de esa dimensión están presentes en el cubo.

| #. ▲ | Structure | Field type | Type | Input field |
|------|---|-----------------|---------|--------------------|
| 1 | Titer | Measure | Double | VALOR_TITER |
| 2 | Fecha Entrada | SharedDimension | Date | FECHA_ENTRADA |
| 3 | Fecha Fin Análisis | SharedDimension | Date | FECHA_FIN_ANALISIS |
| 4 | Fecha Ini Análisis | SharedDimension | Date | FECHA_INI_ANALISIS |
| 5 | Tiempo Resultado | SharedDimension | Date | FECHA_RESULTADO |
| 6 | Reg | SharedDimension | Integer | MUESTRA_ID |
| 7 | Fecha Toma | SharedDimension | Date | FECHA_TOMA |
| 8 | Fecha Oficial | SharedDimension | Date | FECHA_OFICIAL |
| 9 | Orden Determinacion - Determinacion Orden | LevelId | Integer | DET_ORDEN |
| 10 | Num Fila - Num Fila | LevelId | Integer | NUM_FILA |
| 11 | Num Columna - Num Columna | LevelId | Integer | NUM_COLUMNNA |
| 12 | Visible - Visible | LevelId | String | VISIBLE |
| 13 | Determinación | SharedDimension | String | DETER_CODI |
| 14 | Número Muestras | Measure | Integer | COUNT |

Figura 39: Carga a Litebi del Cubo Resultados Origen 1

8.2.3 ETL Resultados Origen 2

Como se había comentado en apartados anteriores, la lectura de este origen de datos se llevaría a cabo a través de diversos ficheros de Excel.

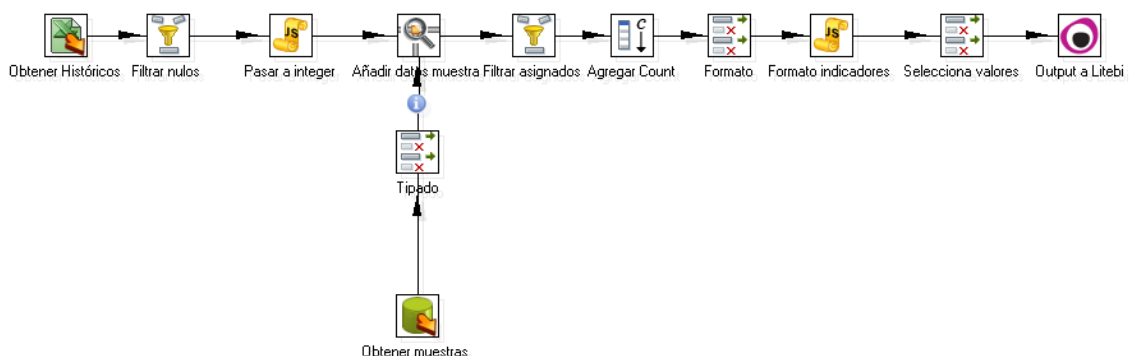


Figura 40: Proceso ETL del Cubo Resultados Origen 2

Comentamos los sencillos pasos de extracción, transformación y carga de los datos de esta fuente:

- Obtener Históricos: Lectura de los ficheros de Excel ubicados en el directorio acordado con el cliente.
- Filtrar nulos: Dejar pasar sólo aquellos registros que el código de registro de la muestra no sea nulo.
- Pasar a integer: Como todos los datos leídos del Excel tienen tipo String, necesitamos tener el código de registro de la muestra como entero para poder compararlo y obtener así datos de la muestra posteriormente. Esto se consigue con un simple `var Muestra= parseInt (REGISTRO) ;`
- Obtener muestras: Lectura de la base de datos donde están contenidas todas las muestras y sus datos.
- Tipado: Damos formato a los datos de las muestras.
- Añadir datos muestra: Asociación entre los datos de las muestras totales y las muestras analizadas en este origen de datos.
- Filtrar asignados: Dejar pasar aquellos registros que se hayan asociado con alguna muestra.
- Agregar count: Se agrega un campo contador para llevar la cuenta de todas las muestras que se analizan.
- Formato: Dar formato a los valores que se van a cargar.
- Formato indicadores: Este Java Script consiste en seleccionar aquello que nos interesa de cada una de las medidas existentes en este tipo de análisis. Ver Anexo 3.
- Selecciona valores: Nos quedamos con los valores que nos van a ser útiles y el resto se desechan.
- Output a Litebi: Como en todos los casos asociamos los campos con los indicadores y las dimensiones que componen el cubo.

| #. ^ | Structure | Field type | Type | Input field |
|------|-------------------------------|-----------------|---------|--------------------|
| 1 | Número Muestras | Measure | Integer | COUNT |
| 2 | Densidad Óptica | Measure | Double | DENS_OPTICA |
| 3 | Titer | Measure | Double | TIT |
| 4 | Grupo de Título | Measure | Double | G_TITULO |
| 5 | S/N | Measure | Double | SN |
| 6 | S/P | Measure | Double | SP |
| 7 | Resultado - Resultado | LevelId | String | RESULTADO |
| 8 | Fecha Toma | SharedDimension | Date | FECHA_TOMA |
| 9 | Fecha Fin Análisis | SharedDimension | Date | FECHA_FIN_ANALISIS |
| 10 | Fecha Ini Análisis | SharedDimension | Date | FECHA_INI_ANALISIS |
| 11 | Fecha Entrada | SharedDimension | Date | FECHA_ENTRADA |
| 12 | Reg | SharedDimension | Integer | CODI |
| 13 | Fecha Oficial | SharedDimension | Date | FECHA_OFICIAL |
| 14 | Log2 | Measure | Double | LOG |
| 15 | Determinación - Determinación | LevelId | String | ENFERMEDAD |

Figura 41: Carga a Litebi del Cubo Resultados Origen 2

8.2.4 ETL Resultados Origen 3

La fuente principal de este origen de datos es un fichero Access al que accedemos a través de una conexión ODBC, ya que se detectaron fallos a la hora de utilizar el paso de Input Access que viene en Kettle.

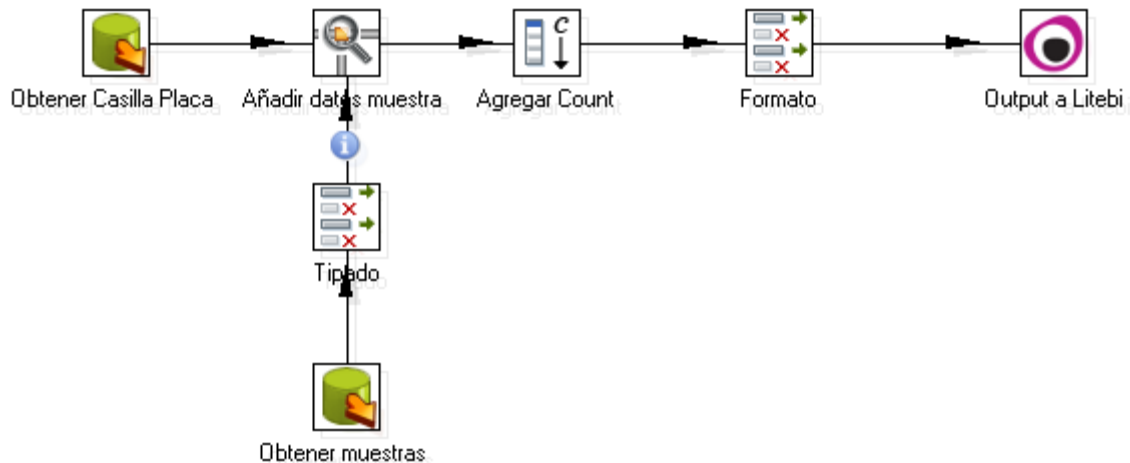


Figura 42: Proceso ETL del Cubo Resultados Origen 3

Detallemos los pasos que se observan en el diagrama:

- Obtener Casilla Placa: Gracias a los modelos de las tablas facilitados por el cliente que pudo componer una consulta SQL (ver Anexo 3) para obtener todos los datos necesarios presentes en esta aplicación.
- Obtener muestras: Como en los casos anteriores se obtienen los datos de las muestras.
- Tipado: Damos formato a los datos obtenidos.
- Añadir datos muestra: Se asocian los códigos de las muestras para añadir más datos relevantes.
- Agregar Count: Se añade un campo contador para llevar la cuenta de las muestras analizadas, algo muy importante para el cliente.
- Formato: Seleccionamos y formateamos aquellos valores que se usarán finalmente.
- Output a Litebi: Volcamos los datos necesarios en nuestra estructura de data mart definida y diseñada en Litebi.

| #. ▲ | Structure | Field type | Type | Input field |
|------|---------------------------------|-----------------|---------|---------------------------|
| 1 | MediaP | Measure | Double | MEDIA_P |
| 2 | Resul2 | Measure | Double | RESULTADOS_INTERPRETACION |
| 3 | MediaN | Measure | Double | MEDIA_N |
| 4 | Resul1 | Measure | Double | RESULTADOS_INTERPRETACION |
| 5 | Densidad Óptica | Measure | Double | DENSIDAD_OPTICA |
| 6 | Número Registros | Measure | Integer | COUNT |
| 7 | Reg | SharedDimension | Integer | MUESTRA_ID |
| 8 | Fecha Fin Análisis | SharedDimension | Date | FECHA_FIN_ANALISIS |
| 9 | Fecha Ini Análisis | SharedDimension | Date | FECHA_INI_ANALISIS |
| 10 | Fecha Oficial | SharedDimension | Date | FECHA_OFICIAL |
| 11 | Fecha Entrada | SharedDimension | Date | FECHA_ENTRADA |
| 12 | Fecha Toma | SharedDimension | Date | FECHA_TOMA |
| 13 | Interpretación - Interpretación | LevelId | String | INTERPRETACION |
| 14 | Rango Ini - Rango Ini | LevelId | Integer | RANGO_INI |
| 15 | Rango Fin - Rango Fin | LevelId | Integer | RANGO_FIN |
| 16 | Determinación - Determinación | LevelId | String | ENFERMEDAD |
| 17 | Y - Y | LevelId | Integer | Y |
| 18 | X - X | LevelId | Integer | X |

Figura 43: Carga a Litebi del Cubo Resultados Origen 3

8.2.5 ETL Resultados Origen 1 Global

Este cubo se debe alimentar de datos contenidos en el origen 1 pero que hacen referencia a todas las muestras que pasan por el centro y varias determinaciones que se analizan para cada una en esta aplicación.

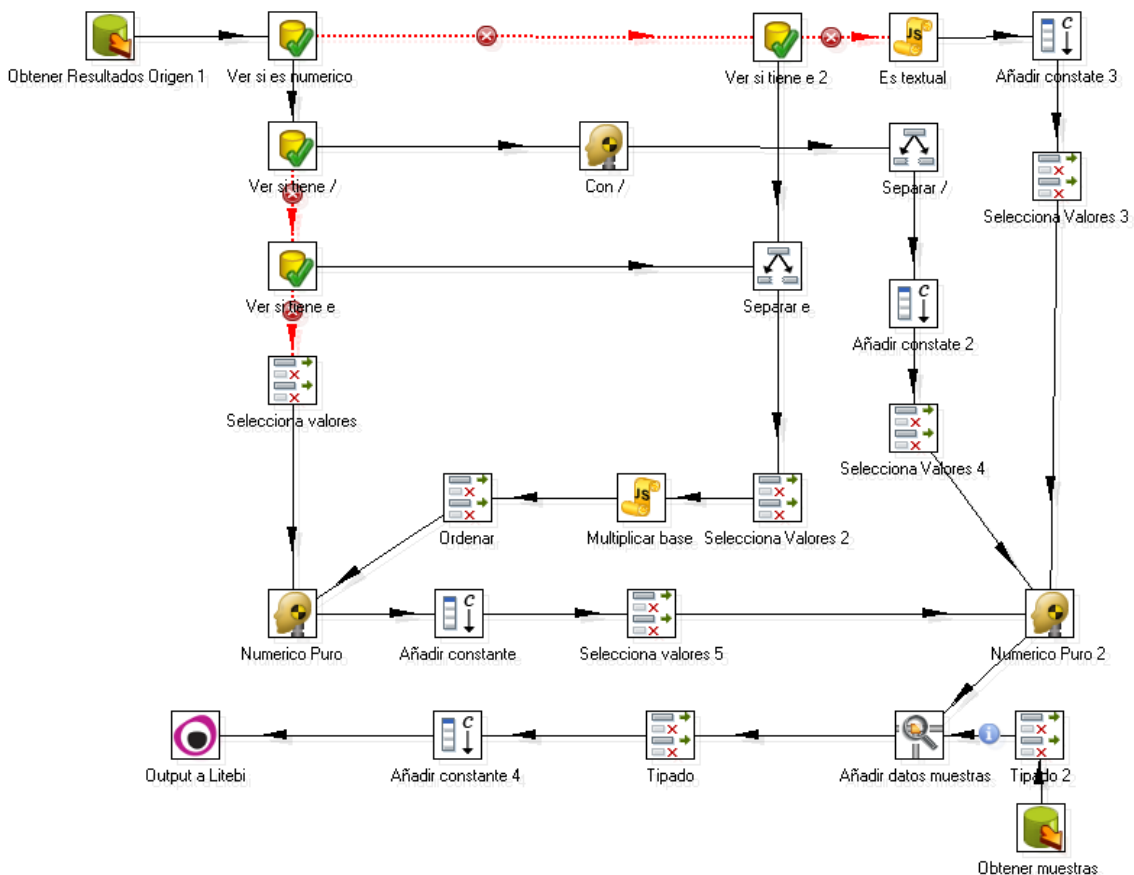


Figura 44: Proceso ETL del Cubo Resultados Origen 1 Global

Vamos a explicar brevemente cada uno de los pasos de este complejo diagrama:

- Obtener Resultados Origen 1: Consulta SQL para obtener los registros de las muestras con sus determinaciones analizadas así como los resultados obtenidos. Ver el código en Anexo 3.
- Ver si es numérico: Filtrar según el resultado obtenido es un número o no.
- Ver si tiene /: Observar si el resultado es de la forma xx/xx.



- Ver si tiene e, Ver si tiene e 2: Comprobar si el resultado está en formato científico (Ej. 6e2).
- Con /: Paso intermedio.
- Separar e: Separar el número en base y exponente.
- Separar /: Por delante de la barra son los positivos, por detrás los totales.
- Multiplicar base: Obtener el resultado numérico en coma flotante. Ver código en Anexo 3.
- Numérico puro, Numérico puro 2: Juntamos todos los registros que tienen un resultado transformado ya como un número.
- Es textual: Guardar el resultado si es un texto en una variable. `var RESULTADO_TEXTUAL = RESUL_VALOR.getString();`
- Selecciona valores, Selecciona valores 2,..., Selecciona valores 5, Ordenar: Renombrar, seleccionar, dar formato y seleccionar el tipo de los campos que sean necesarios.
- Añadir constante, Añadir constante 2,..., Añadir constante 4: Añadir constantes como si los resultados son positivos o totales. Esto se hace porque hora de unir todas las ramas de flujo de información en un mismo punto estamos obligados a que por todas las ramas vengan los mismos campos. En el paso 4 es donde se añade la variable contador para llevar el número de muestras.
- Obtener muestras: Leer los datos asociados a las muestras.
- Añadir datos muestras: Completar los registros añadiendo más datos relacionados con las muestras.
- Tipado: Seleccionar los valores necesarios para la carga.
- Output a Litebi: Asociamos los datos con las dimensiones y los indicadores del cubo.

| #. ▲ | Structure | Field type | Type | Input field |
|------|---|-----------------|---------|---------------------|
| 1 | Número Registros | Measure | Integer | COUNT |
| 2 | Positivos | Measure | Integer | POSITIVOS |
| 3 | Totales | Measure | Integer | TOTALES |
| 4 | Valor Numérico | Measure | Double | VALOR_NUMERICO |
| 5 | Fecha Inicio Análisis | SharedDimension | Date | FECHA_INI_ANALISIS |
| 6 | Fecha Entrada | SharedDimension | Date | FECHA_ENTRADA |
| 7 | Fecha Fin Análisis | SharedDimension | Date | FECHA_FIN_ANALISIS |
| 8 | Fecha Toma | SharedDimension | Date | FECHA_TOMA |
| 9 | Fecha Determinación | SharedDimension | Date | FECHA_DETERMINACION |
| 10 | Determinación | SharedDimension | String | DETERMINACION_ID |
| 11 | Reg | SharedDimension | Integer | MUESTRA_ID |
| 12 | Orden determinación - Orden determinación | LevelId | Integer | ORDEN_DETERMINACION |
| 13 | Resultado Textual - Resultado | LevelId | String | RESULTADO_TEXTUAL |
| 14 | Fecha Oficial | SharedDimension | Date | FECHA_OFICIAL |

Figura 45: Carga a Litebi del Cubo Resultados Origen 1 Global

8.3 Codificación geográfica

Para la generación de mapas con Litebi se generó una base de datos en Access siguiendo las indicaciones del departamento de sistemas.

Como podemos ver en el Anexo 1, Litebi se comunica con una aplicación SIG open source llamada Geoserver, la cual proporciona los servicios de generación de mapas a Litebi.

Se acordó llevar una codificación para los mapas basada en los códigos impuestos para cada municipio, provincia y comunidad autónoma por el Instituto Nacional de Estadística (INE). De esta manera la composición de la base de datos que daría soporte a los mapas sería mucho más sencilla de construir a partir de ficheros Excel obtenidos desde la web del INE (<http://www.ine.es/>).

Vamos a describir los diferentes elementos de la base de datos para que quede claro como Litebi utiliza estos ficheros para comunicarse con Geoserver. Existen tres tablas las cuales se describirán a continuación:

- Tabla Layers: Posee las diferentes capas en las que se puede dividir un mapa geográfico así como el código que identifica a cada capa.

| LayerId | Nombre | Identificador WMS |
|---------|-----------------|-------------------|
| 1 | Comunidades | Comunidades |
| 2 | Provincias de E | Provincias |
| 3 | Municipios de | Municipios |
| 4 | Mundo | Mundo |
| 5 | Comarcas | Comarcas |

Figura 46: Tabla Layers de la base de datos de Litebi-Geoserver

- Tabla Polygons: Tabla maestra donde se recogen diferentes lugares geográficos distinguidos según la capa. Están identificados por un Litebiid que es el campo que se asocia con el nivel geográfico en Litebi y poseen un campo Polygonid que es aquel que identifica el polígono que Geoserver deberá mostrar por pantalla. El campo LitebiParentId identifica al padre inmediatamente anterior del elemento, por ejemplo si estuviésemos en una provincia el LitebiParentId identificaría la comunidad asociada a esa provincia.

| Litebiid | LayerId | LitebiParentId | Nombre | PolygonId |
|----------|---------|----------------|----------------------------|-----------|
| 1 | 1 | | Principado de Asturias | 63 |
| 2 | 1 | | Islas Baleares | 64 |
| 3 | 1 | | Canarias | 65 |
| 4 | 1 | | Cantabria | 66 |
| 5 | 1 | | Castilla y León | 67 |
| 6 | 1 | | Castilla - La Mancha | 68 |
| 7 | 1 | | Cataluña | 69 |
| 8 | 1 | | Extremadura | 70 |
| 9 | 1 | | Galicia | 71 |
| 10 | 1 | | Comunidad de Madrid | 72 |
| 11 | 1 | | Región de Murcia | 73 |
| 12 | 1 | | Comunidad Foral de Navarra | 74 |
| 13 | 1 | | País Vasco | 75 |
| 14 | 1 | | La Rioja | 76 |
| 15 | 1 | | Comunidad Valenciana | 77 |
| 16 | 1 | | Ciudad de Ceuta | 78 |
| 17 | 1 | | Ciudad de Melilla | 79 |
| 18 | 1 | | Andalucía | 61 |

Figura 47: Tabla Polygons de la base de datos de Litebi-Geoserver

- Tabla Padre_Hijo: Esta tabla se realizó para sacar con una simple consulta todas las capas padre que posee un elemento concreto. Por ejemplo buscando un municipio en concreto puedo obtener cuál es su comarca, su provincia y su comunidad en una misma consulta utilizando esta tabla y la de Polygons.

| LitebildPadre | ↕ | LitebildHijo | ↕ |
|---------------|---|--------------|------|
| | 1 | | 66 |
| | 1 | | 768 |
| | 1 | | 827 |
| | 1 | | 838 |
| | 1 | | 851 |
| | 1 | | 893 |
| | 1 | | 895 |
| | 1 | | 897 |
| | 1 | | 898 |
| | 1 | | 903 |
| | 1 | | 906 |
| | 1 | | 1834 |
| | 1 | | 1906 |
| | 1 | | 1947 |
| | 1 | | 1967 |

Figura 48: Tabla Padre_Hijo de la base de datos de Litebi-Geoserver

Por tanto queda claro que para alimentar de datos los niveles geográficos de las dimensiones y generar los mapas sólo basta con realizar la asociación a estas tablas, con los datos que estemos trabajando, a través de unas simples consultas donde obtengamos el Litebild, que se comunicará con Geoserver haciendo la correspondencia internamente con el PolygonId asociado y mostrando el polígono por pantalla de la zona deseada.

8.4 Procesos ETL Incrementales

Una vez construidos todos los procesos de extracción, transformación y carga de información sólo queda ejecutarlos para que tengamos los cubos y las dimensiones cargados con los datos que se poseen actualmente en el centro. Para ello cabe destacar que, al ser la primera vez que se cargan datos, es recomendable seleccionar

en todos los pasos de “Output a Litebi” la opción de carga “Standard”. La ejecución de cada uno de los procesos se realiza pulsando sobre el botón “Play”, situado en la parte superior de la ventana de diseño de Kettle. La carga se debe realizar en este orden obligatoriamente, primero cargaremos las dimensiones y luego los cubos, ya que estos últimos dependen de las dimensiones.

Por fin tenemos nuestras estructuras de datos rellenas con información. Este proceso de carga de datos suele durar bastante tiempo, y de momento, sólo se había conseguido cargar datos una vez, cosa que no es suficiente. ¿Cómo podríamos mantener los datos actualizados constantemente sin necesidad de recargar todos los datos cada vez? Esta pregunta es bien sencilla de responder, utilizando cargas incrementales. Las cargas incrementales vienen a ser pequeñas modificaciones sobre los procesos ETL que únicamente permitan cargar datos de los últimos 15 días, del último mes, del último año...etc. De esta manera con la primera carga poseemos los datos que ya existen en la actualidad y con las cargas incrementales podríamos actualizar los datos añadiendo aquellos que día a día se vayan generando, en cuestión de pocos minutos.

Explicaremos brevemente las pequeñas modificaciones que se realizaron en cada uno de los procesos ETL para conseguir las cargas incrementales que mejoraron el rendimiento de la actualización de datos.

