Introducción al Business Intelligence

Introducción al Business Intelligence

Jordi Conesa Caralt (coord.)
Josep Curto Díaz

Diseño de la colección: Editorial UOC

Primera edición en lengua castellana: mayo, 2010
Primera reimpresión: febrero, 2011
Segunda reimpresión: octubre, 2011

© Josep Curto Díaz y Jordi Conesa i Caralt, del texto.
© Imagen de la portada: Istockphoto
© Editorial UOC, de esta edición
Rambla del Poblenou 156, 08018 Barcelona
www.editorialuoc.com

Realización editorial: El Ciervo 96, S.A.
Impresión:

ISBN: 978-84-9788-886-8
Dipósito legal B.

Ninguna parte de esta publicación, incluyendo el diseño general y el de lacubierta, puede ser copiada, reproducida, almacenada o transmitida de ningún modo ni a través de ningún medio, ya sea electrónico, químico, mecánico, óptico, de grabación, de fotocopia o por otros métodos sin la previa autorización por escrito de los titulares del *copyright*.

Coordinador

Jordi Conesa Caralt
Doctor en Informática por la Universidad Politécnica de Catalunya desde 2008 y vinculado al mundo universitario desde el 2001. Es profesor en la Universitat Oberta de Catalunya desde el 2008, concretamente en los Estudios de Informática, Multimedia y Telecomunicación. Actualmente realiza docencia en asignaturas de bases de datos, trabajos de final de carrera y es el director académico del Máster de Business Intelligence. Sus intereses de investigación caen dentro de los ámbitos del modelado conceptual, las ontologías y la Web Semántica. Profesionalmente, antes de su vida laboral en la universidad, trabajó como programador, analista y jefe de proyectos de aplicaciones web.

Autor

Josep Curto Díaz
Licenciado en Matemáticas por la Universidad Autónoma de Barcelona en el año 2000. Actualmente es Senior Research Analyst en IDC Research Spain. Como analista de IDC cubre el mercado tecnológico en España analizando la evolución de dicho mercado y colaborando en la realización de modelos, previsiones y tendencias. Además también colabora en actividades con medios de comunicación.
Es Master en Dirección de Empresas (International Executive MBA) en IE Business School y Master en Business Intelligence y Master en Dirección y Gestión en Sistemas y Tecnologías de la Información por la UOC.
Ha conjugado su carrera profesional en el ámbito de las tecnologías de la información, en particular en el área del Business Intelligence, con una clara vocación por educación superior siendo profesor en la Universidad Autónoma de Barcelona (UAB) y en la Universitat Oberta de Catalunya (UOC). En el ámbito de la divulgación conviene destacar artículos en BARC Guide, la revista Gestión del Rendimiento, autor del blog Information Management y participaciones como experto para BeyeNETWORK Spain.

Quiero agradecer especialmente a Jordi Conesa por sus inestimables comentarios para la creación de este libro que han mejorado de forma notable su contenido y por la oportunidad de llevar este proyecto a buen puerto. Su aportación facilita, sin duda alguna, la lectura de este libro.

Por otra parte, también quiero agradecer profundamente a Laura Gonzalez su constante apoyo, su gran paciencia y su infinita comprensión. Todas ellas brindadas durante la preparación de cada uno de los capítulos de este libro que nos han robado muchas noches y tantos fines de semana. Es un pilar para todos aquellos proyectos que inicio.

Por último, para mi abuela Pilar Bel, que me enseñó tantas cosas buenas y que siempre tendré en el recuerdo.

Índice

Prólogo

Primero fueron los datos...

Los que hemos nacido en la década de 1970, o en décadas anteriores, aún recordamos cuando las empresas utilizaban libretas, es decir, soporte en papel, para almacenar los datos operativos de sus negocios. En esas libretas se apuntaban las ventas realizadas, los gastos de la empresa, los datos de los clientes... Es cierto que había empresas que utilizaban sistemas informáticos para su gestión, pero la compra y el mantenimiento de dichos sistemas sólo estaba al alcance de las grandes compañías. Afortunadamente, con la aparición de la informática personal el uso de programas informáticos de gestión pasó a ser algo común y a estar al alcance de cualquier empresa. Hoy en día podemos decir que cualquier empresa utiliza programas informáticos para la gestión de los datos de su explotación diaria: compras, ventas, gastos, gestión de clientes... Pero ¿cuál es el siguiente paso?

... después vino la información.

Prácticamente todas las empresas de la actualidad disponen de bases de datos que almacenan datos sobre sus actividades y sus colaboradores (clientes, proveedores...) mediante distintos programas informáticos (programas de contabilidad, de facturación, de gestión de clientes, etc.). Por lo tanto, podemos decir que las empresas disponen, por norma general, de multitud de datos históricos, fiables y rigurosos de todas las actividades realizadas. Es lógico pensar que dichos datos podrían ser refinados, agrupados, tratados y analizados para intentar extraer información que permitiera ayudar en la toma de decisiones de la empresa. Encontrar patrones de conducta en la compra de nuestros clientes, presentar información en tiempo real sobre el rendimiento de las distintas sucursales de una empresa a su dirección, o identificar los clientes que no nos son rentables (su coste de gestión es superior al beneficio que dejan) son ejemplos que muestran qué se podría obtener a partir de los datos de la empresa. Este hecho, la conversión de los

datos operativos de las empresas en información que dé soporte a la toma de decisiones, es lo que se conoce como inteligencia de negocio o Business Intelligence (BI de aquí en adelante).

El objetivo de este libro es introducir al lector en el mundo de la inteligencia de negocio. En el libro se explican los principales conceptos, técnicas y tecnologías utilizadas en los procesos de BI. Explicar en profundidad dichos procesos, tecnologías y técnicas no es el objetivo de este libro ni podría ser abordado en un espacio tan reducido. Por tanto, el libro se centra en los conceptos básicos de inteligencia de negocio (capítulo 1, "Introducción a Business Intelligence"), en cómo crear un data warehouse para almacenar los datos de una empresa en una representación que facilite la extracción de información (capítulo 2, "Diseño de un data warehouse"), en cómo identificar, transformar y cargar los datos de las bases de datos de la empresa en el data warehouse creado en el proceso anterior (capítulo 3, "Diseño de procesos ETL"), en cómo extraer información a partir de los datos almacenados en el data warehouse (capítulo 4, "Diseño de análisis OLAP") y en cómo presentar la información obtenida por los sistemas OLAP para facilitar su lectura y rápida comprensión (capítulos 5 y 6, "Diseño de informes" y "Diseño de cuadros de mando"). Al final del libro, Josep Curto nos presenta claramente cuáles son los factores que durante estos últimos años están dirigiendo la evolución de los sistemas de inteligencia de negocio y las tendencias que estos sistemas están tomando. El libro también presenta y describe el contenido de las principales fuentes de información sobre BI que podemos encontrar en la red.

El objetivo del libro no es sólo introducir conceptos, sino también enseñar a utilizarlos. Con ese objetivo en mente, el primer capítulo presenta un caso de estudio que se irá desarrollando a medida que avance el libro y el lector vaya adquiriendo los conocimientos necesarios para abordar sus distintas fases. Para desarrollar el caso de estudio se ha utilizado un programa informático de código abierto llamado Pentaho. De todas las herramientas libres de BI, Pentaho es posiblemente la suite más completa y madura del mercado en el momento de la redacción de este libro.

Tengo que decir que para mí ha sido una gran satisfacción colaborar con Josep Curto en la elaboración de este libro que ofrece una visión panorámica de la inteligencia de negocio. Considero que Josep ha realizado una magnífica tarea resumiendo en este libro los principales conceptos relacionados con la inteligencia de negocio y sus principales tecnologías. Espero que el

lector adquiera los conocimientos básicos sobre BI a partir de las explicaciones del libro, aprenda a desarrollar proyectos de BI a partir del caso práctico presentado, y pueda expandir sus conocimientos a través de la multitud de referencias presentadas.

Jordi Conesa i Caralt

Capítulo I

Introducción al Business Intelligence

En los últimos años, el mercado Business Intelligence se ha visto marcado por una clara evolución que lo destaca como un mercado maduro:

- Se ha producido una consolidación del mercado mediante la compra de empresas pequeñas por parte de los principales agentes del mercado (SAP, IBM, Microsoft y Oracle).
- Se ha enriquecido con soluciones open source que cubren el espectro de necesidades de una organización para la explotación de la información.
- Han aparecido nuevas empresas con foco en la innovación cubriendo nuevos nichos en el mercado de la inteligencia de negocio como la visualización, el análisis predictivo, las virtual appliances y/o el real-time Business Intelligence.
- A pesar de la crisis económica instaurada mundialmente desde 2008, el mercado de inteligencia de negocio sigue en una fase de crecimiento estable al posicionarse como una necesidad crítica para toda organización.

Este libro se centrará en introducir los diferentes conceptos que engloba la inteligencia de negocio. Otro de los objetivos es ejemplificar el desarrollo de un proyecto de Business Intelligence mediante herramientas open source para facilitar la comprensión de los conceptos presentados en el libro.

Algunas de estas herramientas acumulan diversos años de desarrollo y evolución y están respaldadas por organizaciones que tienen un claro modelo de negocio y que generan sinergias entre ellas. Podemos encontrar tanto herramientas de bases de datos como de minería de datos. Tal es la madurez de dichas soluciones que es posible desarrollar e implementar proyectos de inteligencia de negocio para todo tipo de organizaciones, tanto pymes como grandes organizaciones.

El objetivo de este capítulo es introducir la inteligencia de negocio y enumerar las tecnologías que engloba.

1. ¿Qué es la inteligencia de negocio?

El contexto de la sociedad de la información ha propiciado la necesidad de tener mejores, más rápidos y más eficientes métodos para extraer y transformar los datos de una organización en información y distribuirla a lo largo de la cadena de valor.[1]

La inteligencia de negocio (o Business Intelligence) responde a dicha necesidad, y podemos entender, en una primera aproximación, que es una evolución de los sistemas de soporte a las decisiones (DSS, Decissions Suport Systems). Sin embargo, este concepto, que actualmente se considera crítico en la gran mayoría de empresas, no es nuevo. En octubre de 1958 Hans Peter Luhn, investigador de IBM en dicho momento, acuño el término en el artículo "A Business Intelligence System" como:

> La habilidad de aprehender las relaciones de hechos presentados de forma que guíen las acciones hacia una meta deseada.

No es hasta 1989 que Howard Dresden, analista de Gartner, propone una definición formal del concepto:

> Conceptos y métodos para mejorar las decisiones de negocio mediante el uso de sistemas de soporte basados en hechos.

Desde entonces, el concepto del que estamos hablando ha evolucionado aunando diferentes tecnologías, metodologías y términos bajo su paraguas. Es necesario, por lo tanto, establecer una definición formal de uso en el presente material:

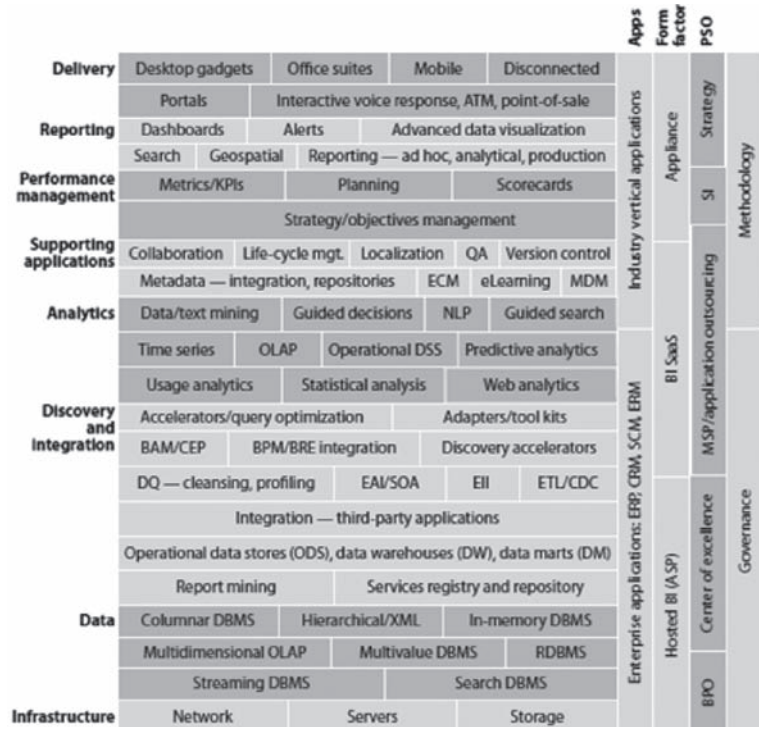
> Se entiende por **Business Intelligence** al conjunto de metodologías, aplicaciones, prácticas y capacidades enfocadas a la creación y administración de información que permite tomar mejores decisiones a los usuarios de una organización.

1. La cadena de valor empresarial, descrita y popularizada por Michael E. Porter en su obra Competitive Advantage: Creating and sustaining superior performance, es un modelo teórico que permite describir las actividades que generan valor en una organización.

Algunas de las tecnologías que forman parte de Business Intelligence son:

- Data warehouse.
- Reporting.
- Análisis OLAP (On-Line Analytical Processing).
- Análisis visual.
- Análisis predictivo.
- Cuadro de mando.
- Cuadro de mando integral.
- Minería de datos.
- Gestión del rendimiento.
- Previsiones.
- Reglas de negocio.
- Dashboards.
- Integración de datos (que incluye ETL, Extract, Transform and Load).

El siguiente diagrama, creado por Forrester Research, ilustra las diferentes tecnologías y aspectos que forman parte de la inteligencia de negocio:



1.1. Beneficios de un sistema de inteligencia de negocio

La implantación de estos sistemas de información proporciona diversos beneficios, entre los que podemos destacar:

- Crear un círculo virtuoso de la información (los datos se transforman en información que genera un conocimiento que permite tomar mejores decisiones que se traducen en mejores resultados y que generan nuevos datos).
- Permitir una visión única, conformada, histórica, persistente y de calidad de toda la información.
- Crear, manejar y mantener métricas, indicadores claves de rendimiento (KPI, Key Perfomance Indicador) e indicadores claves de metas (KGI, Key Goal Indicator) fundamentales para la empresa.
- Aportar información actualizada tanto a nivel agregado como en detalle.
- Reducir el diferencial de orientación de negocio entre el departamento TI y la organización.
- Mejorar comprensión y documentación de los sistemas de información en el contexto de una organización.
- Mejorar de la competitividad de la organización como resultado de ser capaces de:
    a) Diferenciar lo relevante sobre lo superfluo.
    b) Acceder más rápido a información.
    c) Tener mayor agilidad en la toma de las decisiones.

1.2. ¿Cuándo es necesaria la inteligencia de negocio?

Existen situaciones en las que la implantación de un sistema de Business Intelligence resulta adecuada. Destacamos, entre todas las que existen:

- La toma de decisiones se realiza de forma intuitiva en la organización.
- Identificación de problemas de calidad de información.
- Uso de Excel[2] como repositorios de información corporativos o de usuario. Lo que se conoce como Excel caos.
- Necesidad de cruzar información de forma ágil entre departamentos.

2. Se entiende como Excel caos el problema resultante del uso intensivo de Excel como herramienta de análisis. Cada usuario trabaja con un fichero personalizado. Como resultado, la información no cuadra entre departamentos y el coste de sincronización es sumamente elevado.

- Evitar silos de información.
- Las campañas de marketing no son efectivas por la información base usada.
- Existe demasiada información en la organización para ser analizada de la forma habitual. Se ha alcanzado la masa crítica de datos.
- Es necesario automatizar los procesos de extracción y distribución de información.

En definitiva, los sistemas de Business Intelligence buscan responder a las preguntas:

- ¿Qué pasó?
- ¿Qué pasa ahora?
- ¿Por qué pasó?
- ¿Qué pasará?

Tal y como Thomas Davenport en su libro "Competing on Analytics", una nueva forma de estrategia competitiva está emergiendo basada en el uso de la estadística descriptiva, modelos productivos y complejas técnicas de optimización, datos de alta calidad y una toma de decisiones basada en hechos. En dicho contexto, la inteligencia de negocio es el paso previo para dicha estrategia dado que ayuda a sentar las bases para su futuro despliegue.

2. Estrategia de Business Intelligence

Desplegar un proyecto de inteligencia de negocio en el seno de una organización no es un proceso sencillo. Las buenas prácticas indican que, para llegar a buen puerto, es necesario tener una estrategia de inteligencia de negocio que coordine de forma efectiva las tecnologías, el uso, los procesos de madurez.

2.1. ¿Cómo detectar que no existe una estrategia?

Es posible detectar que no existe una estrategia definida a través de los siguientes puntos y percepciones en el seno de una organización:

- Los usuarios identifican el departamento de informática (IT, Information Technology) como el origen de sus problemas de inteligencia de negocio.
- La dirección considera que la inteligencia de negocio es otro centro de coste.
- El departamento de IT continúa preguntando a los usuarios finales sobre las necesidades de los informes.
- El sistema de BI está soportado por help desk.
- No hay diferencia entre BI y gestión del rendimiento.
- No es posible medir el uso del sistema de inteligencia de negocio.
- No es posible medir el retorno de la inversión (ROI, Return On Invest) del proyecto de Business Intelligence.
- Se considera que la estrategia para el data warehouse es la misma que para que el sistema de inteligencia de negocio.
- No hay un plan para desarrollar, contratar, retener y aumentar el equipo de BI.
- No se conoce si la empresa tiene una estrategia para el BI.
- No existe un responsable funcional (o bien el asignado no es el adecuado).
- No existe un centro de competencia.
- Existen múltiples soluciones en la organización distribuidas en diferentes departamentos que repiten funcionalidad.
- No hay un plan de formación real y consistente de uso de las herramientas.
- Alguien cree que es un éxito que la información consolidada esté a disposición de los usuarios finales al cabo de dos semanas.
- Los usuarios creen que la información del data warehouse no es correcta.

El desarrollo de una estrategia de negocio es un proceso a largo plazo que incluye múltiples actividades, entre las que es conveniente destacar:

- Crear un centro de competencia de BI (BICC). Tiene el objetivo de aunar conocimiento en tecnologías, metodologías, estrategia, con la presencia de un sponsor a nivel ejecutivo y con analistas de negocio implicados y que tenga responsabilidad compartida en éxitos y fracasos.
- Establecer los estándares de BI en la organización para racionalizar tanto las tecnologías existentes como las futuras adquisiciones.
- Identificar qué procesos de negocio necesitan diferentes aplicaciones analíticas que trabajen de forma continua para asegurar que no existen silos de funcionalidad.
- Desarrollar un framework de métricas a nivel empresarial como el pilar de una gestión del rendimiento a nivel corporativo.

- Incluir los resultados de aplicaciones analíticas (minería de datos u otras) en los procesos de negocio con el objetivo de añadir valor a todo tipo de decisiones.
- Revisar y evaluar el portafolio actual de soluciones en un contexto de riesgo/recompensas.
- Considerar inversiones tácticas cuyo retorno de inversión estén dentro de un período de tiempo de un año. Además, tener en cuenta los diferentes análisis de mercado, de soluciones e incluso el *hype cycle*[3] de Gartner para conocer el estado del arte.
- Aprender de los éxitos y fracasos de otras empresas revisando casos de estudio y consultando a las empresas del sector para determinar qué ha funcionado y qué no.
- Evangelizar la organización.
- Alinear el departamento IT y el negocio en caso de no poder organizar un BICC, fundamental para trabajar como equipo integrado. El departamento de IT debe entender las necesidades y entregar la mejor solución ajustada a la necesidad particular y escalable a otras futuras.
- Poner atención a las necesidades que requieren BI en la organización porque se acostumbra a satisfacer a los usuarios o departamentos que gritan más fuerte, y esto no significa que den mayor valor a la compañía. Por ejemplo, los departamentos de finanzas son un caso típico de baja atención en soluciones BI.

Dado que este tipo de proyectos suponen una transformación de la cultura de la organización (de una toma de decisiones basada en la intuición a una toma de decisiones fundamentada en datos), una forma adecuada de articular los puntos anteriores es tener una respuesta consistente a diversas preguntas criticas de la compañía con anterioridad a la inversión en este tipo de proyectos:

---

3. El *hype cycle* de Gartner es una representación gráfica de la madurez, adopción y aplicación de negocio de una o varias tecnologías específicas. Es decir, muestra el ciclo de vida de dichas tecnologías. Las etapas que componen el ciclo son:
- Disparador tecnológico: cuando aparece el concepto en el mercado.
- Pico de expectativa inflada: cuando se habla mucho del concepto, pero está poco aplicado.
- Valle de la desilusión: cuando la herramienta está por debajo de lo que se esperaba de ella.
- La pendiente de tolerancia: el camino hacia la madurez.
- Plateau de productividad: cuando alcanza la madurez.

1. Qué problemas o necesidades de negocio se busca resolver mediante la estrategia de Business Intelligence.
2. Dentro de dichos problemas, cuáles son las preguntas a responder y qué acciones se realizarán como resultado de las respuestas.
3. Porqué no se pueden conseguir las respuestas actualmente o, en caso de tener un respuesta porqué ésta no cubre la necesidad de negocio.
4. Cuál es el impacto por la falta de dicha información (toma de decisiones pobre, oportunidades de negocio perdidas, procesos ineficientes,…).
5. Qué fuentes de datos son necesarias para poder responder las preguntas (marketing, finanzas, operaciones, recursos humanos, externas, etc.).
6. En qué medida las diferentes entidades de información (cliente, producto, etc.) están alineadas.
7. Qué diferencial existe entre los datos existentes con los datos necesarios para responder a las preguntas.
8. Cuál es la grado de calidad de los datos.
9. Qué cantidad de datos actual y histórica debe ser guardada y con qué frecuencia hay cambios.
10. Con qué frecuencia deben estar actualizadas las respuestas.

2.2. Business Intelligence Maturity Model

Si bien el objetivo de este libro no es dar pautas para definir una estrategia de Business Intelligence sino una introducción de conceptos, un buen punto de partida es identificar cuál es el grado de madurez de la organización en lo que se refiere a la inteligencia de negocio.

El BIMM (Business Intelligence Maturity Model) es un modelo de madurez que permite clasificar nuestra organización desde el punto de vista del grado de madurez de implantación de sistemas Business Intelligence en la misma (en relación directa con frameworks como COBIT).[4]

- **Fase 1: No existe BI.** Los datos se hallan en los sistemas de procesamiento de transacciones en línea (OLTP, On-Line Transaction Processing), desperdigados en otros soportes o incluso sólo contenidos en el know-how de

4. COBIT (Control Objectives for Information and related Technology) es un conjunto de mejores prácticas para el manejo de información creado por ISACA (Information Systems Audit and Control Association) y ITGI (IT Governance Institute) en 1992.

la organización. Las decisiones se basan en la intuición, en la experiencia, pero no en datos consistentes. El uso de datos corporativos en la toma de decisiones no ha sido detectado y tampoco el uso de una herramienta adecuada al hecho.

- **Fase 2: No existe BI, pero los datos son accesibles.** No existe un procesado formal de los datos para la toma de decisiones, aunque algunos usuarios tienen acceso a información de calidad y son capaces de justificar decisiones con dicha información. Frecuentemente, este proceso se realiza mediante Excel o algún tipo de reporting. Se intuye que deben existir soluciones para mejorar este proceso pero se desconoce la existencia del Business Intelligence.
- **Fase 3: Aparición de procesos formales de toma de decisiones basada en datos.** Se establece un equipo que controla los datos y que permite hacer informes contra los mismos que permiten tomar decisiones fundamentadas. Los datos son extraídos directamente de los sistemas transaccionales sin data cleansing[5] ni modelización, ni existe un data warehouse.
- **Fase 4: Data warehouse.** El impacto negativo contra los sistemas OLTP lleva a la conclusión de que un repositorio de datos es necesario para la organización. Se percibe el data warehouse como una solución deseada. El reporting sigue siendo personal.
- **Fase 5: Data warehouse crece y el reporting se formaliza.** El data warehouse funciona y se desea que todos se beneficien del mismo, de forma que el reporting corporativo se formaliza. Se habla de OLAP, pero sólo algunos identifican realmente sus beneficios.
- **Fase 6: Despliegue de OLAP.** Después de cierto tiempo, ni el reporting ni la forma de acceso al data warehouse es satisfactoria para responder a preguntas sofisticadas. OLAP se despliega para dichos perfiles. Las decisiones empiezan a impactar de forma significativa en los procesos de negocio de toda la organización.
- **Fase 7: Business Intelligence se formaliza.** Aparecen la necesidad de implantar otros procesos de inteligencia de negocio como Data Mining,

5. Data Cleansing consiste en el proceso de detectar y mejorar (o borrar) registros incorrectos e incompletos con el objetivo de conseguir datos coherentes y consistentes.

6. "Gartner's Top Predictions for IT Organizations and Users, 2008 and Beyond: Going Green and Self-Healing". Gartner, enero de 2008.

Balanced ScoreCard..., y procesos de calidad de datos impactan en procesos como Customer Relationship Management (CRM), Supply Chain Management (SCM)... Se ha establecido una cultura corporativa que entiende claramente la diferencia entre sistemas OLTP y DSS.

3. Soluciones Open Source Business Intelligence

El open source es una filosofía de desarrollo de software que cumple los siguientes principios:

- Abierto: la comunidad tiene libre acceso, uso y participación del código fuente, así como la posibilidad de uso de foros para proporcionar feedback.
- Transparencia: la comunidad tiene acceso al roadmap, documentación, defectos y agenda de las milestones.
- Early & Often: la información se publica de manera frecuente y pronto a través de repositorios públicos (incluyendo el código fuente).

El open source ya no es una tendencia emergente sino que es un enfoque que tiene un impacto profundo y que en años venideros tendrá una presencia importante en todos los sectores, tal y como comenta Gartner:

> En 2012, el 80% del SW comercial incluirá algún componente open source. Incluir componentes open source en los productos para abaratar costes es considerado la mínima estrategia que las compañías pueden llevar a cabo para mantener su ventaja competitiva en 5 años.[6]

En los últimos años, el mercado Business Intelligence se ha enriquecido con soluciones open source que cubren todo el espectro de necesidades de una organización para la explotación de la información. Algunas de estas herramientas tienen ya a sus espaldas varios años de recorrido y actualmente se

6. "Gartner's Top Predictions for IT Organizations and Users, 2008 and Beyond: Going Green and Self-Healing". Gartner, enero de 2008.

hallan respaldadas por organizaciones que tienen un claro modelo de negocio orientado a los servicios de valor añadido. Es posible encontrar herramientas solventes y maduras desde el nivel de base de datos hasta el de procesos de minería de datos que pueden, en algunos casos, adaptarse a las necesidades de una organización.

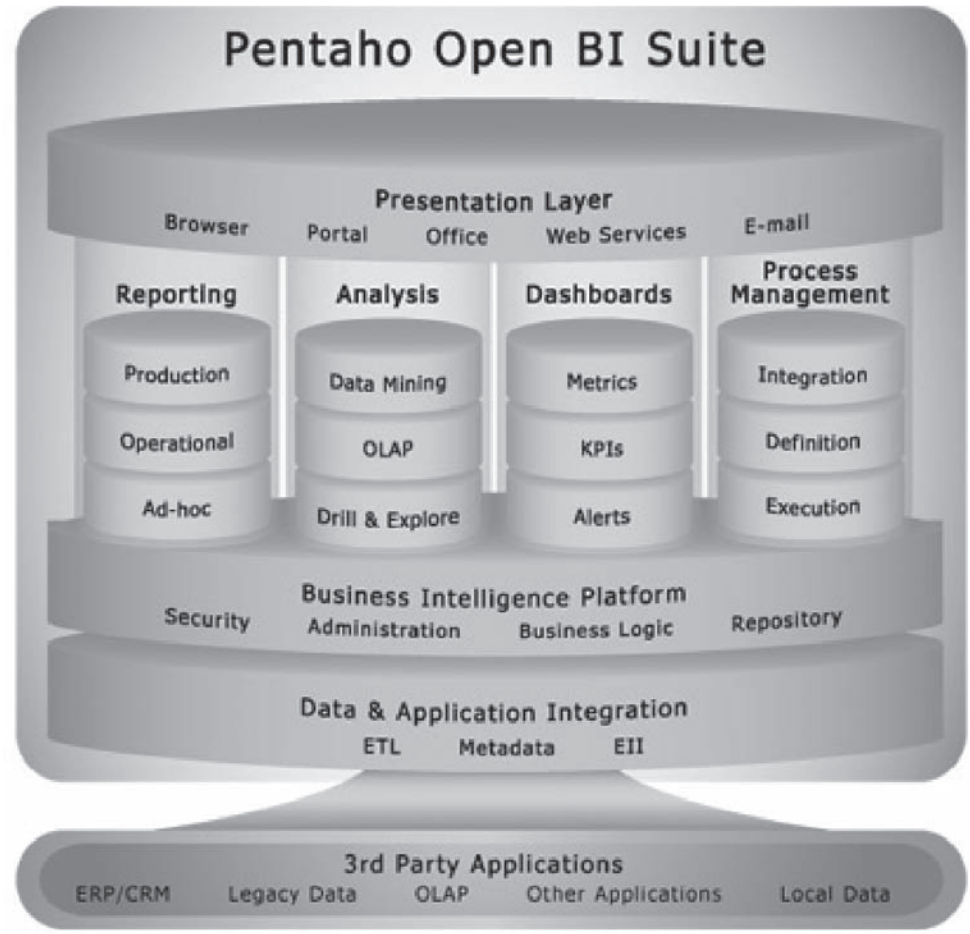
En el mercado de soluciones Open Source Business Intelligence (OSBI) destaca la solución Pentaho.

3.1. Pentaho

Pentaho es una de las suites más completas y maduras del mercado OSBI que existe desde el año 2006. Existen dos versiones: Community y Enterprise. Serán comparadas a lo largo de los diferentes módulos. Está compuesta por diferentes motores incluidos en el servidor de Pentaho:

- Reporting: soporta informes estáticos, paramétricos y ad hoc.
- Análisis: soporta OLAP (mediante Mondrian) y minería de datos (mediante Weka).
- Cuadros de mando: mediante CDF (Community Dashboard Framework).
- ETL: mediante la herramienta Kettle.
- Metadata: que proporciona una capa de acceso de información basada en lenguaje de negocio.
- Workflow: el servidor de Pentaho se basa en acciones que la mayoría de objetos de negocio permite lanzar.

Actualmente Pentaho está siguiendo la estrategia open core, que consiste en que a partir de un núcleo open source se ofrecen servicios y módulos mejorados. Ésta es la razón por la cual existen dos versiones. La principal diferencia entre ambas versiones es que la versión Enterprise se ofrece bajo una modalidad de subscripción y la versión Community es completamente gratuita. En este capítulo siempre se hará referencia a la versión Community.



4. Glosario

BI	Business Intelligence
BICC	Business Intelligence Competency Center
BIMM	Business Intelligence Maturity Model
COBIT	Control Objectives for Information and related Technology
CRM	Customer Relationship Management
DSS	Decision Support Systems
EII	Enterprise Information Integration

ETL	Extract, Transform and Load
IBM	International Business Machines
KPI	Key Performance Indicator
KGI	Key Goal Indicator
ODS	Operational Data Store
OLAP	On-Line Analytical Processing
OLTP	On-Line Transaction Processing
OSBI	Open Source Business Intelligence
SI	Sistemas de Información
SQL	Structured Query Language
TI	Tecnologías de la Información

5. Bibliografía

DAVENPORT, Thomas H., y HARRIS, Jeanne G. (2007). *Competing on Analytics: The New Science of Winning*. Nueva York: Harvard Business Press.

DAVENPORT, Thomas H., HARRIS, Jeanne G., y MORISON, Robert (2010). *Analytics at Work: Smarter Decisions, Better Results*. Nueva York: Harvard Business Press.

MILLER, Dorothy. (2007). *Measuring Business Intelligence Success: A Capability Maturity Model*. Nueva York: D M Morrisey.

MILLER, Gloria J., BRAUTIGAM, Dagmar. V., y GERLACH, Stefanie (2006). *Business Intelligence Competency Centers: A Team Approach to Maximizing Competitive Advantage*. Hoboken: Wiley and SAS Business Series.

Capítulo II

Diseño de un data warehouse

En concepto y el enfoque de la inteligencia de negocio ha evolucionado de sobremanera los últimos años. Uno de los conceptos que más ha evolucionado ha sido el repositorio de datos, también conocido como data warehouse.

En los últimos años han aparecido muchos enfoques de data warehouse tanto a nivel tecnológico como estratégico:

- Aparición de bases de datos especializadas en el despliegue de data warehouse en forma de software o appliances.
- Desarrollo de múltiples metodologías que buscan capacitar la organización para lidiar con el reto de un crecimiento exponencial de datos articulado en tres dimensiones: volumen, velocidad y variedad. Lo que se conoce como Big Data.
- Tendencias SaaS o in-memory que buscan reducir el tiempo de creación de estructuras de datos.

El enfoque que se toma en este capítulo de introducción es el de la metodología ya consolidada en múltiples proyectos y sobre la que se sustentan todas las evoluciones actuales.

El objetivo de este capítulo es introducir el concepto de data warehouse o almacén de datos y la ejemplificación de su diseño usando una solución open source.

Por ello, el contenido cubre desde la definición del concepto de data warehouse, los principales conceptos de una arquitectura de un data warehouse, la presentación del caso práctico y la resolución del mismo mediante las técnicas de modelización introducidas y su cristalización usando una herramienta open source.

1. El núcleo de un sistema de inteligencia de negocio: el data warehouse

Como ya se ha comentado, un sistema de inteligencia de negocio está formado por diferentes elementos: ETL, OLAP, reporting... Pero de todas las piezas, la principal es el data warehouse[1] o almacén de datos.

Un **data warehouse** es un repositorio de datos que proporciona una visión global, común e integrada de los datos de la organización –independientemente de cómo se vayan a utilizar posteriormente por los consumidores o usuarios–, con las propiedades siguientes: estable, coherente, fiable y con información histórica. Al abarcar un ámbito global de la organización y con un amplio alcance histórico, el volumen de datos puede ser muy grande (centenas de terabytes). Las bases de datos relacionales son el soporte técnico más comúnmente usado para almacenar las estructuras de estos datos y sus grandes volúmenes. Resumiendo, presenta las siguientes características:

- Orientado a un tema: organiza una colección de información alrededor de un tema central.
- Integrado: incluye datos de múltiples orígenes y presenta consistencia de datos.
- Variable en el tiempo: se realizan fotos de los datos basadas en fechas o hechos.
- No volátil: sólo de lectura para los usuarios finales.

Frecuentemente el data warehouse está constituido por una base de datos relacional, pero no es la única opción factible, también es posible considerar las bases de datos orientadas a columnas o incluso basadas en lógica asociativa.

Debemos tener en cuenta que existen otros elementos en el contexto de un data warehouse:

- Data Warehousing: es el proceso de extraer y filtrar datos de las operaciones comunes de la organización, procedentes de los distintos sistemas de infor-

1. Según W. H. Inmon (considerado por muchos el padre del concepto), un data warehouse es un conjunto de datos orientados por temas, integrados, variantes en el tiempo y no volátiles, que tienen por objetivo dar soporte a la toma de decisiones.
Según Ralph Kimball (considerado el principal promotor del enfoque dimensional para el diseño de almacenes de datos), un data warehouse es una copia de los datos transaccionales específicamente estructurada para la consulta y el análisis.

mación operacionales y/o sistemas externos, para transformarlos, integrarlos y almacenarlos en un almacén de datos con el fin de acceder a ellos para dar soporte en el proceso de toma de decisiones de una organización.

- Data Mart: es un subconjunto de los datos del data warehouse cuyo objetivo es responder a un determinado análisis, función o necesidad, con una población de usuarios específica. Al igual que en un data warehouse, los datos están estructurados en modelos de estrella o copo de nieve, y un data mart puede ser dependiente o independiente de un data warehouse. Por ejemplo, un posible uso sería para la minería de datos o para la información de marketing. El data mart está pensado para cubrir las necesidades de un grupo de trabajo o de un determinado departamento dentro de la organización.
- Operational Data Store: es un tipo de almacén de datos que proporciona sólo los últimos valores de los datos y no su historial; además, generalmente admite un pequeño desfase o retraso sobre los datos operacionales.
- Staging Area: es el sistema que permanece entre las fuentes de datos y el data warehouse con el objetivo de:
  - Facilitar la extracción de datos desde fuentes de origen con una heterogeneidad y complejidad grande.
  - Mejorar la calidad de datos.
  - Ser usado como caché de datos operacionales con el que posteriormente se realiza el proceso de data warehousing.
  - Uso de la misma para acceder en detalle a información no contenida en el data warehouse.
- Procesos ETL: tecnología de integración de datos basada en la consolidación de datos que se usa tradicionalmente para alimentar data warehouse, data mart, staging area y ODS. Usualmente se combina con otras técnicas de consolidación de datos.
- Metadatos: datos estructurados y codificados que describen características de instancias; aportan informaciones para ayudar a identificar, descubrir, valorar y administrar las instancias descritas.

1.1. Elementos de un data warehouse

La estructura relacional de una base de datos operacional sigue las formas normales en su diseño. Un data warehouse no debe seguir ese patrón de diseño.

La idea principal es que la información sea presentada desnormalizada para optimizar las consultas. Para ello debemos identificar, en el seno de nuestra organización, los procesos de negocio, las vistas para el proceso de negocio y las medidas cuantificables asociadas a los mismos. De esta manera hablaremos de:

- Tabla de hecho: es la representación en el data warehouse de los procesos de negocio de la organización. Por ejemplo, una venta puede identificarse como un proceso de negocio de manera que es factible, si corresponde en nuestra organización, considerar la tabla de hecho ventas.
- Dimensión: es la representación en el data warehouse de una vista para un cierto proceso de negocio. Si regresamos al ejemplo de una venta, para la misma tenemos el cliente que ha comprado, la fecha en la que se ha realizado... Estos conceptos pueden ser considerados como vistas para este proceso de negocio. Puede ser interesante recuperar todas las compras realizadas por un cliente. Ello nos hace entender por qué la identificamos como una dimensión.
- Métrica: son los indicadores de negocio de un proceso de negocio. Aquellos conceptos cuantificables que permiten medir nuestro proceso de negocio. Por ejemplo, en una venta tenemos el importe de la misma.

Existen principalmente dos tipos de esquemas para estructurar los datos en un almacén de datos:

- Esquema en estrella: consiste en estructurar la información en procesos, vistas y métricas recordando a una estrella (por ello el nombre). A nivel de diseño, consiste en una tabla de hechos (lo que en los libros encontraremos como fact table) en el centro para el hecho objeto de análisis y una o varias tablas de dimensión por cada punto de vista de análisis que participa de la descripción de ese hecho. En la tabla de hecho encontramos los atributos destinados a medir (cuantificar): sus métricas. La tabla de hechos sólo presenta uniones con dimensiones.
- Esquema en copo de nieve: es un esquema de representación derivado del esquema en estrella, en el que las tablas de dimensión se normalizan en múltiples tablas. Por esta razón, la tabla de hechos deja de ser la única tabla del esquema que se relaciona con otras tablas, y aparecen nuevas uniones. Es posible distinguir dos tipos de esquemas en copo de nieve:
  - Completo: en el que todas las tablas de dimensión en el esquema en estrella aparecen ahora normalizadas.
  - Parcial: sólo se lleva a cabo la normalización de algunas de ellas.

Es conveniente profundizar en los conceptos de tabla de hecho, dimensión y métrica.

1.2. Tipos de tablas de hecho

Una tabla de hecho es una representación de un proceso de negocio. A nivel de diseño es una tabla que permite guardar dos tipos de atributos diferenciados:

- Medidas del proceso/actividad/flujo de trabajo/evento que se pretende modelizar.
- Claves foráneas hacia registros en una tabla de dimensión (o en otras palabras, como ya sabemos, hacia una vista de negocio).

Existen diferentes tipos de tablas de hecho:

- Transaction Fact Table: representan eventos que suceden en un determinado espacio-tiempo. Se caracterizan por permitir analizar los datos con el máximo detalle. Por ejemplo, podemos pensar en una venta que tiene como resultado métricas como el importe de la misma.
- Factless Fact Tables/Coverage Table: son tablas que no tienen medidas, y tiene sentido dado que representan el hecho de que el evento suceda. Frecuentemente se añaden contadores a dichas tablas para facilitar las consultas SQL. Por ejemplo, podemos pensar en la asistencia en un acto benéfico en el que por cada persona que asiste tenemos un registro pero podríamos no tener ninguna métrica asociada más.
- Periodic Snapshot Fact Table: son tablas de hecho usadas para recoger información de forma periódica a intervalos de tiempo regulares. Dependiendo de la situación medida o de la necesidad de negocio, este tipo de tablas de hecho son una agregación de las anteriores o están diseñadas específicamente. Por ejemplo, podemos pensar en el balance mensual. Los datos se recogen acumulados de forma mensual.
- Accumulating Snapshot Fact Table: representan el ciclo de vida completo –con un principio y un final– de una actividad o un proceso. Se caracterizan por presentar múltiples dimensiones relacionadas con los eventos presentes en un proceso. Por ejemplo, podemos pensar en un proceso de matriculación de un estudiante: recopila datos durante su periodo de vida que suelen sustituir los anteriores (superación y recopilación de asignaturas, por ejemplo).

1.3. Tipos de dimensiones

Las dimensiones recogen los puntos de análisis de un hecho. Por ejemplo, una venta se puede analizar en función del día de venta, producto, cliente, vendedor o canal de venta, entre otros. Respecto al punto de vista de la gestión histórica de los datos, éstos se pueden clasificar como:

- **SCD[2] Tipo 0.** No se tiene en cuenta la gestión de los cambios históricos y no se realiza esfuerzo alguno. Nunca se cambia la información, ni se reescribe.
- **SCD Tipo 1.** No se guardan datos históricos. La nueva información sobrescribe la antigua siempre. La sobrescritura se realiza, principalmente, por errores de calidad de datos. Este tipo de dimensiones son fáciles de mantener, y se usan cuando la información histórica no es importante.
- **SCD Tipo 2.** Toda la información histórica se guarda en el data warehouse. Cuando hay un cambio se crea una nueva entrada con fecha y surrogate key apropiadas. A partir de ese momento será el valor usado para las futuras entradas. Las antiguas usarán el valor anterior.
- **SCD Tipo 3.** Toda la información histórica se guarda en el data warehouse. En este caso se crean nuevas columnas con los valores antiguos y los actuales son remplazados con los nuevos.
- **SCD Tipo 4.** Es lo que se conoce habitualmente como tablas históricas. Existe una tabla con los datos actuales y otra con los antiguos o los cambios.
- **SCD Tipo 6/Híbrida.** Combina las aproximaciones de los tipos 1, 2 y 3 (y, claro, entonces 1+2+3=6). Consiste en considerar una dimensión de tipo 1 y añadir un par de columnas adicionales que indican el rango temporal de validez de una de las columnas de la tabla. Si bien su diseño es complejo, entre sus beneficios podemos destacar que reduce el tamaño de las consultas temporales. Existe otra variante para este tipo de dimensión que consiste en tener versiones del registro de la dimensión (numerados de 0 a n+1, donde 0 siempre es la versión actual).

Existen otros tipos de dimensiones, cuya clasificación es funcional:

- **Degeneradas:** se encuentran como atributos en la tabla de hecho, si bien tiene el significado de un punto de vista de análisis. Contiene información

2. SCD son las siglas de Slowly Changing Dimension, y se refiere a la política de actualización de datos en una dimensión.

de baja cardinalidad formada por relaciones dicotómicas. Frecuentemente contienen sólo un atributo y, por ello, no se crea una tabla aparte. Por ejemplo, el sexo de un paciente.

- **Monster:** es conveniente comentar que algunas dimensiones pueden crecer desmesuradamente. Una buena práctica es romper la dimensión en dos tablas: una que contenga los valores estáticos y otra que contenga los valores volátiles. Un ejemplo claro puede ser la información de cliente. Debemos ser conscientes de cuál es la información primordial del mismo y cuál la que sólo se usa puntualmente en los informes u otros análisis.
- **Junk:** que contiene información volátil que se usa puntualmente y que no se guarda de forma permanente en el data warehouse.
- **Conformadas:** que permite compartir información entre dimensiones. Consiste en dimensiones definidas correctamente para que sean usadas por dos tablas y poder así realizar consultas comunes. El ejemplo más fácil es la dimensión temporal.
- **Bridge:** que permiten definir relaciones n a m entre tablas de hecho. Necesarias para definir por la relación entre un piloto y sus múltiples patrocinadores.
- **Role-playing:** que tienen asignado un significado. Por ejemplo, podemos tener la dimensión fecha, pero también fecha de entrega.
- **Alta cardinalidad:** que contienen una gran cantidad de datos difícilmente consultables en su totalidad. Por ejemplo, cada uno de los habitantes de un país.

1.4. Tipos de métricas

Podemos distinguir diferentes tipos de medidas, basadas en el tipo de información que recopilan así como su funcionalidad asociada:

- Métricas: valores que recogen el proceso de una actividad o los resultados de la misma. Estas medidas proceden del resultado de la actividad de negocio.
  - Métricas de realización de actividad (leading): miden la realización de una actividad. Por ejemplo, la participación de una persona en un evento.
  - Métricas de resultado de una actividad (lagging): recogen los resultados de una actividad. Por ejemplo, la cantidad de puntos de un jugador en un partido.

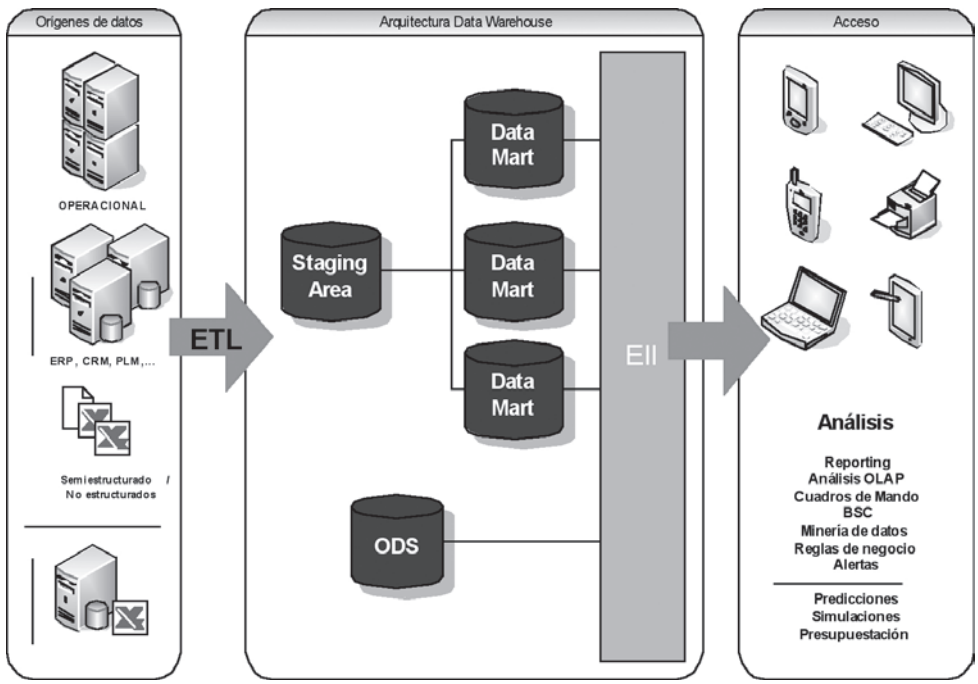
- Indicadores clave: valores correspondientes que hay que alcanzar y que suponen el grado de asunción de los objetivos. Estas medidas proporcionan información sobre el rendimiento de una actividad o sobre la consecución de una meta.
  - Key Performance Indicator (KPI): indicadores clave de rendimiento. Más allá de la eficacia, se definen unos valores que nos explican en qué rango óptimo de rendimiento nos deberíamos situar al alcanzar los objetivos. Son métricas del proceso.
  - Key Goal Indicator (KGI): indicadores de metas. Definen mediciones para informar a la dirección general si un proceso TIC ha alcanzado sus requisitos de negocio, y se expresan por lo general en términos de criterios de información.

Debemos añadir que existen también indicadores de desempeño. Los indicadores clave de desempeño (en definitiva, son KPI) definen mediciones que determinan cómo se está desempeñando el proceso de TI para alcanzar la meta. Son los indicadores principales que señalan si será factible lograr una meta o no, y son buenos indicadores de las capacidades, prácticas y habilidades. Los indicadores de metas de bajo nivel se convierten en indicadores de desempeño para los niveles altos.

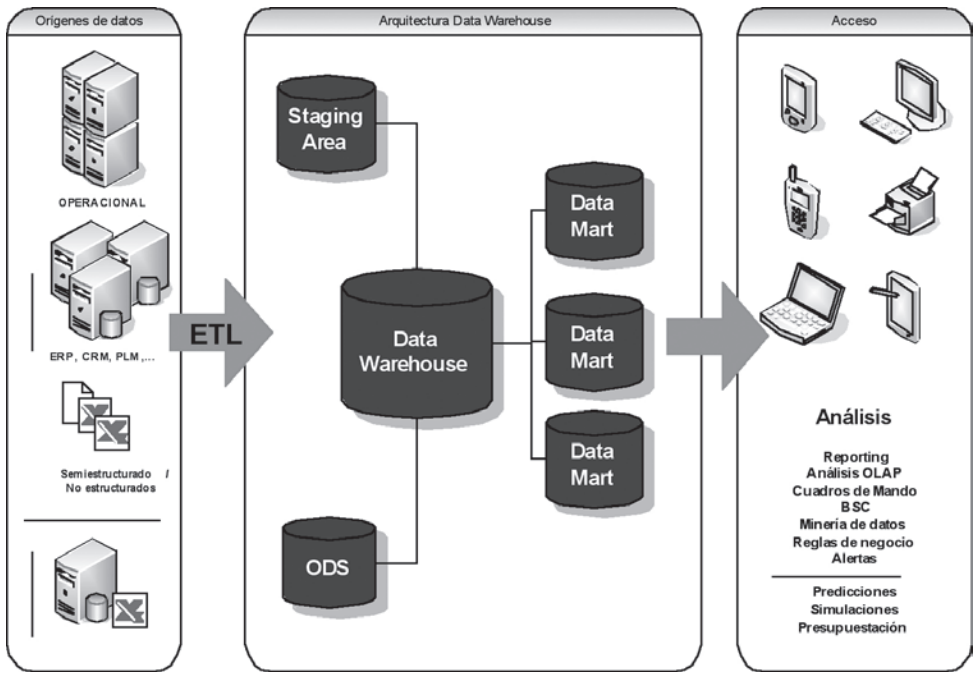
1.5. Arquitectura de un data warehouse

Existen principalmente tres enfoques en la arquitectura corporativa de un data warehouse:

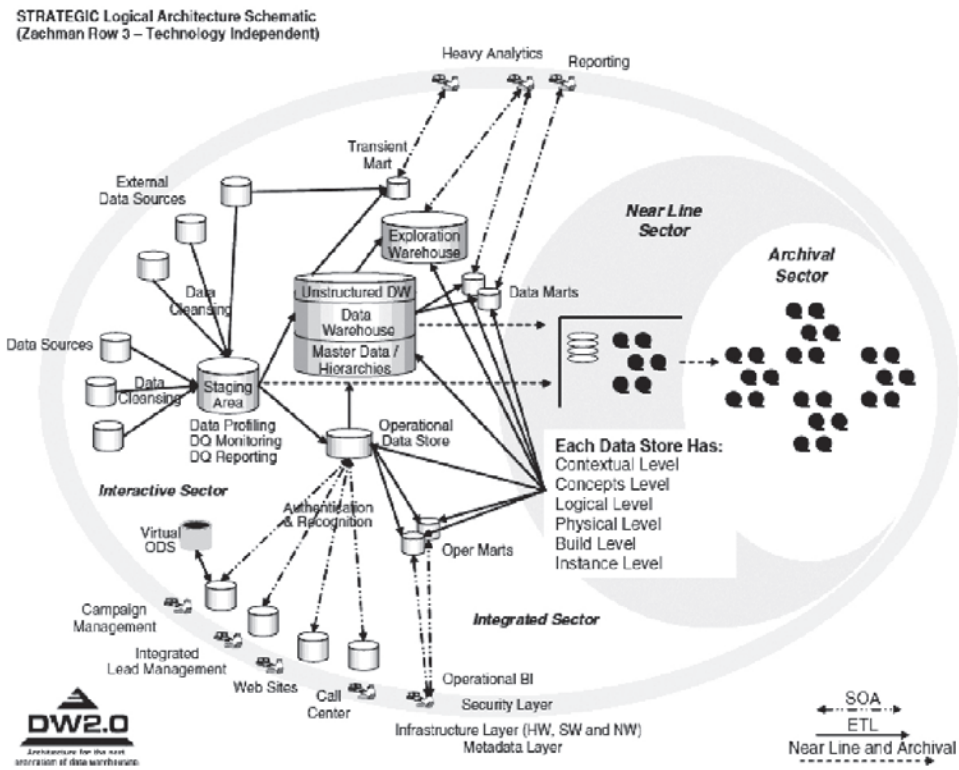
- Enterprise Bus Architecture (o Data Warehouse Virtual/Federado): también conocido como MD (Multidimensional Arquitechture), consiste en una arquitectura basada en data marts independientes federados que pueden hacer uso de una staging area en el caso de ser necesario. Federados quiere decir que se hace uso de una herramienta EII (Enterprise Information Integration) para realizar las consultas como si se tratara de un único data warehouse. Puede existir en el caso de ser necesario un ODS.