- En las dimensiones compartidas “Determinación” y “Tiempo” se optó por dejarlas como cargas estándar, ya que se realizaban en un corto plazo de tiempo y se observaron una serie de problemas en los datos si estas dimensiones se cargaban de manera incremental.
- Dimensión Reg: En el paso de “Obtener Muestras” se añadió una condición en la consulta SQL que limitará los datos obtenidos a los del último mes según la fecha oficial. Esta expresión era: `WHERE add_months(SYSDATE, -1) <= FECHA_OFICIAL.`
- Cubo Resultados Origen 1: En el paso de “Obtener Resultados Origen 1” se añadió una expresión a la consulta SQL para obtener la fecha de hace un mes y se le llamo como HOY (`add_months(SYSDATE, -1) as HOY`). Como todavía no se tenía

ninguna fecha con la que comparar en el primer paso, tuvimos que realizar este filtrado de información justo un paso antes de cargar los datos en Litebi por medio de un paso de filtrado y una condición `FECHA_OFICIAL => HOY`.

- Cubo Resultados Origen 2: Este caso fue el más complicado de tratar. A partir del nombre que tenía cada hoja de Excel, según el formato indicado al cliente para cada archivo, se obtuvo un nuevo campo con el nombre del fichero o de la hoja (era el mismo en ambos casos) llamado `FECHA_HOJA`. A partir de este campo se generó un paso de Java Script donde obtener campos para el año y el mes actual, y el año y el mes del fichero Excel. Una vez obtenidos estos campos, se realizaba un paso de filtrado dejando pasar aquellos ficheros de Excel que fuesen como mucho de hace un mes. Podemos observar el código y las condiciones en el Anexo 3.
- Cubo Resultados Origen 3: Se utilizó el mismo recurso que para el caso del Cubo de Resultados Origen 1.
- Cubo Resultados Origen 1 Global: Se utilizó la misma condición que para la dimensión Reg pero en el paso “Obtener Resultados Origen 1” `WHERE (add_months(SYSDATE, -1) <= FECHA)`.

Gracias a esto, las cargas de datos se realizaban en tiempos mucho menores, ya que únicamente se cargaban los datos del último mes en todos los cubos. El último paso a tener en cuenta era que para todos los pasos de “Output a Litebi” esta vez la opción de carga a escoger debería ser “Actualizar datos”, de esta manera los datos antiguos se mantienen en los data marts, los que ya estaban se sobrescriben y los nuevos se añaden.

8.5 Cargas periódicas

Al realizarse los procesos de transformación de datos de manera incremental se solventó el problema del coste computacional que suponía trabajar con una gran cantidad de datos día tras día para mantenerlos actualizados. El próximo objetivo a conseguir era que la actualización se realizara de manera automática sin necesidad de

acción humana y sobretodo que se realizara fuera de horas de trabajo para no sobrecargar el sistema.

Para conseguir este objetivo se diseñaron los Jobs, procesos que automatizaban las cargas periódicas de datos y que ejecutaban de manera secuencial cada una de las transformaciones de datos anteriormente diseñadas. Este paso se compone de tres jobs: uno para cargar todas las dimensiones compartidas, otro para cargar los cubos y otro que ejecutará a estos dos de manera secuencial, es decir, un job general. A continuación se muestra el diseño de estos tres procesos:



Figura 49: Job de carga de dimensiones compartidas

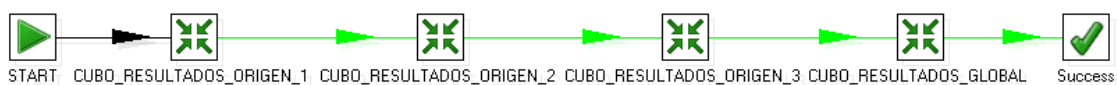


Figura 50: Job de carga de cubos OLAP

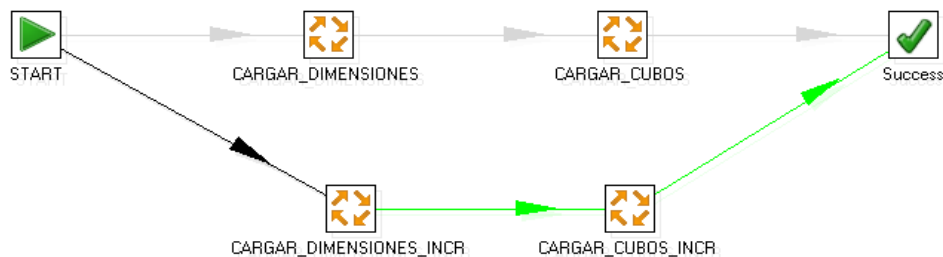


Figura 51: Job de carga de información general

Como se puede observar en los diagramas, cada uno se comunica y ejecuta los procesos ETL que se han diseñado (pasos con un aspa verde) ya sean cubos o dimensiones. El último diagrama es el job general, el cual se comunica con los dos anteriores (pasos en naranja llaman a otros jobs). De esta manera tenemos comunicación con todos los procesos que engloban la solución en un solo fichero o diagrama de flujo.

Como se puede observar en el diagrama del job general, existen dos líneas de flujo de información, una activa y otra inactiva. La inactiva se refiere a las cargas estándar de datos (cargados los datos una vez, se eligió la opción de carga “Borrar datos primero” para estos casos en los “Output a Litebi”) la cual se dejó como recurso de emergencia por si en las cargas incrementales ocurría algún error. La línea de flujo activa hace referencia a las cargas incrementales realizadas en el apartado anterior y será la línea de flujo principal para cada carga diaria que se ejecute.

Estos procesos se pueden ejecutar de manera manual o de manera automática preferiblemente, utilizando la herramienta KITCHEN de Kettle en un pequeño script y ejecutando este script como tarea programada del sistema con la periodicidad que nosotros le asignemos.

Este fichero llamado li_run.bat se componía de la siguiente línea de script:

```
kitchen /file:/ETL/GENERAL.kjb /level:basic > li_run.log
```

Con este simple script se conseguía ejecutar el job general sin necesidad de abrir Kettle y además podíamos observar los fallos que se produjeran durante la carga de datos en un archivo log donde aparece toda la información de la ejecución del proceso.

Una vez realizado este script solamente nos quedaba darle una periodicidad de ejecución a través de una sencilla tarea programada de Windows. Para realizar una tarea programada vamos a Inicio>Programas>Accesorios>Herramientas del Sistema>Tareas Programadas y agregamos una tarea nueva diciéndole el archivo a ejecutar, la hora y los días que queremos que se ejecute. En este caso se definió a las 2:00 diariamente para tener así los datos actualizados de manera que se recargaran cuando nadie estuviese utilizando el sistema.

| Nombre | Programa | Hora de próxi... | Hora de últim... |
|-----------------------------|-------------------------|------------------|------------------|
| Agregar tarea programada | | | |
| export | A las 23:00 Diariam... | 23:00:00 15/... | 23:00:00 14/... |
| li_run | A las 2:00 Diariame... | 2:00:00 16/0... | 2:00:00 15/0... |
| ShadowCopyVolume{0a3f91d... | Horario múltiple de ... | 7:00:00 16/0... | 12:00:00 15/... |

Figura 52: Tarea programada automática li_run

8.6 Miembros calculados y conjuntos

El último paso en la composición de esta solución de BI, fue definir en Litebi los miembros calculados y los conjuntos necesarios para la realización de los informes del centro. Como se explica en el Anexo 1, los miembros calculados y los conjuntos son datos que se pueden calcular a partir de los datos existentes en los cubos a través de fórmulas multidimensionales en lenguaje MDX.

Litebi posee una forma muy sencilla de generar estos elementos a través de un editor de fórmulas. De esta manera el usuario no necesita conocer todos aquellos aspectos técnicos del lenguaje, sino que gracias al editor podemos hacernos una idea de la construcción de las fórmulas y las operaciones que realizan cada una de sus funciones.

Para el caso de CECAV se definieron dos miembros calculados y dos conjuntos que explicaremos a continuación:

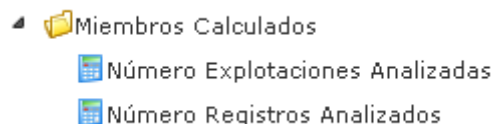


Figura 53: Miembros calculados de la solución en Litebi

- Número Explotaciones Analizadas: Como se había definido al identificar las necesidades de información del cliente, este factor era uno de los importantes, pero no teníamos la posibilidad de obtenerlo a nivel de ETL dada su complejidad.

podíamos distinguir y contar los diferentes registros que formaban los diferentes grupos de muestras. A continuación mostramos el código en MDX (se realizan productos cartesianos con las jerarquías que se seleccionarán en los informes que afecten al número de registros):

```
COUNT(
    Filter(
        Crossjoin(
            Descendants([123557892_123636907_47.123636907].CurrentMember,[12355789
            2_123636907_47.123636907].[123557893])
            ,
            Crossjoin(
                {
                    [123557892_123635181_47.123635181].CurrentMember
                    :
                    [123557892_123635181_47.123635181].CurrentMember
                }
                ,
                {
                    [123557892_123634743_47.123634743].CurrentMember
                    :
                    [123557892_123634743_47.123634743].CurrentMember
                }
            )
        )
    , NOT ( IsEmpty (
        [Measures].[123601004_Sum] ) ) ) )
```

Esta fórmula se utilizó únicamente para los casos de análisis de muestras en placas y así obtener el número de placas que se habían analizado (se definió en todos los cubos menos el de “Resultados Origen 1 Global” ya que trataba con muestras independientes).

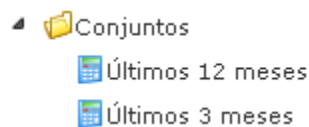


Figura 54: Conjuntos de la solución en Litebi

- Conjuntos Últimos 12 meses y Últimos 3 meses: Estos conjuntos se definieron para obtener, siempre que el usuario lo deseara, la selección de forma sencilla del conjunto de meses que necesitara (fue solicitado por el usuario ya que esta era la periodicidad de la mayoría de los informes). A continuación se muestra la fórmula para los últimos 12 meses realizada con el editor de Litebi:

Cola (Miembros del nivel (Nivel de la jerarquía (Jerarquía del miembro (Miembro actual de la dimensión (Tiempo_Fecha Fin Análisis)) 'Mes')) ,12)



La fórmula para los últimos 3 meses sería similar, únicamente cambiando el 12 por el 3. Esto supone una gran ventaja, ya que estos conjuntos se actualizan a medida que pasa el tiempo y de esta manera tendríamos informes actualizados independientemente del tiempo que pase.

9 RESULTADOS DE LA SOLUCIÓN

En el momento en el que la implantación de la solución se hizo efectiva cubriendo todas las necesidades de información demandadas por el cliente y los datos estaban siendo actualizados a diario, quedaba la parte más importante de todas, la presentación de resultados.

En esta última fase, se procedió a analizar aquellos informes que la empresa cliente estaba realizando anteriormente a la implantación de la solución de BI, y de esta manera, se presentó al usuario final aquellos resultados que inicialmente demandaba. Además de presentar a la empresa cliente los resultados que la solución ofrecía, se les impartieron una serie de formaciones sobre la aplicación para que pudieran manejarse libremente por ella y realizar sus propios informes sin la necesidad de contactar con la empresa implantadora. Para ello, se les otorgo unas credenciales de autenticación de la aplicación a aquellas personas que iban a ser los encargados de generar los informes en el centro, así como a aquellas personas de cargos más altos para únicamente acceder a observar la información que necesitaran.

9.1 Informes

Como hemos dicho, el objetivo final de esta solución es el de presentar una serie de resultados que satisfagan las necesidades que el cliente demandaba inicialmente. Basándonos en el material proporcionado por CECAV, se confeccionaron una serie de informes, tanto gráficos como geográficos, que sirvieran como primer ejemplo para la posterior generación de más reportes por parte de la organización.

Aquí mostramos una serie de informes confeccionados en los diferentes orígenes de datos que fueron tratados, lo cual fue de gran utilidad para el centro. Como se observa en las figuras, las tablas de datos aparecen vacías respetando así el acuerdo de confidencialidad pactado con CECAV.

| Reg x Comarca | | Indicadores | Número Registros (Sum) |
|---------------|-----------------------|-------------|---------------------------|
| (All) | Comarca | | |
| | Alcalatén | ▶ | |
| | Alto Maestrazgo | ▶ | |
| | Alto Mijares | ▶ | |
| | Alto Palancia | ▶ | |
| | Bajo Maestrazgo | ▶ | |
| | Bajo Segura | ▶ | |
| | Campo de Turia | ▶ | |
| | Canal de Navarrés | ▶ | |
| | Condado de Cocentaina | ▶ | |
| | Costera | ▶ | |
| | Hoya de Alcoy | ▶ | |
| Total ▶ | Hoya de Buñol | ▶ | |
| | Huerta Sur | ▶ | |
| | Los Serranos | ▶ | |
| | Marina Alta | ▶ | |
| | Plana Alta | ▶ | |
| | Plana Baja | ▶ | |
| | Puertos de Morella | ▶ | |
| | Ribera Alta | ▶ | |
| | Safor | ▶ | |
| | Utiel - Requena | ▶ | |
| | Valle de Albaida | ▶ | |
| | Vinalopó Medio | ▶ | |

Figura 55: Informe Número de registros por Comarca

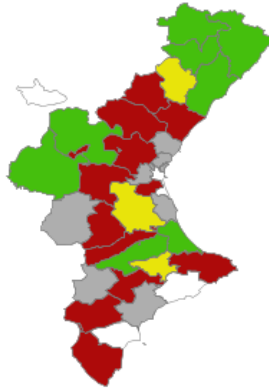


Figura 56: Mapa de número de registros por Comarca

| Reg x Provincia | Indicadores | Número Muestras (Sum) | Titer (Avg) | Titer (Max) | Número Explotaciones Analizadas | Número Registros Analizados |
|-----------------|------------------------|-----------------------|-------------|-------------|---------------------------------|-----------------------------|
| (All) | Provincia | | | | | |
| | Albacete | | | | | |
| | Alicante | | | | | |
| | Almería | | | | | |
| | Badajoz | | | | | |
| | Castellón | | | | | |
| | Cuenca | | | | | |
| | Desconocida | | | | | |
| | Granada | | | | | |
| | Guadalajara | | | | | |
| Total | Islas Baleares | | | | | |
| | León | | | | | |
| | Madrid | | | | | |
| | Murcia | | | | | |
| | Santa Cruz de Tenerife | | | | | |
| | Tarragona | | | | | |
| | Teruel | | | | | |
| | Toledo | | | | | |
| | Valencia | | | | | |
| | Zaragoza | | | | | |
| | Total | | | | | |

Figura 57: Informe Diferentes indicadores por Provincia

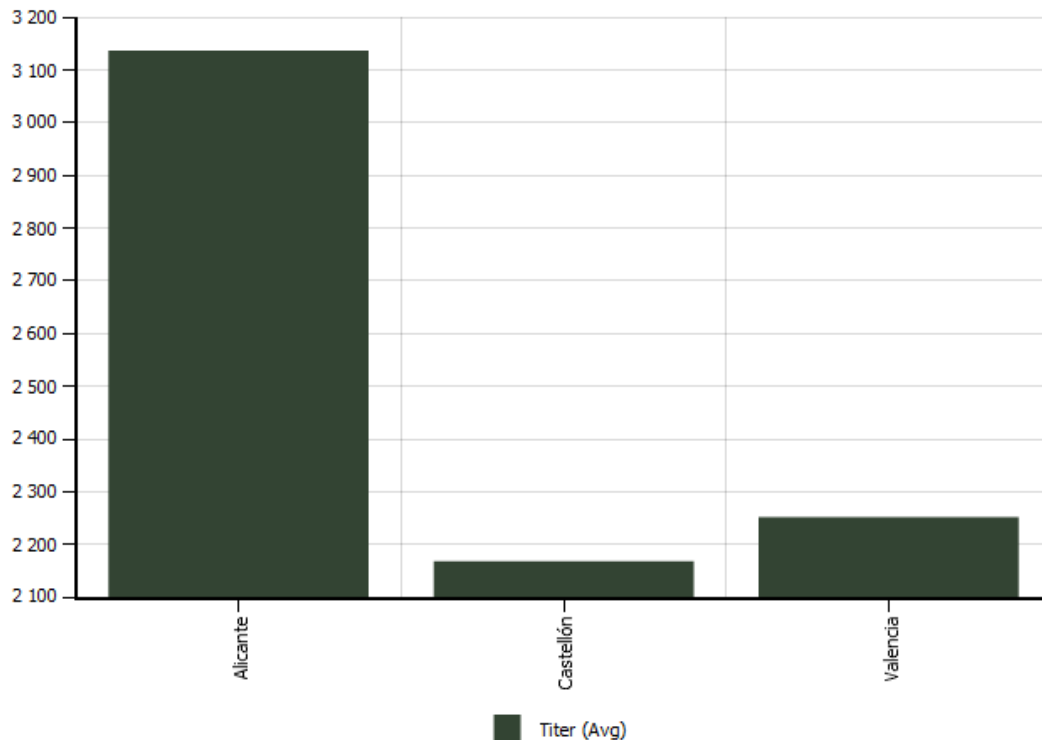


Figura 58: Gráfica de media del Títer en Provincias de la Comunidad Valenciana

Cabe destacar que estos informes, como bien se ha dicho en capítulos anteriores, están dotados de una gran capacidad dinámica a través de la selección de información gracias al filtrado. Únicamente cambiando una dimensión, como puede ser la enfermedad a tratar, podemos obtener diferentes reportes que aportan la información necesaria para los análisis que demandaba el cliente en unos pocos clics de ratón.

9.2 Usuarios y seguridad

Una vez realizados todos aquellos informes que demandó el cliente, pasamos a darles acceso pleno a la aplicación con el fin de sacarle el mayor provecho a la solución y para que sirva de sustitutivo a los otros sistemas de reportes que se estaban utilizando.

Principalmente los usuarios destinados a utilizar la aplicación fueron los investigadores de los departamentos de análisis de muestras, que serían los encargados de realizar los informes y presentarlos al nivel superior del centro, y a la dirección del centro, los

cuales utilizarían la aplicación como una herramienta únicamente de consulta de información para obtener de un simple golpe de vista información sobre los análisis que se realizan en el centro.

Desde la pestaña de configuración de Litebi se crearon dos roles:

- Usuario: Este rol de usuario se agregó a todos los cubos generados con permisos de lectura sobre ellos. No se dotó de permisos en escritura y eliminación para evitar problemas.
- Agroalimed: Rol especial para ciertos usuarios que solo tendrían acceso a una serie de carpetas de la aplicación con el fin de observar los informes que se ubicaran en ellas.

Una vez creados los roles se les proporcionó acceso a las personas que iban a ocupar esos roles. A cada usuario se le proporcionó un nombre de usuario (coincidente con su dirección de correo electrónico, normalmente para hacerlo de manera estandarizada) y una contraseña de acceso a su elección.

Las personas ubicadas en el centro (investigadores y dirección) se agregaron al rol de “Usuario” y personas ajenas a la organización a las cuales se les creó un usuario se ubicaron en el rol de “Agroalimed” (aquí se situaban clientes y dueños de explotaciones sobre todo).



Figura 59: Usuarios y Roles de la solución

9.3 Planificación de Informes

Otorgando a los usuarios permisos de acceso y habiéndoles proporcionado una formación sobre la aplicación fueron capaces de explotar sus funcionalidades, como la planificación del envío de informes.

Gracias a esta funcionalidad se les proporcionó mayor independencia para la consulta de información, no siendo necesario acceder continuamente a la aplicación. La planificación de informes es bien sencilla (como se puede ver el en Anexo 1) consiste en enviar aquellos informes que planifiquemos a una serie de personas en un periodo concreto. Estos informes pueden llegar al correo electrónico del destinatario en los diferentes formatos de exportación que la aplicación ofrece: Excel, PDF o una url con el enlace directo al informe sobre la aplicación por si se desea realizar modificaciones o explorarlo en detalle (este tipo de envío requiere que el usuario destinatario posea permisos de acceso a la aplicación).

En primera instancia se planificaron una serie de informes en los días de formación que sirvieran como ejemplo y de utilidad para el cliente.

10 CONCLUSIONES

Aún a falta de observar la evolución en el tiempo de la solución implantada, se puede asegurar que se cumplieron los objetivos propuestos y además se dotó a la empresa de un sistema de inteligencia de negocio fácil, potente y asequible. Esta implantación ha servido para observar y corroborar de primera mano la gran utilidad de estos sistemas en las organizaciones así como mostrarlo como escaparate para posibles futuras implantaciones, ya que se trata de un caso de éxito claro.

En la solución se han aprovechado todas las capacidades y las herramientas de las que dispone Litebi en la actualidad, siendo posible mayor funcionalidad en un futuro, aumentando así la inteligencia de negocio de la organización. Se ha demostrado la potencia de acceso a la información, al acceso a los cubos y al diseño de informes, y lo más importante, las posibilidades de acceso vía portal web a toda la información. CECAV quedó totalmente satisfecha de la usabilidad, fiabilidad y robustez que ha ofrecido esta tecnología. Los usuarios destacaron sobre todo la rapidez y la calidad de los informes generados (destacando enormemente los informes a nivel geográfico), así como un aumento de la comunicación por parte de los diferentes laboratorios con la dirección del centro.

Gracias a esta solución se eliminó completamente la dependencia de los anteriores sistemas heterogéneos de reporte, integrando toda la información de los análisis de los laboratorios de la organización en un único lugar, más potente, rápido, sencillo y fiable. Además cabe destacar la escalabilidad de la solución, que permitirá en un futuro de una forma sencilla, disfrutar del sistema a otros departamentos, si se diera el caso, así como a otras áreas de la empresa (finanzas, producción, administración...etc.).

A nivel personal y profesional debo decir que este proyecto me ha aportado muchos conocimientos y muchas ganas de seguir por el camino de la inteligencia de negocios, ya que me ha parecido realmente apasionante. Este proyecto me ha permitido jugar diferentes roles a lo largo de su desarrollo, tanto técnicos como empresariales, adquiriendo una experiencia que pienso que me será realmente útil en mi futuro profesional.



11 BIBLIOGRAFÍA

- [1] Análisis Business Intelligence “*Varios artículos*” (Mayo – Junio 2010)
<http:// analisisbi.blogspot.com/>
- [2] Emilio Arias “*Éxito en la implantación de un sistema Business Intelligence*” (Mayo 2010)
<http://www.monografias.com/trabajos29/sistema-business-intelligence/sistema-business-intelligence.shtml>
- [3] Bi-Spain.com “*¿Qué es el Corporate Performance Management? Scoreboards, Business Intelligence y Enterprise Planning*” (Junio 2010)
<http://www.bi-spain.com/>
- [4] Carlos Borrás “*Business Intelligence*” (Mayo 2010)
<http://firmas.lasprovincias.es/carlosborras/business-intelligence/>
- [5] *Business Intelligence.com: The Resource for Business Intelligence* (Mayo – Junio 2010)
<http://www.businessintelligence.com/>
- [6] CECAV Centro de calidad avícola y alimentación animal de la Comunidad Valenciana (Junio 2010)
<http://www.cecav.es>
- [7] Corporate Information Factory (CIF) Resources by Bill Inmon (Junio 2010)
<http://www.inmoncif.com/home/>
- [8] CRM, BI, Call Center, Customer Data: News, Experts & Learning Guides (Junio 2010)
<http://www.searchcrm.com>
- [9] Luz María Duarte Barroso “*Importancia del ERP en las corporaciones*” (Mayo 2010)
<http://www.gestiopolis.com/canales3/ger/impoerporq.htm>
- [10] Garnett, Jody y Pumphrey, Mike “*Geoserver*” (Junio 2010)
<http://geoserver.org/display/GEOS/Welcome>
- [11] Santiago Gracia Esteban (Valencia, 1996) “*Proyecto de final de carrera: Análisis de la implantación de una sistema de información, en una empresa del sector de la automoción*” Editado por la UPV
- [12] Ibermática “*Business Intelligence. El conocimiento compartido*” (Mayo 2010)
http://lam.e-gim.net/gimmaster/ftp_lam/docs/articulobusinessintelligence.pdf



- [13] Bill Inmon vs. Ralph Kimball (Mayo 2010)
<http://www.1keydata.com/datawarehousing/inmon-kimball.html>
- [14] Instituto Nacional de Estadística (Marzo 2010)
<http://www.ine.es/>
- [15] Lite Internet Solutions S.L. Litebi. “*Litebi*” (Enero -Julio 2010)
<http://www.litebi.com>
- [16] Lite Internet Solutions S.L. Litebi. “*Plan de negocio*” (*Documentación interna*)
- [17] Lite Internet Solutions S.L. Litebi. “*Manual de usuario*” (*Documentación interna*)
- [18] Alfonso López “Balanced Scorecard” (Junio 2010)
<http://www.ciberconta.unizar.es/LECCION/bsc/INICIO.HTML>
- [19] Ministerio de administraciones públicas “*El Data warehouse*” (Mayo 2010)
<http://www.csi.map.es/csi/silice/DW1.html>
- [20] Monografias.com “*Data mining*” (Junio 2010)
<http://www.monografias.com/trabajos/datamining/datamining.shtml>
- [21] Nase, Inteligencia de Negocios “*5 pasos para lograr un proyecto de Business Intelligence exitoso*” (Junio 2010)
<http://www.nase-it.com/pdf/PasosparaBI.pdf>
- [22] Pentaho Data Integration “*Kettle Documentation*” (Enero – Mayo 2010)
[http://wiki.pentaho.com/display/EAI/Latest+Pentaho+Data+Integration+\(aka+Kettle\)+Documentation](http://wiki.pentaho.com/display/EAI/Latest+Pentaho+Data+Integration+(aka+Kettle)+Documentation)
- [23] J.E.Pereira “*Cuadro de Mando Integral, CMI*” (Junio 2010)
http://www.mercadeo.com/41_scorecard.htm
- [24] Javier Racca “*La hora del BPM*” (Junio 2010)
<http://www.tynmagazine.com/641-La-hora-del-BPM.note.aspx>
- [25] Dolors Reig “*¿Qué es el Cloud Computing?*” (Junio 2010)
<http://www.dreig.eu/caparazon/2008/10/30/¿que-es-el-cloud-computing-definicion-tendencias-y-precauciones/>
- [26] Sales force (Junio 2010)
<http://www.salesforce.com>



[27] Sinnexus “*Business Intelligence*” (Mayo 2010)

http://www.sinnexus.com/business_intelligence/index.aspx

[28] TodoBI <business intelligence> “*Los 10 factores clave de un sistema BPM*” (Junio 2010)

<http://todobi.blogspot.com/2005/09/los-10-factores-clave-de-un-sistema.html>

[29] Elizabeth Vitt, Michael Luckevich, Stacia Mismar (2003 – McGraw-Hill)

“*Business Intelligence. Técnicas de análisis para la toma de decisiones estratégicas*”

[30] Wikipedia (Mayo - Junio 2010)

<http://es.wikipedia.org>

ANEXO 1: PLATAFORMA BI SAAS, LITEBI

1 MÓDULOS Y CARACTERÍSTICAS BÁSICAS

Litebi es una aplicación Web que permite integrar datos de cualquier origen de datos, relacionarlos entre ellos y disponer de potentes posibilidades de análisis más allá de las posibilidades actuales de la inteligencia de negocios.

Se trata de una completa plataforma SaaS de Inteligencia de negocios basada en cubos OLAP, Reportes, Cuadros de Mando y una Herramienta ETL de integración de datos. Esta plataforma BI permite definir un data warehouse completo, o simplemente cargar y analizar datos de un excel en pocos minutos.

La funcionalidad de Litebi está dividida en módulos y varía desde funcionalidades de comprobada eficacia como sistemas OLAP y Cuadros de Mando, hasta tecnologías innovadoras y propias que hacen de Litebi una herramienta de Inteligencia de negocios de nueva generación, cómo el uso de técnicas de integración de información, inteligencia artificial o web semántica aplicadas a la toma decisiones empresariales.

1.1 *liteSpace*

Es el diccionario de datos de Litebi, contiene datos y estructuras de metadatos que serán utilizadas en los procesos de análisis de información por el usuario, divididos en dos familias:

- **Estructurados:** Cubos y dimensiones para el análisis OLAP y la minería de datos (todavía no disponible). Enfocados al análisis cuantitativo.
- **No estructurados:** Para análisis semánticos, textuales y de contenido. Enfocado al análisis conceptual.

Una característica fundamental de liteSpace es la posibilidad de que el usuario, de forma sencilla y visual defina su propio diccionario de datos, creándose

automáticamente las estructuras físicas y lógicas necesarias para albergar la información.

Esta innovación supone un gran salto respecto al modo tradicional de desarrollar sistemas de BI (en concreto las tareas relativas al diseño e implementación de Data Warehouse) y supone una gran diferencia respecto a sus competidores. Algunas de las características principales de este módulo son las siguientes:

- Cada cliente dispone de su propio liteSpace (espacio) con sus datos analíticos (Cubos) e informes (Vistas de análisis).
- Editor de Cubos y Dimensiones: Permite definir los modelos analíticos en los que se cargaran los datos.
- Seguridad basada en roles.
- Sistema de gestión de contenidos que permite organizar los elementos de análisis en carpetas. Cada usuario posee unas carpetas privadas de uso personal y existe un espacio compartido de carpetas públicas.

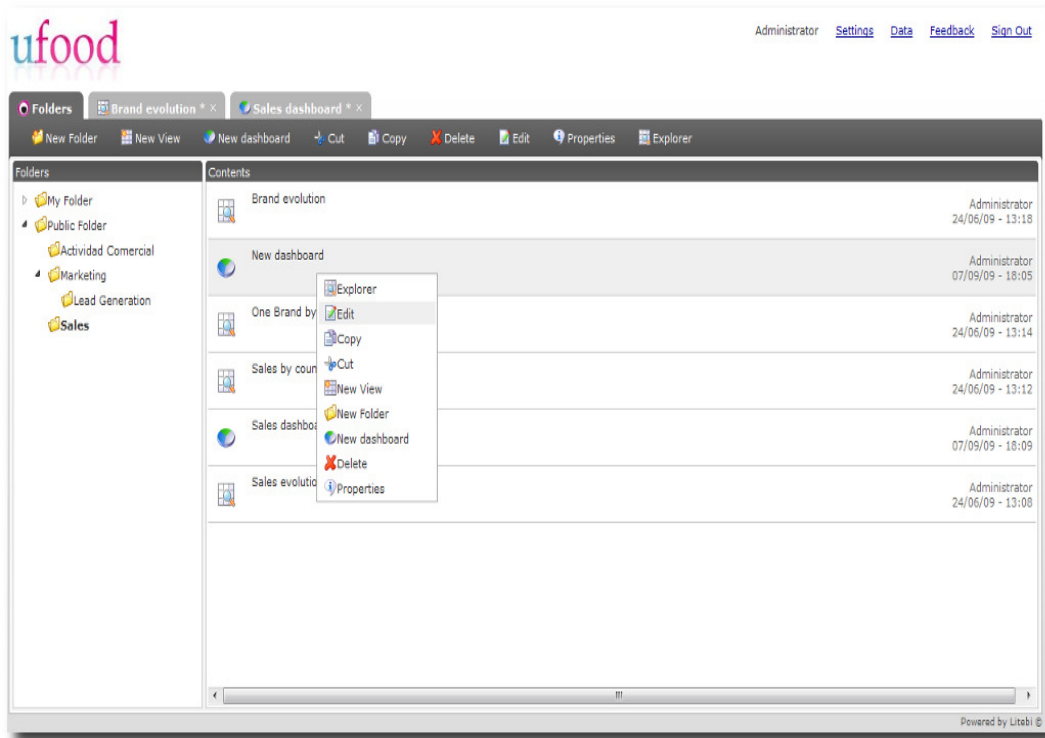


Figura 60: liteSpace (Sistema de archivos y carpetas)

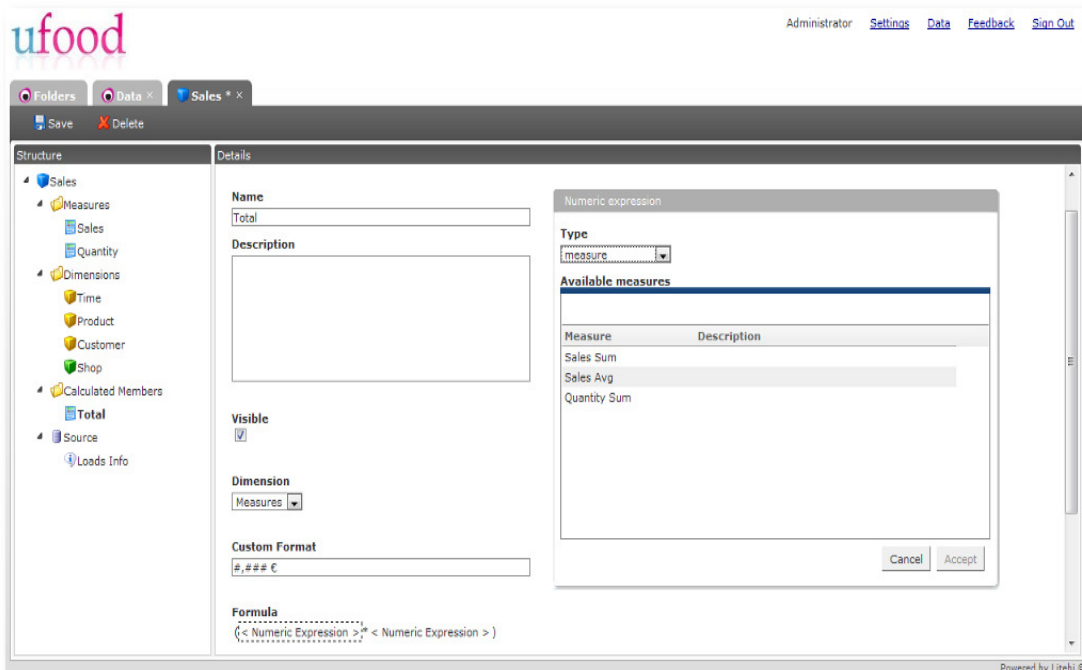


Figura 61: liteSpace (Editor de cubos)



1.2 *liteIntegrator*

Herramienta de integración de datos basada en la popular y potente herramienta de ETL Open Source Kettle de Pentaho, que pretende llevar la integración de información en organizaciones al entorno actual, en el que cada vez hay más información distribuida en múltiples formatos y fuentes en Internet, que una organización bien informada ha de tener en cuenta junto con los orígenes de datos existentes en sus sistemas.

Este módulo permite:

- Definir procesos para integrar, transformar y preparar la información para ser analizada en liteSpace, capaces de ejecutarse de forma periódica en la plataforma de Litebi.
- Integrar datos provenientes de la infraestructura del cliente, situados “detrás del firewall”. Por ejemplo: BBDD corporativas, CRM, ERP, ficheros excel, texto plano...etc.
- Integrar datos provenientes de orígenes situados en internet: Por ejemplo: Otras aplicaciones SaaS, Plataformas de Cloud Computing, Feeds RSS, servicios web...etc.
- Integrar datos estructurados (Orígenes relacionales, XML, RSS) y no estructurados (Webs, PDFs, Texto...etc.).

liteIntegrator está compuesto de tres componentes fundamentales:

1. Capa Servicios Web: Desarrollados usando la tecnología AXIS2, permiten carga a través de canales securizados de datos en liteSpace desde cualquier lugar a través de Internet. Suponen la base para la construcción de una API de Servicios Web que permita embeber con facilidad la funcionalidad de Litebi en productos de terceros, apoyando la línea de negocio de OEM partners.

2. Agente de Integración: Accede a cualquier origen de datos y ejecuta los procesos de integración de Información que cargan liteSpace a través de la capa de servicios web.
3. Servidor de Integración: Es responsable de coordinar los procesos de integración de información, ya estén estos ejecutándose en un agente o en el propio servidor. Es capaz de ejecutar procesos de integración en la plataforma de Litebi.

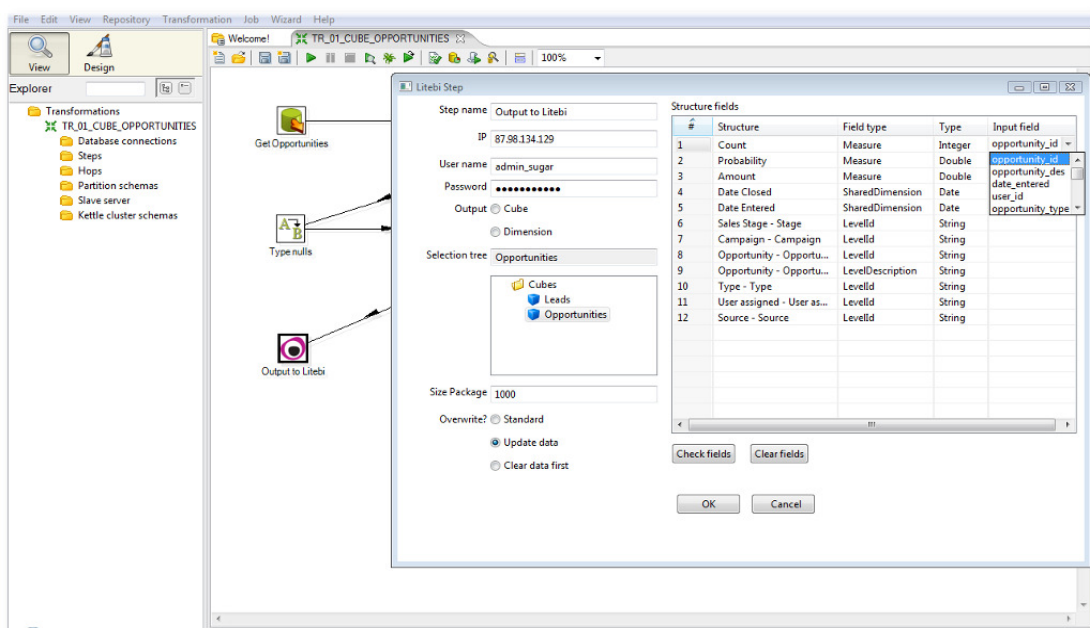


Figura 62: liteIntegrator

1.3 Módulos de análisis

Una vez los datos están cargados en liteSpace mediante liteIntegrator es posible utilizar diferentes herramientas analíticas a través de la plataforma.

A continuación se exponen cada una de estas herramientas.

1.3.1 liteExplorer

Potente herramienta OLAP de análisis y exploración de datos estructurados, de uso sencillo orientada al usuario final, permite:



- Interactuar con la información residente en liteSpace de forma intuitiva, dinámica y potente, ayudando al análisis de grandes cantidades de información para la toma de decisiones.
- Realizar operaciones de exploración multidimensional, *drill down*, *drill up*, *axis swapping*, filtros, jerarquización...etc.
- Visualizar gráficamente los datos explorados, mediante múltiples tipos de gráficas y mapas.
- Exportar los informes generados a formatos habituales como Excel y PDF.
- Realizar operaciones de minería de datos (aún en desarrollo):
 - o Predicción. Obtener automáticamente previsiones de resultados futuros en función de los disponibles (*ej. proyección de ventas*)
 - o Extrapolación: En función de valores numéricos de un área información, predecir posibles valores para otras variables (*ej. Teniendo en cuenta el comportamiento de este producto en esta zona, ¿que he de esperar en esta otra región?*).

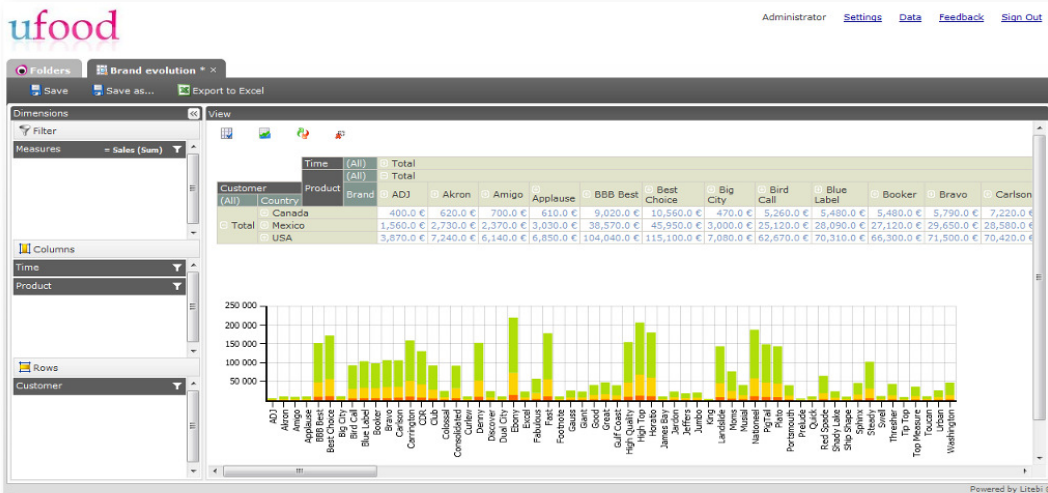
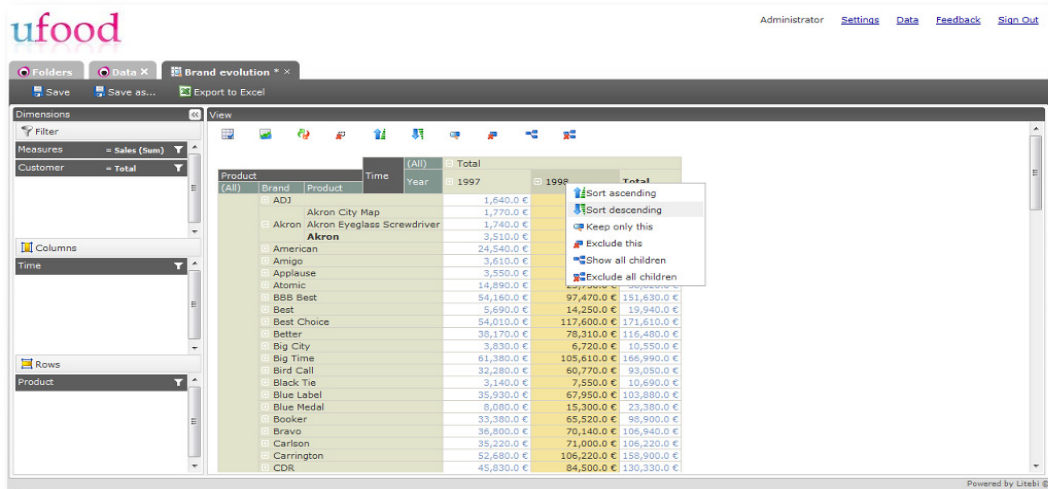


Figura 63: liteExplorer

1.3.2 liteMonitor

Es la herramienta de cuadros de mando de Litebi (dashboards), permite al usuario diseñar cuadros de mando utilizando controles visuales para representar la información existente en LiteSpace: Tablas de datos OLAP, gráficas, alarmas, datos textuales, patrones, scorecards...etc., permitiendo la interacción con el mismo mediante filtros y operaciones de drill-through. En definitiva, LiteMonitor permite al usuario monitorizar los aspectos clave del negocio, integrando recursos.

Con LiteMonitor es posible:

- Obtener una vista resumida de la información más relevante de tu empresa.
- Controlar la evolución de tus indicadores clave de rendimiento (KPIs) a menudo.
- Crear cuadros de mandos ricos y potentes a través de una Interfaz Web muy usable.
- Profundizar (Drill-through) desde cualquier dashboard a una vista de detalle en LiteExplorer.

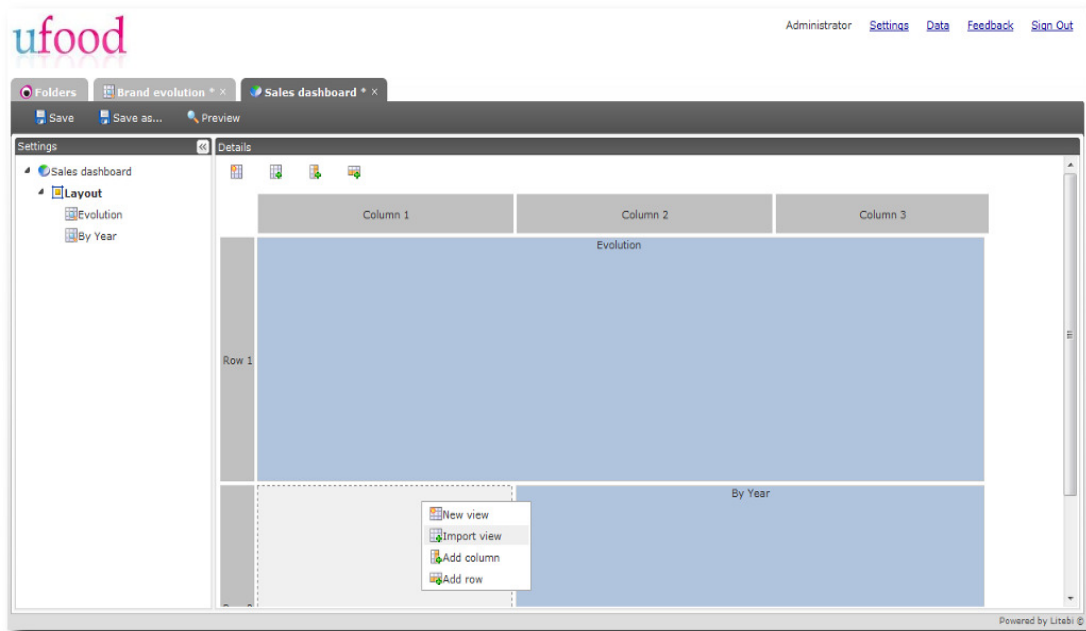


Figura 64: liteMonitor (Gestor de diseño)

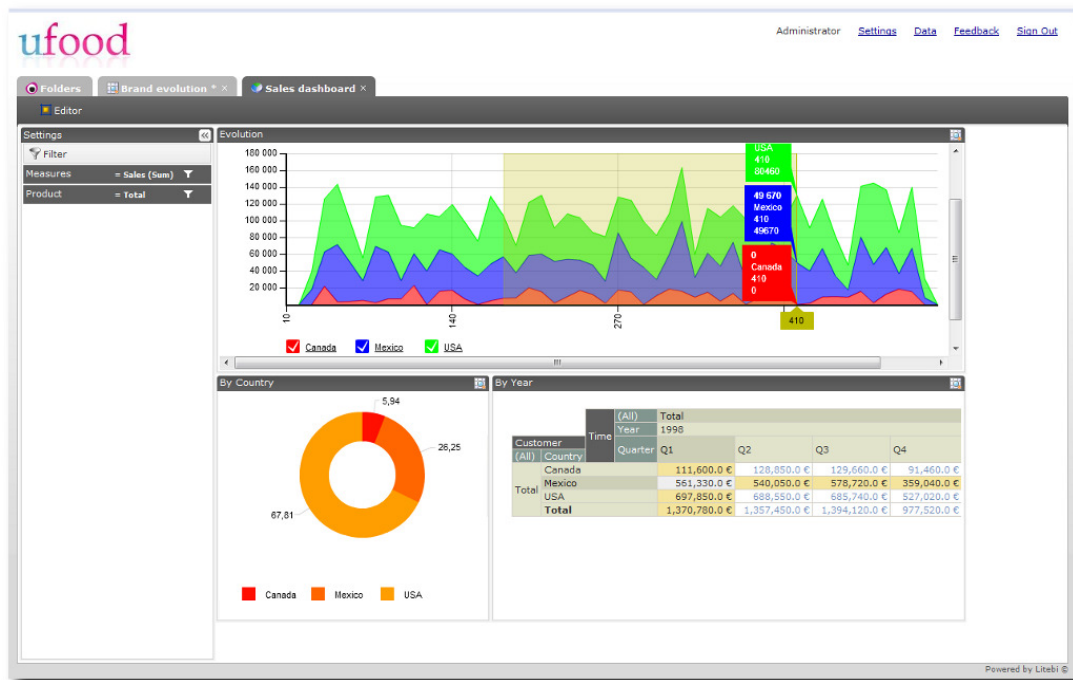


Figura 65: liteMonitor (Visualización)

1.4 Características de la plataforma

En un principio los todos los módulos fueron desarrollados para el usuario final, pero en la mayoría de los casos el usuario final no tiene los conocimientos suficientes para generar modelos analíticos eficientes para el manejo de sus datos.