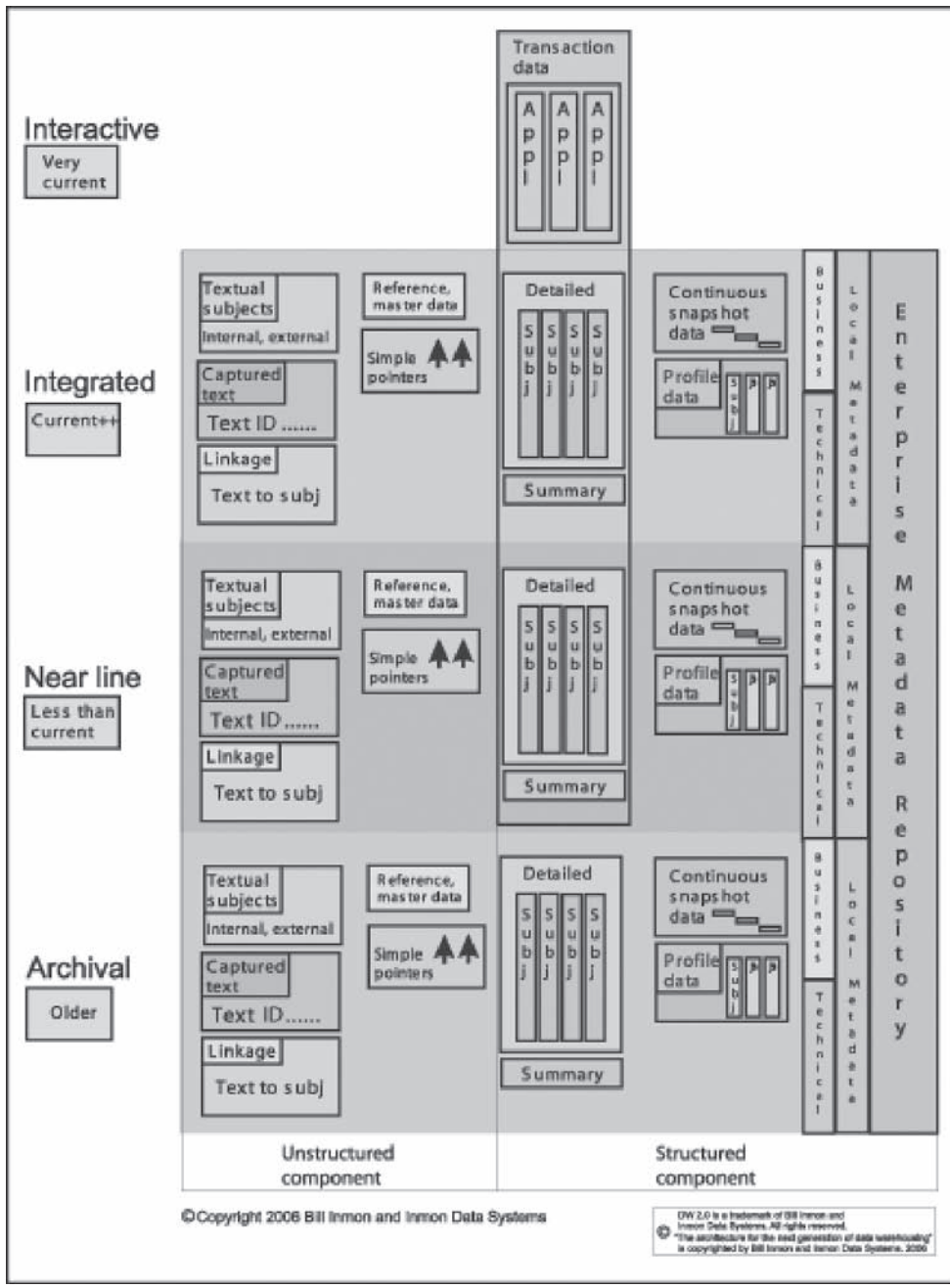
- Corporate Information Factory (o Enterprise Data Warehouse): consiste en una arquitectura en la que existe un data warehouse corporativo y unos data marts (o incluso cubos OLAP) dependientes del mismo. El acceso a datos se realiza a los data marts o a la ODS en caso de existir, pero nunca al propio data warehouse. Puede existir en el caso de ser necesaria una staging area.



- Enterprise Data Warehouse 2.0: consiste en la revisión de la metodología de Bill Inmon para incluir toda la experiencia de los últimos veinte años. El punto diferencial es que se separa la información por la edad de la misma y se clasifica por su uso. Se caracteriza por completar tanto la inclusión de información estructurada como no estructurada y por focalizarse en el objetivo de responder a todas las necesidades actuales de negocio. El siguiente gráfico representa una arquitectura completa:



Es decir, distingue diferentes entornos en función de lo actual que es la información y de su uso.

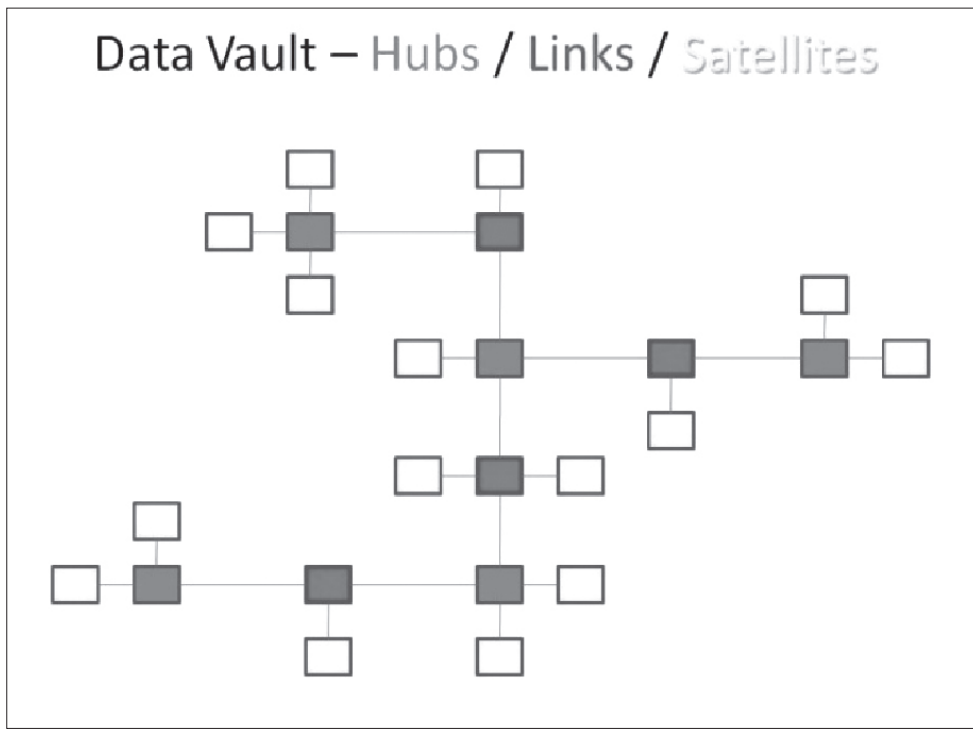

Interactive
Very current
Integrated
Current++
Near line
Less than current
Archival
Older
Transaction data
Appl
Appl
Appl
Textual subjects
Internal, external
Captured text
Text ID .....
Linkage
Text to subj
Reference, master data
Simple pointers
Detailed
Subj
Subj
Subj
Subj
Summary
Continuous snapshot data
Profile data
Business
Technical
Local Metadata
Enterprise Metadata Repository
Unstructured component
Structured component
©Copyright 2006 Bill Inmon and Inmon Data Systems

Esta metodología aún está en proceso de despliegue en el contexto empresarial.

Se combina, frecuentemente, con Data Vault, un modelo de datos basado en tres tipos de entidades:

- Hubs: contiene los indicadores claves de negocio.
- Links: contiene las relaciones.
- Satellits: contiene las descripciones.

Este tipo de diseño busca la máxima flexibilidad del data warehouse con el objetivo que éste sea adaptable a cualquier evolución del modelo de negocio de la organización.



2. Presentación caso práctico: análisis de estadísticas web

Actualmente la mayoría de las organizaciones tienen aplicaciones web mediante las cuales despliegan tanto soluciones de negocio como de soporte al mismo. Dichas aplicaciones generan ficheros de texto donde se guardan los datos de acceso y consulta, comúnmente llamados archivos de log. El formato en el que están guardados estos datos y la cantidad generada de información no permite que puedan ser analizados directamente. Sin embargo, la información contenida en dichos archivos permite conocer las tendencias de los usuarios, los servicios más usados... Además, es conveniente recordar que la legislación vigente obliga a almacenar la información durante al menos un año.

El caso práctico del presente libro consistirá en desarrollar un sistema de **análisis de estadísticas web** mediante herramientas open source de inteligencia de negocio.

En particular, las herramientas que se usarán son:

- MySQL para el data warehouse.
- Pentaho y sus correspondientes herramientas de diseño para la solución de inteligencia de negocio.

El objetivo de este sistema es recopilar la información de los logs en un único repositorio de datos (el data warehouse) y construir una solución de negocio para el análisis consistente en informes, análisis OLAP y cuadros de mando.

A lo largo de los diferentes módulos se explicará progresivamente cómo construir la solución de inteligencia de negocio. El contenido de los capítulos es:

1. Capítulo 1: Diseño de un data warehouse.
2. Capítulo 2: Diseño de procesos ETL.
3. Capítulo 3: Diseño de análisis OLAP.
4. Capítulo 4: Diseño de informes.
5. Capítulo 5: Diseño de cuadros de mando.

2.1. Formato de un log

Cualquier servidor web guarda sus logs conforme a una información básica común (extraída de los datos de cabecera del protocolo http). En resumen, los datos más relevantes son los siguientes:

- Ip: dirección desde la que se hace la visita.
- Fecha/hora: cuándo se realiza la visita.
- URL: dirección del recurso visitado o accedido. La URL contiene la información sobre el protocolo utilizado, el puerto, el dominio accedido, parámetros, etc.
- Resultado: código que indica si se tuvo éxito en el acceso o por el contrario hubo un error.
- Tamaño: en bytes del recurso accedido.
- Agente de usuario: indica por ejemplo el navegador utilizado para acceder. Este dato también puede tener codificado el sistema operativo sobre el que está ejecutándose dicho navegador.
- URL previa: indica la URL a partir de la cual se hizo la visita.

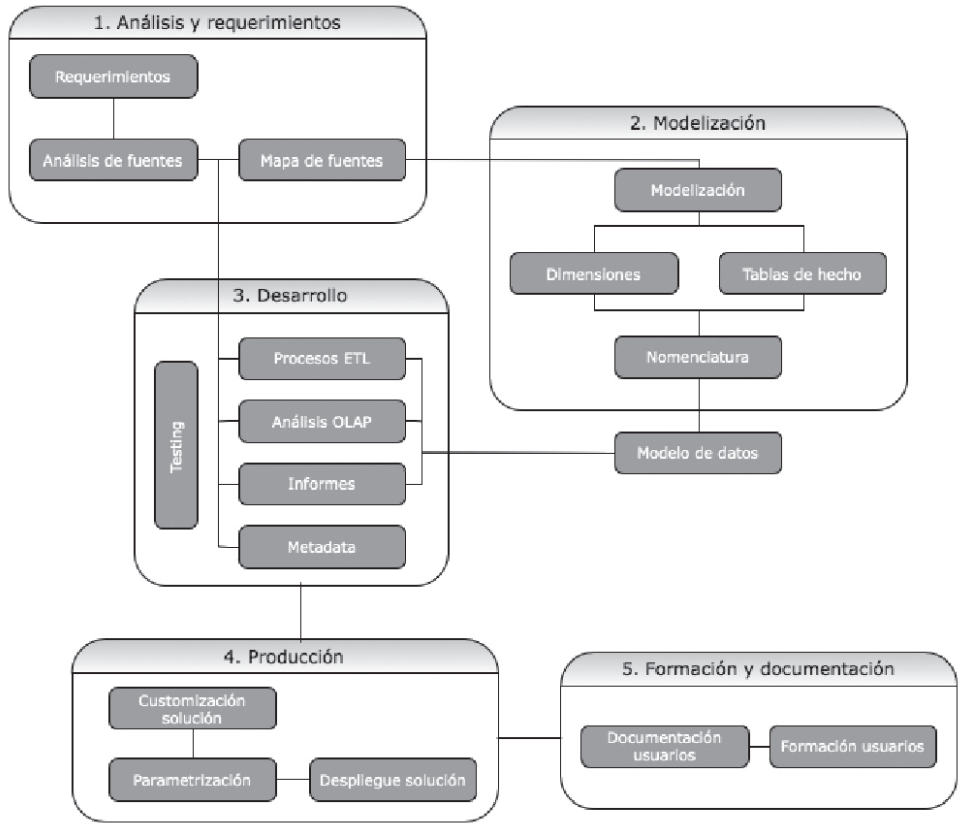
2.2. Necesidades de negocio

Algunas de las cuestiones de negocio que es posible responder a través de una herramienta de análisis de estadísticas web son las siguientes:

1) Días con más visitas.
2) Horas con mayor afluencia de visitas.
3) Páginas más visitadas.
4) Número de visitas.
5) Número de usuarios.
6) Número de sesiones.

2.3. Fases de un proyecto de BI

A través de los diferentes módulos que componen este material recorreremos de forma resumida las diferentes fases de un proyecto de inteligencia de negocio, si bien algunas de ellas no serán necesarias. A modo de resumen se incluye un gráfico para tenerlas presentes.



3. Resolución caso práctico con MySQL

3.1. Modelo conceptual de datos

El modelo conceptual se basa en identificar qué tipo de procesos y vistas de negocio proporcionan la respuesta a las preguntas que tienen los usuarios finales. Normalmente en esta fase, se debe ser previsor y pensar más allá de las necesidades actuales y de poder cubrir las futuras. Un buen consultor Business Intelligence conoce múltiples modelos de negocio y puede aportar opiniones clave en el diseño.

En el análisis de estadísticas web, los datos se extraen de archivos de texto.

Dado que las aplicaciones se despliegan en diferentes plataformas, no todos los logs están unificados. Ello dificulta la consolidación de la información. Cuando esto sucede, lo habitual es crear una staging area. El objetivo de crear la staging area es facilitar el proceso de transformación de los datos. Si bien para el caso práctico la staging no es necesaria, sus beneficios son claros:

- Realizar otro tipo de análisis a posteriori distinguiendo los datos mediante ciertos criterios, por ejemplo si el acceso ha sido realizado por robots o por una amenaza (lo que se conoce como accesos no autorizados).
- Mejorar el rendimiento de la carga de datos al realizar la carga previa a la staging area y, a posteriori, realizar las transformaciones de los datos entre bases de datos.

En el caso de crear una staging area, su modelo sería:

En el momento de hacer el diseño conceptual es necesario identificar las

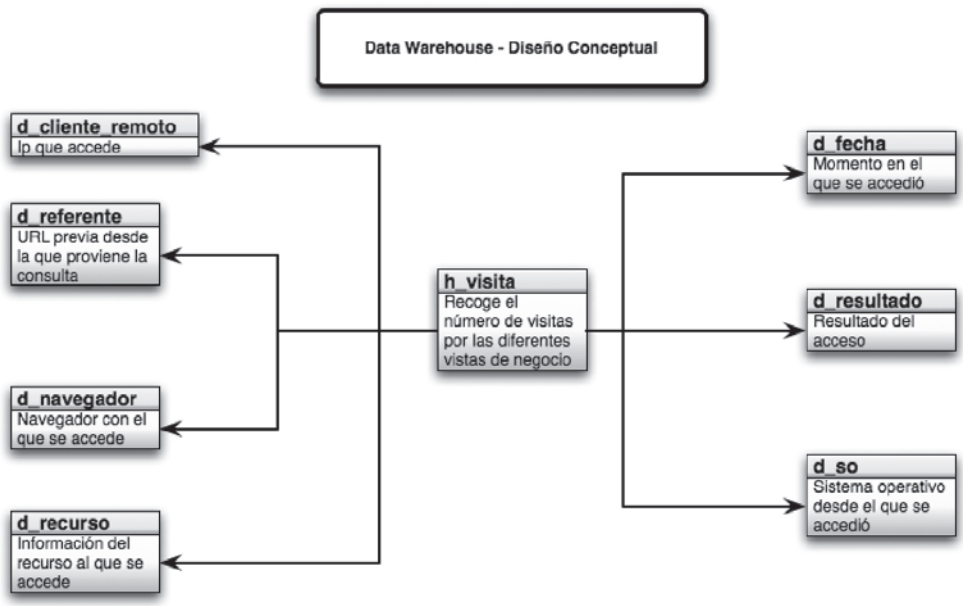
SA_AEW: Datos extraidos de los logs para el análisis de estadisticas web.

tablas de hecho y las dimensiones que se pueden deducir del caso práctico. Teniendo en cuenta la información que se extrae identificamos para nuestro ejemplo una tabla de hecho o proceso de negocio: la visita. Cada acceso se traduce en una visita.

Cada visita puede analizarse desde diferentes puntos de vista (lo que nos proporciona las dimensiones del proceso de negocio):

- Dimensión cliente remoto: la ip desde la que se realiza la visita.
- Dimensión referente: la URL previa desde la que se accede.
- Dimensión navegador: el navegador desde el que se realiza la visita.
- Dimensión recurso: información del recurso al que se accede.
- Dimensión fecha: momento en el que se realiza la visita.
- Dimensión resultado: resultado de la visita.
- Dimensión sistema operativo: sistema operativo desde el que se realiza la visita.

De manera que se obtiene el siguiente diseño conceptual:



3.2. Modelo lógico de datos

Después del modelo conceptual, a través del cual hemos identificado las tablas de hecho y las dimensiones, es necesario realizar el diseño lógico con el que se identifican las métricas de las tablas de hecho y los atributos de las dimensiones.

La tabla de hecho contiene la clave subrogada que identifica de manera única cada registro, las claves foráneas a las dimensiones relacionadas con la tabla de hecho y las métricas. Existen otras métricas, como el tiempo de respuesta, el número de bytes servidos, el número de sesiones...

Consideraremos la medida más natural en este caso práctico: el número de visitas.

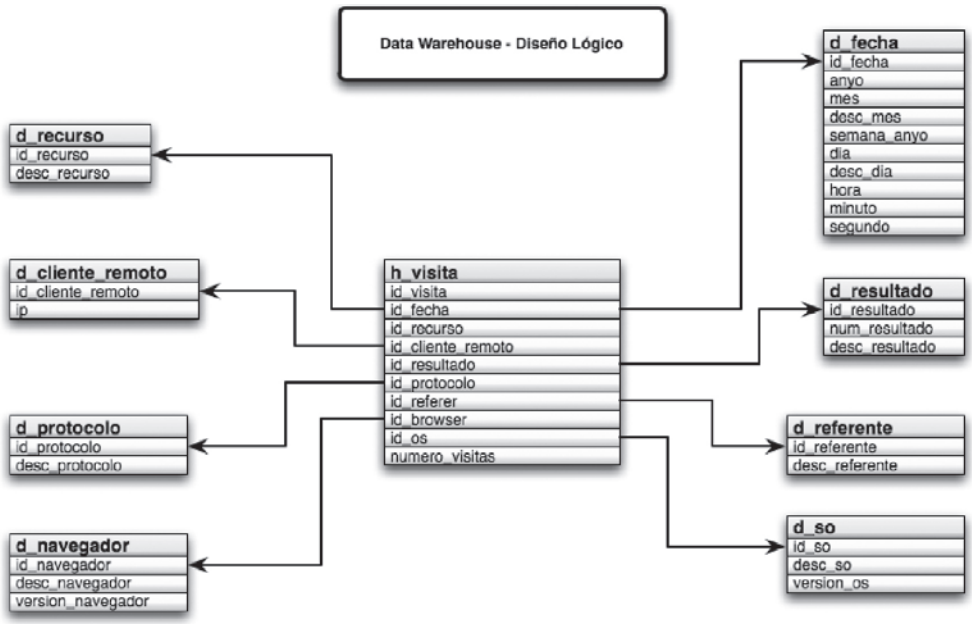
Así, de esta manera, para la tabla de hecho h_visita tenemos:

Tabla de hecho	Claves foráneas	Métricas
h_visita	id_os, id_resultado, id_temporal, id_recurso, id_browser, id_referer, id_cliente_remoto	Número de visitas

Y los atributos de cada una de las dimensiones son:

Dimensión	Clave primaria	Atributos
d_recurso	id_recurso	desc_recurso
d_cliente_remoto	id_cliente_remoto	ip
d_protocolo	id_protocolo	desc_protocolo
d_navegador	id_navegador	desc_navegador, versio_navegador
d_fecha	id_fecha	Anyo, mes, desc_mes, semana_anyo, dia, desc_dia, hora, minute, segundo
d_resultado	id_resultado	desc_resultado, num_resultado
d_referente	id_referente	desc_navegador, versio_navegador

Por lo que el diseño lógico resultante es el siguiente esquema en estrella:



3.3. Modelo físico de datos

El siguiente paso es el diseño físico. En nuestro caso, trabajaremos con MySQL como la base de datos que será usada como data warehouse y con una herramienta de modelización de base de datos. En este caso particular vamos a usar MySQL Workbench.[3]

Un data warehouse está formado por una colección de tablas. El objetivo es definir, para cada tabla, el formato de cada clave y atributo.

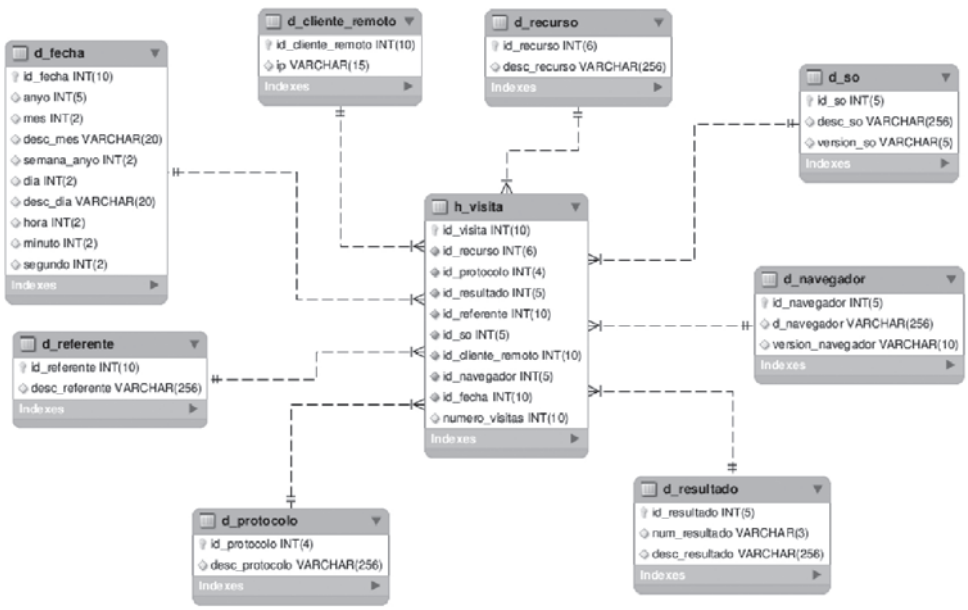
Debemos recordar los siguientes criterios:

- Se recomienda que las claves sean enteros y que sean independientes de las fuentes de origen.

3. MySQL Workbench es una herramienta de modelización de base de datos optimizada para los diferentes motores de MySQL. Permite reingenieria inversa. Es la herramienta de modelización que se encuentra instalada en la imagen virtual. Si bien eso no es limitante para usar otra.

- Las métricas pueden ser aditivas (números), semiaditivas (con particularidades en el momento de acumular las cantidades) o no aditivas (que entonces estamos hablando de atributos cualitativos). En las tablas de hecho deberíamos decantarnos por que todas las métricas fueran de las aditivas.
- En un caso real, sería necesario incluir campos que permitieran la trazabilidad del dato, por ejemplo fecha de carga, fecha de modificación, autor, fuente de origen. Para simplificar el modelo, no se incluyen.

Si consideramos cada una de las tablas, entonces el diseño físico resultante es:



Este diseño es un ejemplo que puede extenderse para responder muchas más preguntas y, por lo tanto, incluir mucha más información consolidada.

4. Glosario

BI	Business Intelligence
DSS	Decision Support Systems
EII	Enterprise Information Integration
ETL	Extract, Transform and Load
IBM	International Business Machines
KPI	Key Performance Indicator
KGI	Key Goal Indicator
ODS	Operational Data Store
OLAP	On-Line Analytical Processing
SI	Sistemas de Información
SQL	Structured Query Language
SCD	Slowly Changing Dimension
TI	Tecnologías de la Información
URL	Uniform Resource Locator

5. Bibliografía

BOUMAN, R., y VAN DONGEN, J. (2009). *Pentaho® Solutions: Business Intelligence and Data Warehousing with Pentaho® and MySQL.* Indianapolis: Wiley Publishing.

INMON, W. H. (2005). *Building the Data Warehouse,* 4th Edition. Hoboken: John Wiley & Sons.

INMON, W. H., STRAUSS, D., y NEUSHLOSS, G. (2008). *DW 2.0: The Architecture for the next generation of Data Warehousing.* Burlington: Morgan Kaufman Series.

KIMBALL, R. (2009). *Data Warehouse Toolkit Classics: The Data Warehouse Toolkit, 2nd Edition; The Data Warehouse Lifecycle Toolkit, 2nd Edition; The Data Warehouse ETL Toolkit.* Hoboken: John Wiley & Sons.

KNIGHT, B. (2009). *Professional Microsoft SQL Server 2008 Integration Services.* Indianapolis: Wrox.

Capítulo III

Diseño de procesos ETL

En un contexto empresarial, la integración puede darse en cuatro grandes áreas:

- **Integración de datos:** proporciona una visión única de todos los datos de negocio, se encuentren donde se encuentren. Éste es el ámbito del presente documento y, en particular, en el contexto de la inteligencia de negocio.
- **Integración de aplicaciones:** proporciona una visión unificada de todas las aplicaciones tanto internas como externas a la empresa. Esta integración se consigue mediante la coordinación de los flujos de eventos (transacciones, mensaje o datos) entre aplicaciones.
- **Integración de procesos de negocio:** proporciona una visión unificada de todos los procesos de negocio. Su principal ventaja es que las consideraciones de diseño del análisis e implementación de los procesos de negocio son aislados del desarrollo de las aplicaciones.
- **Integración de la interacción de los usuarios:** proporciona una interfaz segura y personalizada al usuario del negocio (datos, aplicaciones y procesos de negocio).