Por lo tanto, los módulos referentes a la manipulación de los modelos analíticos (cubos y/o dimensiones) estarán a cargo de los consultores y/o partners de Litebi, quienes serán capaces de generar modelos analíticos eficientes para la posterior consulta, manipulación y creación de almacén de datos que podrá estar a cargo del usuario final, cuyos informes consultará para la toma de decisiones inteligentes para su negocio.

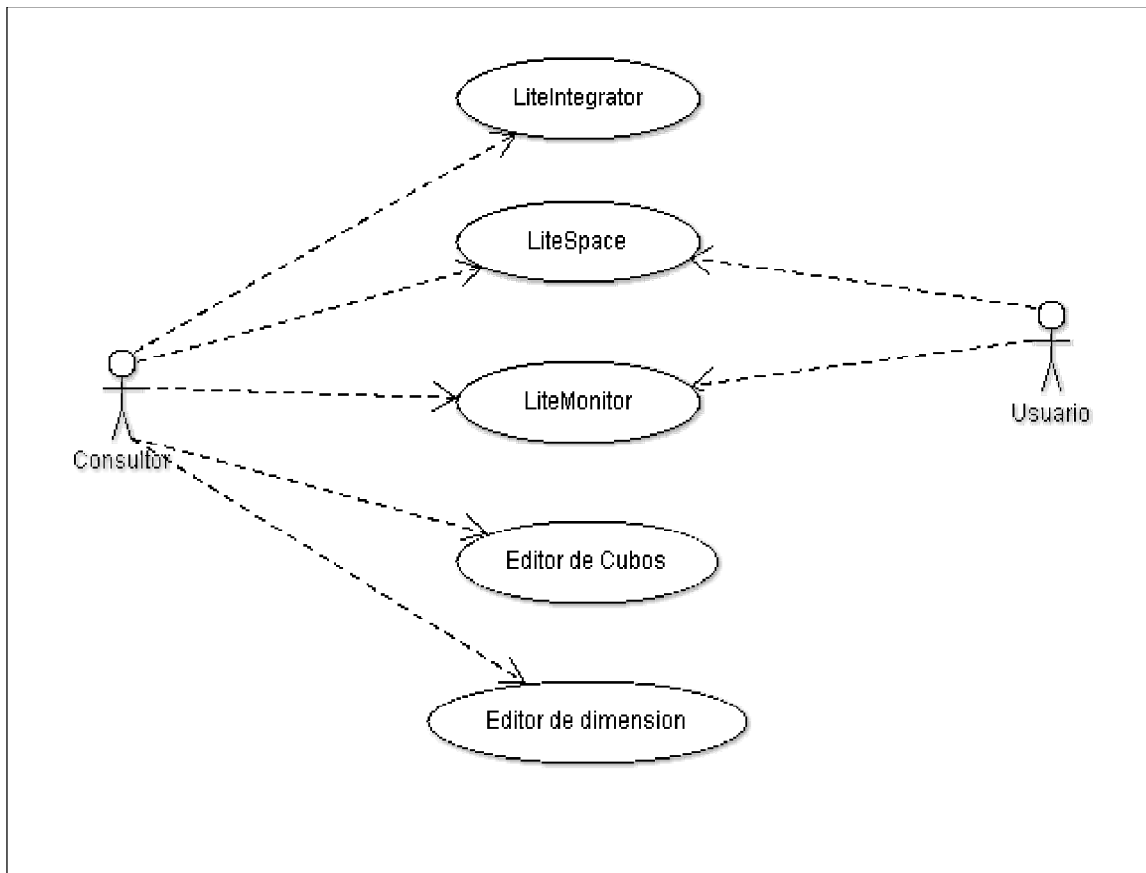


Figura 66: Uso módulos de Litebi

En la Figura 66 se puede observar los módulos que, en la mayoría de los casos, tienen a cargo el Consultor y el Usuario final. A continuación se explica las características de cada uno de estos módulos.

Básicamente Litebi funciona en tres pasos:

1. **Definir** la información que se desea analizar a través de una interfaz web. Esto genera todas las estructuras de datos y metadatos (datawarehouse y modelos analíticos) necesarios para satisfacer las necesidades de información del cliente. Es a lo que se denomina como **liteSpace**.
2. **Cargar** la información proveniente de los sistemas del cliente (ERP, CRM, Excel, Web Services, etc.) a los modelos definidos en **liteSpace**. Para hacer esto se apoya en un producto open source de ETL y una capa de servicios web



que permite construir de forma sencilla procesos de integración de datos muy potentes.

A este conjunto, que nos permite integrar la información que se desea analizar, se le denomina **liteIntegrator**.

3. **Analizar** una vez los datos residen en **liteSpace**, se cuenta con una herramienta de reporting avanzado (OLAP) con la que el usuario final puede, de forma muy sencilla y potente analizar desde cualquier punto de vista la información y generar sus propios informes que pueden ser compartidos a lo largo de toda la organización. Es el fin de los informes hecho a medida o el caos semi-gestionado a base de excels.

Para almacenar la información necesaria para el funcionamiento de Litebi, se tiene actualmente por un lado en la base de datos lo referente a la configuración de cada espacio y la estructura de los diferentes modelos analíticos y por otro lado los datos cargados de fuentes externas mapeados para cada modelo analíticos diseñado en Litebi.

2 DESARROLLO DE SOLUCIONES CON LITEBI

En esta sección trataremos de ilustrar de qué manera se diseña e implementa una solución de BI con Litebi siguiendo los tres pasos fundamentales a través de sus diferentes módulos:

- **Definir** modelos analíticos con Litebi: Diseño de cubos y dimensiones.
- **Cargar** datos en esos modelos analíticos.
- Cómo utilizar las herramientas de **análisis**.

Cabe destacar que la información contenida en este apartado estará destinada a consultores y partners encargados de desarrollar soluciones de BI con Litebi. En el siguiente apartado se mostrara la aplicación desde la perspectiva del uso a cargo del cliente.

2.1 Definir: liteSpace

Internamente Litebi utiliza los conceptos tradicionales del modelado dimensional en Business Intelligence, veamos algunas definiciones:

- **Cubo**: Elemento fundamental de liteSpace. Conjunto de datos organizado orientado al análisis, estructurado de forma dimensional. Los cubos están formados por **métricas** (measures) y **dimensiones**. *Ej. Cubo de ventas, Cubo de contabilidad...etc.*
- **Métrica**: Cantidades numéricas que deseamos controlar en un cubo. Es lo que aparece en las celdas al analizar un cubo. *Ej. Cantidad vendida, Cantidad facturada, Precio unitario...etc.*
- **Dimensión**: Punto de vista de la información que deseamos en el análisis. *Ej. Dimensión geográfica (¿dónde sucedió la venta?), Dimensión Cliente (¿a quién le vendí?), Dimensión Producto (¿Qué vendí?)*. Las dimensión pueden ser **compartidas** (entre varios cubos) o **embebidas** (pertenecientes a un sólo cubo).



- **Jerarquías:** Una dimensión puede tener una o varias jerarquías, una jerarquía es un conjunto de **niveles** cada uno expresando un nivel de profundidad en la información. Permiten analizar la información de forma agrupada (ventas por año) o al detalle (ventas diarias). *Ej. La dimensión tiempo está formada por la jerarquía "Por Trimestre", que tiene los niveles "Año > Trimestre > Mes > Día".* Es posible separar cada una de las jerarquías definidas en una dimensión compartida como dimensiones compartidas nuevas y así ganar potencia y flexibilidad a la hora de analizar los datos.

En el centro de Litebi se encuentra liteSpace, que es como un "espacio analítico" o un "Data Warehouse gestionado". La idea es bien sencilla, ser capaz de ofrecer las ventajas de un Data Warehouse clásico (responder a las necesidades analíticas del cliente, integrar información de cualquier origen, dar cobertura a toda la organización, etc.) pero sin la complejidad, el riesgo y el coste que este tipo de proyectos suelen conllevar.

Gracias al modelo SaaS se elimina la necesidad de gestionar el Data Warehouse por el cliente (y el hardware, y el despliegue...), y en parte gracias a que se tiene una tecnología capaz de permitir a cualquiera definir el "qué" quiere analizar a través de interfaces web (modelos dimensionales) y dejar que sea Litebi el que haga todo el trabajo duro entre bastidores (construcción de modelos, optimización de los mismos, diseño de metadatos...etc.).

Una vez creado el nuevo cubo, el funcionamiento es muy sencillo:

- A la izquierda tenemos un árbol con la estructura del cubo, a la derecha las propiedades del elemento seleccionado.
- Podemos cambiar el nombre del cubo y su definición.
- Podemos crear nuevas métricas y definir para cada métrica:
 - o **Nombre y descripción**
 - o **Formato:** Sigue el estándar de Java (#,###,###.0)
 - o **Agregador:** Las formas de agregación que deseamos para la métrica al totalizarla (que se sume, que se haga la media de los hijos, que se escoja el máximo o el mínimo de los hijos)
 - o **El tipo de datos** (habitualmente Integer o Number para una métrica).
- Podemos crear dos tipos de dimensiones:
 - o **Compartidas:** Indicamos que el cubo utilizará una dimensión compartida (definida aparte en la carpeta Dimensiones, ap. 3.1.2) y fijamos un alias para el uso. Es posible utilizar una misma dimensión en un cubo más de una vez (ej: *Fecha de venta, Fecha de entrega utilizando la dimensión compartida Tiempo*).
 - o **Embebidas:** Son dimensiones que sólo existen para un cubo en concreto. Es habitual utilizarlas para definir propiedades (Ej. Estado de una factura) o simplemente para simplificar el modelado. Al crear una dimensión embebida podemos:
 - Definir su **nombre y descripción**
 - Especificar si tiene “**Total**” es decir un elemento superior que totalice toda la estructura actuando como cúspide de la jerarquía
 - Una **serie de niveles** de detalle que forman una única jerarquía. Para cada nivel podemos definir
 - Nombre y descripción

- Tipo de datos del nivel (de la columna código)
- Si tiene columna descripción (siempre es de tipo String).
Para más información sobre las columnas código y las columnas descripción de un nivel ver edición de dimensiones compartidas.
- Además cada nivel puede tener una serie de propiedades asociadas (*Ver edición de dimensiones compartidas*)



Figura 69: Editando un cubo

- **Miembros calculados y Sets:** Permiten realizar cálculos avanzados y cálculo de conjuntos sobre los objetos del cubo, utilizan el lenguaje MDX (*Multidimensional Expressions*) y es posible definir desde cálculos sencillos (Facturación = Precio unitario * Cantidad vendida) hasta fórmulas complejas (Incrementos entre periodos



de tiempo, porcentajes del total, y un amplio etc.). La edición de miembros calculados requiere de un conocimiento del lenguaje MDX a priori (aunque con la ayuda del editor de fórmulas se simplifica) y queda fuera del ámbito del presente manual. Aquí el manual sobre el lenguaje: MDX Language Reference <http://msdn.microsoft.com/en-us/library/ms145595.aspx>

2.1.2 Definiendo una dimensión compartida

Desde el Diccionario de Datos de Litebi es posible definir dimensiones compartidas que serán utilizadas por diferentes cubos. Típicamente en los proyectos hay dimensiones (Estructuras jerárquicas) que son utilizadas en varios cubos (sets de datos) cómo por ejemplo la dimensión temporal.

Es recomendable siempre que sea posible crear dimensiones compartidas para los cubos, ya que incrementa la facilidad de trabajo en el análisis de éstos.

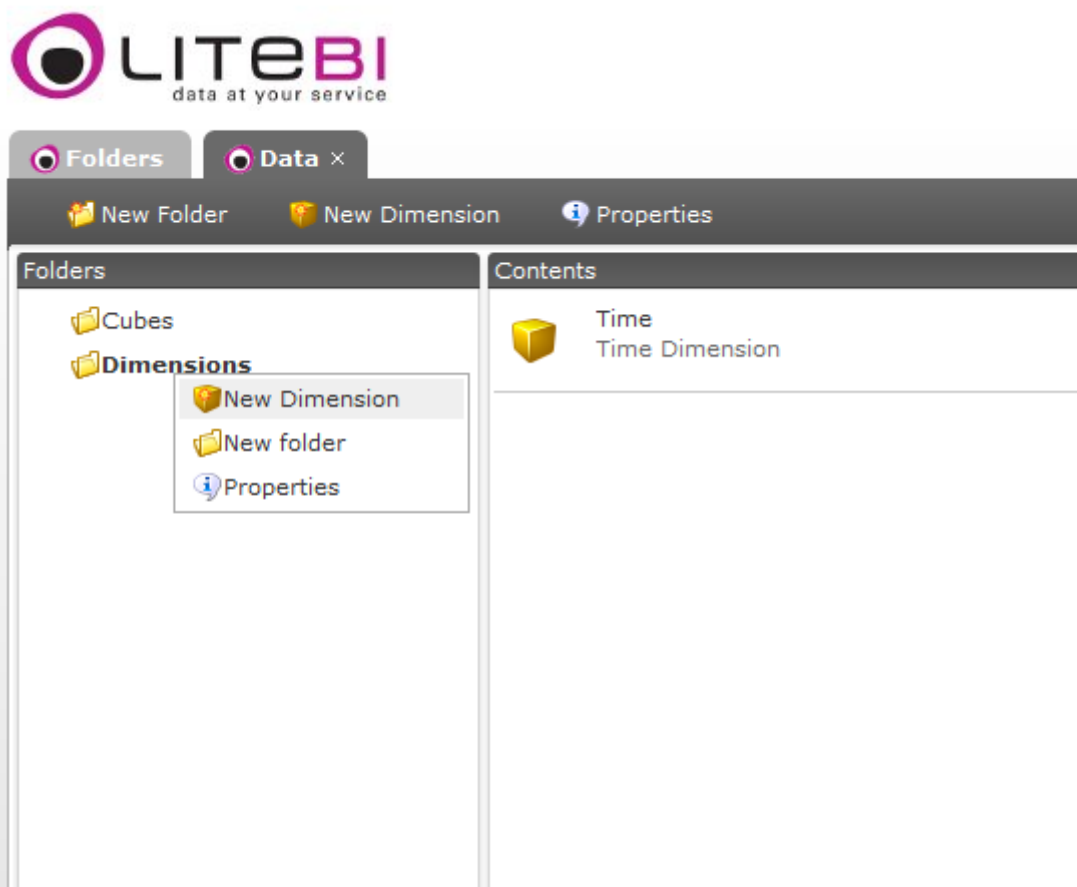


Figura 70: Creando una dimensión

Para crear una dimensión hay que seguir tres pasos:

1. Definir cuál es el **nivel base** de la dimensión.
 - El nivel base es el nivel de máximo detalle de la dimensión, el que posteriormente se utilizará para relacionar la dimensión con el cubo.
 - *Ej. En una dimensión temporal el nivel base podría ser el día, si en el cubo tenemos la información a nivel de día.*

2. Crear los **niveles** que van a ser usados (independientemente de las jerarquías que posteriormente los agrupen).
 - *Ej. Año, Trimestre, Mes, Día de la semana, Semana del año, Quincena...etc.*

- Al definir estos niveles hay que tener en cuenta que un nivel tiene dos tipos de campos
- **Campo Id:**
 - Es obligatorio.
 - Puede ser de cualquier tipo de datos.
 - Define al nivel.
 - *Ej. En el nivel "Producto" sería el "Código de producto" en el nivel "año" sería el número del año "2007".*
- **Campos Descripción:**
 - Es opcional.
 - Siempre es de tipo String.
 - Si existe es utilizado para mostrar el elemento al usuario.
 - Si no existe se utiliza el campo Id para mostrar el elemento al usuario.
 - *Ej. En el nivel "producto" sería el "Nombre del producto", pero en un nivel "año" no sería necesario.*

Es importante tener claros estos conceptos de cara a cargar datos en liteSpace a través de liteIntegrator.

3. **Definir las jerarquías:** Una vez creados los niveles que van a utilizarse en la dimensión, se definen las jerarquía que los van a agrupar.
- Una dimensión puede tener tantas jerarquías como se desee, para dar más alternativas de análisis al usuario.
 - Toda dimensión ha de tener al menos una jerarquía.
 - Todas las jerarquías incluyen el nivel base cómo máximo nivel de detalle.
 - *Ej. Jerarquía "Por Trimestre" formada por los niveles "Año > Trimestre > Mes > Nivel base día"*

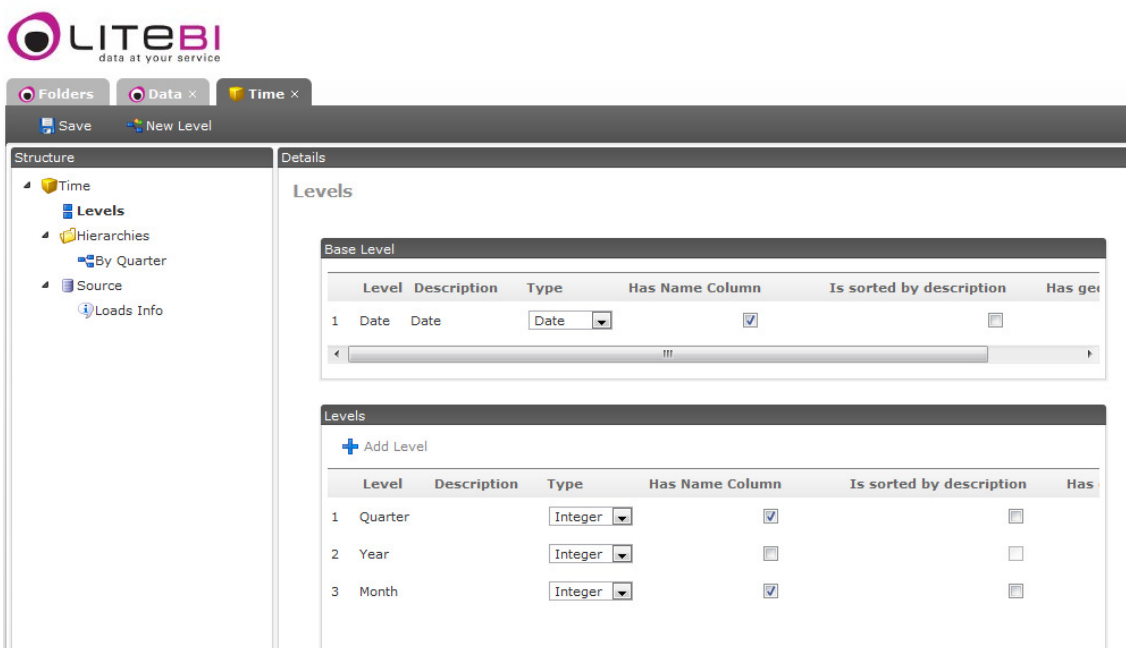


Figura 71: Creando niveles de una dimensión

2.2 Cargar: liteIntegrator

Litebi cuenta con una potente herramienta de ETL a la que llamamos liteIntegrator, que consta de una parte que se ejecuta en el lado del cliente, la cual se apoya en la fantástica herramienta open source Kettle, y una capa de servicios web en el lado de Litebi.

Es una herramienta que sirve para tomar datos de cualquier origen, mapearlos y cargarlos en los diferentes modelos analíticos definidos en Litebi para un espacio privado del usuario, además permite configurarlo de tal manera que los datos sean cargados diariamente, para siempre tener datos actualizados en los modelos analíticos.

Con Litebi el objetivo es construir los procesos que obtengan la información de los orígenes deseados y los organicen de la forma deseada de manera que éstos puedan ser cargados en los cubos y dimensiones de liteSpace.



Los problemas que pueden aparecer a la hora de construir procesos de integración son de diversa índole, así que se centrará aquí el funcionamiento principal de la herramienta.

2.2.1 Instalación de liteIntegrator

Para instalar liteIntegrator hay que seguir los siguientes pasos:

- Tener instalada una máquina de java de versión igual o superior a las 1.6.
- Descargar la herramienta open source base de liteIntegrator (Kettle/Pentaho Data Integration), desde: <http://sourceforge.net/projects/pentaho/>
- Descargar el plugin de Litebi para Kettle de la dirección <https://app.litebi.com/plugin/plugin.zip>
- Descomprimir el plugin en el directorio de instalación de Kettle, en /libext, de forma que el contenido del archivo quede en este directorio.

2.2.2 Funcionamiento de liteIntegrator

Para ver un funcionamiento completo de Kettle lo recomendable es su propia y extensa documentación:

[http://wiki.pentaho.com/display/EAI/Latest+Pentaho+Data+Integration+\(aka+Kettle\)+Documentation](http://wiki.pentaho.com/display/EAI/Latest+Pentaho+Data+Integration+(aka+Kettle)+Documentation)

En resumen, Kettle es una completa herramienta de ETL que utiliza dos tipos de procesos:

- **Transformations:** Hacen el trabajo base de lectura, transformación, limpieza filtrado, etc... de los datos. Con el plugin de Litebi son capaces de comunicarse con

la capa de servicios web de Litebi para cargar datos y obtener información de liteSpace.

- **Jobs:** Coordinan la ejecución de otras transformaciones, definen el flujo de ejecución global de los procesos.

Cómo consejos básicos de trabajo con Kettle:

- NO utilizar el repositorio de transformaciones, es contraproducente, es preferible almacenar las transformaciones y jobs cómo ficheros de XML.
- Construir transformaciones breves y ordenadas.
- Utilizar los JOBS para controlar el flujo completo de ejecución.

Con Litebi el objetivo es construir los procesos que obtengan la información de los orígenes deseados, los organicen de la forma deseada para cargarlos en los cubos y dimensiones de liteSpace.

2.2.3 Paso de Output a Litebi

Además de todos los pasos de lectura, transformación, lookup, etc., de Kettle el paso fundamental para cargar datos a Litebi es el “Output a Litebi”:

- Permite asociar los campos recibidos en la transformación a la estructura de un cubo o una dimensión en liteSpace de un espacio privado, para ello se necesita el usuario y clave del cliente.
- Una vez los campos están correctamente asociados, Kettle es capaz de subir los datos a los cubos y dimensiones quedando estos listos para su análisis.
- La **IP a utilizar** para cargar datos al entorno de Litebi es: “**app.litebi.com**”
- Se puede configurar el número de datos por paquete de envío, por defecto está de 1000 datos por paquete.
- Es posible seleccionar el modo de carga: *Standard* si es la primera vez que se produce la carga, *Actualizar datos* cuando ya se ha hecho alguna carga y solo queremos añadir los registros nuevos, o *Borrar datos primero* para cargar totalmente los datos de nuevo en las estructuras de datos.

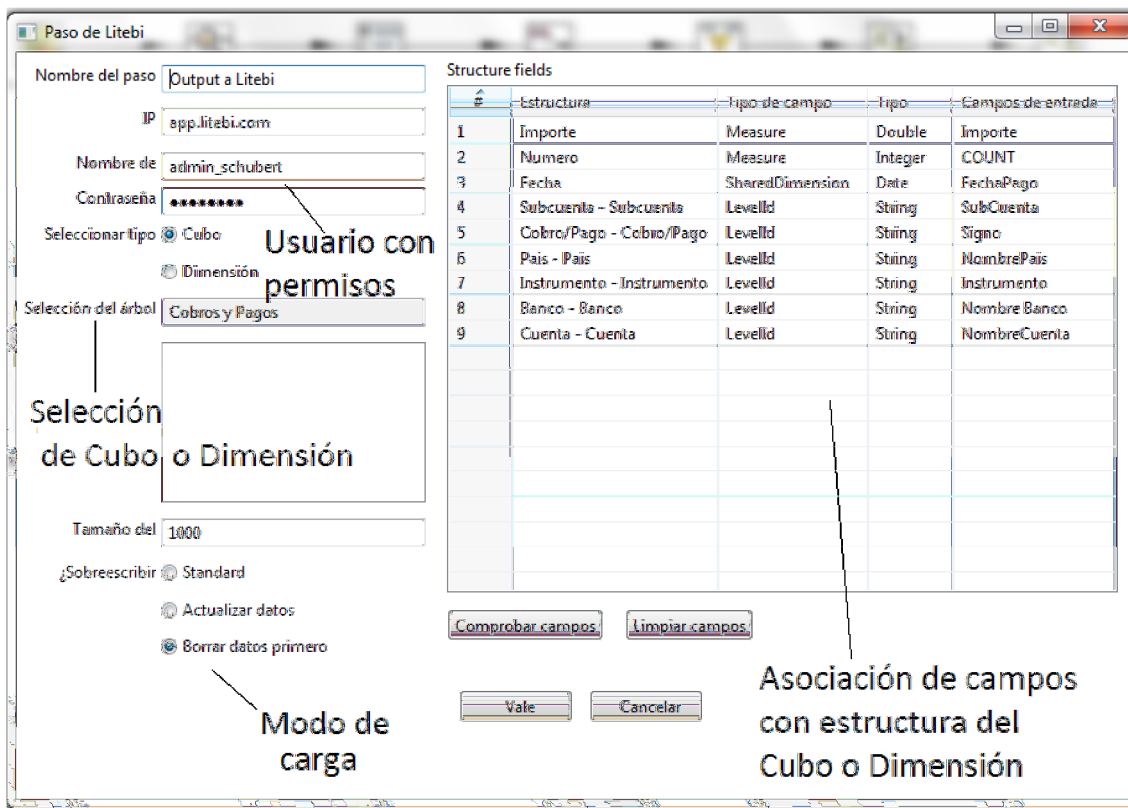
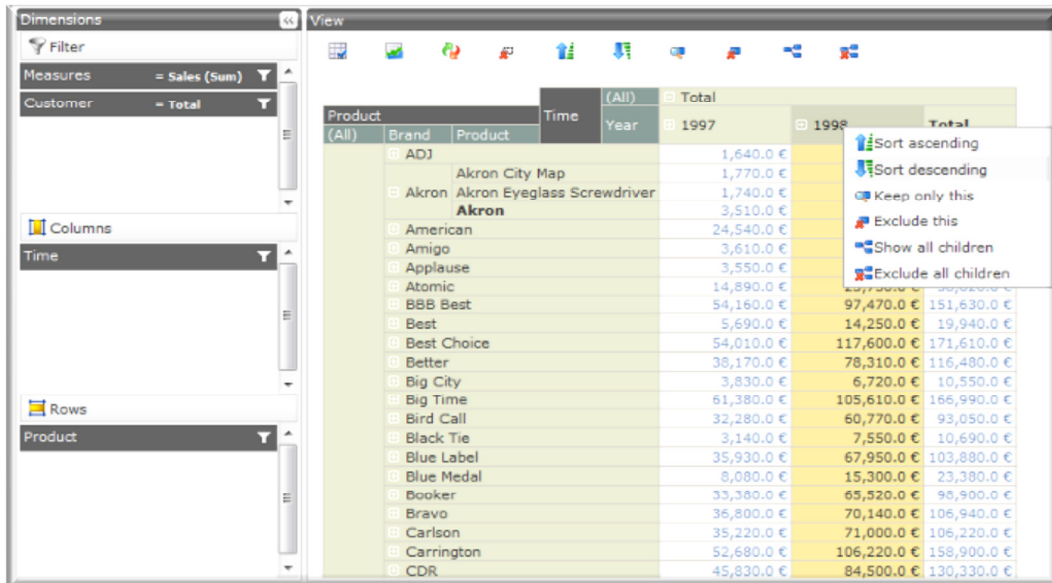


Figura 72: Paso de Output a Litebi (Kettle)

2.3 Análisis: liteExplorer y liteMonitor

2.3.1 liteExplorer

El funcionamiento es muy sencillo, al explorar un cubo o reporte (vista de cubo) se abre liteExplorer quedando en el panel izquierdo disponibles las dimensiones que conforman el cubo y que es posible utilizar para el análisis.



| Product | Time | (All) | Total |
|----------------------------|------|------------|-------------|
| (All) | Year | 1997 | 1998 |
| ADJ | | 1,640.0 € | |
| Akron City Map | | 1,770.0 € | |
| Akron Eyeglass Screwdriver | | 1,740.0 € | |
| Akron | | 3,510.0 € | |
| American | | 24,540.0 € | |
| Amigo | | 3,610.0 € | |
| Applause | | 3,550.0 € | |
| Atomic | | 14,890.0 € | |
| BBB Best | | 54,160.0 € | 97,470.0 € |
| Best | | 5,690.0 € | 14,250.0 € |
| Best Choice | | 54,010.0 € | 117,600.0 € |
| Better | | 38,170.0 € | 78,310.0 € |
| Big City | | 3,830.0 € | 6,720.0 € |
| Big Time | | 61,380.0 € | 105,610.0 € |
| Bird Call | | 32,280.0 € | 60,770.0 € |
| Black Tie | | 3,140.0 € | 7,550.0 € |
| Blue Label | | 35,930.0 € | 67,950.0 € |
| Blue Medal | | 8,080.0 € | 15,300.0 € |
| Booker | | 33,360.0 € | 65,520.0 € |
| Bravo | | 36,800.0 € | 70,140.0 € |
| Carlson | | 35,220.0 € | 71,000.0 € |
| Carrington | | 52,680.0 € | 106,220.0 € |
| CDR | | 45,830.0 € | 84,500.0 € |
| | | | 151,630.0 € |
| | | | 19,940.0 € |
| | | | 171,610.0 € |
| | | | 116,480.0 € |
| | | | 10,550.0 € |
| | | | 166,990.0 € |
| | | | 96,900.0 € |
| | | | 106,940.0 € |
| | | | 106,220.0 € |
| | | | 158,900.0 € |
| | | | 130,330.0 € |

Figura 73: liteExplorer, Visor Olap de Litebi

Éste panel tiene tres áreas:

- **Filtros:** Dimensión que no se están visualizando en la vista pero sobre las que es posible seleccionar filtros (acotar la información que se ve). Ej. No deseo ver los años en filas o columnas, pero deseo ver datos únicamente del año 2008.
- **Columnas:** Dimensiones que deseo ver en la vista del cubo cómo columnas.
- **Filas:** Dimensiones que deseo ver en la vista del cubo cómo filas.
 - o **Edición de Filtros:** Sobre cualquier dimensión es posible activar (mediante botón derecho del ratón) los filtros, pudiendo seleccionar los elementos de la dimensión que quiero utilizar en la vista.

Además con esta herramienta se da la posibilidad de generar gráficos con respecto a los datos observados en el Visor Olap y además, si existen datos geográficos, es posible obtener un mapa que represente estos datos, para ello se utiliza el servidor de mapas GeoServer (Explicado más adelante).

2.3.2 liteMonitor

Con LiteMonitor es posible crear cuadros de mando, con los datos cargados en Litebi, en minutos y modificarlos en tiempo record, todo desde una sencilla interfaz web.

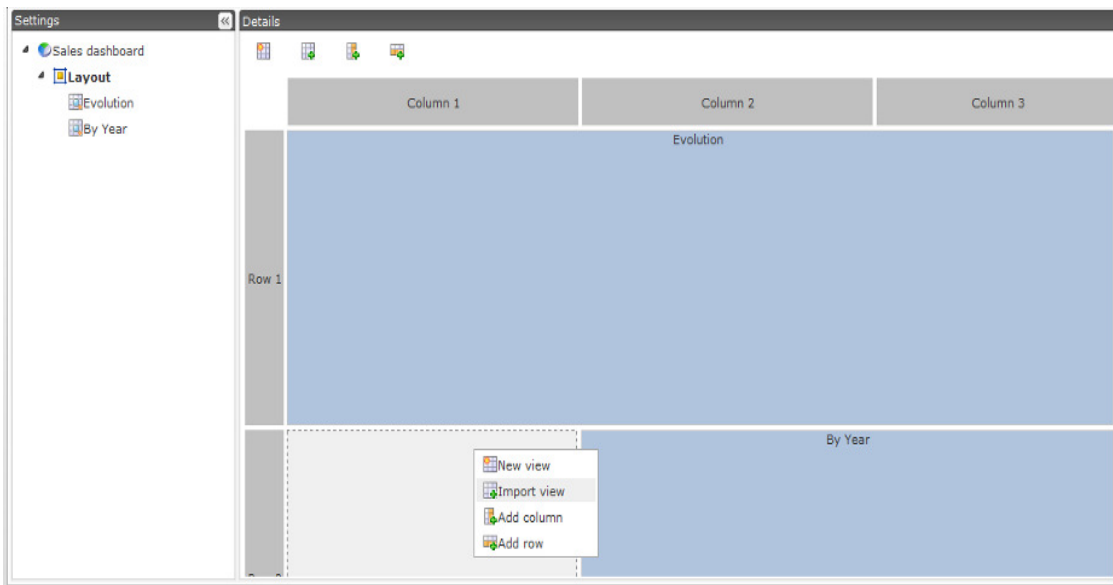


Figura 74: Creación de cuadros de mando

- Cada cuadro de mando está asociado a una vista o informe.
- La vista o informe asociada puede ser importada o creada desde LiteMonitor.
- En el panel de la izquierda aparecerá en forma de árbol las diferentes vistas asociadas al tablero de los cuadros de mando.
- Se podrá decidir posición y tamaño de cada cuadro de mando.
- Se tiene la opción de vista previa.

Como conclusión este apartado, en la Figura 75 podemos observar el diagrama de servicios de Litebi, algunos ya implementados y otros de implementación futura.

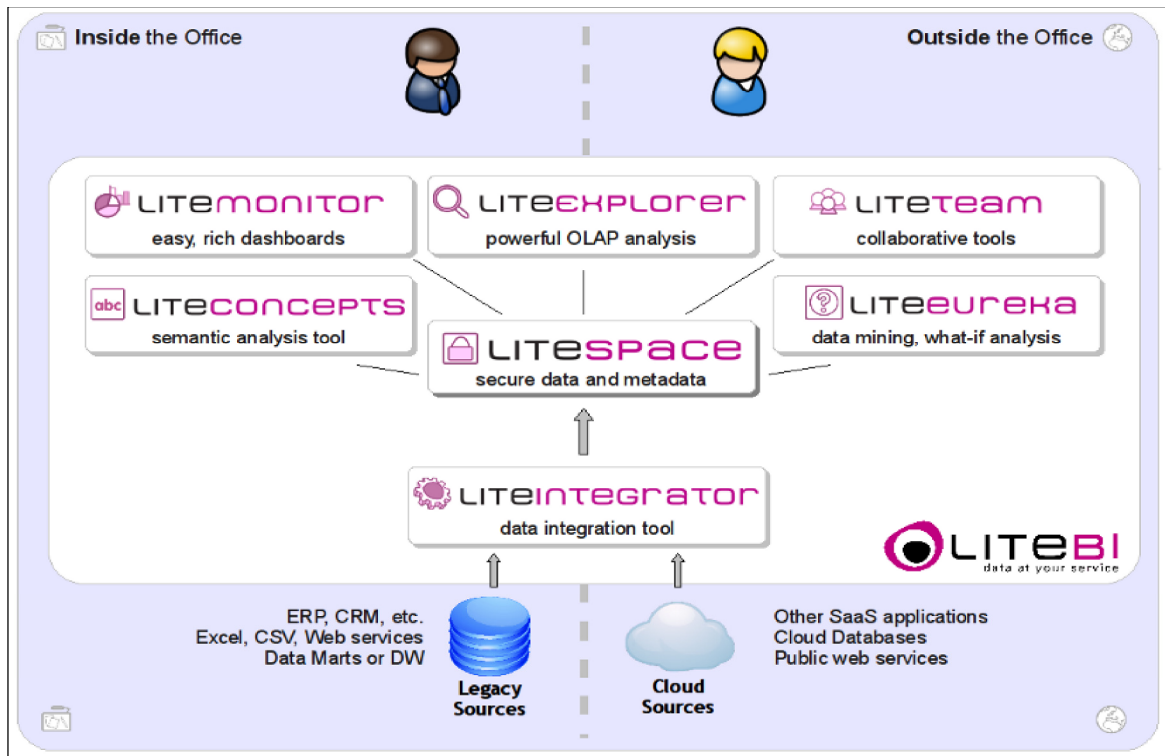


Figura 75: Diagrama servicios de Litebi

3 TECNOLOGÍAS UTILIZADAS EN LITEBI

3.1 Servidor de mapas

Los servidores de mapas permiten la interacción con la información espacial almacenada en servidores de datos espaciales accesibles vía web. El usuario accede a la información de manera que puede visualizarla, consultarla y, en función de las características de los servidores y de los servicios prestados, descargarla o realizar análisis espaciales.

El usuario se conecta a los servicios prestados por estos servidores de mapas a través de cliente tanto ligeros, aplicaciones web que permiten la consulta de estos servidores de mapas desde el navegador, como pesados, aplicaciones SIG de escritorio con módulos que permiten la conexión a servidores de mapas.

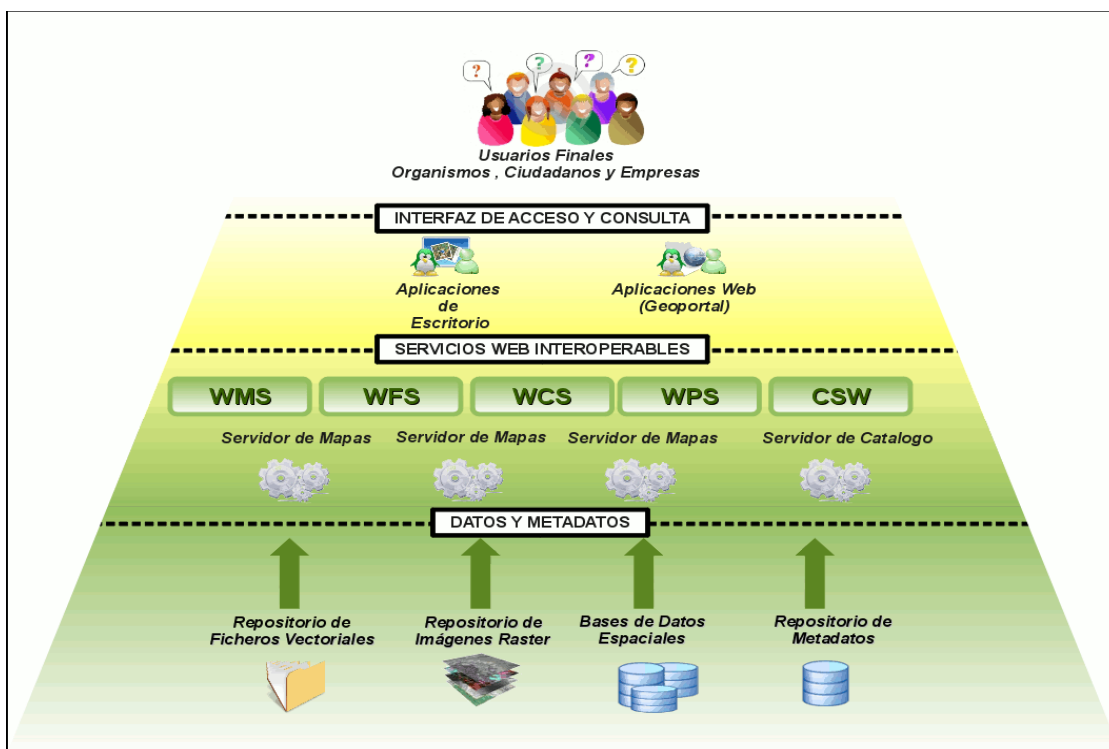


Figura 76: Infraestructura servidores de mapas

Para Litebi se utiliza el servidor GeoServer para el manejo de mapas dinámicos en el Visor Olap de la aplicación.

Geoserver es un servidor de código abierto desarrollado en Java, lo que le hace ser multiplataforma, que permite a los usuarios compartir y editar datos geoespaciales.

Diseñado para la interoperabilidad, publica datos de cualquier fuente de datos espaciales con estándares abiertos.

Geoserver está desarrollado sobre la base Geotools, una biblioteca de sistemas de información geográfica. Geoserver lee una variedad de formatos de datos, incluyendo PostGIS, Oracle Spatial, ArcSDE, DB2, MySQL, Shapefiles, GeoTIFF, GTOPO30, ECW, MrSID y JPEG2000. A través de protocolos estándares es capaz de generar KML, GML, Shapefile, GeoRSS, PDF, GeoJSON, JPEG, GIF, SVG, PNG y otros. Además, se puede editar datos a través de WFS transaccionales (WFS-T). Geoserver incluye un cliente integrado OpenLayers capaz de visualizar datos para obtener una vista previa.

Geoserver permite la publicación eficiente de datos geoespaciales de Google Earth a través de la utilización de enlaces de red, utilizando KML. Permite utilizar las funciones avanzadas de Google Earth para incluir plantillas de salida, pop-ups, el tiempo, altura de visualizaciones, y "super-overlays".

Geoserver es la implementación de referencia del Open Geospatial Consortium (OGC) para las normas Web Feature Service (WFS) y Web Coverage Service (WCS), además está certificado como servidor de alto rendimiento para Web Map Service (WMS). Geoserver es un componente básico de la Web Geoespacial.

GeoServer le permite mostrar la información espacial del mundo. Implementando el estándar de la Web Map Service (WMS), GeoServer puede crear mapas en una variedad de formatos de salida. OpenLayers, una biblioteca de cartografía gratuita, está integrada en GeoServer, lo que hace la generación de mapas rápidos y fáciles. GeoServer se basa en Geotools, un conjunto de herramientas de código abierto de Java SIG.

GeoServer también se ajusta al estándar Web Feature Services (WFS), que permite la participación real y edición de los datos que se utiliza para generar los mapas.



GeoServer es software libre. Esto reduce significativamente las barreras económicas en comparación con los productos tradicionales de los SIG.

Además, no sólo está disponible de forma gratuita, también es de código abierto. Las correcciones de errores y mejoras de características en el software de código abierto están más aceleradas en comparación con las soluciones de software tradicional.

GeoServer puede mostrar los datos en cualquiera de las aplicaciones de mapas populares, tales como Google Maps, Google Earth, Yahoo Maps y Microsoft Virtual Earth. Además, GeoServer puede contactar con las arquitecturas tradicionales de los SIG como ESRI ArcGIS.

3.2 Mondrian

Mondrian, ahora rebautizado como Pentaho Analysis Services, es el motor OLAP integrado en la suite de Business Intelligence Open Source Pentaho.

Mondrian es un proyecto Open Source, licenciado bajo la Mozilla Public License (MPL). Esta licencia es una de las “Business Friendly” lo cual implica que es de las menos restrictivas para su uso desde la mayor parte de los puntos de vista (al igual que la resto de la suite de Pentaho), permitiendo Modificar, Embeber, Modularizar, el software sin restricciones; dejando al parecer de la organización el aporte o no de los cambios realizados al proyecto.