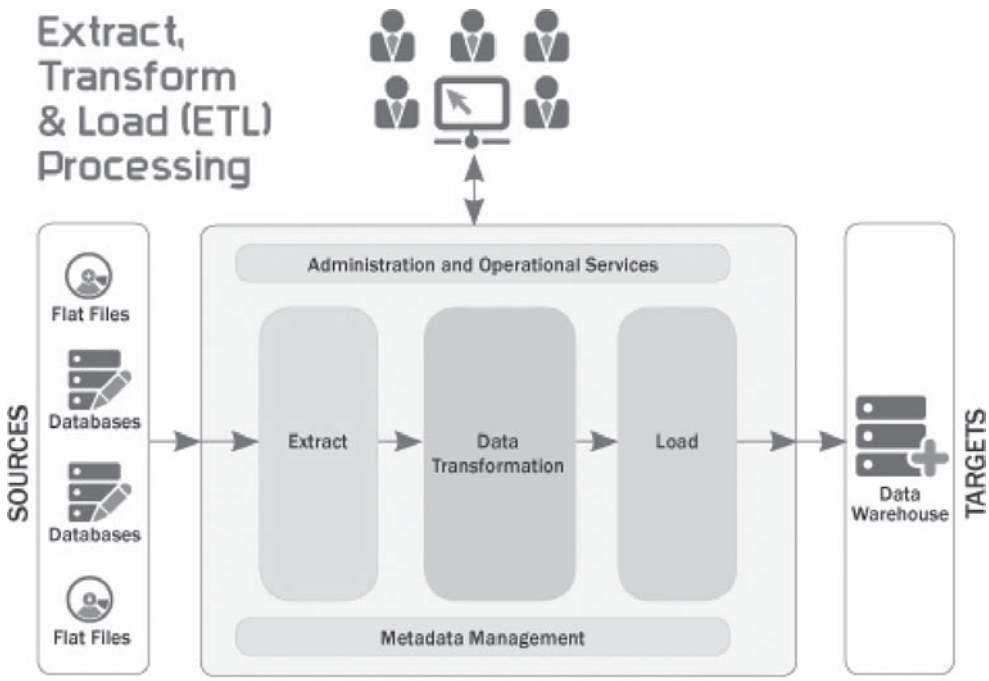
Este capítulo se centrará en la integración de datos en general y en los procesos ETL (Extracción, Transformación y Carga) en particular, que es una de las tecnologías de integración de datos que se usa en los proyectos de implantación de Business Intelligence.

El objetivo de este capítulo es conocer las diferentes opciones de integración de datos en el ámbito de la inteligencia de negocio y, en particular, conocer el diseño de procesos ETL.

1. Integración de datos: ETL

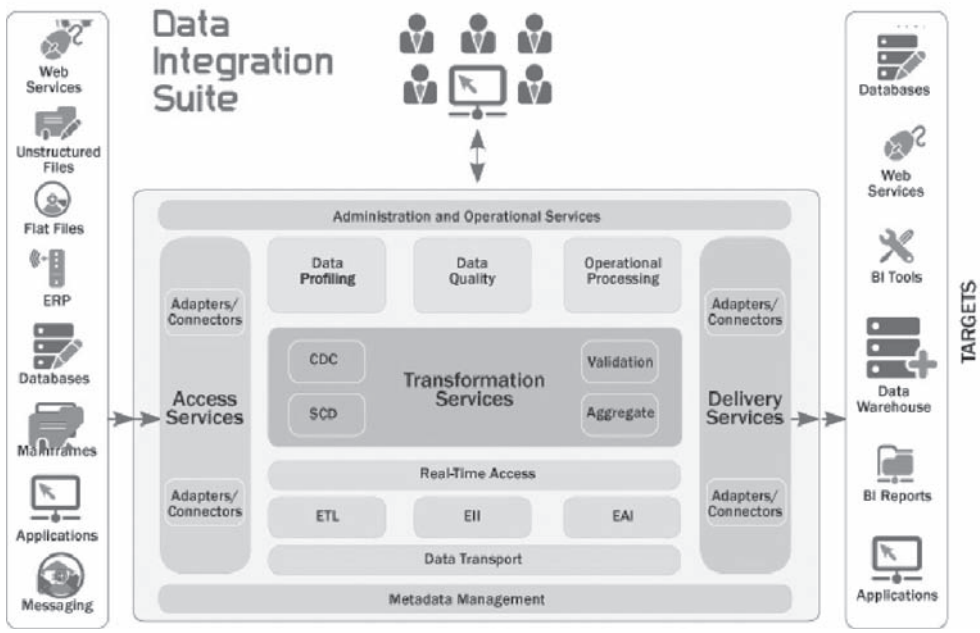
En el contexto de la inteligencia de negocio, las herramientas ETL han sido la opción usual para alimentar el data warehouse. La funcionalidad básica de estas herramientas está compuesta por:

- Gestión y administración de servicios.
- Extracción de datos.
- Transformación de datos.
- Carga de datos.
- Gestión de datos.



En los últimos años, estas herramientas han evolucionado incluyendo más funcionalidades propias de una herramienta de integración de datos. Podemos destacar:

- Servicios de acceso/entrega de datos (vía adaptadores/conectores).
- Gestión de servicios.
- Data profiling.
- Data quality.
- Procesos operacionales.
- Servicios de transformación: CDC, SCD, validación, agregación.
- Servicios de acceso a tiempo real.
- Extract, Transform and Load (ETL).
- Enterprise Information Integration (EII).
- Enterprise Application Integration (EAI).
- Capa de transporte de datos.
- Gestión de metadatos.



Esta evolución es consecuencia de diversos motivos, entre los que podemos destacar los diferentes tipos de datos existentes:

- Estructurados: contenidos en bases de datos.
- Semiestructurados: en formatos legibles para máquinas, si bien no están completamente estructurados: HTML tabulado, Excel, CSV..., que pueden obtenerse mediante técnicas estándar de extracción de datos.
- No estructurados: en formatos legibles para humanos, pero no para máquinas: Word, HTML no tabulado, PDF..., que pueden obtenerse mediante técnicas avanzadas como text mining u otras.

Así como la evolución de las necesidades de negocio.

Por ello, el punto de partida adecuado es definir formalmente el concepto de integración de datos.

Se entiende por **integración de datos** al conjunto de aplicaciones, productos, técnicas y tecnologías que permiten una visión única consistente de nuestros datos de negocio.

Respecto la definición:

- Las aplicaciones son soluciones a medida que permiten la integración de datos en base al uso de productos de integración.
- Los productos comerciales desarrollados por terceros capacitan la integración mediante el uso de tecnologías de integración.
- Las tecnologías de integración son soluciones para realizar la integración de datos.

1.1. Técnicas de integración de datos

Existen diferentes técnicas de integración de datos:

- **Propagación de datos:** consiste en copiar datos de un lugar de origen a un entorno destino local o remoto. Los datos pueden extraerse del origen mediante programas que generen un fichero que debe ser transportado al destino, donde se utilizará como fichero de entrada para cargar en la base de datos de destino. Una aproximación más eficiente es descargar sólo los datos que han cambiado en origen respecto a la última propagación reali-

zada, generando un fichero de carga incremental que también será transportado al destino. Este tipo de procesos son habitualmente de tipo en línea y trabajan con una arquitectura de push.[1] Puede realizarse como:

– Distribución.

– Intercambio bidireccional. Puede ser master-slave o peer-to-peer.

- **Consolidación de datos:** consiste en capturar los cambios realizados en múltiples entornos origen y propagarlos a un único entorno destino, donde se almacena una copia de todos estos datos. Ejemplos son un data warehouse o un ODS, alimentado por varios entornos de producción. Con esta técnica es difícil trabajar con tiempos de latencia[2] bajos:
  - Cuando no se requiere latencia baja, se suele proveer los datos mediante procesos batch en intervalos prefijados (superior a varias horas). Se usan consultas SQL para conseguir los datos (lo que se denomina técnica pull).
  - Cuando se requiere latencia baja, se utiliza la técnica push. En este caso, la aplicación de integración de datos debe identificar los cambios producidos en origen para transmitir sólo esos cambios, y no todo el conjunto de datos del origen. Para ello, se suele emplear algún tipo de técnica de tipo CDC (change data capture).
- **Federación de datos:** proporciona a las aplicaciones una visión lógica virtual común de una o más bases de datos. Esta técnica permite acceder a diferentes entornos origen de datos, que pueden estar en los mismos o en diferentes gestores de datos y máquinas, y crear una visión de este conjunto de bases de datos como si fuese en la práctica una base de datos única e integrada. Cuando una aplicación de negocio lanza una consulta SQL contra esta vista virtual, el motor de federación de datos descompone la consulta en consultas individuales para cada uno de los orígenes de datos físicos involucrados y la lanza contra cada uno de ellos. Cuando ha recibido todos los datos respuesta a las consultas, integra los resultados parciales en un resultado único, realizando las sumarizaciones, agregaciones y/o ordenaciones necesarias para resolver la consulta original, y devuelve los

1. La técnica push consiste en la actualización continua en linea del entorno destino mediante aplicaciones de integración de datos que capturan los cambios en origen y los transmiten a destino, donde son almacenados, en la que los datos son automáticamente enviados al entorno remoto.

2. En redes informáticas de datos, se denomina latencia a la suma de retardos temporales dentro de una red. Un retardo es producido por la demora en la propagación y trasmisión de paquetes dentro de la red.

datos a la aplicación que lanzó la consulta original. Uno de los elementos clave del motor de federación es el catálogo de datos común. Este catálogo contiene información sobre los datos: su estructura, su localización y, en ocasiones, su demografía (volumen de datos, cardinalidad de las claves, claves de clustering, etc.). Ello permite que se pueda optimizar la división de la consulta original al enviarla a los gestores de bases de datos, y que se elija el camino más eficiente de acceso global a los datos.

- **CDC (Change Data Capture):** se utilizan para capturar los cambios producidos por las aplicaciones operacionales en las bases de datos de origen, de tal manera que pueden ser almacenados y/o propagados a los entornos destino para que éstos mantengan la consistencia con los entornos origen. A continuación trataremos las cuatro principales técnicas de change data capture.
  - CDC por aplicación: consiste en que la propia aplicación es la que genera la actualización de datos en origen, y se encarga de actualizar directamente los entornos destino, o almacenar localmente los cambios en una tabla de paso (staging) mediante una operación de INSERT dentro de la misma unidad lógica de trabajo.
  - CDC por timestamp: se puede emplear cuando los datos de origen incorporan un timestamp (por ejemplo a nivel de fila si el origen es una tabla relacional) de la última actualización de ésta. El CDC se limitará a escanear los datos de origen para extraer los datos que posean un timestamp posterior al de la última vez que se ejecutó el proceso de CDC: estos datos son los que han cambiado desde la última captura de datos y, por tanto, son los que deben actualizarse en los entornos destino.
  - CDC por triggers: los triggers o disparadores son acciones que se ejecutan cuando se actualizan (por UPDATE, DELETE o INSERT) los datos de una determinada tabla sobre la que están definidos. Estos triggers pueden utilizar estos datos de la actualización en sentencias SQL para generar cambios SQL en otras tablas locales o remotas. Por lo tanto, una forma de capturar cambios es crear triggers sobre las tablas de origen, cuyas acciones modifiquen los datos de las tablas destino.
  - CDC por captura de log: consiste en examinar constantemente el fichero de log de la base de datos de origen en busca de cambios en las tablas que se deben monitorizar. Estos programas basan su eficiencia en la

lectura de los buffers de memoria de escritura en el log, por lo que la captura de la información no afecta al rendimiento del gestor relacional al no requerir acceso al disco que contiene el fichero de log.

- **Técnicas híbridas:** la técnica elegida en la práctica para la integración de datos dependerá de los requisitos de negocio para la integración, pero también en gran medida de los requisitos tecnológicos y de las probables restricciones presupuestarias. A la práctica se suelen emplear varias técnicas de integración constituyendo lo que se denomina una técnica híbrida.

1.2. Tecnologías de integración de datos

Existen diferentes tecnologías de integración de datos basadas en las técnicas presentadas:

- ETL: permite extraer datos del entorno origen, transformarlos según nuestras necesidades de negocio para integración de datos y cargar estos datos en los entornos destino. Los entornos origen y destino son usualmente bases de datos y/o ficheros, pero en ocasiones también pueden ser colas de mensajes de un determinado middleware, así como ficheros u otras fuentes estructuradas, semiestructuradas o no estructuradas. Está basada en técnicas de consolidación. Las herramientas de ETL en la práctica mueven o transportan datos entre entornos origen y destino, pero también documentan cómo estos datos son transformados (si lo son) entre el origen y el destino almacenando esta información en un catálogo propio de metadatos; intercambian estos metadatos con otras aplicaciones que puedan requerirlos y administran todas las ejecuciones y procesos de la ETL: planificación del transporte de datos, log de errores, log de cambios y estadísticas asociadas a los procesos de movimiento de datos. Este tipo de herramientas suelen tener una interfaz de usuario de tipo GUI y permiten diseñar, administrar y controlar cada uno de los procesos del entorno ETL.
  - ETL de generación de código: constan de un entorno gráfico donde se diseñan y especifican los datos de origen, sus transformaciones y los entornos destino. El resultado generado es un programa de tercera generación (típicamente COBOL) que permite realizar las transformaciones de datos. Aunque estos programas simplifican el proceso

ETL, incorporan pocas mejoras en cuanto al establecimiento y automatización de todos los flujos de procesos necesarios para realizar la ETL. Usualmente son los administradores de datos los encargados de distribuir y administrar el código compilado, planificar y ejecutar los procesos en lotes, y realizar el transporte de los datos.

– ETL basados en motor: permite crear flujos de trabajo en tiempo de ejecución definidos mediante herramientas gráficas. El entorno gráfico permite hacer un mapping de los entornos de datos de origen y destino, las transformaciones de datos necesarios, el flujo de procesos y los procesos por lotes necesarios. Toda esta información referente a diseño y procesos del ETL es almacenada en el repositorio del catálogo de metadatos. Se compone por diversos motores:
    a) Motor de extracción: utiliza adaptadores como ODBC, JDBC, JNDI, SQL nativo, adaptadores de ficheros planos u otros. Los datos pueden ser extraídos en modo pull planificado, típicamente soportando técnicas de consolidación en proceso por lotes, o mediante modo push, típicamente utilizando técnicas de propagación en procesos de tipo en línea. En ambos casos se pueden utilizar técnicas de changed data capture (CDC) ya vistas.
    b) Motor de transformación: proporciona una librería de objetos que permite a los desarrolladores transformar los datos de origen para adaptarse a las estructuras de datos de destino, permitiendo, por ejemplo, la sumarización de los datos en destino en tablas resumen.
    c) Motor de carga: utiliza adaptadores a los datos de destino, como el SQL nativo, o cargadores masivos de datos para insertar o actualizar los datos en las bases de datos o ficheros de destino.
    d) Servicios de administración y operación: permiten la planificación, ejecución y monitorización de los procesos ETL, así como la visualización de eventos y la recepción y resolución de errores en los procesos.
– ETL integrado en la base de datos: algunos fabricantes incluyen capacidades ETL dentro del motor de la base de datos (al igual que lo hacen con otro tipo de características, como soporte OLAP y minería de datos). En general, presentan menos funcionalidades y comple-

jidad, y son una solución menos completa que los ETL comerciales basados en motor o de generación de código. Por ello, a los ETL integrados en base de datos se les clasifica en tres clases en relación con los ETL comerciales (basados en motor o de generación de código):

e) ETL cooperativos: con ellos, los productos comerciales pueden usar funciones avanzadas del gestor de base de datos para mejorar los procesos de ETL. Ejemplos de ETL cooperativos son aquellos que pueden utilizar procedimientos almacenados y SQL complejo para realizar las transformaciones de los datos en origen de una forma más eficiente, o utilizar paralelismo de CPU en consultas para minimizar el tiempo de los procesos ETL.

f) ETL complementarios: cuando los ETL de bases de datos ofrecen funcionalidades complementarias a los ETL comerciales. Por ejemplo, hay gestores de bases de datos que ofrecen soporte a MQT (Materializad Query Tables) o vistas de sumarización precalculadas, mantenidas y almacenadas por el gestor que pueden usarse para evitar transformaciones de datos realizadas por el ETL comercial. Además, otros gestores permiten la interacción directa mediante SQL con middleware de gestión de mensajes (por ejemplo, leyendo una cola de mensajes mediante una UDF o permitiendo la inserción de nuevos mensajes en colas mediante SQL) o con aplicaciones que se comunican mediante web services.

g) ETL competitivos: algunos gestores ofrecen herramientas gráficas integradas que explotan sus capacidades ETL en lo que claramente es competencia con los ETL comerciales.

- EII: el objetivo de la tecnología EII es permitir a las aplicaciones el acceso a datos dispersos (desde un data mart hasta fichero de texto o incluso web services) como si estuviesen todos residiendo en una base de datos común. Por lo tanto se basa en la federación. El acceso a datos dispersos implica la descomposición de la consulta inicial (habitualmente en SQL) direccionada contra la vista virtual federada en subcomponentes, que serán procesados en cada uno de los entornos donde residen los datos. Se recogen los resultados individuales de cada uno de los subcomponentes de la consulta, se combinan adecuadamente y se devuelve el resultado a la aplicación que lanzó la consulta. Los productos de EII han evolucionado

desde dos entornos origen diferenciados: las bases de datos relacionales y las bases de datos XML. Actualmente, la tendencia en productos EII es que soporten ambas interfaces a datos, SQL (ODBC y JDBC) y XML (XQuery y XPath). Los productos comerciales que implementan EII varían considerablemente en las funcionalidades que aportan; el área más diferenciadora es la optimización de las consultas distribuidas. Las características básicas de los productos que implementan soluciones de integración de datos EII son:

- Transparencia: los datos parecen estar en un origen único.
- Heterogeneidad: integración de datos de diferentes fuentes (relacionales, XML, jerárquicos) y también no estructurados.
- Extensibilidad: posibilidad de federar cualquier fuente de datos.
- Alta funcionalidad: acceso en lectura y escritura a cualquier fuente soportada.
- Autonomía: acceso no disruptivo para los datos o las aplicaciones.
- Rendimiento: posibilidad de optimizar las consultas dependiendo del tipo y fuente de datos.

- EDR: tiene el objetivo de detectar los cambios que suceden las fuentes de origen. Está soportada por las técnicas de integración de datos de CDC (Change Data Capture) y por la técnica de propagación de datos. Consta básicamente de los siguientes elementos:
  - Programa de captura: se encarga de recuperar los cambios producidos en la base de datos de origen. Esta captura puede ser realizada a través de una salida que lea constantemente el log de recuperación de la base de datos, a través de triggers o mediante una aplicación externa de usuario. El programa de captura se apoya en una serie de tablas donde se almacena información de control del proceso de captura, como por ejemplo las tablas que son orígenes de replicación.
  - Sistema de transporte: los sistemas de transporte más comunes son a través de tablas de paso (staging), que dan lugar a la denominada replicación de tipo SQL, o a través de un middleware de gestión de colas, la denominada queue-replication o Q-replication.
  - Programa de aplicación de cambios: es la pieza que, o bien lee mediante SQL de las tablas de staging los cambios pendientes de aplicar en la SQL-replication, o lee del sistema de colas en la Q-replication, y mediante la información de control almacenada en tablas realiza el

mapeo entre datos de origen y de destino, realiza las transformaciones necesarias a los datos y actualiza los datos de destino mediante SQL si se trata de destinos relacionales, o publica un registro XML para que pueda ser tratado por aplicaciones de propósito general.
– Programa de administración: permite las definiciones necesarias de origen de datos y destinos, mapeos, transformaciones y establecer los intervalos de aplicación de cambios. Usualmente es una herramienta de tipo gráfico.
– Utilidades: programas de utilidad que sirven para, por ejemplo, planificar una carga de datos inicial del destino a partir de los datos de origen.

1.3. Uso de la integración de datos

Los procesos de integración de datos se usan en múltiples tipologías de proyectos. Podemos destacar los siguientes:

- Migración de datos.
- Procesos de calidad de datos.
- Corporate Performance Management (CPM).
- Master Data Management (MDM).
- Customer Data Integration (CDI).
- Product Information Management (PIM).
- Enterprise Information Management (EIM).
- Data Warehousing.
- Business Intelligence (BI).

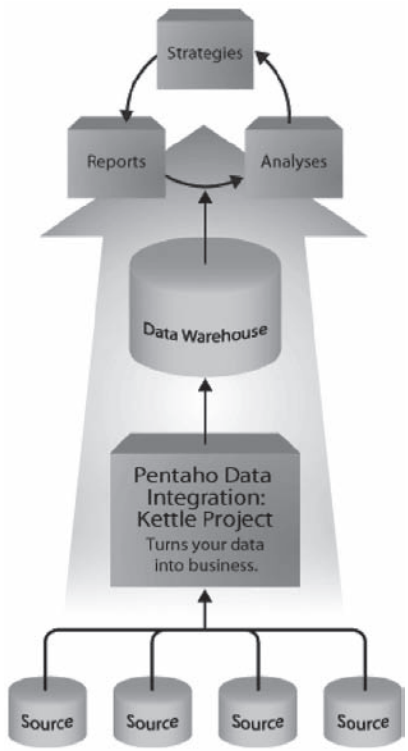
2. ETL en el contexto de Pentaho

Pentaho Data Integration (PDI), anteriormente llamado Kettle, fue iniciado en 2001 por Matt Casters. En el año 2006, Pentaho adquirió Kettle y lo renombró después de que éste pasara a ser open source. De esta forma continuaba con

la política de crear una suite completa de inteligencia de negocio open source. Matt Casters pasó a formar parte del equipo de Pentaho.

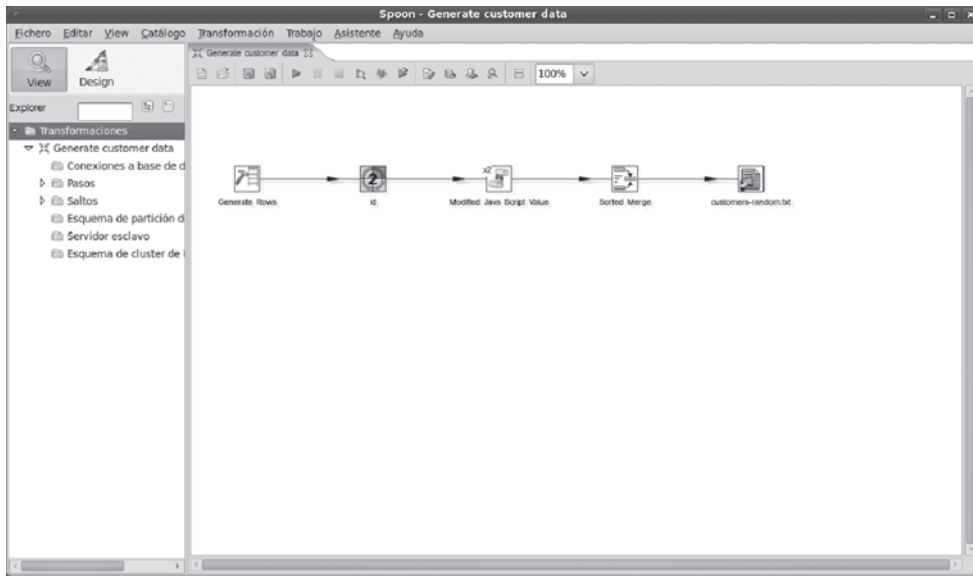
PDI es una solución de integración de datos programada en java orientada completamente al usuario y basada en un enfoque de metadatos. Los procesos ETL se encapsulan en metadatos que se ejecutan a través del motor ETL.

Esta herramienta permite cargar datos de múltiples fuentes de origen, cargar dichos datos en un data warehouse para que posteriormente la información consolidada sea de utilidad a nivel operativo, táctico y estratégico.



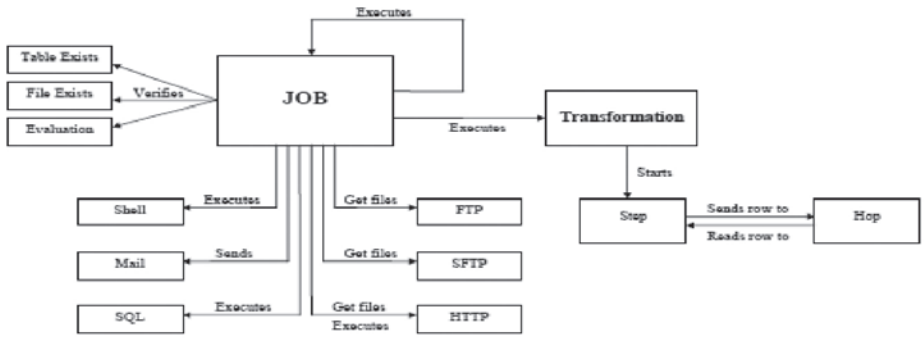
Las principales características de PDI son:

- Entorno gráfico orientado al desarrollo rápido y ágil basado en dos áreas: la de trabajo y la de diseño/vista.



- Multiplataforma.
- Incluye múltiples conectores a bases de datos, tanto propietarias como comerciales. Así como conectores a ficheros planos, Excel, XML u otros.
- Arquitectura extensible mediante plugins.
- Soporta uso de cluster, procesos ETL en paralelo y arquitecturas servidor maestro-esclavo.
- Completamente integrado con la suite de Pentaho.
- Basado en el desarrollo de dos tipos de objetos:
  - Transformaciones: permiten definir las operaciones de transformación de datos.
  - Trabajos: permiten gestionar y administrar procesos ETL a alto nivel.

El siguiente diagrama nos proporciona una idea de cómo se relacionan trabajos y transformaciones.

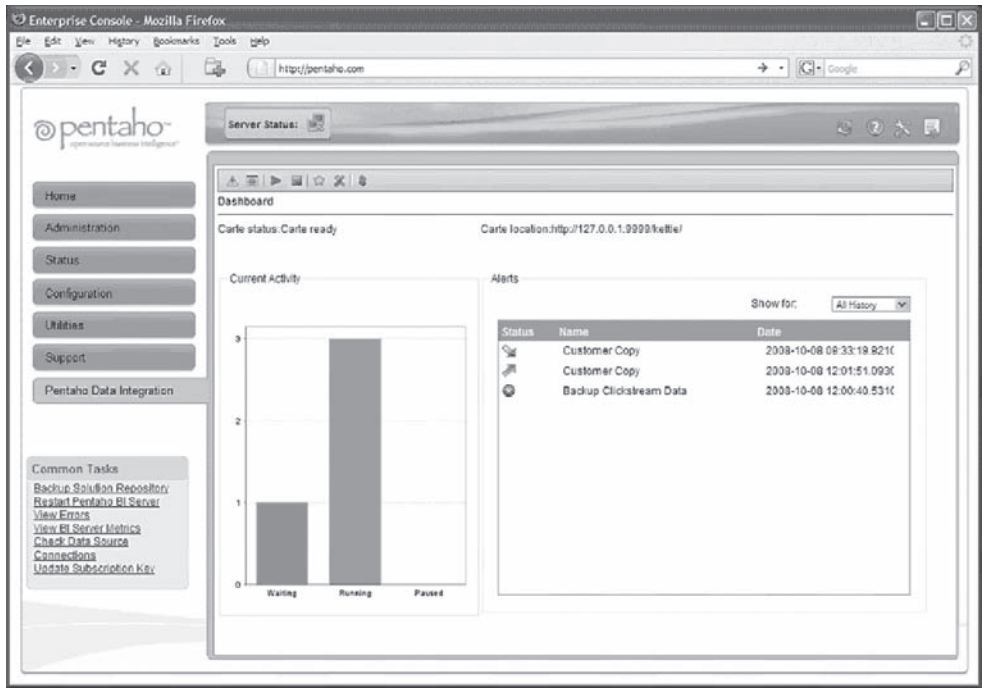


- Está formado por cuatro componentes:
  - Spoon: entorno gráfico para el desarrollo de transformaciones y trabajos.
  - Pan: permite ejecutar transformaciones.
  - Kitchen: permite ejecutar trabajos.
  - Carte: es un servidor remoto que permite la ejecución de transformaciones y trabajos.
- Pasos disponibles para trabajos:
  - Generales: permite iniciar un trabajo, ejecutar transformaciones o trabajos entre otras operaciones.
  - Correo: permite enviar correos, recuperar cuentas o validarlas.
  - Gestión de ficheros: permite realizar operaciones con ficheros como crear, borrar, comparar o comprimir.
  - Condiciones: permite realizar comprobaciones necesarias para procesos ETL como la existencia de un fichero, una carpeta o una tabla.
  - Scripting: permite crear scripts de JavaScript, SQL y Shell.
  - Carga bulk: permite realizar cargas bulk a MySQL, MSSQL, Acces y ficheros.
  - XML: permite validar XML y XSD.
  - Envío de ficheros: permite enviar o coger ficheros desde FTP y SFTP.
  - Repositorio: permite realizar operaciones con el repositorio de transformaciones y trabajos.
- Pasos disponibles para transformaciones:
  - Entrada: permite recuperar datos desde bases de datos (JDBC), Acces, CSV, Excel ficheros, LDAP, Mondrian, RSS u otros.

- Salida: permite cargar datos en bases de datos u otros formatos de salida.
- Transformar: permite realizar operaciones con datos como filtrar, ordenar, partir añadir nuevos campos, mapear...
- Utilidades: permite operar con filas o columnas y otras operaciones como enviar un email, escribir a un log.
- Flujo: permite realizar operaciones con el flujo de datos como fusionar, detectar flujos vacíos, realizar operaciones diferentes en función de una condición...
- Scripting: permiten crear scripts de JavaScript, SQL, expresiones regulares, fórmulas y expresiones java.
- Búsqueda de datos: permite añadir información al flujo de datos mediante la búsqueda en bases de datos y otras fuentes.
- Uniones: permite unir filas en función de diferentes criterios.
- Almacén de datos: permite trabajar con dimensiones SCD.
- Validación: permite validar tarjetas de crédito, datos, direcciones de correo o XSD.
- Estadística: permite realizar operaciones estadísicas sobre un flujo de datos.
- Trabajos: permite realizar operaciones própias de un trabajo.
- Mapeado: permite realizar el mapeo entre campos de entrada y salida.
- Embebido: permite realizar operaciones con sockets.
- Experimental: incluye los pasos en fase de validación.
- Obsoleto: incluye los pasos que desapareceran en la siguiente versión del producto.
- Carga bulk: permite realizar cargas bulk a Infobright, LucidDB, MonetDB y Oracle.
- Historial: recopila los pasos frecuentemente usados por el desarrollador.

Diferencias con la versión Enterprise:

- Inclusión de una consula web para la administración y monitorización de procesos ETL.



– Soporte profesional.

3. Caso práctico

3.1. Contexto

Con el objetivo de entender cómo se diseñan los procesos ETL, en el caso práctico se partirá de una situación real simplificada.

Consideremos que se administra una única aplicación que está alojada en un servidor apache. El archivo log resultante está en combined log format, lo que significa que los campos que incluye en cada línea del archivo son:

• Ip: desde la que se accede a un recurso de la aplicación.

- RFC 1413: identificador de la máquina en la red (uso interno). Este valor para aplicaciones web externas suele estar vacío.
- Usuario remoto: identificador del usuario. Sucede lo mismo que en el caso anterior.
- Fecha: en formato [dd/MM/yyyy:HH:mm:ss -XXXX].
- Recurso: aquello a lo que se accede.
- Resultado.
- Tiempo: segundos que se tarda en acceder al recurso.
- Referente: desde donde se accede al recurso.
- User-agent: información del sistema operativo y del navegador que han sido usados para acceder al recurso.

En nuestro caso, se prescindirá de la información que proporcionan los campos: RFC 1413, usuario remoto y tiempo.

También se considerará que, conociendo las necesidades informacionales, se ha preparado una colección de ficheros que contienen información que permitirá facilitar la carga de algunas de las dimensiones de nuestro data warehouse.

Resumiendo, la situación de partida está formada por cinco ficheros con los que se procederá a una carga inicial del data warehouse:

- access.log: que contiene la información de acceso a nuestra aplicación web.
- navegador.csv: que contiene un listado de navegadores base.
- protocolo.csv: que contiene los protocolos de acceso estándar.
- resultado.csv: que contiene el resultado que puede proporcionar el servidor a un acceso.
- so.cv: que contiene un listado de sistemas operativos base.

Estos últimos ficheros han sido creados partiendo de la situación normal, cuando las dimensiones se precargan en la carga inicial (la que se realizará en este caso).

La estrategia que se seguirá en el proceso ETL será:

1) Cargar las dimensiones navegador, protocolo, resultado y so a partir de los ficheros anteriores.
2) Complementar las dimensiones restantes a partir de la información presente en el fichero access.log y alimentar la tabla de hecho de visitas.
3) Crear un trabajo para lanzar todas las transformaciones de una manera única.

Se usará la siguiente notación:

- Para las transformaciones: TRA_ETL_INI_[nombre de la dimensión o tabla de hecho a cargar].
- Para los trabajos: JOB_CARGA_INI_[nombre de la dimensión o tabla de hecho a cargar].

3.2. Diseño con Pentaho Data Integration

Éste es el primer contacto con la herramienta de Pentaho. Antes de nada, es necesario estructurar el proyecto o solución de negocio. Las soluciones de negocio se guardan dentro de la carpeta pentaho-solutions.[3]

De esta manera creamos una carpeta padre llamada AEW (Análisis de Estadísticas Web), y en ella, diversas carpetas que contienen los materiales de los diferentes módulos:

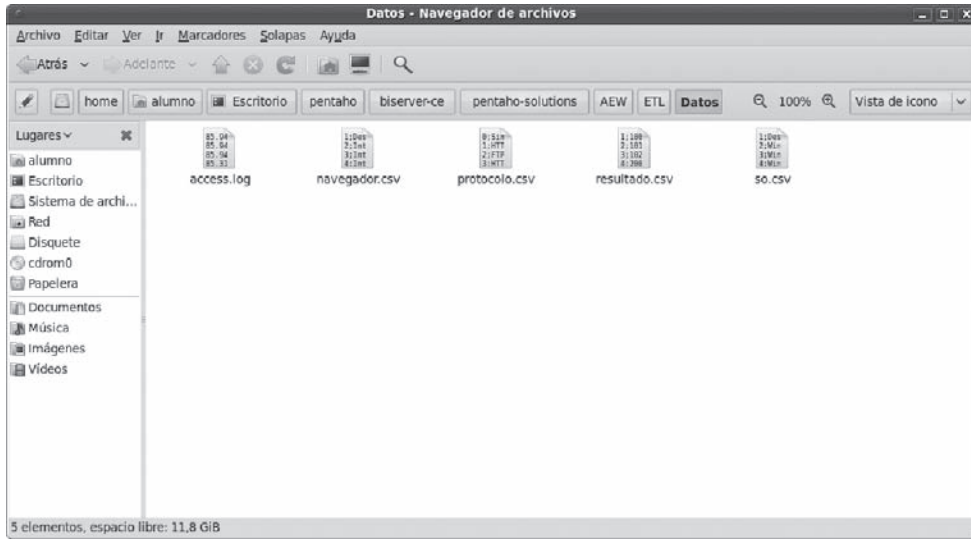
- ETL
- OLAP
- Reporting
- Dashboard

En el caso de Pentaho, se puede trabajar de dos maneras: mediante el repositorio de datos (que permite guardar las transformaciones en base de datos) o mediante ficheros. Trabajaremos de esta segunda manera para compartir de manera más fácil los cambios.

Dentro de la carpeta ETL existen dos carpetas llamadas datos y procesos ETL; su propio nombre describe su funcionalidad.

Dentro de datos tenemos los cinco ficheros presentados en el contexto.

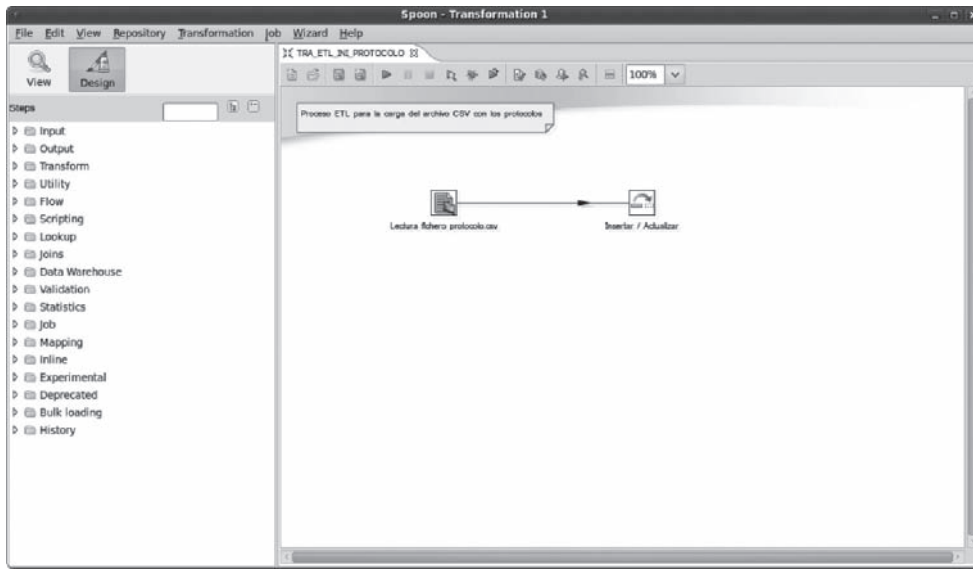
3. Se puede acceder a la carpeta pentaho-solutions desde el escritorio, a través del enlace directo llamado Pentaho, y luego entrando en biserver-ce.



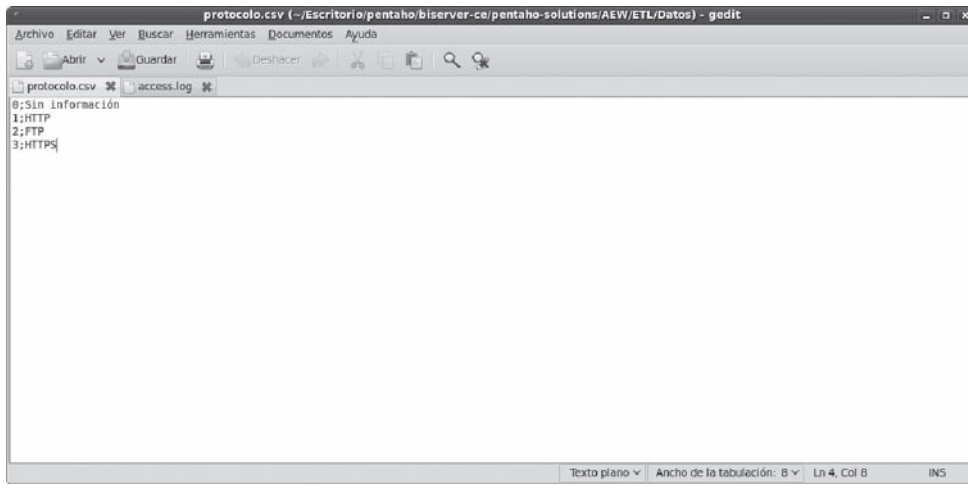
Siguiendo la estrategia que hemos definido anteriormente, el primer paso es la carga de los ficheros CSV. Son ficheros ya preparados en el formato de la base de datos, de manera que cargan directamente. Así, para el resto es suficiente haber entendido cómo se ha creado uno de los procesos ETL.

La transformación ETL llamada TRA_ETL_INI_PROTOCOLO tiene dos pasos:

- Lectura del fichero CSV.
- Insertar/actualizar de la base de datos a partir de la información extraída del fichero.

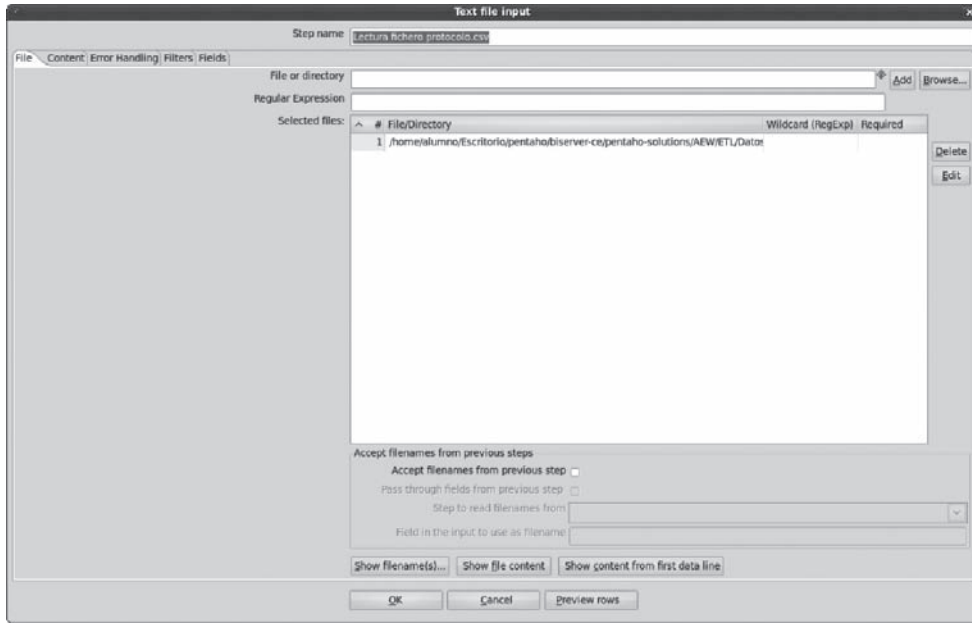


El fichero tiene esta forma:

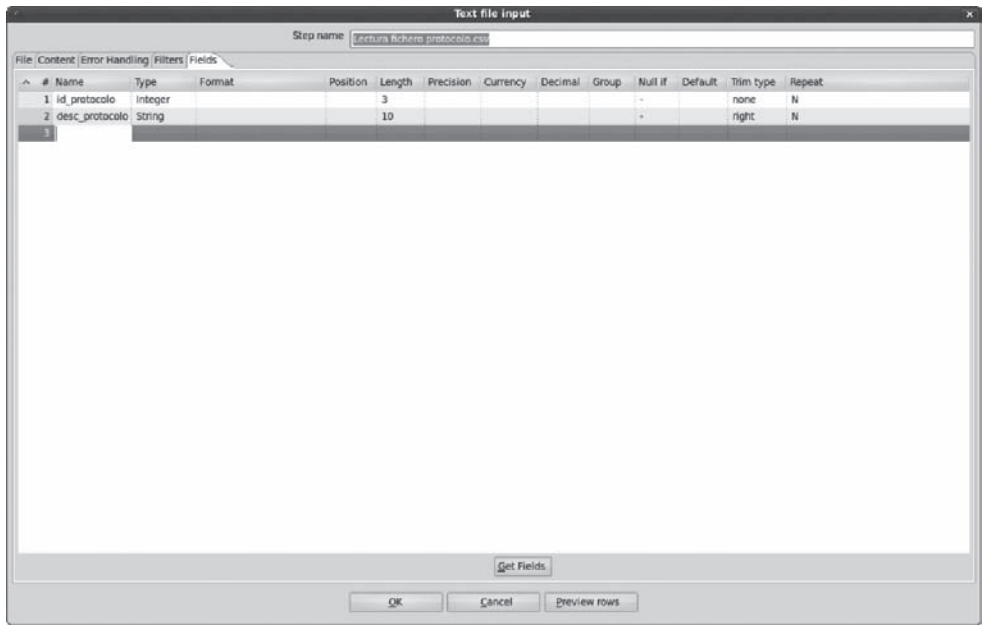


Para parametrizar el paso de lectura se usa el paso *text file input,* y entonces:

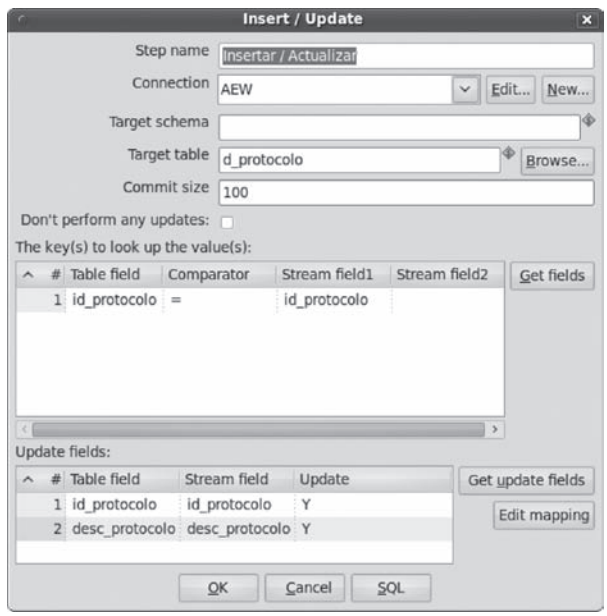
- Se define el fichero que es la fuente de origen en la pestaña file.



- Se define cuál es el separador y cómo está encapsulado el texto en la pestaña content.
- Se definen los campos a cargar en la campaña fields. Para facilitar la tareas se pulsa el botón get fields.



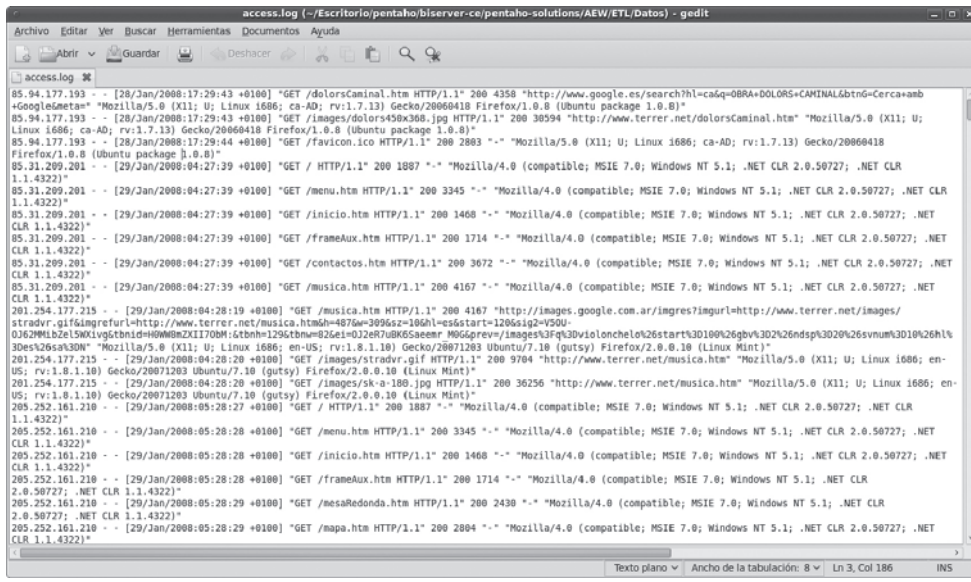
- Después se inserta la información mediante el paso insert/update, cuya parametrización:



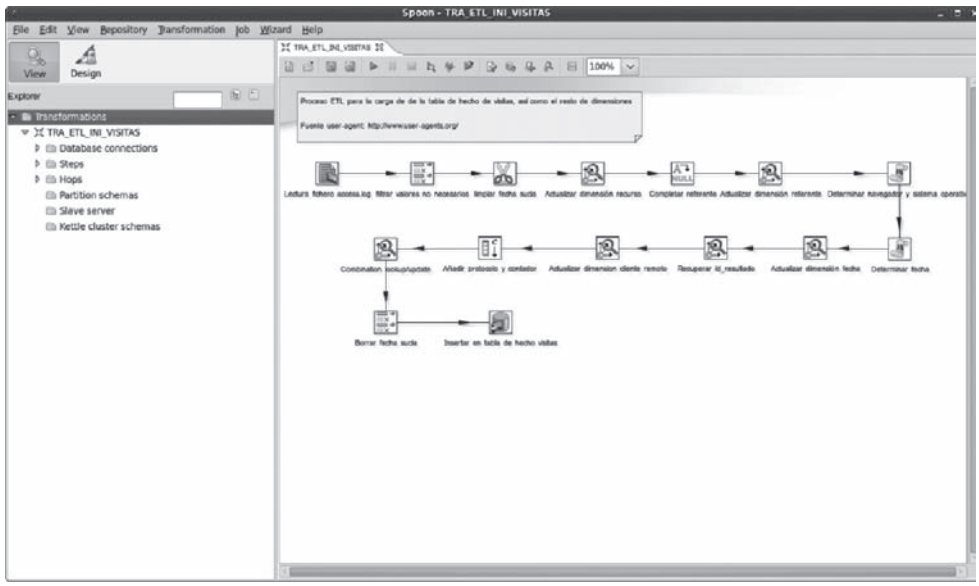
De manera equivalente se realizan las otras dimensiones que se cargan a partir de los ficheros CSV: navegador, resultado y so.

Una vez cargadas estas cuatro dimensiones, a partir del fichero access.log se cargará la información de las dimensiones restantes primero y posteriormente la tabla de hecho.

La información del fichero access.log está en la forma de un log de un servidor de Apache.

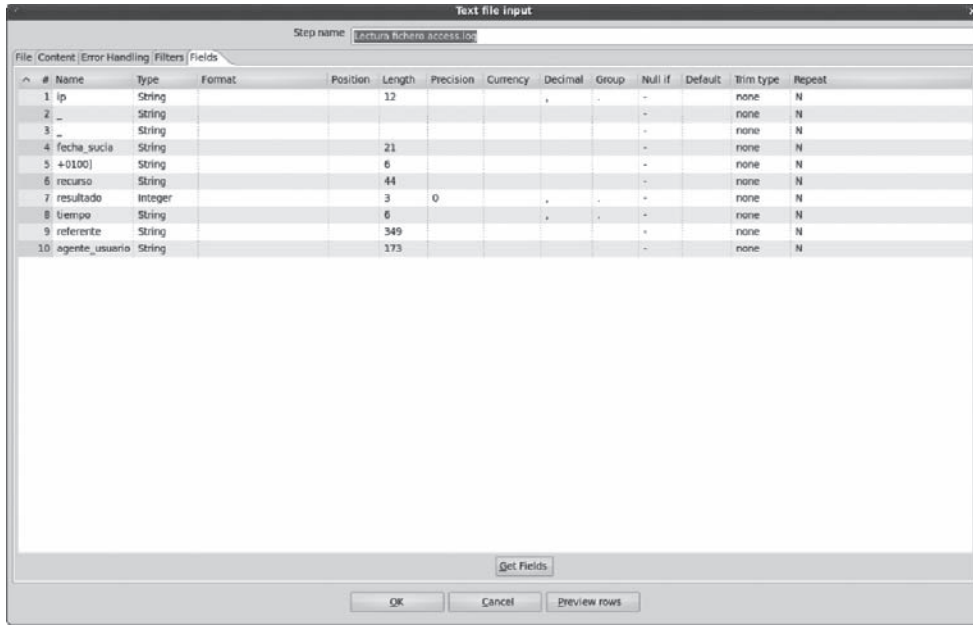


La transformación creada carga el resto de dimensiones y la tabla de hecho, y su forma es:

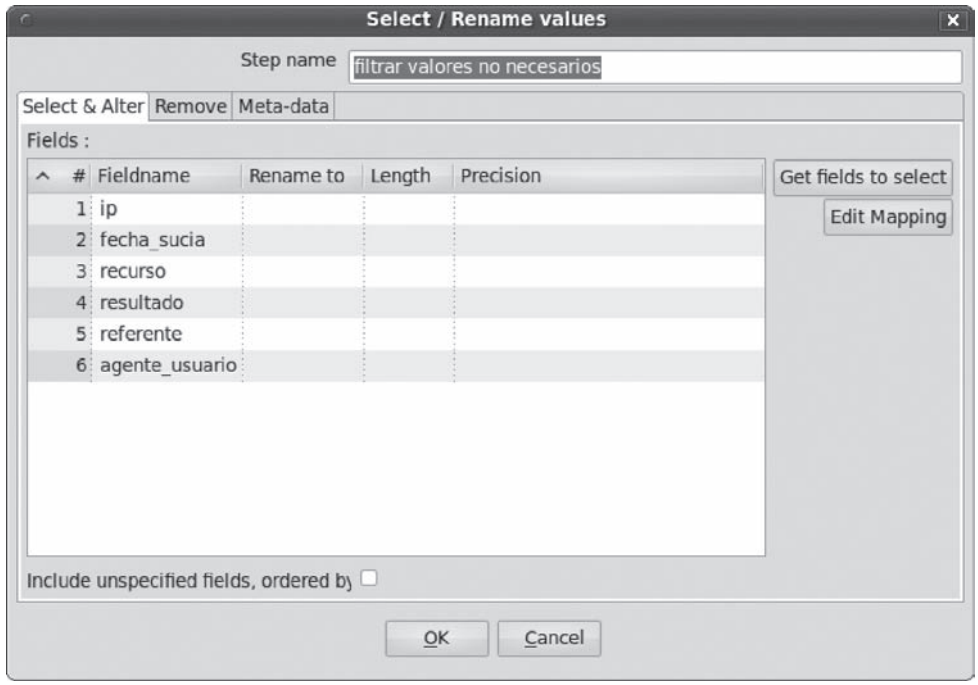


Analicemos los diferentes pasos que la componen:

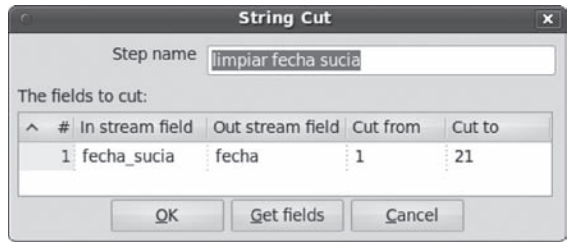
- Lectura del fichero CSV con el que recuperamos los campos que forman parte del flujo de información usando el paso *text file input*.