El esquema de Mondrian es el utilizado para la exploración de cubos Olap en LiteExplorer

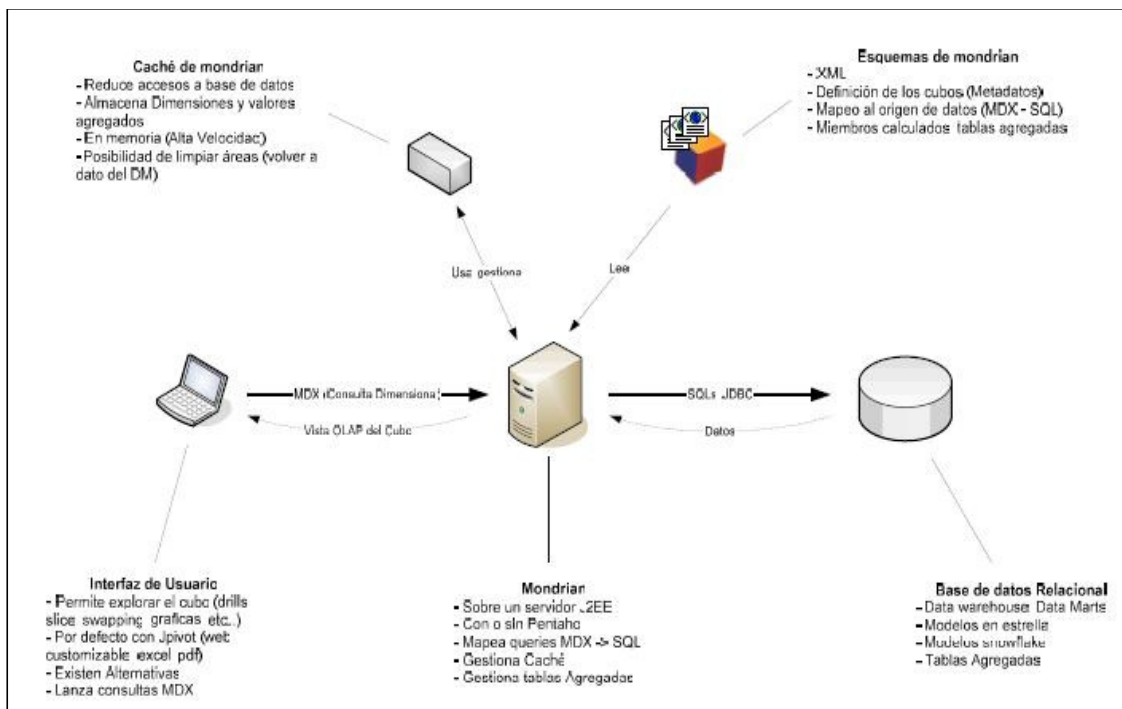


Figura 77: Funcionamiento de Mondrian

Mondrian se encarga de recibir consultas dimensionales (lenguaje MDX) y devolver los datos de un cubo, sólo que este cubo no es algo físico sino un conjunto de metadatos que definen como se han de “mapear” estas consultas que tratan conceptos dimensionales a sentencias SQL ya tratando con conceptos relacionales que obtengan de la base de datos la información necesario para satisfacer la consulta dimensional.

Es utilizado como gestor de metadatos dimensionales para liteSpace.

- **Potente**, rico en características.
- **Utilizado ampliamente** por organizaciones y también, de forma embebida, en productos comerciales, incluso otras alternativas SaaS BI.
- Producto **sólido** y altamente testado (7 años de historia)
- **Independiente** del SGDB
- **Adaptable** a las necesidades concretas de Litebi (Open Source)
- Basado en **estándares** del sector (MDX, XMLA, etc.), posibilidades de interoperación con otros sistemas.

```

<Schema name="T0Name">
  <Dimension type="StandardDimension" name="7" caption="Product">
    <Hierarchy name="11" hasAll="true" allMemberName="Total" primaryKey="BASE_LEVEL_ID" caption="Product">
      <Table name="DIMENSION_7">
        </Table>
        <Level name="10" column="LEVEL_10_ID" nameColumn="LEVEL_10_DESC" ordinalColumn="LEVEL_10_DESC" type="String" uniqueMembers="true" levelType="Regular">
          </Level>
        <Level name="9" column="LEVEL_9_ID" ordinalColumn="LEVEL_9_ID" type="String" uniqueMembers="false" levelType="Regular">
          </Level>
        <Level name="8" column="BASE_LEVEL_ID" nameColumn="BASE_LEVEL_DESC" ordinalColumn="BASE_LEVEL_DESC" type="String" uniqueMembers="true" levelType="Regular">
          </Level>
        </Hierarchy>
      </Dimension>
    <Dimension type="StandardDimension" name="1" caption="Customer">
      <Hierarchy name="6" hasAll="false" primaryKey="BASE_LEVEL_ID" caption="Customer">
        <Table name="DIMENSION_1">
          </Table>
          <Level name="3" column="LEVEL_3_ID" ordinalColumn="LEVEL_3_ID" type="String" uniqueMembers="false" levelType="Regular">
            </Level>
          <Level name="4" column="LEVEL_4_ID" ordinalColumn="LEVEL_4_ID" type="String" uniqueMembers="false" levelType="Regular">
            </Level>
          <Level name="5" column="LEVEL_5_ID" ordinalColumn="LEVEL_5_ID" type="String" uniqueMembers="false" levelType="Regular">
            </Level>
          <Level name="2" column="BASE_LEVEL_ID" nameColumn="BASE_LEVEL_DESC" ordinalColumn="BASE_LEVEL_DESC" type="String" uniqueMembers="true" levelType="Regular">
            </Level>
          </Hierarchy>
        </Dimension>
      <Dimension type="StandardDimension" name="12" caption="Tiempo">
        <Hierarchy name="19" hasAll="true" allMemberName="Total" primaryKey="BASE_LEVEL_ID" caption="Tiempo">
          <Table name="DIMENSION_12">
            </Table>
            <Level name="17" column="LEVEL_17_ID" ordinalColumn="LEVEL_17_ID" type="String" uniqueMembers="false" levelType="Regular">
              </Level>
            <Level name="13" column="BASE_LEVEL_ID" nameColumn="BASE_LEVEL_DESC" ordinalColumn="BASE_LEVEL_DESC" type="String" uniqueMembers="true" levelType="Regular">
              </Level>
            </Hierarchy>
          </Dimension>
        <Dimension type="Cube" name="123" caption="Sales" cache="false" enabled="true">
          <Table name="CUBE_123">
            </Table>
            <DimensionUsage source="7" name="7_1" caption="Product" foreignKey="DIMENSION_7_1_ID">
              </DimensionUsage>
            <DimensionUsage source="1" name="1_2" caption="Customer" foreignKey="DIMENSION_1_2_ID">
              </DimensionUsage>
            <DimensionUsage source="12" name="12_3" caption="Tiempo" foreignKey="DIMENSION_12_3_ID">
              </DimensionUsage>
            <Measure name="126_Sum" column="MEASURE_124" aggregator="sum" caption="Ventas (Sum)" visible="true">
              </Measure>
            </Dimension>
        <Dimension type="Cube" name="136" caption="Nuevo Cubo" cache="false" enabled="true">
          <Table name="CUBE_136">
            </Table>
            <DimensionUsage source="7" name="7_4" caption="Product" foreignKey="DIMENSION_7_4_ID">
              </DimensionUsage>
            <DimensionUsage source="1" name="1_2" caption="Customer" foreignKey="DIMENSION_1_2_ID">
              </DimensionUsage>
            <Measure name="137_Sum" column="MEASURE_137" aggregator="sum" caption="Nueva medida (Sum)" visible="true">
              </Measure>
            </Dimension>
          </Schema>

```

Figura 78: Ejemplo fichero XML con el esquema de Mondrian en Litebi

4 MANUAL DE USUARIO LITEBI

En el presente capítulo se presentan las funcionalidades de Litebi de cara al cliente. Su objetivo es capacitar a usuarios finales para que sean capaces de utilizar y aprovechar todas las funcionalidades que Litebi otorga.

En la presente sección se muestra cómo:

- **Manejar** los menús y ventanas de la aplicación.
- **Crear** informes multidimensionales con **liteExplorer** personalizados de forma sencilla incluyendo gráficas y exportaciones.
- **Crear** cuadros de mando con **liteMonitor** para componer representaciones visuales integrales con diferentes puntos de vista. Ayuda a controlar el funcionamiento de la organización.

4.1 El espacio de solución: liteSpace

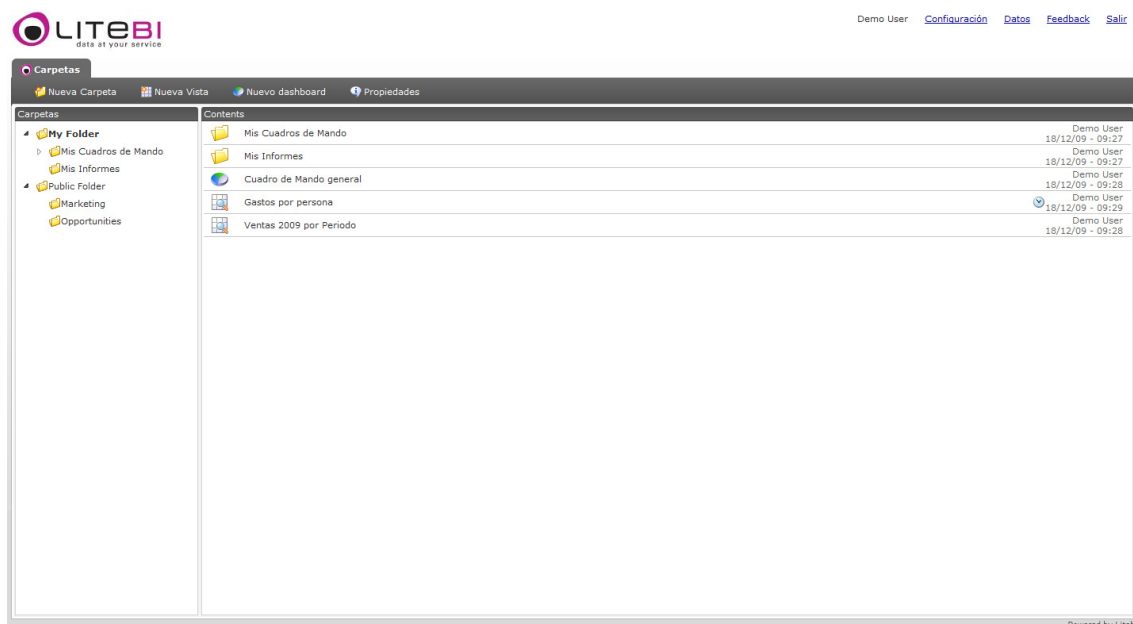


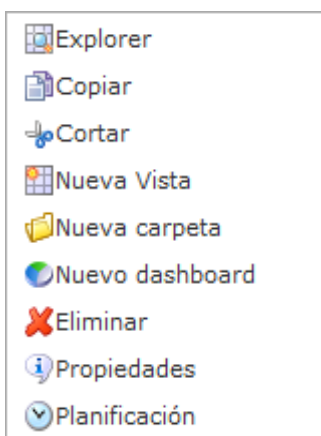
Figura 79: liteSpace, gestión de archivos y carpetas

Cada usuario dispondrá de dos carpetas. Una carpeta privada y una pública.

La carpeta privada será un lugar dónde podrá almacenar sus informes y cuadros de mando personales, que únicamente tendrá acceso el usuario que los haya creado. Será su espacio personal.

La carpeta pública será el lugar dónde se crearán aquellos informes y cuadros de mando que se quieran compartir con el resto de usuarios.

4.1.1 Interacciones



La aplicación de Litebi ha sido diseñada con un estilo Windows. Se podrá interactuar con los elementos de tu espacio con el botón derecho del ratón sobre los elementos.

Al pulsar sobre el espacio vacío podremos crear:

- Nueva vista: Para crear nuevos informes.
- Nueva carpeta: Para crear una nueva carpeta dónde almacenar informes y cuadros de mando.
- Nuevo dashboard (Cuadro de mando): Para crear nuevos cuadros de mando.

Si pulsamos sobre cualquiera de los elementos ya creados se añadirán las siguientes opciones:

- Explorar: En el caso de que el elemento se trate de un informe o de un cuadro de mando, con esa opción pasaremos a explorarlo. Es la misma acción que pulsando con el botón izquierdo sobre el elemento.

- Copiar: Copiará el elemento por si queremos tener una copia en otro lugar.
- Cortar: Cortará el elemento por si queremos mover el elemento a otro lugar.
- Pegar: Pegará el elemento que haya sido previamente copiado o cortado a lugar indicado.
- Eliminar: Eliminará el elemento.
- Propiedades: Abrirá las opciones del elemento, dónde podremos cambiar el nombre y la descripción.
- Planificación: Este es un elemento muy interesante, nos servirá para planificar envíos de los informes a un correo electrónico. Lo explicaremos a continuación.

4.1.2 Planificación de correos

Cómo hemos comentado en el apartado anterior, al pulsar con el botón derecho sobre un informe podremos planificar su envío programado al correo electrónico.

La primera pantalla que nos aparece, que la vemos a continuación, es dónde configuraremos la programación.

Primero deberemos **Añadir** una nueva planificación, le pondremos el nombre y descripción deseada.

En el desplegable **Tarea** elegiremos el tipo de periodicidad que vamos a tratar.

- El caso de **Diariamente** lo vemos en la imagen. Deberemos seleccionar cada cuantos días queremos que se ejecute.

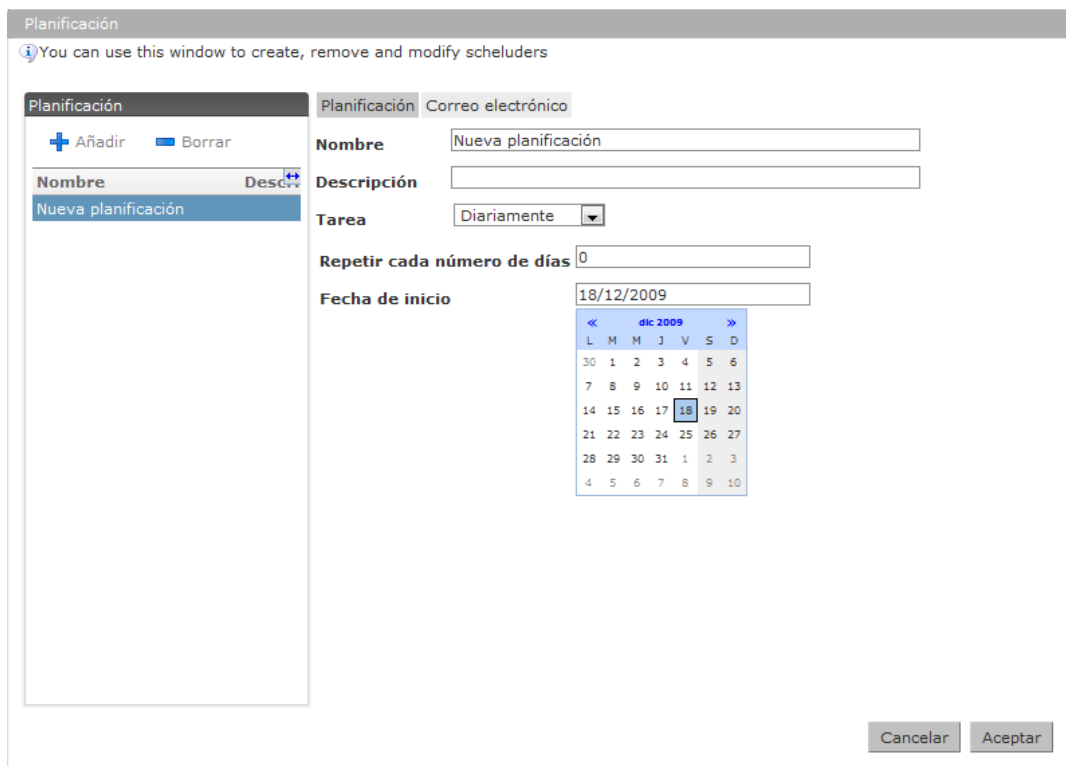
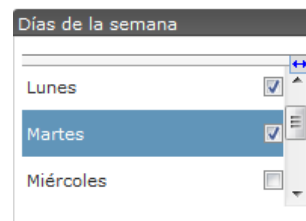
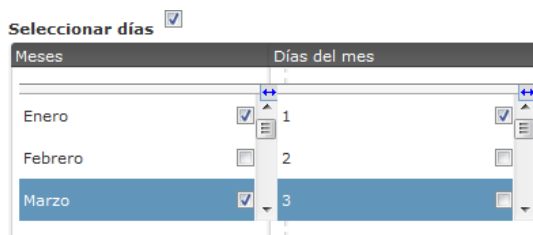


Figura 80: Planificación de informes en Litebi (Planificación)

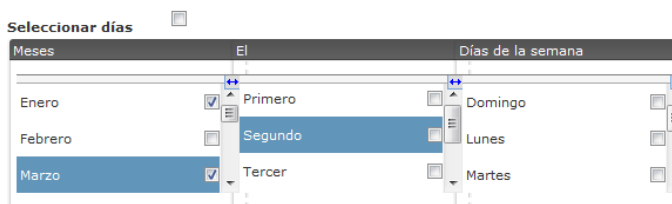
- En el caso **Semanalmente** nos aparecerá unas nuevas opciones para elegir qué días de la semana queremos que se ejecute y cada cuántas semanas.



- En el caso **Mensualmente** nos aparecerá también un nuevo cuadro de selección dónde podremos seleccionar los meses que se ejecutará y en qué días del mes.



- También podremos hacer una selección de días más **avanzada**, personalizando por ejemplo **el segundo domingo** de los meses seleccionados.



En todos los casos deberemos seleccionar el **día de inicio** mediante el calendario, seleccionando el día, o escribiendo el día de inicio de forma manual.

La siguiente pestaña de configuración es la del **Correo electrónico**. En ella seleccionaremos la dirección de correo destino, el asunto del mensaje que se enviará, el cuerpo del mensaje y si queremos el informe en **Excel** o **Pdf**.

Planificación

You can use this window to create, remove and modify schedulers

Planificación | Correo electrónico

+ Añadir - Borrar

Nombre Desc

Nueva planificación

De Litebi

Enviar a dirección ext

Asunto

Cuerpo de mensaje

Enviar excel

Enviar pdf

Cancelar Aceptar

Figura 81: Planificación de informes en Litebi (Envío)

4.2 Construir informes: liteExplorer

La herramienta liteExplorer nos permitirá crear informes avanzados OLAP que permite realizar consultas multidimensionales sobre los cubos construidos y almacenar informes.

El funcionamiento es muy sencillo, al explorar un cubo o reporte (vista de cubo) se abre liteExplorer quedando en el panel izquierdo disponibles las dimensiones que conforman el cubo y que es posible utilizar para el análisis, este panel tiene tres áreas:

- **Filtros:** Dimensión que no se están visualizando en la vista pero sobre las que es posible seleccionar filtros (acotar la información que se ve). *Ej. No deseo ver los años en filas o columnas, pero deseo ver datos únicamente del año 2008*
- **Columnas:** Dimensiones que deseo ver en la vista del cubo cómo columnas.
- **Filas:** Dimensiones que deseo ver en la vista del cubo cómo filas.

Edición de Filtros: Sobre cualquier dimensión es posible activar (mediante botón derecho del ratón) los filtros, pudiendo seleccionar los elementos de la dimensión que quiero utilizar en la vista.

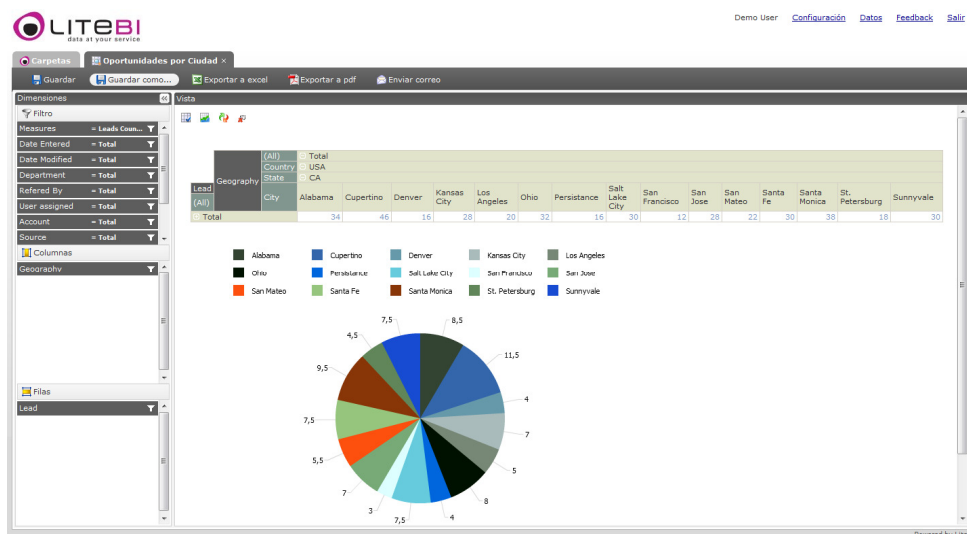


Figura 82: Explorando el cubo de oportunidades (leads) con liteExplorer

4.2.1 Operaciones básicas

Vamos a ver cómo crear un informe multidimensional con liteExplorer. Lo primero que debemos hacer será pulsar sobre el cubo o vista que deseemos explorar. También podemos pulsar con el botón derecho y pinchar sobre la opción Explorar.

The screenshot shows the LITEBI interface with the following components and annotations:

- Métricas:** Points to the 'Measures' section in the sidebar, currently set to '= Leads Count'.
- Dimensiones:** Points to the 'Dimensions' section in the sidebar, which includes 'Geography' and 'Lead'.
- Opciones de Vista:** Points to the 'View Options' section, showing a tree structure for 'Geography' (All, Country, State, City) and 'Lead' (All).
- Dimensiones explorada en profundidad:** Points to the expanded 'City' dimension in the table header.
- Dimension Lead colapsada. + -> Expandir:** Points to the collapsed 'Lead' dimension in the table header.
- La métrica número de leads por ciudad:** Points to the numerical value '16' in the 'Persistence' column of the 'Total' row.
- Elementos a mostrar:** Points to the 'Columns' section in the sidebar, where 'Geography' is selected.
- Dimensiones seleccionadas en Columnas/Filas:** Points to the 'Geography' and 'Lead' dimensions in the sidebar.

| Lead | Geography | Alabama | Cupertino | Denver | Kansas City | Los Angeles | Ohio | Persistence | Salt Lake City | San Francisco | San Jose | San Mateo | Santa Fe | Santa Monica | St. Petersburg | Sunnyvale |
|-------|-----------|---------|-----------|--------|-------------|-------------|------|-------------|----------------|---------------|----------|-----------|----------|--------------|----------------|-----------|
| Total | (All) | 34 | 46 | 16 | 28 | 20 | 32 | 16 | 30 | 12 | 28 | 22 | 30 | 38 | 18 | 30 |

Figura 83: Disposición de los elementos en liteExplorer

Para generar cualquier informe lo que deberemos de hacer será arrastrar las dimensiones deseadas hacia las columnas o filas.

Una vez ya tengamos una dimensión en filas y otra en columnas se mostrará el informe resultante. Ese será un informe básico que podremos personalizar con multitud de opciones.

La selección de elementos a mostrar y filtros los explicaremos en un apartado específico.

Las opciones de vista son las siguientes:

- Opciones de la tabla: aquí podremos personalizar si queremos que la tabla esté visible y el tamaño de la paginación. Esto último es para cuándo existen muchos registros que mostrar seleccionar un máximo con la finalidad de no tener muchos datos a la vista que no nos permitan analizar la situación con claridad.
- Opciones de la gráfica: aquí personalizaremos todo lo referente a la gráfica. Este paso se explicará más adelante.
- Cambiar ejes: aquí cambiará las dimensiones de lugar, las que están en filas pasarán a columnas y viceversa.
- Ocultar vacías: esta opción es muy útil visualmente. Nos eliminará los registros vacíos de la tabla.

4.2.2 Gráficas

Al pulsar sobre opciones de la gráfica nos aparecerá una ventana donde podremos configurar a nuestro gusto el tipo de gráfica que deseemos.