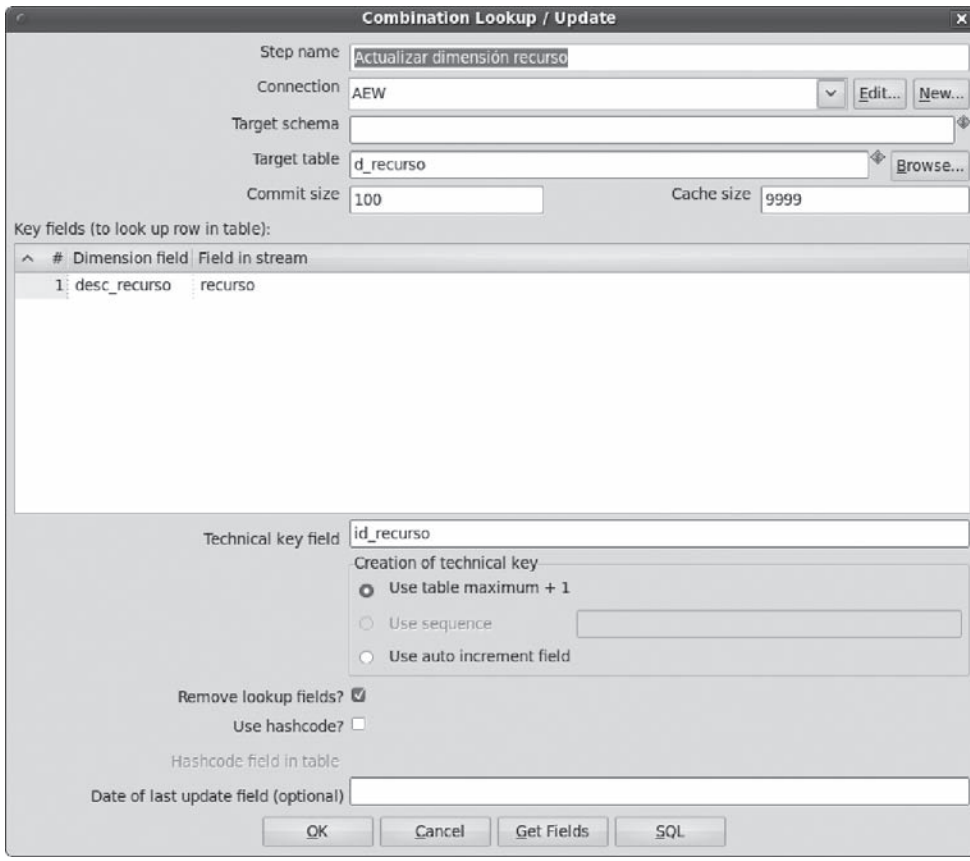
- Dado que existe información que no será usada, se descarta mediante el paso *select/rename values* (los campos que no se seleccionan desaparecen del flujo).



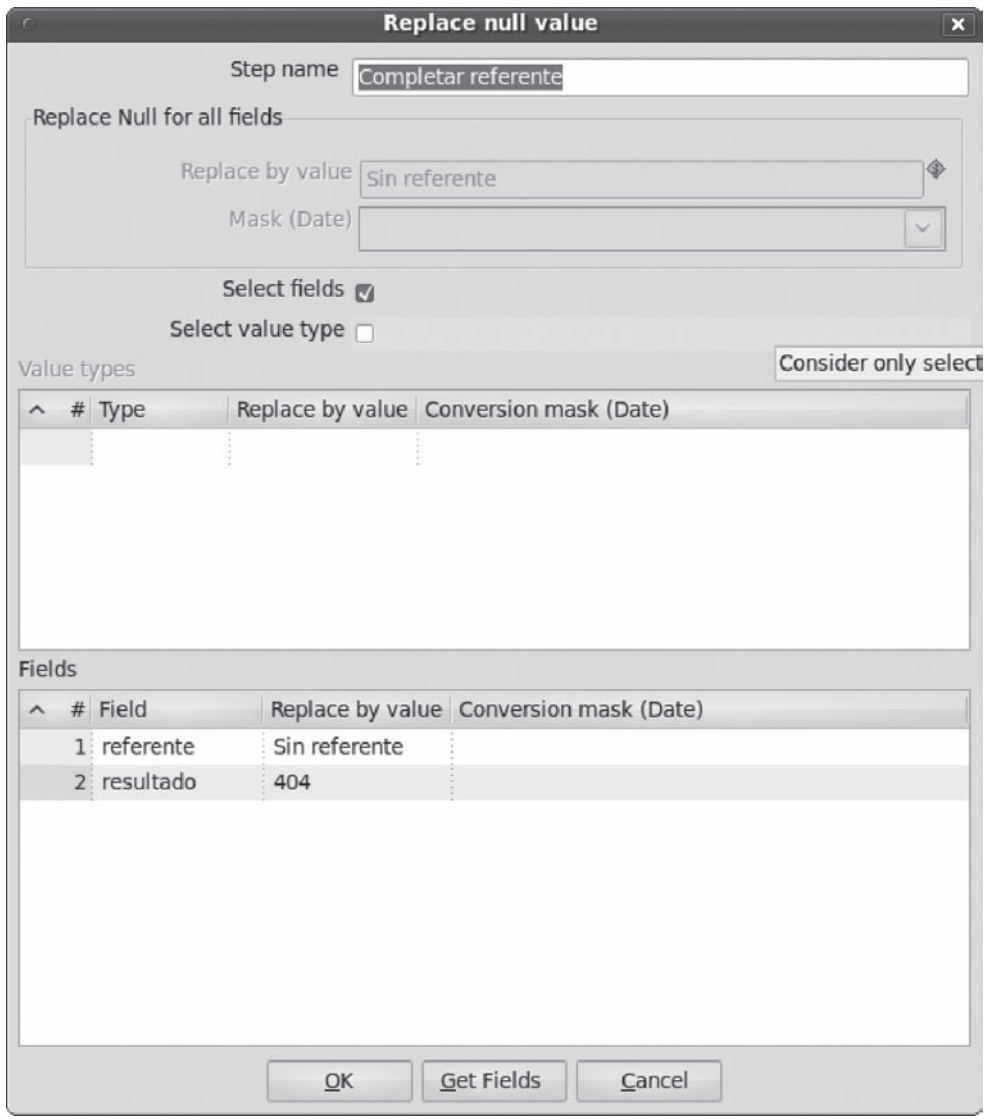
- La fecha se ha extraído como una cadena y se borra el carácter sobrante mediante el paso *string cut*.



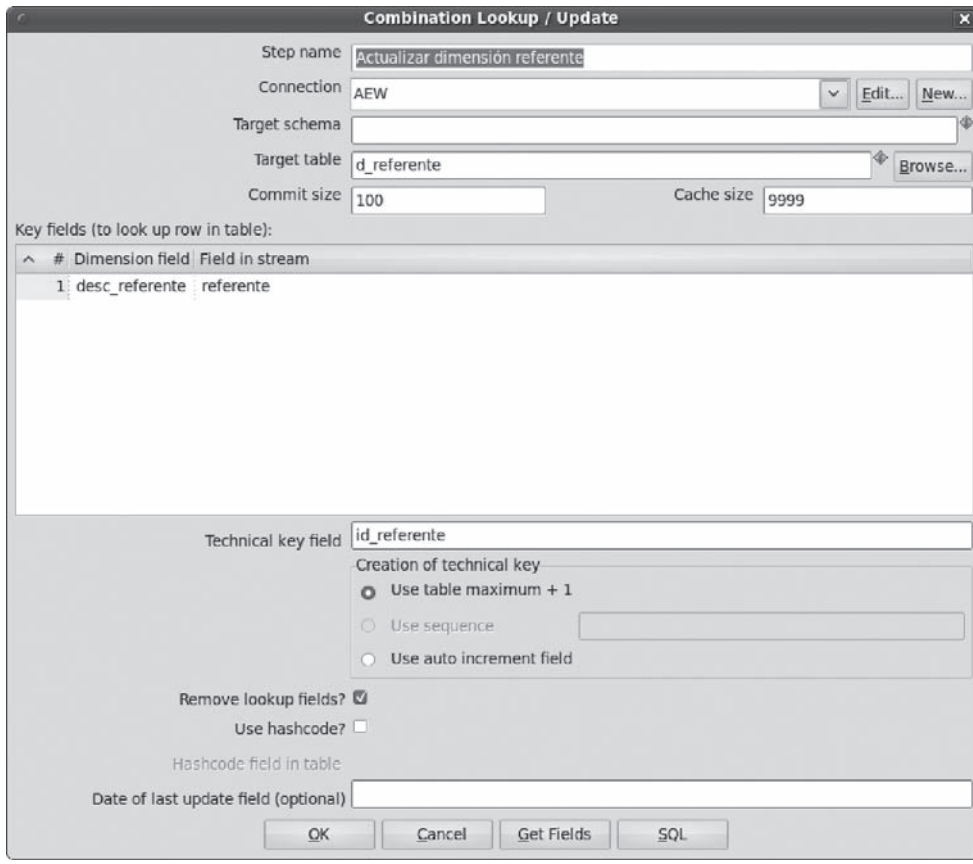
- Se actualiza la dimensión recurso a partir de la información entrante y se recupera el id_recurso mediante el paso *combination lookup/update*, que hace una búsqueda o actualiza.



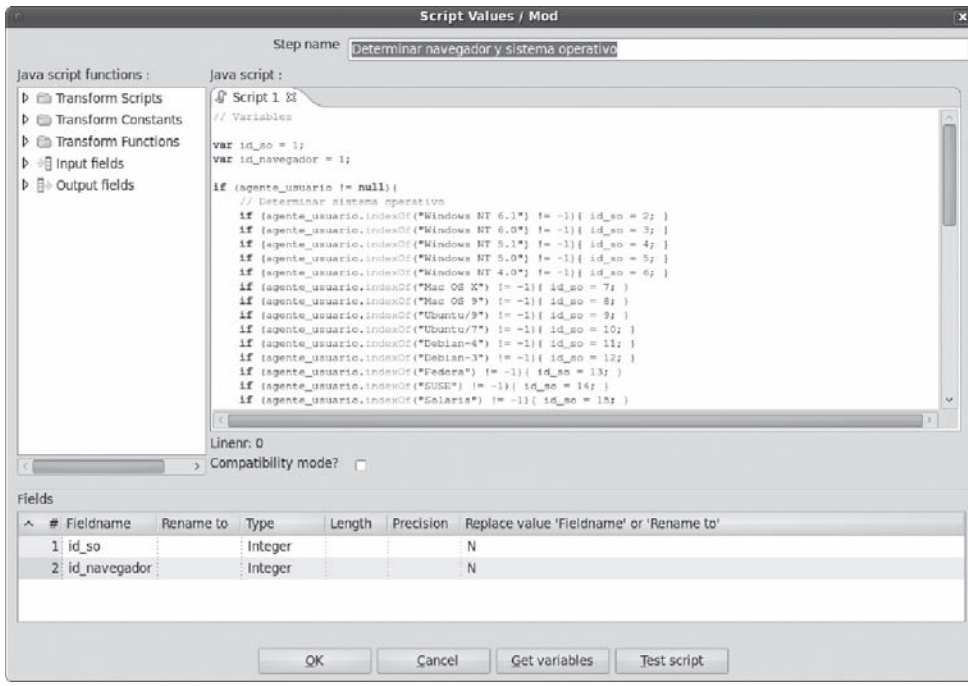
- Completamos los nulos del campo referente mediante el paso *replace null value*.



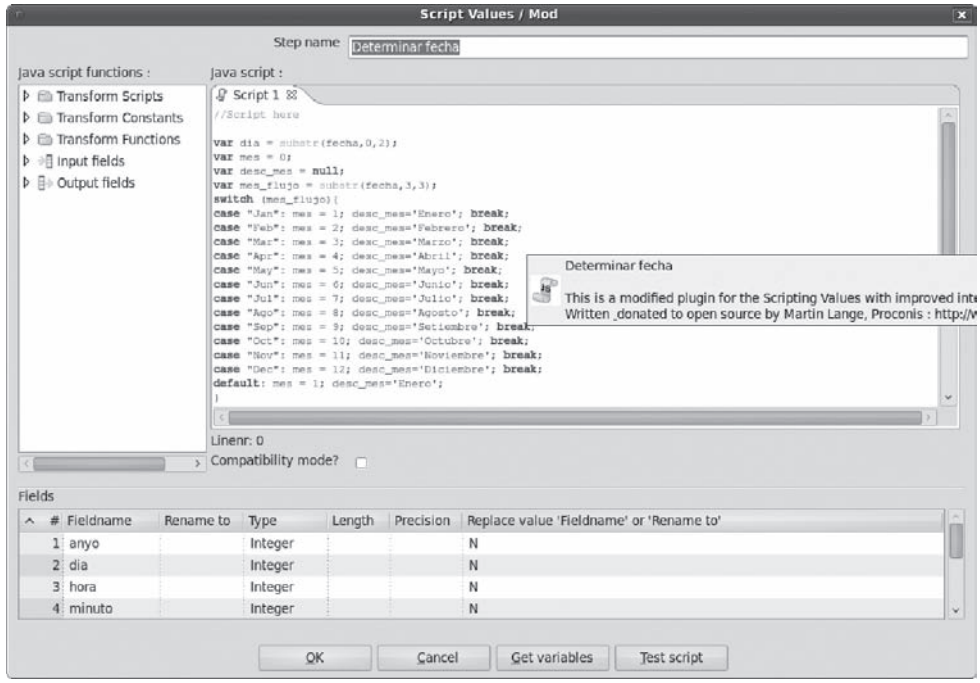
- Se actualiza la dimensión referente a partir de la información entrante y se recupera el id_referente mediante el paso *combination lookup/update*.



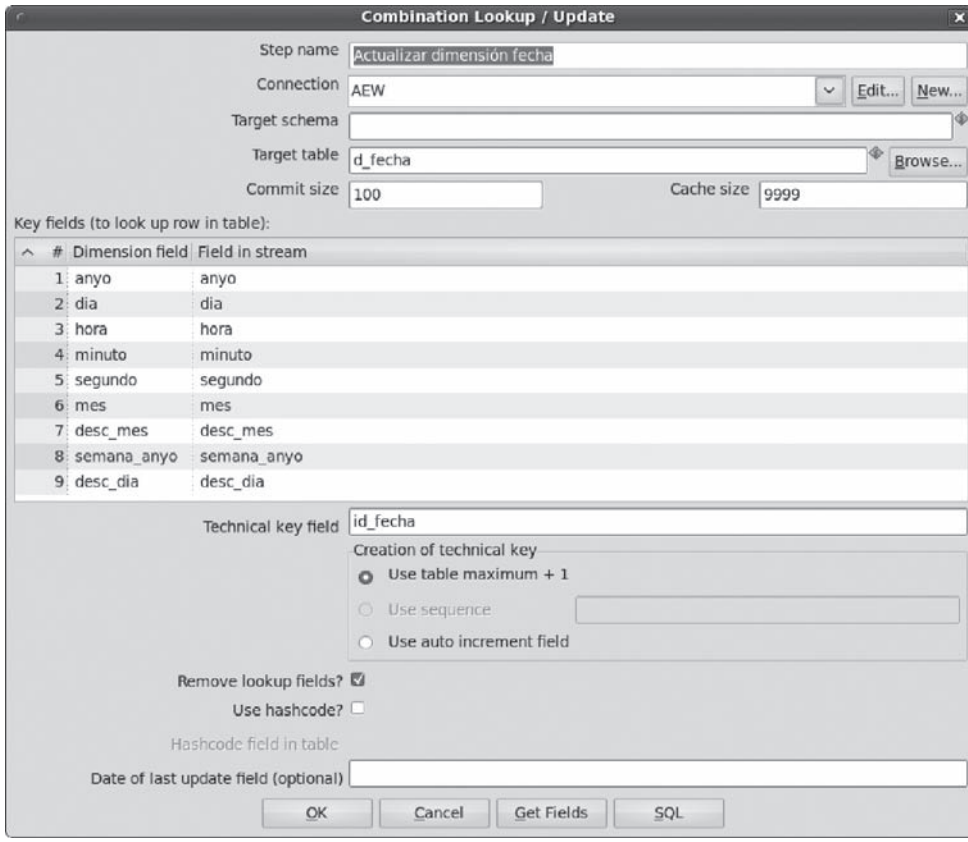
- A partir de la información en el campo agente_usuario, se determina el navegador y el sistema operativo mediante el paso *script value modified javascript*.



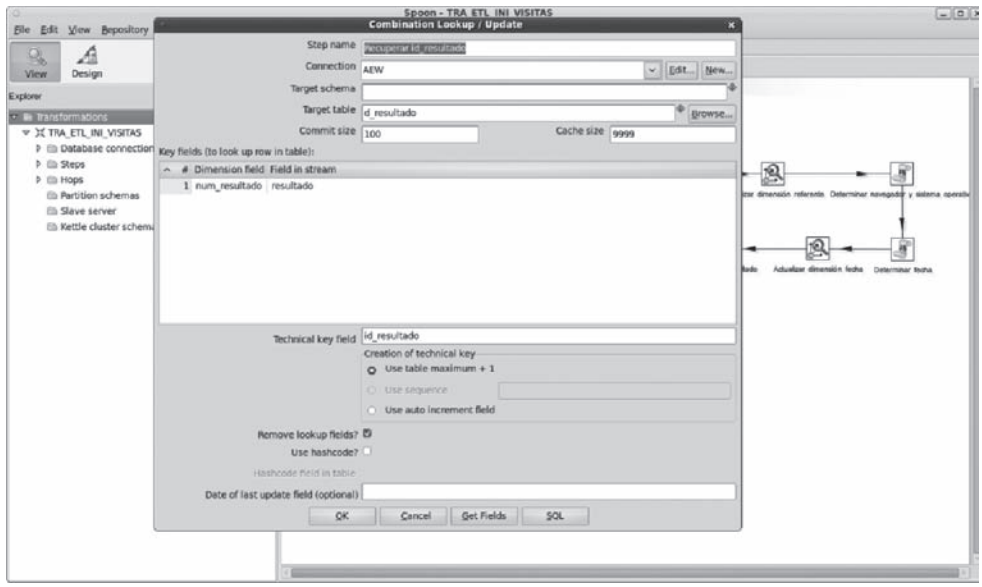
- Se determina la fecha mediante el paso *script value modified javascript*.



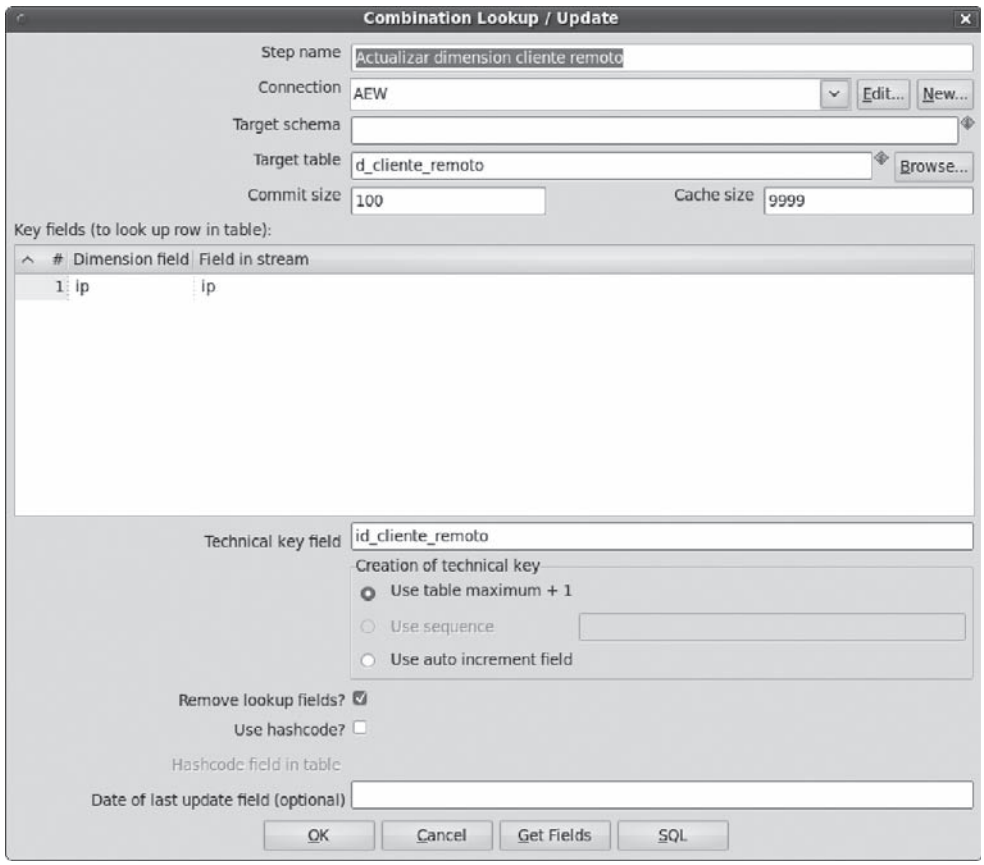
- Se actualiza la fecha y se recupera el id_fecha mediante el paso *combination lookup/update*.



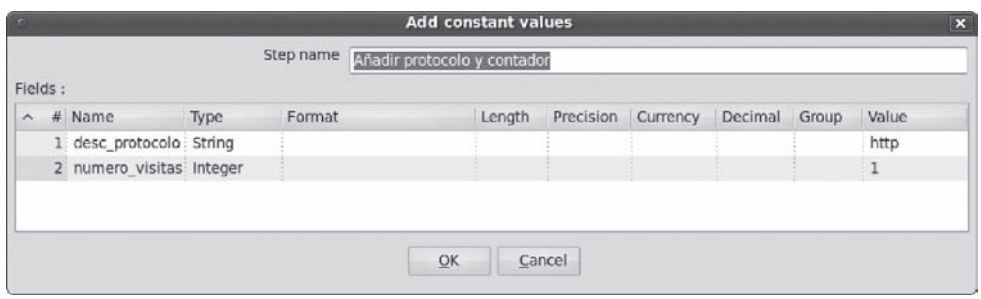
- Se recupera id_resultado de la base de datos a partir de la información en el flujo mediante el paso *combination lookup/update*.



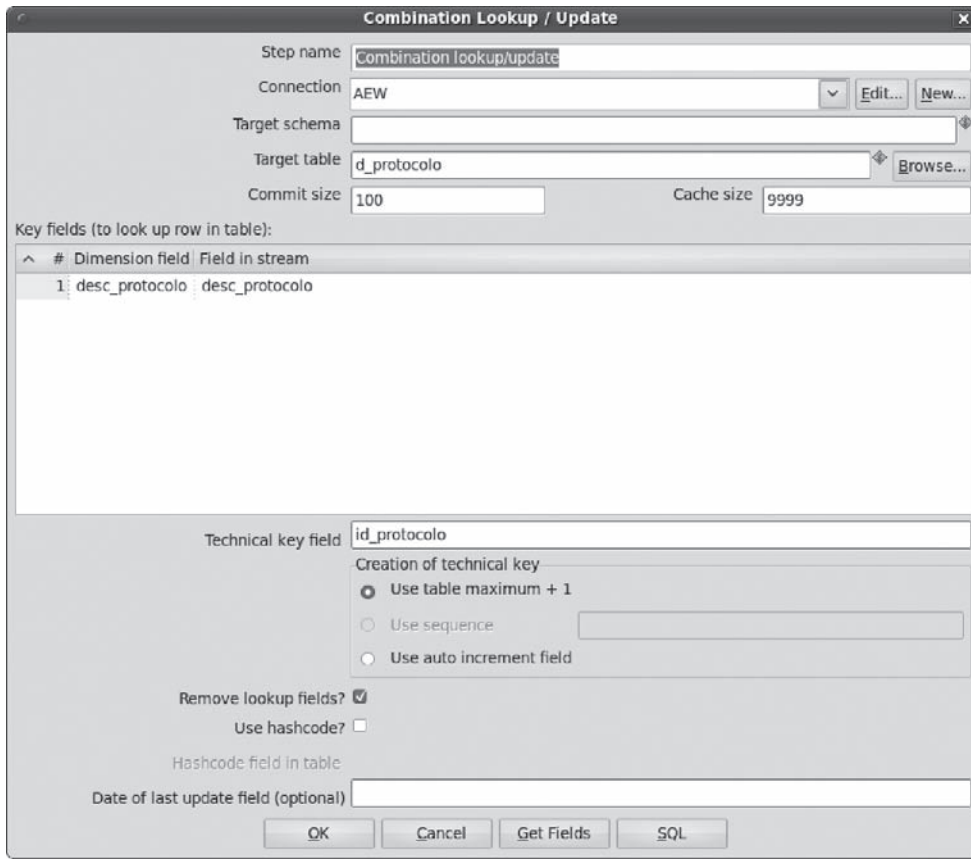
- Se actualiza la dimensión cliente remoto y se recupera el id_cliente_remoto mediante el paso *combination lookup/update*.



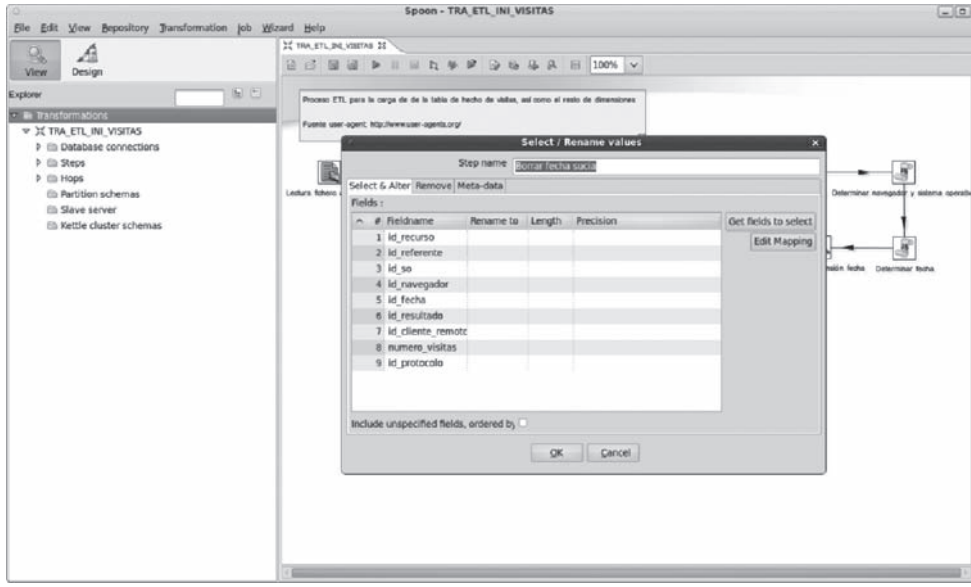
- Se añade el protocolo y el contador mediante el paso *add constant values*.



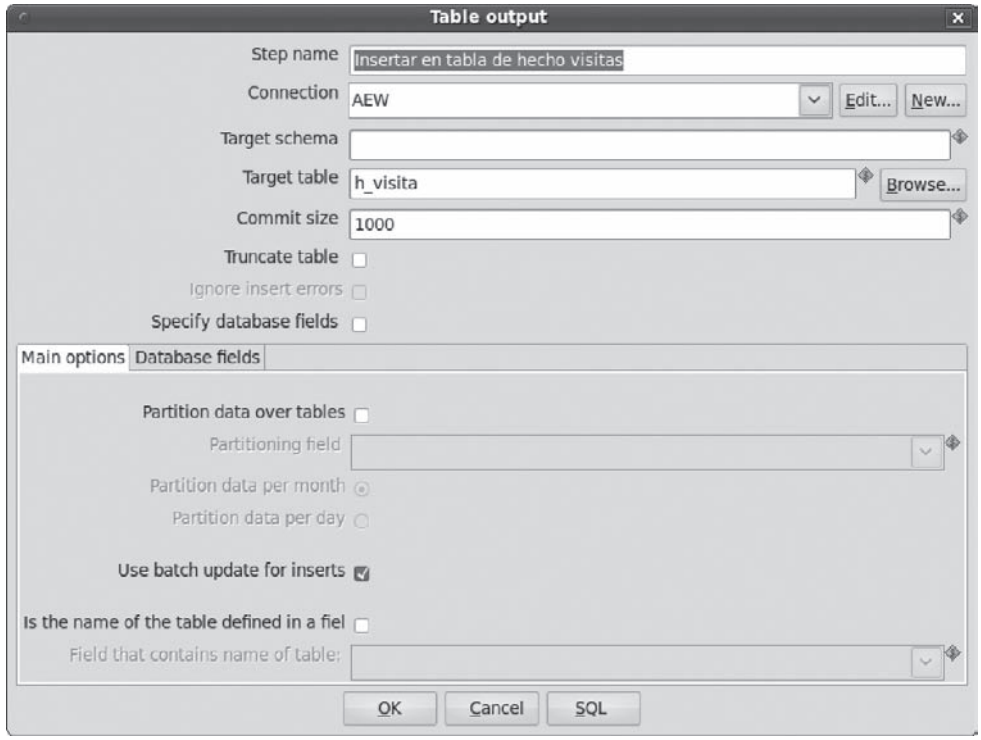
- Se actualiza la dimensión protocolo y se recupera el id_protocolo mediante el paso *combination lookup/update*.



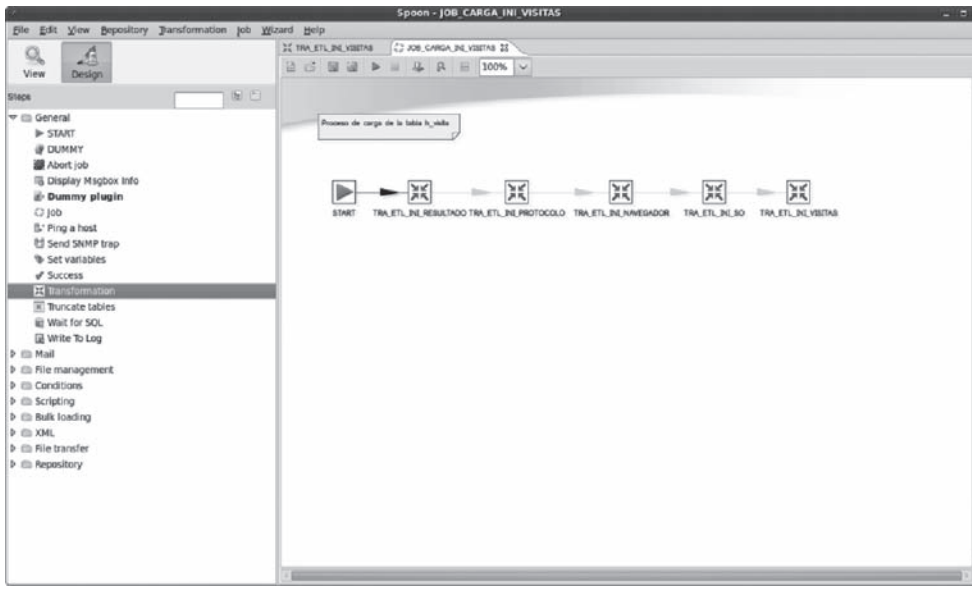
- Se quitan del flujo los campos que no son necesarios mediante el paso *select/rename values*.



- Se inserta la información en la tabla de hecho de visitas mediante el paso *table output*.



Finalmente se diseña un trabajo para lanzar de forma secuencial todas las transformaciones.



4. Anexo 1: 34 subsistemas ETL de Kimball

Existen múltiples sistemas ETL. Según la clasificación de Ralph Kimball, existen 34 clasificados en cuatro grandes grupos:

- Extracción: extrae la información de la fuente de origen.
- Limpieza y conformación: consiste en acciones que permiten validar y aumentar la calidad de la información.
- Entrega: consiste en la preparación de la información para su posterior entrega.
- Gestión: aúna las tareas de administración de procesos ETL.
- Extracción:
  - Data Profiling (subsistema 1): consiste en la exploración de los datos para verificar su calidad y si cumple los estándares conforme los requerimientos.

  - Change Data Capture (subsistema 2): detecta los cambios para refinar los procesos ETL y mejorar su rendimiento.
  - Sistema de extracción (subsistema 3): permite la extracción de datos desde la fuente de origen a la fuente destino.
- Limpieza y conformación:
  - Data Cleansing (subsistema 4): implementa los procesos de calidad de datos que permiten detectar las incoherencias de calidad.
  - Rastreo de eventos de errores (subsistema 5): captura todos los errores que proporcionan información valiosa sobre la calidad de datos y permiten la mejora de los mismos.
  - Creación de dimensiones de auditoría (subsistema 6): permite crear metadatos asociados a cada tabla. Estos metadatos permiten validar la evolución de la calidad de los datos.
  - Deduplicación (subsistema 7): eliminar información redundante de tablas importantes como cliente o producto. Requiere cruzar múltiples tablas en múltiples sistemas de información para detectar el patrón que permite identificar cuando una fila está duplicada.
  - Conformación: permite identificar elementos equivalentes que permiten compartir información entre tablas relacionadas.
- Entrega:
  - Slowly Changing Dimension (SCD) (subsistema 9): implementa la lógica para crear atributos de variabilidad lenta a lo largo del tiempo.
  - Surrogate Key (subsistema 10): permite crear claves subrogadas independientes para cada tabla.
  - Jerarquías (subsistema 11): permite hacer inserciones en estructuras jerárquicas de tablas.
  - Dimensiones especiales (subsistema 12): permite crear dimensiones especiales como junk, mini o de etiquetas.
  - Tablas de hecho (subsistema 13): permite crear tablas de hecho.
  - Pipeline de claves subrogadas (subsistema 14): permite remplazar las claves operacionales por las claves subrogadas.
  - Constructor de tablas multivaluadas (subsistema 15): permite construir tablas puente para soportar las relaciones N:M.
  - Gestión para información tardía (subsistema 16): permite aplicar modificaciones a los procesos en caso de que los datos tarden en llegar.

  - Gerente de dimensión (subsistema 17): autoridad central que permite crear y publicar dimensiones conformadas.
  - Aprovisionador de tablas de hecho (susbsistema 18): permite la gestión de las tablas de hecho.
  - Creador de agregadas (subsistemas 19): permite gestionar agregadas.
  - Creador de cubos OLAP (susbsistema 20): permite alimentar de datos a esquemas OLAP desde esquemas dimensionales relacionales.
  - Propagador de datos (subsistema 21): permite preparar información conformada para ser entregada para cualquier propósito especial.
- Gestión:
  - Programador de trabajos (subsistema 22): permite gestionar ETL de la categoría de trabajos.
  - Sistema de backup (subsistema 23): realiza copias de respaldo de los procesos ETL.
  - Reinicio y recuperación (subsistema 24): permite reiniciar un proceso ETL en el caso de error.
  - Control de versiones (subsistema 25): permite hacer control de versiones de un proyecto ETL y de los metadatos asociados.
  - Migración de versiones (subsistema 26): permite pasar proyectos en fase test a producción mediante versionado.
  - Monitorización de workflow (subsistema 27): dado que un proceso de ETL es un workflow, es necesario monitorizarlos para medir su rendimiento.
  - Ordenación (subsistema 28): permite calibrar los procesos ETL para mejorar su rendimiento.
  - Linealidad y dependencia (subsistema 29): identifica elementos dependientes. Permite identificar las transformaciones en las que participa o ha participado. Permite la trazabilidad del dato.
  - Escalado de problemas (subsistemas 30): suporta la gestión de incidencias.
  - Paralelismo/Clustering (susbsitema 31): permite el uso de procesos en paralelo, grid computing y clustering para mejorar el rendimiento y reducir tiempo del proceso.
  - Seguridad (subsistemas 32): gestiona el acceso a ETL y metadatos.
  - Compliance Manager (subsistema 33): permite soportar la legislación vigente respecto a la custodia y la responsabilidad de datos que debe aplicarse a la organización.

– Repositorio de metadatos (subsistema 34): captura los metadatos de los procesos ETL, de los datos de negocio y de los aspectos técnicos.

5. Glosario

BI	Business Intelligence
CDC	Change Data Capture
CDI	Customer Data Integration
CPM	Corporate Performance Management
CPU	Central Processing Unit
CSV	Comma Separated Value
EAI	Enterprise Application Integration
EDR	Environmental Data Record
EII	Enterprise Information Integration
EIM	Enterprise Information Management
ETL	Extract, Transform and Load
HTML	HyperText Markup Language
JDBC	Java Database Connectivity
JNDI	Java Naming and Directory Interface
ODBC	Open Database Connectivity
ODS	Operational Data Store
OLAP	On-Line Analitycal Processing
PDF	Portable Document Format
PDI	Pentaho Data Integration
PIM	Product Information Management
SCD	Slowly Changing Dimension
SQL	Structured Query Language
XML	eXtensible Markup Language

6. Bibliografía

BOUMAN, R., y VAN DONGEN, J. (2009). *Pentaho® Solutions: Business Intelligence and Data Warehousing with Pentaho® and MySQL.* Indianapolis: Wiley Publishing.

BOUMAN, R. CASTER, MATT y VAN DONGEN, J. (2010). *Pentaho® Kettle Solutions*. Indianapolis: Wiley Publishing.

INMON, W. H. (2005). *Building the Data Warehouse,* 4th Edition. Hoboken: John Wiley & Sons.

INMON, W. H., STRAUSS, D., y NEUSHLOSS, G. (2008). *DW 2.0: The Architecture for the Next Generation of Data Warehousing.* Burlington: Morgan Kaufman Series.

KIMBALL, R. (2009). *Data Warehouse Toolkit Classics: The Data Warehouse Toolkit,* 2nd Edition; *The Data Warehouse Lifecycle Toolkit,* 2nd Edition; *The Data Warehouse ETL Toolkit.* Hoboken: John Wiley & Sons.

KNIGHT, B. (2009). *Professional Microsoft SQL Server 2008 Integration Services*. Indianapolis: Wrox.

PULVIRENTI, ADRIÁN SERGIO y ROLDAN, MARIA CARINA (2011). *Pentaho Data Integration 4 Cookbook*. Birmingham: Packt Publishing.

ROLDAN, MARIA CARINA (2010). *Pentaho 3.2 Data Integration - Beginner's Guide*. Birmingham: Packt Publishing.

Capítulo IV

Diseño de análisis OLAP

Es bien sabido que el concepto de Business Intelligence engloba múltiples conceptos. Uno de los más importantes es el concepto OLAP (On-Line Analytical Processing), acuñado por Edgar F. Codd.

Una manera sencilla de explicar este concepto es decir que es una tecnología que permite un análisis multidimensional[1] a través de tablas matriciales o pivotantes.

Si bien el término OLAP se introduce por primera vez en 1993, los conceptos base del mismo, como por ejemplo el análisis multidimensional, son mucho más antiguos.

A pesar de ser una tecnología que ya tiene más de cuatro décadas, sus características y su evolución han provocado que la gran mayoría de soluciones del mercado incluya un motor OLAP.

Es necesario comentar:

- Las herramientas OLAP de los diferentes fabricantes, si bien son similares, no son completamente iguales dado que presentan diferentes especificaciones del modelo teórico.
- La última tendencia en OLAP es la tecnología in-memory.
- Las soluciones open source OLAP han sido las últimas a añadirse a la lista y, por ahora, no tienen tanta variedad como su contrapartida propietaria.
- En el mercado open source OLAP sólo hay dos soluciones actualmente: el motor ROLAP Mondrian y el motor MOLAP PALO.

Este capítulo se centrará en presentar el concepto OLAP y sus diferentes opciones.

1. En 1962, se introduce el análisis multidimensional en el libro de Ken Iverson A Programming Language.

1. OLAP como herramienta de análisis

OLAP forma parte de lo que se conoce como sistemas analíticos, que permiten responder preguntas como: ¿por qué paso? Estos sistemas pueden encontrarse tanto integrados en suites de Business Intelligence o ser simplemente una aplicación independiente.

Es necesario, antes de continuar, introducir una definición formal de OLAP:

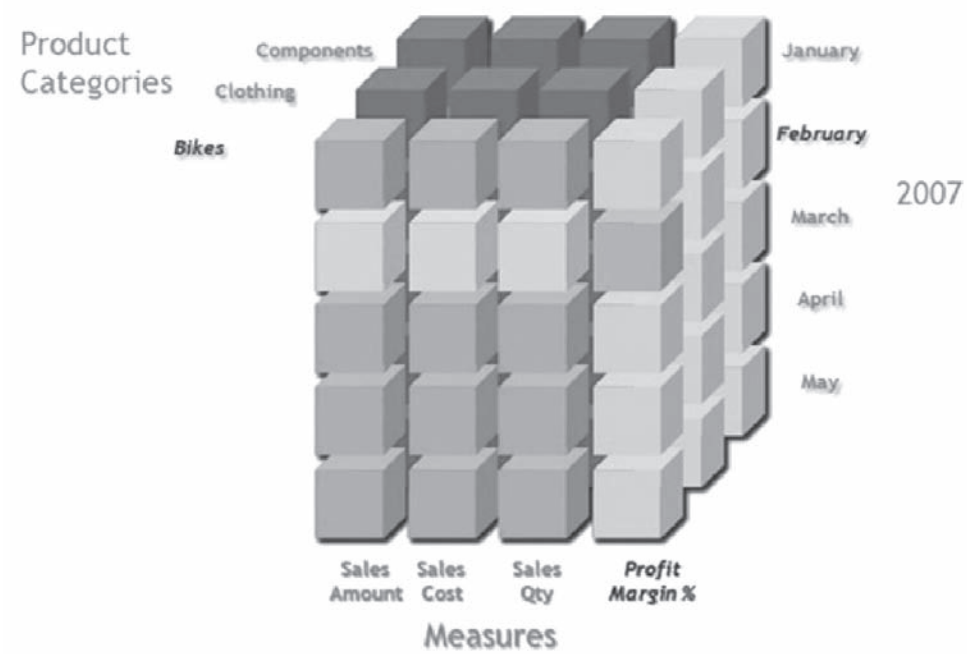
> Se entiende por OLAP, o proceso analítico en línea, al método ágil y flexible para organizar datos, especialmente metadatos, sobre un objeto o jerarquía de objetos como en un sistema u organización multidimensional, y cuyo objetivo es recuperar y manipular datos y combinaciones de los mismos a través de consultas o incluso informes.

Una herramienta OLAP está formada por un motor y un visor. El motor es, en realidad, justo el concepto que acabamos de definir. El visor OLAP es una interfaz que permite consultar, manipular, reordenar y filtrar datos existentes en una estructura OLAP mediante una interfaz gráfica de usuario que dispone funciones de consulta MDX[2] y otras.

Las estructuras OLAP permiten realizar preguntas que serían sumamente complejas mediante SQL.

Consideremos un ejemplo gráfico que nos permitirá entender la potencia de este tipo de herramientas.

2. MDX es Multidimensional Expressions. Se considera el lenguaje de consulta OLAP estándar de facto en el mercado.



Imaginemos que queremos responder a la siguiente pregunta: ¿cuál es el margen de beneficios de la venta de bicicletas para febrero de 2007?

Si tenemos un cubo, como el de ejemplo, formado por el tiempo, los productos y las medidas, la respuesta es la intersección entre los diferentes elementos.

Cabe observar que una estructura de esta forma permite consultas mucho más completas, como por ejemplo comparar el margen de beneficios de febrero y mayo, entre diferentes productos, etc.

Además, el visor OLAP proporciona libertad a los usuarios finales para realizar dichas consultas de forma independiente al departamento de IT.

1.1. Tipos de OLAP

Existen diferentes tipos de OLAP, que principalmente difieren en cómo se guardan los datos:

- **MOLAP (Multidimensional OLAP):** es la forma clásica de OLAP y frecuentemente es referida con dicho acrónimo. MOLAP utiliza estructuras de bases de datos generalmente optimizadas para la recuperación de los mismos. Es lo que se conoce como bases de datos multidimensionales (o, más coloquialmente, cubos). En definitiva, se crea un fichero que contiene todas las posibles consultas precalculadas. A diferencia de las bases de datos relacionales, estas formas de almacenaje están optimizadas para la velocidad de cálculo. También se optimizan a menudo para la recuperación a lo largo de patrones jerárquicos de acceso. Las dimensiones de cada cubo son típicamente atributos tales como periodo, localización, producto o código de la cuenta. La forma en la que cada dimensión será agregada se define por adelantado.
- **ROLAP (Relational OLAP):** trabaja directamente con las bases de datos relacionales, que almacenan los datos base y las tablas dimensionales como tablas relacionales mientras se crean nuevas tablas para guardar la información agregada.
- **HOLAP (Hybrid OLAP):** no hay acuerdo claro en la industria en cuanto a qué constituye el OLAP híbrido, exceptuando el hecho de que es una base de datos en la que los datos se dividen en almacenaje relacional y multidimensional. Por ejemplo, para algunos vendedores, HOLAP consiste en utilizar las tablas relacionales para guardar las cantidades más grandes de datos detallados, y utiliza el almacenaje multidimensional para algunos aspectos de cantidades más pequeñas de datos menos detallados o agregados.
- **DOLAP (Desktop OLAP):** es un caso particular de OLAP ya que está orientado a equipos de escritorio. Consiste en obtener la información necesaria desde la base de datos relacional y guardarla en el escritorio. Las consultas y los análisis son realizados contra los datos guardados en el escritorio.
- **In-memory OLAP:** es un enfoque por el que muchos nuevos fabricantes están optando. Consiste en que la estructura dimensional se genera sólo a nivel de memoria y se guarda el dato original en algún formato que potencia su despliegue de esta forma (por ejemplo, comprimido o mediante una base de datos de lógica asociativa). En este último punto es donde cada fabricante pone su énfasis.

Cada tipo tiene ciertas ventajas, aunque hay desacuerdo sobre las ventajas específicas de los diferentes proveedores.

- MOLAP es mejor en sistemas más pequeños de datos, es más rápido para calcular agregaciones y retornar respuestas y necesita menos espacio de almacenaje. Últimamente, in-memory OLAP está apuntalándose como una opción muy válida al MOLAP.
- ROLAP se considera más escalable. Sin embargo, el preproceso de grandes volúmenes es difícil de implementar eficientemente, así que se desecha con frecuencia. De otro modo, el funcionamiento de las consultas puede ser no óptimo.
- HOLAP está entre los dos en todas las áreas, pero puede preprocesar rápidamente y escalar bien.

Todos los tipos son, sin embargo, propensos a la explosión de la base de datos. Éste es un fenómeno que causa la cantidad extensa de espacio de almacenaje que es utilizado por las bases de datos OLAP cuando se resuelven ciertas, pero frecuentes, condiciones: alto número de dimensiones, de resultados calculados de antemano y de datos multidimensionales escasos.

La dificultad en la implementación OLAP deviene en la formación de las consultas, elegir los datos base y desarrollar el esquema. Como resultado, la mayoría de los productos modernos vienen con bibliotecas enormes de consultas preconfiguradas. Otro problema está en la baja calidad de los datos, que deben ser completos y constantes.

1.2. Elementos OLAP

OLAP permite el análisis multidimensional. Ello significa que la información está estructurada en ejes (puntos de vista de análisis) y celdas (valores que se están analizando).

En el contexto OLAP existen diferentes elementos comunes a las diferentes tipologías OLAP (que en definitiva se diferencian a nivel práctico en que en MOLAP se precalculan los datos, en ROLAP no, y en in-memory se generan al iniciar el sistema):

- Esquema: un esquema es una colección de cubos, dimensiones, tablas de hecho y roles.
- Cubo: es una colección de dimensiones asociadas a una tabla de hecho. Un cubo virtual permite cruzar la información entre tablas de hecho a partir de sus dimensiones comunes.

- Tabla de hecho, dimensión y métrica.
- Jerarquía: es un conjunto de miembros organizados en niveles. En cuanto a bases de datos, se puede entender como una ordenación de los atributos de una dimensión.
- Nivel: es un grupo de miembros en una jerarquía que tienen los mismos atributos y nivel de profundidad en la jerarquía.
- Miembro: es un punto en la dimensión de un cubo que pertenece a un determinado nivel de una jerarquía. Las métricas (medidas) en OLAP se consideran un tipo especial de miembro que pertenece a su propio tipo de dimensión. Un miembro puede tener propiedades asociadas.
- Roles: permisos asociados a un grupo de usuarios.
- MDX: es un acrónimo de Multidimensional eXpressions (aunque también es conocido como Multidimensional Query eXpression). Es el lenguaje de consulta de estructuras OLAP, fue creado en 1997 por Microsoft y, si bien no es un lenguaje estándar, la gran mayoría de fabricantes de herramientas OLAP lo han adoptado como estándar de hecho.

1.3. 12 reglas OLAP de E. F. Codd

La definición de OLAP presentada anteriormente se basa en las 12 leyes que acuñó Edgar F. Codd en 1993. Estas reglas son las que, en mayor o menor medida, intentan cumplir todos los fabricantes de software:

- Vista conceptual multidimensional: se trabaja a partir de métricas de negocio y sus dimensiones.
- Transparencia: el sistema OLAP debe formar parte de un sistema abierto que soporta fuentes de datos heterogéneas (lo que llamamos actualmente arquitectura orientada a servicios).
- Accesibilidad: se debe presentar el servicio OLAP al usuario con un único esquema lógico de datos (lo que, en definitiva, nos indica que debe presentarse respecto una capa de abstracción directa con el modelo de negocio).
- Rendimiento de informes consistente: el rendimiento de los informes no debería degradarse cuando el número de dimensiones del modelo se incrementa.
- Arquitectura cliente/servidor: basado en sistemas modulares y abiertos que permitan la interacción y la colaboración.

- Dimensionalidad genérica: capacidad de crear todo tipo de dimensiones y con funcionalidades aplicables de una dimensión a otra.
- Dynamic sparse-matrix handling: la manipulación de datos en los sistemas OLAP debe poder diferenciar valores vacíos de valores nulos y además poder ignorar las celdas sin datos.
- Operaciones cruzadas entre dimensiones sin restricciones: todas las dimensiones son creadas igual y las operaciones entre dimensiones no deben restringir las relaciones entre celdas.
- Manipulación de datos intuitiva: dado que los usuarios a los que se destinan este tipo de sistemas son frecuentemente analistas y altos ejecutivos, la interacción debe considerarse desde el prisma de la máxima usabilidad de los usuarios.
- Reporting flexible: los usuarios deben ser capaces de manipular los resultados que se ajusten a sus necesidades conformando informes. Además, los cambios en el modelo de datos deben reflejarse automáticamente en esos informes.
- Niveles de dimensiones y de agregación ilimitados: no deben existir restricciones para construir cubos OLAP con dimensiones y con niveles de agregación ilimitados.

2. OLAP en el contexto de Pentaho

Mondrian es el motor OLAP integrado en Pentaho y ha sido renombrado como Pentaho Analysis Services. Este motor se combina con un visor OLAP –que es diferente si consideramos la versión Community o Enterprise– y con dos herramientas de desarrollo.

Resumiendo:

	Community	Enterprise
Motor OLAP	Mondrian	
Visor OLAP	Jpivot, PAT	Pentaho Analyser (anteriormente Clearview)
Herramientas de desarrollo	Pentaho Schema Workbench, Pentaho Agreggation Designer	

Trataremos cada uno de esos puntos en este apartado.

2.1. Mondrian

Mondrian es un servidor/motor OLAP escrito en java que está licenciado bajo la Eclipse Public License (EPL). Existe como proyecto desde 2003 y fue adquirido por Pentaho en 2005.

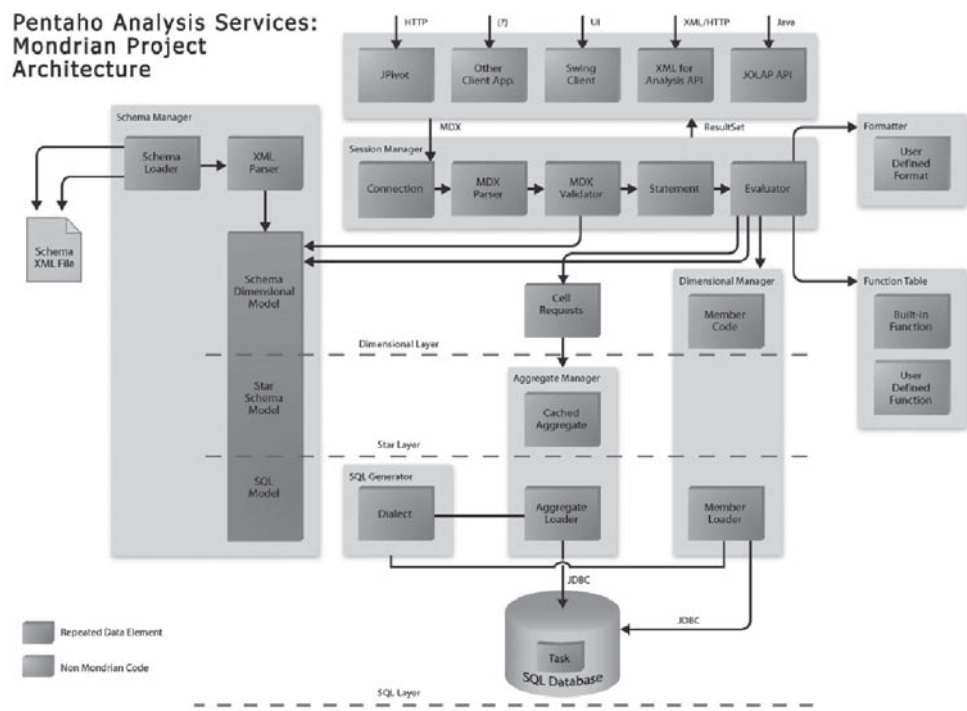
Mondrian se caracteriza por ser un motor ROLAP con caché, lo cual lo sitúa cerca del concepto de HOLAP. ROLAP significa que en Mondrian no residen datos (salvo en la caché) sino que éstos están en una base de datos en la que existen las tablas que conforman la información multidimensional con la que el motor trabaja. El lenguaje de consulta es MDX.