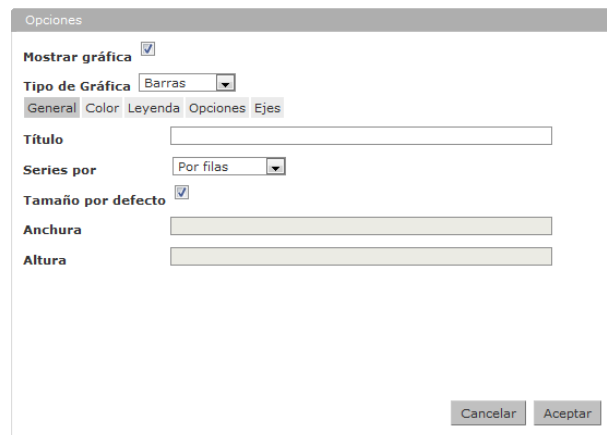


Figura 84: Menú de gráfica

Existen cuatro tipos de gráficas:

- Barras
- Geográfica.
- Líneas.
- Tarta.

En la pestaña general podremos configurar si queremos mostrar la gráfica, el tipo de gráfica, el título, si las series van a ser por filas o por columnas y si queremos el tamaño por defecto o un tamaño personalizado de la gráfica.

Común a todos los tipos de las gráficas serán las opciones de color y de leyenda. Aquí personalizaremos el color de cada una de las series y si deseamos mostrar la leyenda y en qué lugar queremos que aparezca.

La pestaña Opciones y Ejes será distinta dependiendo del tipo de gráfico y nos permitirá personalizar con más detalle los gráficos.

4.2.3 Exportar y Enviar correo

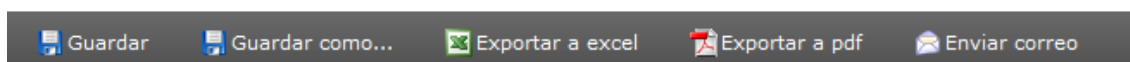


Figura 85: Opciones de envío y exportación

Cuándo tenemos nuestro informe listo podemos guardarlo, exportarlo a Excel, exportarlo a pdf o enviarlo por correo. Estas acciones se realizan mediante la barra superior que vemos en la imagen.

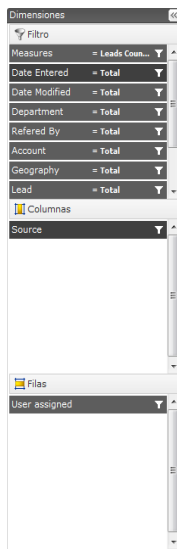
- Guardar y Guardar como...: Con estas acciones guardaremos el informe dentro de liteSpace bien en nuestra carpeta privada o en la carpeta pública.
- Exportar a Excel: al pulsar sobre este botón automáticamente el navegador nos dirá dónde queremos guardar el archivo Excel que se ha generado.

- Exportar a pdf: el comportamiento de la exportación a pdf es la misma que en Excel. El navegador nos pedirá dónde almacenar el fichero pdf con el informe generado.
- Enviar correo: con la opción de enviar correo se nos abrirá una ventana muy similar a la explicada en planificación. Nos permitirá enviar tanto el archivo en Excel cómo en pdf.

4.2.4 Selección de elementos

La selección de los elementos que van a intervenir en los informes se realiza completamente en la columna de la izquierda. Tendremos tres formas de interactuar con la selección de elementos para el informe:

- Selección de dimensiones en filas y columnas



Para la selección de dimensiones se utiliza un sistema de *drag & drop*, arrastrar y soltar. Pulsaremos sobre la dimensión deseada y la arrastraremos hasta la casilla de columnas o filas, según deseemos.

Estamos en un análisis multidimensional por lo que podemos poner más de una dimensión en columnas o filas.

Una vez tengamos una columna y una fila, automáticamente se generará el informe en la parte de la derecha.

- Elementos mostrados de las dimensiones seleccionadas.

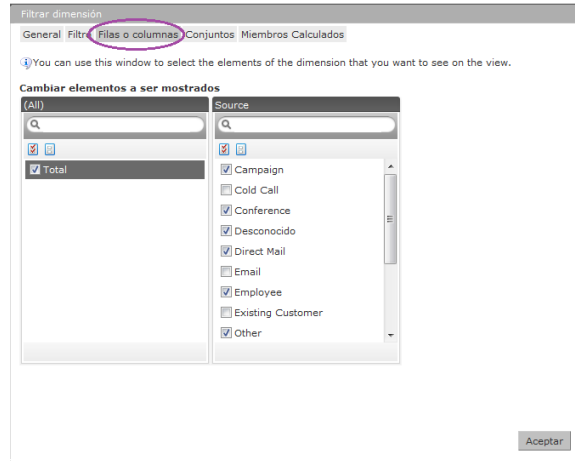


Figura 86: Selección de elementos de una dimensión

Los elementos a mostrar los seleccionaremos de las dimensiones que hemos puesto en filas y columnas pulsando sobre el embudo. En este tipo de filtro no se muestran los datos que no hayamos seleccionado pero no se dejan de tener en cuenta en el análisis.

- Filtros.

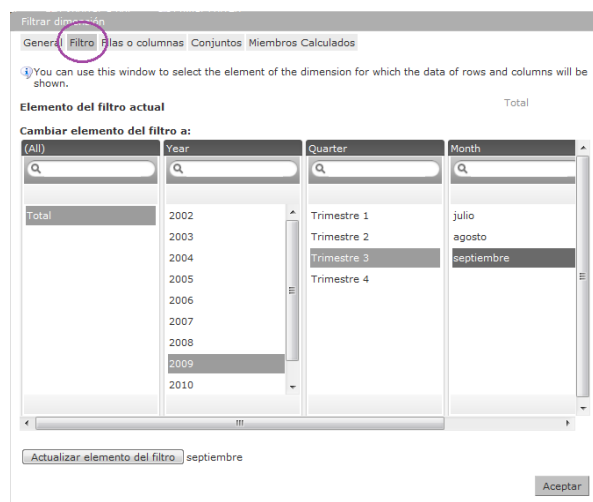


Figura 87: Selección de filtros

Los filtros se pueden seleccionar a cualquier dimensión, pulsando sobre su embudo, estén o no seleccionadas en filas y columnas, ya que de todas formas afectará a los datos.

Por ejemplo podemos hacer un filtrado por fecha sin que la dimensión temporal esté mostrada, ya que los datos que aparezcan en el informe serán únicamente de la fecha que hayamos seleccionado en el filtro. Los datos que filtremos no se tendrán cuenta en el análisis.

4.3 Construir cuadros de mando: liteMonitor

La herramienta que Litebi utiliza para el desarrollo de cuadros de mando es liteMonitor. Nos permitirá componer representaciones visuales integrales con diferentes puntos de vista. Es una herramienta de control.

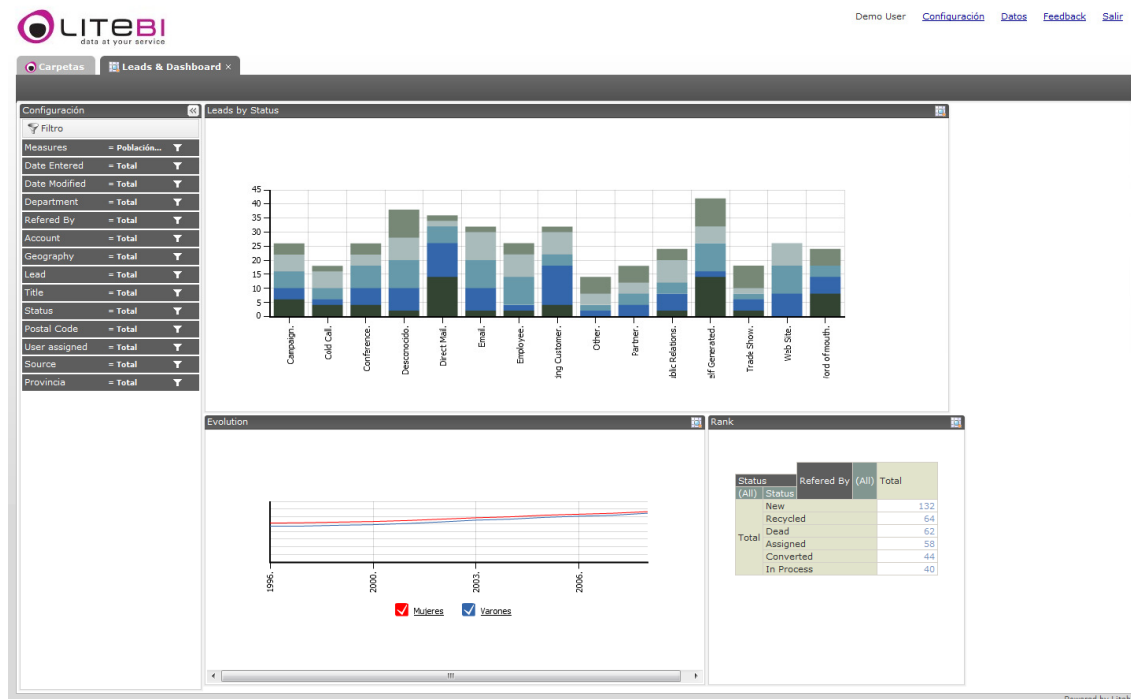


Figura 88: Cuadro de mando con Litebi

4.3.1 Construir la estructura

El primer paso para construir nuestro cuadro de mando es definir la estructura que va a tener. Los cuadros de mando se dividen en columnas y en filas, pero no existen restricciones. Es decir, la primera fila puede ser una columna sólo por si queremos una tabla, la segunda fila puede tener tres columnas dónde colocaremos por ejemplo tres pequeños gráficos. Tiene una flexibilidad total.

Veamos cómo funciona el editor de la estructura.

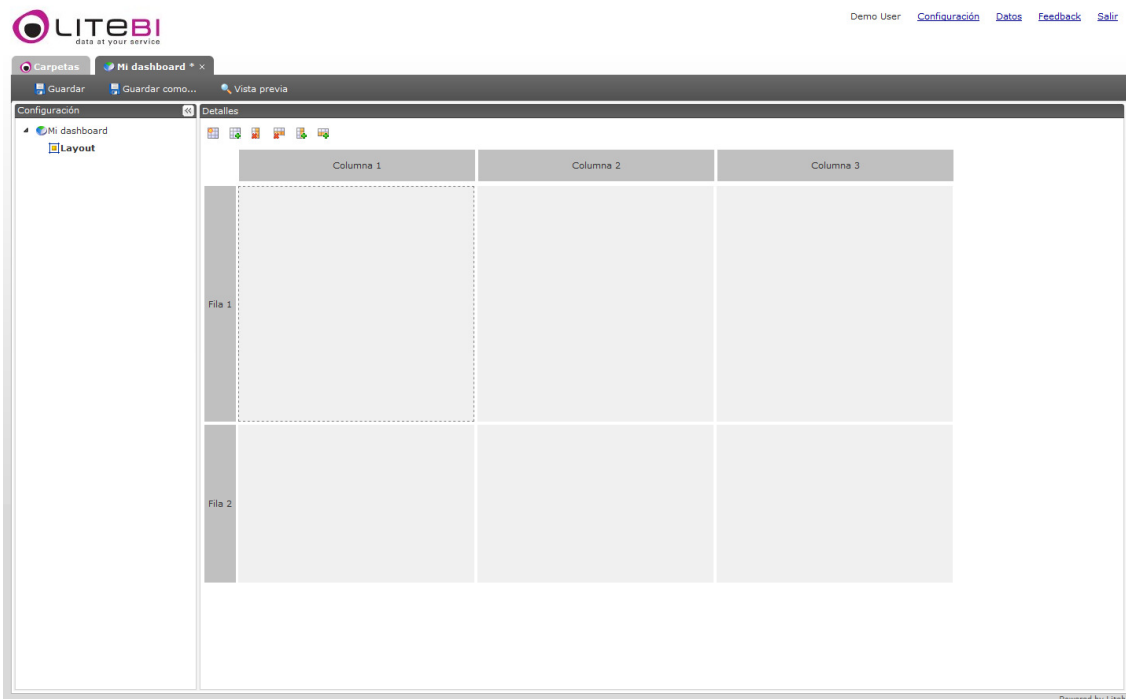


Figura 89: Construyendo un cuadro de mando

Las opciones del editor irán variando dependiendo de la zona que hayamos seleccionado. Una vez hayamos decidido cuanta secciones debe tener el cuadro de mando, nos quedamos con el mayor número de filas y de columnas que vaya a tener, ya que posteriormente podremos unir filas y columnas para que quede unificado y vamos añadiendo o eliminando filas y columnas a nuestro gusto.

Si pulsamos con el botón derecho sobre la cabecera de la Fila o de la Columna podremos modificar también su tamaño

4.3.2 Agregar vistas

Una vez tengamos los tamaños podemos importar vistas ya creadas a nuestro cuadro de mando o crear nuevas. Para ello pulsaremos sobre el espacio que queremos y pulsamos botón derecho. Nos aparecerán las opciones Nueva vista o Importar vista.

Una vez tengamos una vista puesta en su lugar aparecerán nuevas opciones de edición. Podremos tanto mover de sitio la vista por si no estamos contentos con su ubicación actual, cómo expandir la vista. Por ejemplo esto será muy útil para que las tablas ocupen más de una columna.

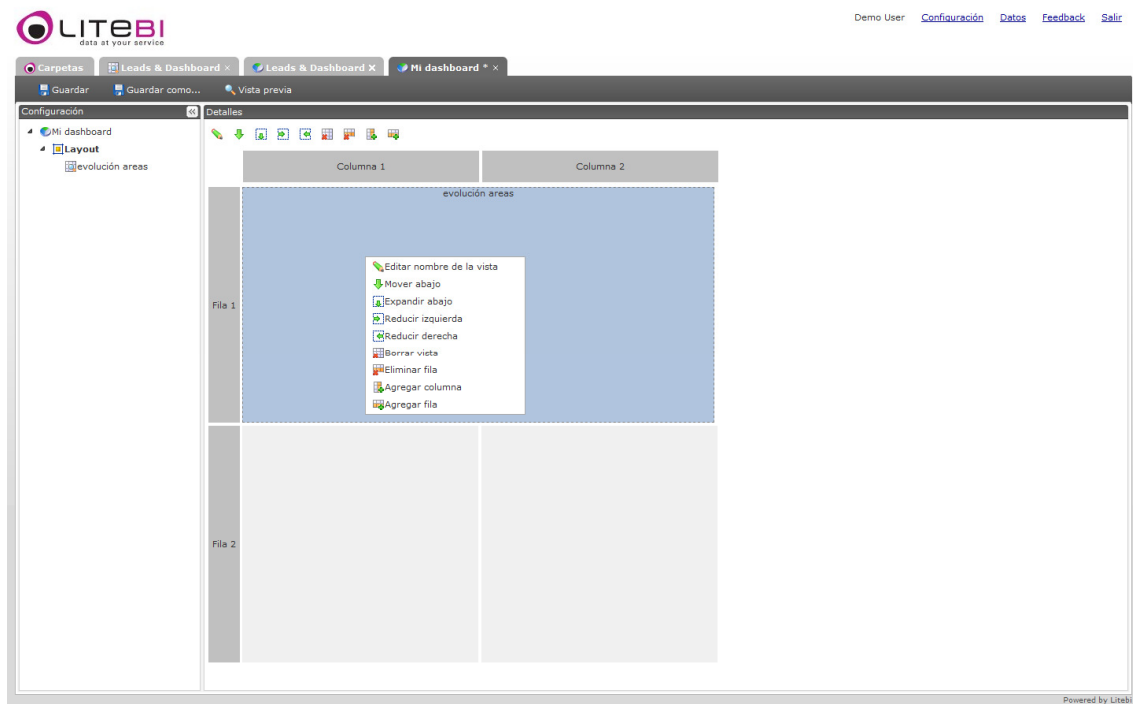


Figura 90: Definiendo la estructura de un cuadro de mando

Cuando ya tengamos nuestra estructura lista con todas las vistas en el sitio que deseamos la guardamos en la carpeta deseada. Podemos realizar una vista previa del cuadro de mando y ver si está de acorde con lo que queremos.

4.3.3 Filtrado y profundización en cuadros de mando

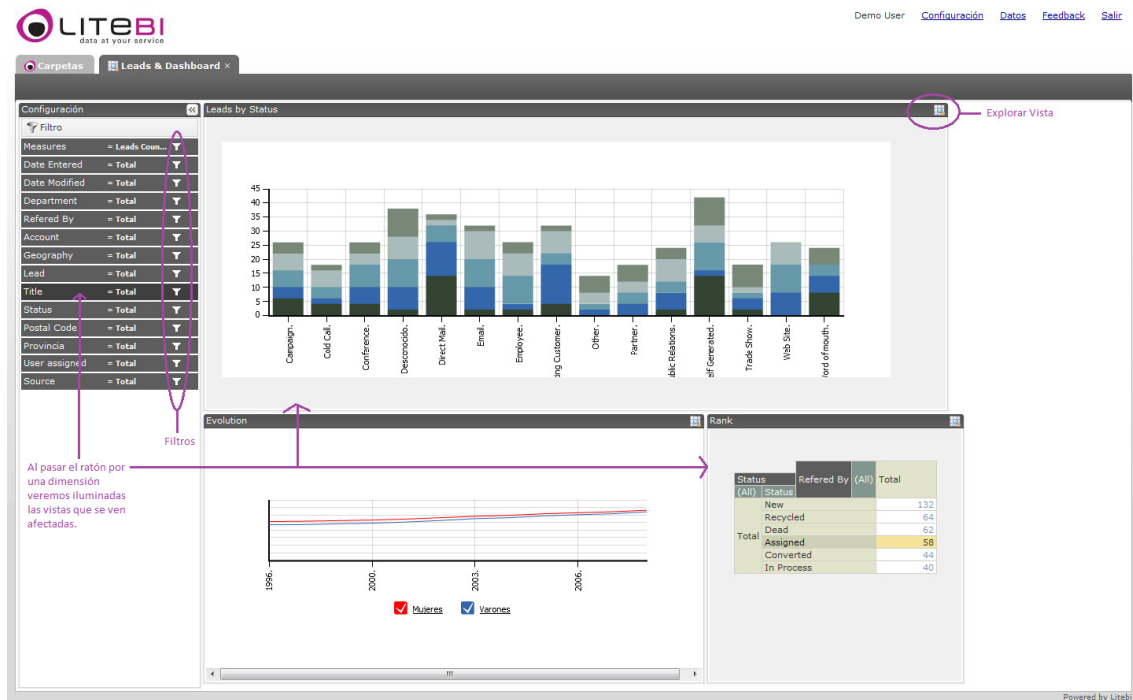


Figura 91: Exploración de un cuadro de mando

Una vez ya tenemos el cuadro de mando finalizado vamos a ver cómo explorarlo. El objetivo de un cuadro de mando es que en un pequeño vistazo veamos cómo está la situación. Es un escenario de control. Pero si vemos algo raro, algo que no está controlado, desde la misma ventana del cuadro de control podemos profundizar.

Existen dos formas de profundizar, mediante los **filtros** que cambiarán los datos que se muestran en el cuadro de mando y mediante la **exploración de la vista** para ya realizar un análisis exhaustivo.

5 FUTUROS MÓDULOS DE LITEBI

En este apartado trataremos de explicar los módulos que se encuentran en desarrollo actualmente para Litebi.

5.1 *liteEureka*

A partir de la información contenida en liteSpace, liteEureka aplica **técnicas de minería de datos** e **inteligencia artificial** para localizar **patrones de información** que puedan ser útiles al usuario. El objetivo es el **descubrimiento automático** de información relevante entre los datos estructurados residentes en liteSpace utilizando las estructuras dimensionales.

liteEureka funciona como un framework para crear **sistemas expertos** que se adapten a las necesidades del usuario (bien entrenándolo, bien definiendo reglas). Será posible crear sistemas expertos verticalizados adaptados a las necesidades de distintos ámbitos (*Ej. Sistema experto de control de calidad en una línea de fabricación*).

El concepto clave en liteEureka es el de “**descubrimiento**” el sistema funciona como un sugeridor/descubridor capaz de adaptarse a las necesidades del cliente.

Se plantea el **desarrollo propio desde cero de liteEureka** apoyándose en librerías y técnicas de IA y Minería de Datos que permita al sistema trasladar al usuario descubrimientos como:

- **Clasificación:** Dadas unas variables determinadas (para un cliente: sexo, edad y región), genera y propone una serie de “arquetipos” para comprender la información (Ej. Un 12% son mujeres entre 25-35 años de Madrid, Un 9% hombres de 40-50 años de Barcelona, etc.)
- **Información potencialmente relevante.** En base a las reglas definidas en el sistema experto mostrar información similar a la información que anteriormente se consideró valiosa.

- **Datos anómalos** (limpieza de datos, detección de casos exóticos). *Ej. El cliente A efectuó una compra 20 veces superior a su media en determinado día, etc.*

5.2 *liteTeam*

Como parte del enfoque de Software como Servicio, Litebi incorpora funcionalidades que optimizan el uso de la aplicación por parte de equipos de trabajo. *liteTeam* aporta:

- **Mail Integrado:** El sistema es capaz de enviar correos electrónicos con **informes adjuntos**. Se puede programar el envío de estos correos o bien enviarlos en el momento del análisis.
- **Alarmas:** Es posible definir Alarmas de forma que, ante determinados eventos (*ej. Un valor supera el umbral indicado*) desencadenen un suceso (*ej. envío de un correo o aviso a un usuario*). Permite controlar la información de forma pasiva.
- **Colaboración:** Es posible agregar comentarios a los elementos de análisis, o llamar la atención de otro usuario sobre datos determinados. (*ej. proceso de revisión de objetivos de ventas en el que el responsable va marcando elementos de datos y los comerciales explican los motivos detrás del valor*)

5.3 *liteConcepts*

Herramienta para el **análisis de los datos no estructurados** (textuales) almacenados en *liteSpace* o existentes en Internet.

LiteConcepts supone la creación un lenguaje de **representación de ontologías** relacionadas con el modelo dimensional almacenado en *liteSpace*. Planteamos un **desarrollo propio** apoyándose en librerías y técnicas de minería web y tecnologías semánticas que permitirá:

- **Relacionar información no estructurada con información estructurada.** Por ejemplo mientras se realiza un análisis OLAP ser capaz de ofrecer un contexto relacionado de Noticias, documentos o webs relacionadas.



- **Vigilar** un determinado **sector** de Internet. El **sector** se define como un conjunto de **conceptos** construidos usando técnicas de **minería web** a partir de un input suministrado por el usuario (keywords, webs, etc.). (*Ej. Vigilancia competitiva del sector del calzado, posibilitando recibir noticias relacionadas, descubrimiento de información relevante, monitorización de webs, etc.*).
- Usar para el proceso de toma de decisiones información proveniente de diferentes **orígenes ya existentes en internet** (Google Trends, Noticias, Google Sets, Digg, Freebase, Technorati, etc.) (*Ej. Uso de noticias de internet, datos internos de ventas, y tendencias detectadas en Google Trends para presentar un Cuadro de Mando sobre tendencias de marketing en un determinado sector*).

6 BENEFICIOS DE LITEBI Y COMPARATIVA CON OTROS SISTEMAS

6.1 Beneficios de Litebi

Podríamos catalogar los beneficios que supone Litebi para un cliente en **tres categorías**:

- **Respecto a no tener sistema de BI:**
 - Posibilita **análisis mucho más potentes** sobre los datos ya existentes de forma sencilla (**OLAP, Cuadros de Mando, Informes, Predicciones**, etc.).
 - **Mejora** el proceso de **toma de decisiones**, haciendo a la **organización** más **eficaz y competitiva**.
 - Facilita el que la **información llegue** a quien la **necesita**.
 - Facilita el que la empresa tenga una “**vista única de la verdad**” más realista y controlable.
- **Respecto a alternativas BI tradicionales:**
 - **Reduce el costo total** de disponer de un sistema de BI, así como la posibilidad de realizar un **gasto controlado** sólo conforme se va considerando **oportuno**.
 - **Reduce el tiempo** que ha de pasar hasta poder comenzar a **analizar los datos**.
 - **Simplifica** enormemente el **despliegue** del sistema de BI: Sin hardware, sin grandes equipos de consultoría, etc.
 - **Simplifica** drásticamente el **mantenimiento**. Actualizaciones y mejoras automáticas y gratuitas.

- Posibilita el **acceso seguro** al análisis de **datos desde cualquier lugar** con acceso a **Internet**.
- **Respecto a otras alternativas BI + SaaS:**
 - **Realmente es BI + SaaS**. Creado desde 0 para ser SaaS, **no reciclado** a partir de un producto preexistente.
 - El **mejor modelo de carga y gestión de datos** del mercado. La velocidad y potencia de los despliegues nos permitirá ofrecer menores inversiones iniciales y más soluciones paquetizadas.
 - **Autonomía total** por parte del cliente, en caso de que lo desee, para construirse su propio sistema BI.
 - **Basado** en tecnología **Google** con las garantías de calidad, escalabilidad y permanencia que supone.
 - **Pago por uso**, atractivo para los clientes.
 - Posibilidad de **integrar datos** provenientes de prácticamente cualquier origen de datos.
 - Posibilidad de **crear** de forma sencilla y **en función** de las **necesidades** todos los **elementos de análisis** que se necesiten.

6.2 Comparativa con la competencia

Los principales competidores / productos sustitutos que se han identificado son:

- **Excel**: Uso intensivo de excel para el análisis.
- **Informes**: Reportes desarrollados y mantenidos por un departamento de informática o incluidos en el ERP.



- **QlikView:** Producto de BI ágil de origen Sueco con rápido crecimiento.
- **Microsoft:** Sobre todo la parte de Analysis Services de SQL Server y módulos de su ERP Navision cómo “Business Analytics”. Por lo general no tienen costes de licencia.
- **Pentaho/Open Source:** Plataforma de Business Intelligence de software libre, por lo tanto sin coste de licencias.
- **Plataforma tradicional:** Plataformas cómo IBM-Cognos, SAP-Business Objects o Microstrategy.

Estos son algunos de los mensajes en los que Litebi se apoya para hacer frente a cada uno de estos competidores:

6.2.1 Excel

- “En Business Intelligence hablamos del “infierno de excel” o “excel-hell” para referirnos a lo complicado que es mantener y fiarse de sistemas de análisis basados en excel”.
- “Los sistemas basados en excel son un punto de partida habitual y muy bueno para implantar Litebi”.
- “Con Excel hay que dedicar mucho tiempo a preparar la información, con Litebi ahorras ese tiempo que puedes dedicarlo a tareas de mayor valor añadido”.
- “Con Litebi vas a poder automatizar todo el proceso desde recoger el datos más actualizado hasta recibirlo en forma de Informe en PDF en un correo electrónico todos los días o semanas”.

6.2.2 Informes

- “Los informes no dan flexibilidad ninguna y es necesario construir uno para cada necesidad concreta con Litebi el usuario puede hacerse sus propios informes en segundos y responder a preguntas de negocio de forma interactiva”.
- “Los informes a medida dan información de detalle, no global o histórica, y tampoco pueden utilizar por lo general información de diferentes orígenes de datos”.
- “El departamento de informática puede estar empleando mucho tiempo en desarrollar y mantener sistemas de informes, tiempo que podría emplear en tareas de mayor valor añadido”.
- “Los sistemas basados en informes son un punto de partida habitual y muy bueno para implantar Litebi”.

6.2.3 QlikView

- “QlikView tiene calidad visual muy grande pero carece de la potencia y el enfoque empresarial de otras soluciones cómo Litebi, por ejemplo que varias personas puedan compartir sus informes en un portal web de forma sencilla o disponer de sistemas de mantenimiento y carga de datos potentes y automáticos”.
- “Somos más rápidos en el despliegue y considerablemente más fáciles de mantener que un sistema basado en QlikView”.
- “QlikView arrastra una tecnología de escritorio debido a los 14 años que lleva en el mercado Litebi es un producto pensado para la web lo cual nos permite ofrecer más valor a un precio menor, gracias al modelo de Software como servicio”.
- “En definitiva, creemos que QlikView nos supera en atractivo visual pero que nuestro producto ofrece más valor, mejor tecnología y más potencia a menor precio y con menos quebraderos de cabeza. Nosotros somos una solución más seria y potente”.

6.2.4 Microsoft

- “Microsoft no dispone de un espacio web donde compartir informes, análisis, planificar el envío de informes por mail o desarrollar cuadros de mando, suele ser necesario adquirir otra tecnología para complementar los “cubos” de Analysis Services”.
- “Microsoft puede estar bien si todo lo que tienes es Microsoft, los problemas comienzan cuando quieres utilizarlo con otras tecnologías, Litebi es una plataforma abierta y agnóstica”.

6.2.5 Pentaho / Open Source

- “Conocemos muy bien los sistemas Open Source y sabemos que, al final, pese a la ausencia de costes de licencia, el coste de despliegue, de mantenimiento y de hardware es tremendo y la calidad del sistema resultante muy baja”.
- “Al final los sistemas de software libre tienen alguna ventaja para grandes corporaciones donde pueden preocuparse de la tremenda complejidad de implantar algo así y de las adaptaciones necesarias, pero para una Pyme es un despilfarro y una garantía de insatisfacción, lo hemos visto en muchos casos”.
- “Los sistemas de software libre sólo tienen bajo licencia gratuita las partes más básicas del sistema, pero en cuanto quieres hacer algo que en Litebi está incluido de serie, te exigen pagar licencias adicionales que son considerablemente caras”.

6.2.6 Plataforma tradicional

- “Los productos tradicionales cómo Cognos, Business Objects o Microstrategy, son plataformas fantásticas a las que nos gusta parecernos en el sentido de la potencia y la funcionalidad que tienen pero en una empresa de tamaño mediano es una inversión tremenda y matar moscas a cañonazos”.



- “Litebi ofrece soluciones que van al núcleo del problema y que ofrecen la misma funcionalidad que una solución tradicional a una fracción del coste y sin necesidad de preocuparse de grandes despliegues o mantenimientos”.
- “Creemos honestamente que tenemos la mejor propuesta en Calidad/Precio del mercado”.



ANEXO 2: USO DE IMAGEN CORPORATIVA

La persona D. Javier Giménez Aznar con NIF 20445537D en representación de la empresa LITE INTERNET SOLUTIONS S.L autoriza mediante este documento a D. Rafael Matamoros Zapata con NIF 44524528-R el uso de la imagen corporativa de la empresa (logo, nombre y otros datos) así como toda la información que aparece asociada a su empresa (productos, información adicional) a lo largo del presente proyecto de final de carrera.

En Valencia, 10 de Junio de 2010

DIRECTOR DE DESARROLLO
DE NEGOCIO LITEBI

D.JAVIER GIMÉNEZ AZNAR

EL ALUMNO

RAFAEL MATAMOROS ZAPATA

La persona D. Pablo Catalá Gregori con NIF 22578675N en representación de la empresa CECAV autoriza mediante este documento a D. Rafael Matamoros Zapata con NIF 44524528-R el uso de la imagen corporativa de la empresa (logo, nombre y otros datos) así como toda la información que aparece asociada a su empresa (productos, información adicional) a lo largo del presente proyecto de final de carrera.

En Valencia, 10 de Junio de 2010

DIRECTOR DE CECAV

D. PABLO CATALÁ GREGORI

EL ALUMNO

RAFAEL MATAMOROS ZAPATA



UNIVERSIDAD
POLITECNICA
DE VALENCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica



ÍNDICE DE CONTENIDO

| | |
|---|-----------|
| 1 INTRODUCCIÓN | 7 |
| 1.1 Introducción y motivaciones | 7 |
| 1.2 Objetivos | 9 |
| 1.3 Contenido del proyecto | 10 |
| 1.4 Actividad de la empresa, metodología de trabajo y recursos a utilizar | 11 |
| 1.4.1 Actividad de la empresa | 11 |
| 1.4.2 Metodología de trabajo y recursos a utilizar | 12 |
| 1.4.3 Planificación temporal de las actividades | 14 |
| 2 LA INTELIGENCIA DE NEGOCIO | 15 |
| 2.1 Introducción a la Inteligencia de Negocio | 15 |
| 2.1.1 ¿Qué es la Inteligencia de Negocio? | 16 |
| 2.1.2 La Inteligencia de Negocio: Cifras y Beneficios | 18 |
| 2.1.3 Tipos de usuarios BI | 21 |
| 2.2 Herramientas y técnicas | 23 |
| 2.2.1 OLTP (On-line Transaction Processing) | 23 |
| 2.2.2 OLAP (On-line Analytical Processing) | 23 |
| 2.2.3 Query & Reporting | 29 |
| 2.2.4 EIS y DSS | 29 |
| 2.2.5 Cuadro de mando integral | 30 |
| 2.2.6 Data Mining | 32 |
| 2.2.7 CRM | 33 |
| 2.2.8 BPM y CPM | 34 |
| 2.2.9 CIF y GIF | 34 |
| 2.2.10 Gestión del conocimiento | 35 |
| 2.3 El Data warehouse | 35 |
| 2.3.1 Introducción | 35 |
| 2.3.2 Definiciones | 36 |
| 2.3.3 Modelos de tablas | 38 |



| | | |
|-------|---|-----------|
| 2.3.4 | Procesos ETL | 39 |
| 2.4 | El Sistema informacional | 41 |
| 3 | CLOUD COMPUTING | 43 |
| 4 | LA EMPRESA CLIENTE: CECAV | 46 |
| 4.1 | Introducción | 46 |
| 4.2 | Historia | 46 |
| 4.3 | Estructura de la empresa | 47 |
| 4.4 | Actividades de la empresa | 48 |
| 5 | PROYECTO BI CECAV: MAPA SANITARIO | 49 |
| 5.1 | Introducción | 49 |
| 5.2 | Situación previa a la solución | 49 |
| 5.3 | Objetivos de la solución | 49 |
| 5.4 | Solución propuesta | 50 |
| 5.5 | Planificación | 51 |
| 6 | ANÁLISIS DE REQUERIMIENTOS | 52 |
| 6.1 | Toma de requerimientos | 54 |
| 6.1.1 | Requerimientos genéricos | 54 |
| 6.1.2 | Requerimientos funcionales | 55 |
| 6.1.3 | Requerimientos de Datos | 56 |
| 6.1.4 | Requerimientos de Negocio | 56 |
| 6.1.5 | Fallos comunes en las soluciones de Business Intelligence | 57 |
| 6.2 | Estado previo de los sistemas de información | 58 |
| 6.2.1 | Origen de datos 1 | 59 |



| | |
|--|------------|
| 6.2.2 Origen de datos 2 | 60 |
| 6.2.3 Origen de datos 3 | 61 |
| 6.3 Generación previa de informes y problemas | 63 |
| 6.4 Necesidades de información | 64 |
| 7 DISEÑO DE LA SOLUCIÓN | 67 |
| 7.1 Objetivos y beneficios | 67 |
| 7.2 Primer diseño | 69 |
| 7.3 Diseño definitivo | 72 |
| 7.3.1 Resultados Origen 1 | 73 |
| 7.3.2 Resultados Origen 2 | 76 |
| 7.3.3 Resultados Origen 3 | 78 |
| 7.3.4 Resultados Origen 1 Global | 80 |
| 8 IMPLEMENTACIÓN DE LA SOLUCIÓN | 82 |
| 8.1 Pentaho™ Data Integration (Kettle) | 82 |
| 8.2 Procesos ETL | 82 |
| 8.2.1 ETLs Dimensiones Compartidas | 82 |
| 8.2.2 ETL Resultados Origen 1 | 90 |
| 8.2.3 ETL Resultados Origen 2 | 91 |
| 8.2.4 ETL Resultados Origen 3 | 93 |
| 8.2.5 ETL Resultados Origen 1 Global | 95 |
| 8.3 Codificación geográfica | 97 |
| 8.4 Procesos ETL Incrementales | 99 |
| 8.5 Cargas periódicas | 101 |
| 8.6 Miembros calculados y conjuntos | 104 |
| 9 RESULTADOS DE LA SOLUCIÓN | 108 |

| | |
|--|------------|
| 9.1 Informes | 108 |
| 9.2 Usuarios y seguridad | 111 |
| 9.3 Planificación de Informes | 113 |
| 10 CONCLUSIONES | 114 |
| 11 BIBLIOGRAFÍA | 115 |
| ANEXO 1: PLATAFORMA BI SAAS, LITEBI | 118 |
| 1 MÓDULOS Y CARACTERÍSTICAS BÁSICAS | 118 |
| 1.1 liteSpace | 118 |
| 1.2 liteIntegrator | 121 |
| 1.3 Módulos de análisis | 122 |
| 1.3.1 liteExplorer | 122 |
| 1.3.2 liteMonitor | 124 |
| 1.4 Características de la plataforma | 126 |
| 2 DESARROLLO DE SOLUCIONES CON LITEBI | 129 |
| 2.1 Definir: liteSpace | 129 |
| 2.1.1 Definiendo un cubo | 131 |
| 2.1.2 Definiendo una dimensión compartida | 134 |
| 2.2 Cargar: liteIntegrator | 137 |
| 2.2.1 Instalación de liteIntegrator | 138 |
| 2.2.2 Funcionamiento de liteIntegrator | 138 |
| 2.2.3 Paso de Output a Litebi | 139 |
| 2.3 Análisis: liteExplorer y liteMonitor | 140 |
| 2.3.1 liteExplorer | 140 |
| 2.3.2 liteMonitor | 142 |



| | | |
|----------|--|------------|
| 3 | TECNOLOGÍAS UTILIZADAS EN LITEBI | 144 |
| 3.1 | Servidor de mapas | 144 |
| 3.2 | Mondrian | 146 |
| 4 | MANUAL DE USUARIO LITEBI | 149 |
| 4.1 | El espacio de solución: liteSpace | 149 |
| 4.1.1 | Interacciones | 150 |
| 4.1.2 | Planificación de correos | 151 |
| 4.2 | Construir informes: liteExplorer | 154 |
| 4.2.1 | Operaciones básicas | 155 |
| 4.2.2 | Gráficas | 156 |
| 4.2.3 | Exportar y Enviar correo | 157 |
| 4.2.4 | Selección de elementos | 158 |
| 4.3 | Construir cuadros de mando: liteMonitor | 160 |
| 4.3.1 | Construir la estructura | 161 |
| 4.3.2 | Agregar vistas | 162 |
| 4.3.3 | Filtrado y profundización en cuadros de mando | 163 |
| 5 | FUTUROS MÓDULOS DE LITEBI | 164 |
| 5.1 | liteEureka | 164 |
| 5.2 | liteTeam | 165 |
| 5.3 | liteConcepts | 165 |
| 6 | BENEFICIOS DE LITEBI Y COMPARATIVA CON OTROS SISTEMAS | 167 |
| 6.1 | Beneficios de Litebi | 167 |
| 6.2 | Comparativa con la competencia | 168 |
| 6.2.1 | Excel | 169 |
| 6.2.2 | Informes | 170 |
| 6.2.3 | QlikView | 170 |



| | |
|------------------------------|-----|
| 6.2.4 Microsoft | 171 |
| 6.2.5 Pentaho / Open Source | 171 |
| 6.2.6 Plataforma tradicional | 171 |

| | |
|---|------------|
| ANEXO 2: USO DE IMAGEN CORPORATIVA | 173 |
|---|------------|

ÍNDICE DE FIGURAS

| | |
|--|----|
| Figura 1: Extracción del conocimiento a partir de los datos | 7 |
| Figura 2: Logo de Litebi | 11 |
| Figura 3: Esquema de una solución de BI | 18 |
| Figura 4: Toma de decisiones de acuerdo a un tipo de usuario | 22 |
| Figura 5: Ejemplo de Cubo OLAP | 24 |
| Figura 6: Sistema M-OLAP | 24 |
| Figura 7: Sistema R-OLAP | 25 |
| Figura 8: Sistema H-OLAP | 25 |
| Figura 9: Sistema D-OLAP | 26 |
| Figura 10: Ejemplo de esquema en estrella | 27 |
| Figura 11: Ejemplo de análisis OLAP | 28 |
| Figura 12: Perspectivas de un cuadro de mando integral | 31 |
| Figura 13: Ejemplo de modelo en estrella | 38 |
| Figura 14: Etapas de una solución de BI | 40 |
| Figura 15: Estructura de una aplicación de BI según Pentaho | 42 |
| Figura 16: Esquema de computación en la nube | 45 |
| Figura 17: Logo del CECAV | 46 |
| Figura 18: Aplicación de análisis de muestras | 60 |
| Figura 19: Exportación a Excel de datos de análisis de muestras | 61 |
| Figura 20: Base de datos Origen 3 Parte 1 | 62 |
| Figura 21: Base de datos Origen 3 Parte 2 | 62 |
| Figura 22: Ejemplo de Mapa Sanitario | 63 |
| Figura 23: Modelo estrella de la primera solución | 71 |
| Figura 24: Modelo estrella del diseño del cubo para Resultados Origen 1 | 73 |
| Figura 25: Estructura del cubo para Resultados Origen 1 | 75 |
| Figura 26: Modelo estrella del diseño del cubo para Resultados Origen 2 | 76 |
| Figura 27: Estructura del cubo para Resultados Origen 2 | 77 |
| Figura 28: Modelo estrella del diseño del cubo para Resultados Origen 3 | 78 |
| Figura 29: Estructura del cubo para Resultados Origen 3 | 79 |
| Figura 30: Modelo estrella del diseño del cubo para Resultados Origen 1 Global | 80 |
| Figura 31: Estructura del cubo para Resultados Origen 1 Global | 81 |
| Figura 32: Proceso ETL de la Dimensión Tiempo | 83 |



| | |
|--|-----|
| Figura 33: Carga a Litebi de la Dimensión Tiempo | 85 |
| Figura 34: Proceso ETL de la Dimensión Determinación | 85 |
| Figura 35: Carga a Litebi de la Dimensión Determinación | 86 |
| Figura 36: Proceso ETL de la Dimensión Registros | 87 |
| Figura 37: Carga a Litebi de la Dimensión Registros | 89 |
| Figura 38: Proceso ETL del Cubo Resultados Origen 1 | 90 |
| Figura 39: Carga a Litebi del Cubo Resultados Origen 1 | 91 |
| Figura 40: Proceso ETL del Cubo Resultados Origen 2 | 91 |
| Figura 41: Carga a Litebi del Cubo Resultados Origen 2 | 93 |
| Figura 42: Proceso ETL del Cubo Resultados Origen 3 | 93 |
| Figura 43: Carga a Litebi del Cubo Resultados Origen 3 | 94 |
| Figura 44: Proceso ETL del Cubo Resultados Origen 1 Global | 95 |
| Figura 45: Carga a Litebi del Cubo Resultados Origen 1 Global | 97 |
| Figura 46: Tabla Layers de la base de datos de Litebi-Geoserver | 98 |
| Figura 47: Tabla Polygons de la base de datos de Litebi-Geoserver | 98 |
| Figura 48: Tabla Padre_Hijo de la base de datos de Litebi-Geoserver | 99 |
| Figura 49: Job de carga de dimensiones compartidas | 102 |
| Figura 50: Job de carga de cubos OLAP | 102 |
| Figura 51: Job de carga de información general | 102 |
| Figura 52: Tarea programada automática li_run | 104 |
| Figura 53: Miembros calculados de la solución en Litebi | 104 |
| Figura 54: Conjuntos de la solución en Litebi | 106 |
| Figura 55: Informe Número de registros por Comarca | 109 |
| Figura 56: Mapa de número de registros por Comarca | 110 |
| Figura 57: Informe Diferentes indicadores por Provincia | 110 |
| Figura 58: Gráfica de media del Títer en Provincias de la Comunidad Valenciana | 111 |
| Figura 59: Usuarios y Roles de la solución | 112 |
| Figura 60: liteSpace (Sistema de archivos y carpetas) | 120 |
| Figura 61: liteSpace (Editor de cubos) | 120 |
| Figura 62: liteIntegrator | 122 |
| Figura 63: liteExplorer | 124 |
| Figura 64: liteMonitor (Gestor de diseño) | 125 |
| Figura 65: liteMonitor (Visualización) | 126 |
| Figura 66: Uso módulos de Litebi | 127 |



| | |
|---|-----|
| Figura 67: Explorando un cubo con liteExplorer | 131 |
| Figura 68: Creando un nuevo cubo en liteSpace | 131 |
| Figura 69: Editando un cubo | 133 |
| Figura 70: Creando una dimensión | 135 |
| Figura 71: Creando niveles de una dimensión | 137 |
| Figura 72: Paso de Output a Litebi (Kettle) | 140 |
| Figura 73: liteExplorer, Visor Olap de Litebi | 141 |
| Figura 74: Creación de cuadros de mando | 142 |
| Figura 75: Diagrama servicios de Litebi | 143 |
| Figura 76: Infraestructura servidores de mapas | 144 |
| Figura 77: Funcionamiento de Mondrian | 147 |
| Figura 78: Ejemplo fichero XML con el esquema de Mondrian en Litebi | 148 |
| Figura 79: liteSpace, gestión de archivos y carpetas | 149 |
| Figura 80: Planificación de informes en Litebi (Planificación) | 152 |
| Figura 81: Planificación de informes en Litebi (Envío) | 153 |
| Figura 82: Explorando el cubo de oportunidades (leads) con liteExplorer | 154 |
| Figura 83: Disposición de los elementos en liteExplorer | 155 |
| Figura 84: Menú de gráfica | 156 |
| Figura 85: Opciones de envío y exportación | 157 |
| Figura 86: Selección de elementos de una dimensión | 159 |
| Figura 87: Selección de filtros | 159 |
| Figura 88: Cuadro de mando con Litebi | 160 |
| Figura 89: Construyendo un cuadro de mando | 161 |
| Figura 90: Definiendo la estructura de un cuadro de mando | 162 |
| Figura 91: Exploración de un cuadro de mando | 163 |