Mondrian se encarga de recibir consultas dimensionales a un cubo mediante MDX y de devolver los datos. El cubo es, en este caso, un conjunto de metadatos que definen cómo se ha de mapear la consulta por sentencias SQL al repositorio que contiene realmente los datos.

Esta forma de trabajar tiene ciertas ventajas:

- No se generan cubos/estructuras OLAP estáticas y por lo tanto se ahorra en tiempo de generación y en espacio.
- Se trabaja con datos actualizados siempre al utilizar la información residente en la base de datos.
- Mediante el uso del caché y de tablas agregadas, se pretende simular el mejor rendimiento de los sistemas MOLAP.

El siguiente diagrama presenta la arquitectura de Mondrian:



Mondrian funciona sobre las bases de datos estándar del mercado: Oracle, DB2, SQL Server, MySQL, PostgreSQL, LucidDB, Teradata..., lo que habilita y facilita el desarrollo de negocio.

Los últimos desarrollos de Mondrian se caracterizan por incluir olap4j.[3] Es una iniciativa del mismo desarrollador de Mondrian: Julian Hyde.

Mondrian es un caso atípico en el contexto OSBI. Para la gran mayoría de herramientas de inteligencia de negocio existen una o varias opciones. En el caso de soluciones ROLAP, Mondrian es el único producto.

Varios fabricantes han incluido Mondrian en sus soluciones (JasperSoft, Openi, OpenReports, SpagoBI, SQLPower e incluso la defenestada Lucidera). El hecho de existir una única solución y de existir toda una comunidad de fabricantes y usuarios a su alrededor, hace que el equipo de desarrollo de Mondrian (dirigido por Julian Hyde) tenga ciclos de desarrollo mucho menores que otras soluciones.

3. Olap4j es una API java cuyo objetivo es permitir la creación de aplicaciones OLAP intercambiables entre los diferentes motores OLAP del mercado.

2.2. Visores OLAP

Como ya se ha comentado anteriormente, en el contexto de Pentaho, existen tres visores OLAP: JPivot, PAT y Pentaho Analyser.

JPivot

JPivot es un cliente OLAP basado en JSP que empezó en 2003. Puede considerarse un proyecto hermano de Mondrian dado que combinado con él permite realizar consultas tanto MDX como a partir de elementos gráficos que se renderizan en un navegador web. Durante largo tiempo ha sido el único visor existente para Mondrian. Pentaho ha adaptado su estilo para diferenciarlo de la interfaz original.

Las características principales de este visor analítico son:

- Capacidades de análisis interactivo a través de un acceso web basado en Excel, lo que le proporciona una alta funcionalidad.
- Está basado en estándares de la industria (JBDC, JNDI, SQL, XML/A y MDX).
- Posibilidad de extenderse mediante desarrollo.

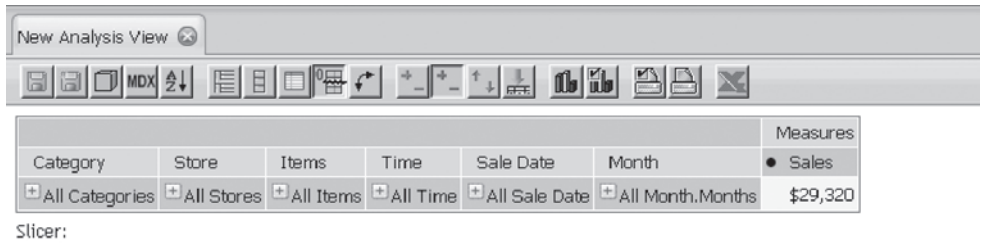
						Measures		
Region	My Image	P	P0	P1	P2	Measures[0]	Measures[1]	Measures[2]
-All Region[0]		0	V00			856.44	820.95	983.55
+Region[0]		0		V10		1,095.05	1,087.79	958.23
-Region[1]		1		V11		1,052.05	1,107.59	980.32
+City[0]	✓	0			V20	1,088.31	884.90	1,085.73
+City[1]	✓	1			V21	1,336.43	943.27	1,117.62
+City[2]	✓	2			V22	969.34	1,086.70	1,094.71
+City[3]	✓	3			V23	964.30	900.58	953.65
+City[4]	✓	4			V24	915.59	1,032.98	876.19
+City[5]	✓	5			V25	1,023.12	1,242.39	988.85
+City[6]	✓	6			V26	949.16	941.62	1,077.48
+City[7]	✓	7			V27	1,055.92	1,095.56	1,078.26
+Region[2]		2		V12		1,027.36	1,084.88	1,009.71
+Region[3]		3		V13		972.10	932.27	836.34
+Region[4]		4		V14		919.44	926.47	873.20

El menú de JPivot ofrece diversas opciones al usuario final:

- Navegador OLAP: permite determinar las dimensiones que aparecen en las filas y/o columnas así como los filtros aplicados y las métricas por aparecer.
- Consulta MDX: permite visualizar y editar la consulta MDX que genera el informe OLAP.
- Configurar tabla OLAP: permite configurar aspectos por defecto de la tabla OLAP como el orden ascendente o descendente de los elementos.
- Mostrar miembros padre: permite mostrar u ocultar el padre del miembro de una jerarquía.
- Ocultar cabeceras: muestra u oculta las cabeceras repetidas para facilitar la compresión del contenido.
- Mostrar propiedades: muestra o oculta las propiedades de los miembros de una jerarquía.
- Borrar filas o columnas vacías: muestra u oculta los valores sin contenido (conveniente para ciertos informes).
- Intercambiar ejes: intercambia filas por columnas.
- Drill buttons: member, position y replace:
  - Member: permite expandir todas las instancias de un miembro.
  - Position: permite expandir la instancia seleccionada de un miembro.
  - Replace: permite sustituir un miembro por sus hijos.

- Drill through: permite profundizar en el detalle de información a partir de un nivel de información agregado superior.
- Mostrar gráfico: muestra un gráfico asociado a los datos. No todos los datos son susceptibles de generar gráficos consistentes.
- Configuración del gráfico: permite configurar las propiedades del gráfico. Desde el tipo del mismo hasta propiedades de estilo como tipo de letra, tamaño o color.
- Configuración de las propiedades de impresión/exportación: permite configurar las propiedades de la impresión como el título, disposición del papel, tamaño...
- Exportar a PDF: permite generar un PDF con el contenido del informe.
- Exportar a Excel: permite generar un Excel con el contenido del informe.

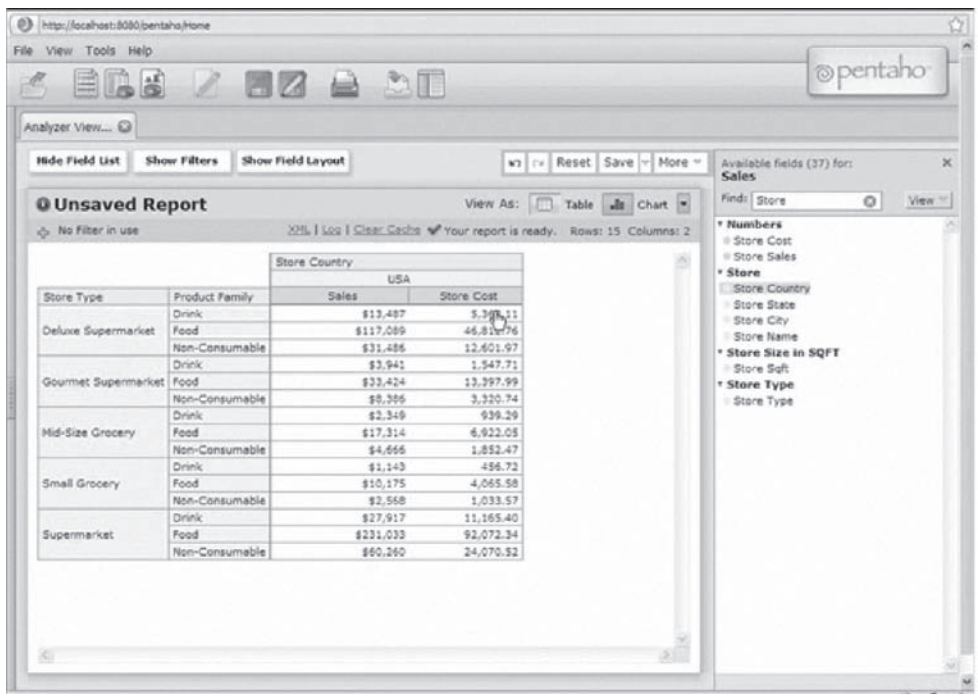
En el caso de la integración con Pentaho se añade la capacidad de crear y guardar nuevas vistas analíticas. Cabe comentar que la creación no ofrece muchas capacidades de configuración.



Cabe comentar que JPivot no es una herramienta muy orientada al usuario. No dispone de funcionalidades drag & drop ni tampoco otras funcionalidades como la creación de nuevas métricas o jerarquías.

Pentaho Analyser

Con el objetivo de ofrecer un servicio de alto valor añadido en la versión Enterprise, Pentaho ha adquirido el producto Clearview de Lucidera y lo ha renombrado como Pentaho Analyser sustituyendo JPivot.



Es una solución que ofrece:

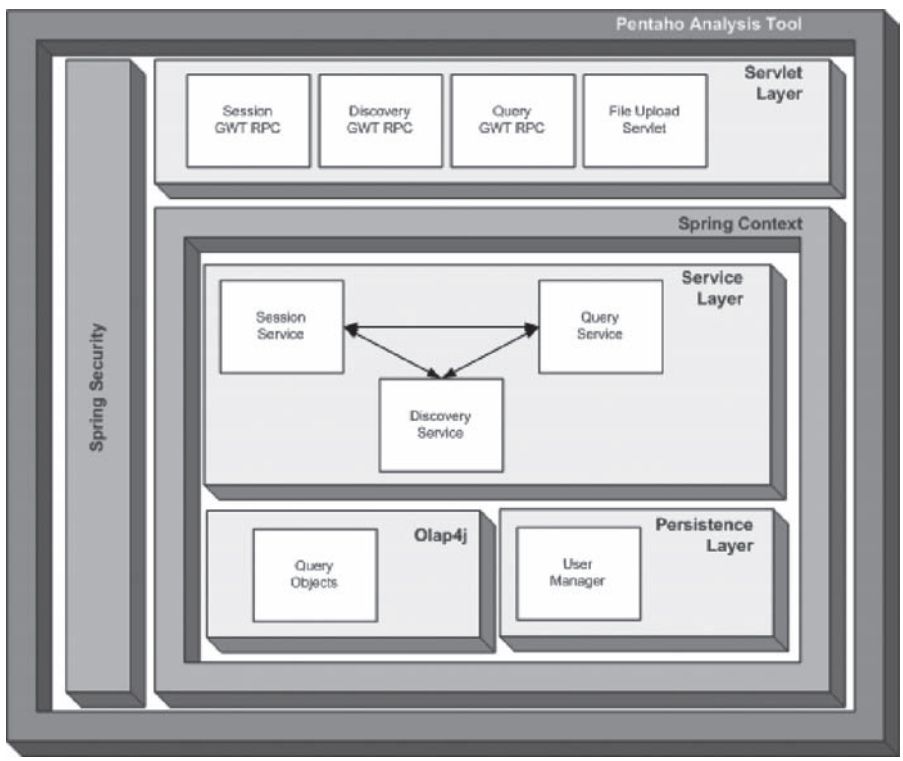
- Capacidades de drag & drop.
- Creación de nuevas medidas calculadas.
- Soporte para soluciones BI SaaS.
- Buscador de objetos.
- Funcionalidades encapsuladas (como, por ejemplo, consultas por jerarquía temporal).

Esta solución está completamente orientada al usuario facilitando su trabajo con la misma.

PAT

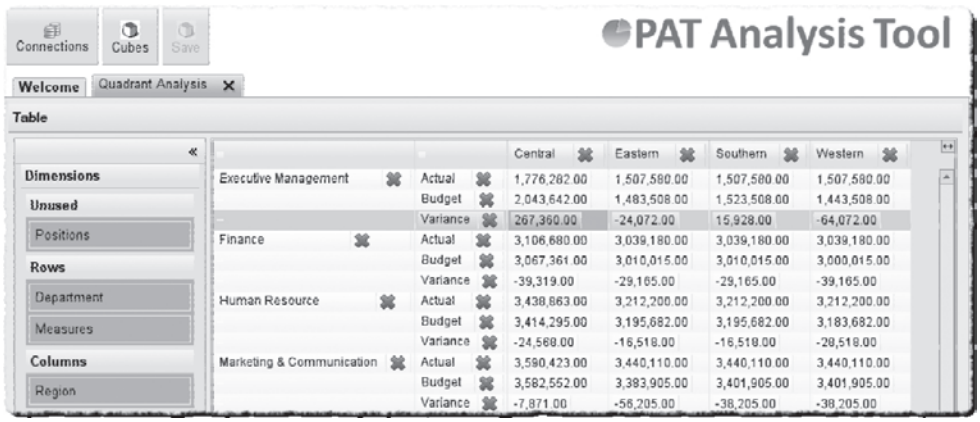
PAT es el acrónimo de Pentaho Analysis Tool. Es una herramienta desarrollada con GWT por parte de la comunidad de Pentaho para sustituir JPivot. La versión actual aún no está disponible para su uso en producción, sin embargo se espera que para el año 2010 salga la primera versión.

El siguiente diagrama muestra la arquitectura de PAT:



Las características que se contemplan en el roadmap para la versión 1.0 son:

- Capacidades de drag & drop.
- Uso de temas.
- Uso de olap4j para ser usado como visor OLAP de múltiples motores (Mondrian, Microsoft Analysis Services u otros).
- Duplicación de todas las funcionalidades de JPivot.
- Extensión de dichas capacidades, como por ejemplo soporte de gráficos en formato flash.



2.3. Herramientas de desarrollo

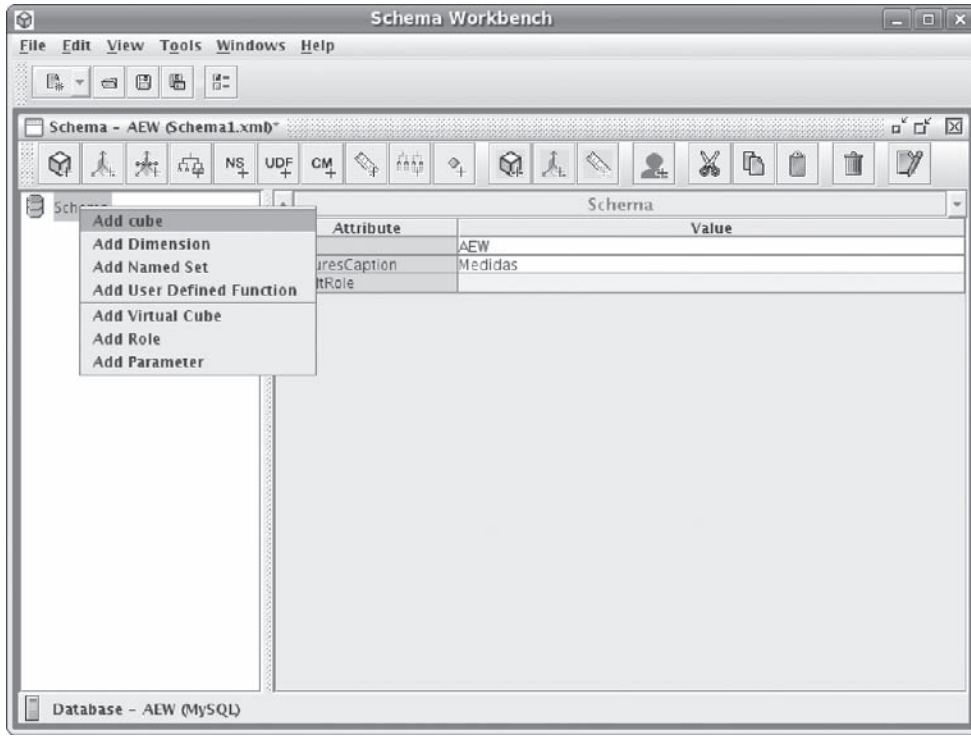
Pentaho Schema Workbench

Pentaho Schema Workbench (PSW) es una herramienta de desarrollo que permite crear, modificar y publicar un esquema de Mondrian. Es un programa java multiplataforma.

Es una herramienta muy orientada al desarrollador conocedor de la estructura de un esquema de Mondrian. Permite crear todos los objetos que soporta Mondrian: esquema, cubo, dimensiones, métricas...

Tiene dos áreas: la zona en la que se muestra la estructura jerárquica del esquema OLAP y la zona de edición de las propiedades de cada elemento.

Presenta un menú superior para crear cubos, dimensiones, dimensiones conformadas, métricas, miembros calculados, subconjuntos (named set) y roles, así como operaciones estándar como cortar, copiar y pegar.



Además, entre sus características incluye:

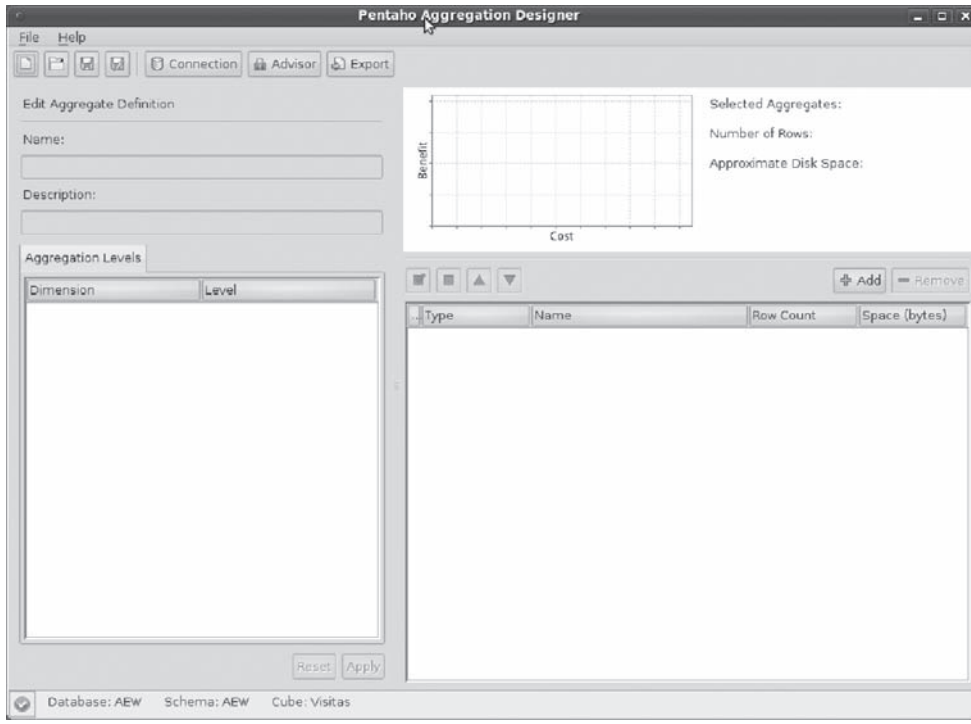
- Realizar consultas MDX contra el esquema creado (requiere conocer la sintaxis del lenguaje).
- Consultar la base de datos que sirve de origen para el esquema de Mondrian.
- Publicar directamente el esquema en el servidor de Pentaho.

Pentaho Aggregation Designer

Un método para optimizar Mondrian, aparte de configurar la caché del mismo adecuadamente, es la creación de tablas agregadas.

Desde hace pocos meses, Pentaho ofrece una herramienta de diseño orientada a dicha función: Pentaho Aggregation Designer.

Esta herramienta java permite analizar la estructura del esquema de Mondrian contra la cantidad de datos que recuperar y, a partir de dicho análisis, recomendar la creación de tablas agregadas.



En nuestro caso particular no haremos uso de esta herramienta. Aunque se incluye para que pueda ser conocida.

3. Caso práctico

3.1. Diseño de OLAP con Schema Workbench

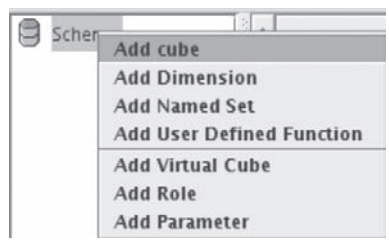
El diseño de estructuras OLAP es común en Pentaho y otras soluciones open source del mercado dado que las herramientas que proporcionan sólo difieren en pequeños puntos de rediseño de la interfaz GUI. El punto realmente diferente es cómo se publican en una u otra plataforma.

En el proceso de creación de una estructura OLAP debemos tener presente que lo que haremos es mapear el diseño de la base de datos (tablas de hecho y dimensiones) con nuestro diseño, de forma que:

- Es posible crear un esquema con menos elementos que los existentes en la base de datos (no interesa contemplar todos los puntos de vista de análisis, por ejemplo).
- Es posible crear un esquema con la misma cantidad de elementos. Se consideran todas las tablas de hecho y las dimensiones.
- Es posible crear un esquema con más elementos que los existentes en la base de datos. Por ejemplo, es posible crear dimensiones u otros objetos que sólo existen en el esquema OLAP y que se generan en memoria.

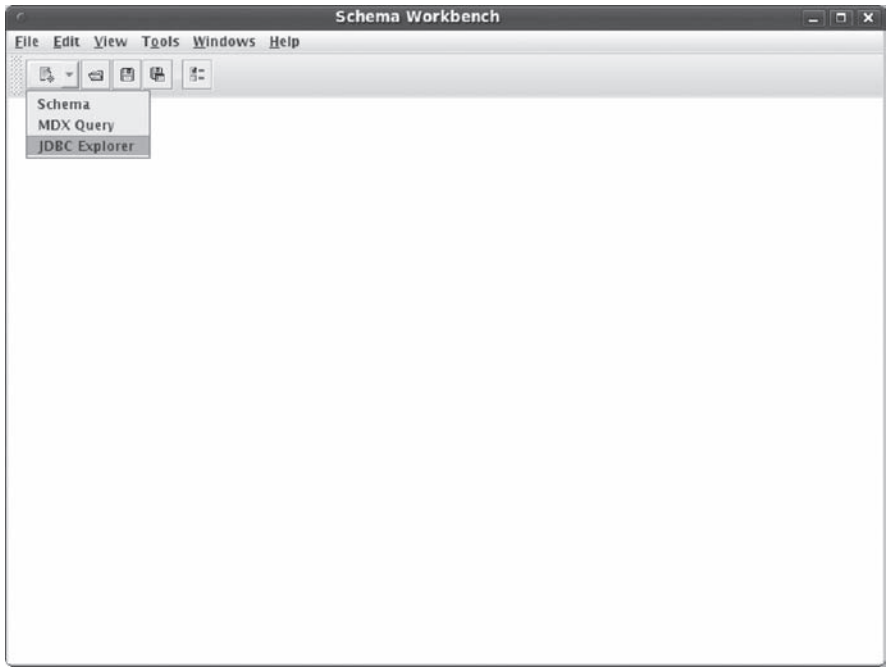
Para este primer ejemplo, vamos a considerar un mapeo uno a uno (todos los elementos del data warehouse tendrán su correspondencia en la estructura OLAP).

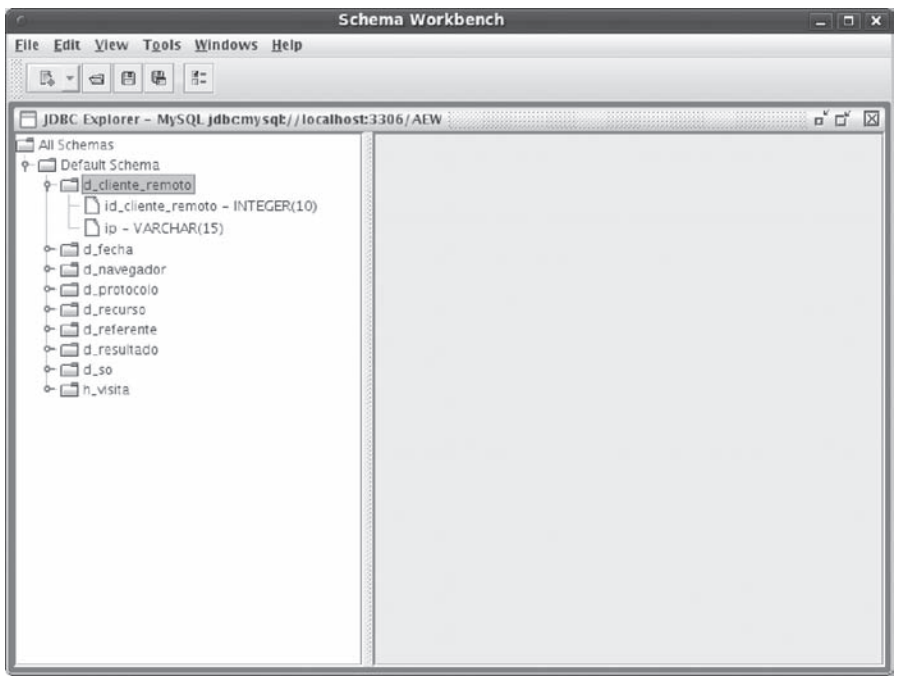
Esta herramienta permite crear elementos o bien a través del despliegue de los elementos disponibles en cada elemento de la arquitectura



o bien a través del menú superior que incluye la creación de cubos, dimensiones, jerarquías, niveles, medidas, medidas calculadas, elementos virtuales (cubos, dimensiones y métricas), roles y operaciones estándar como copiar, cortar, pegar, e incluso la edición del XML de forma directa.

Por otra parte, esta herramienta incluye un explorador de la base de datos que, una vez creada la conexión a la base de datos, nos permite explorar la estructura de las tablas para recordar cuál es nombre de los campos y atributos a usar.

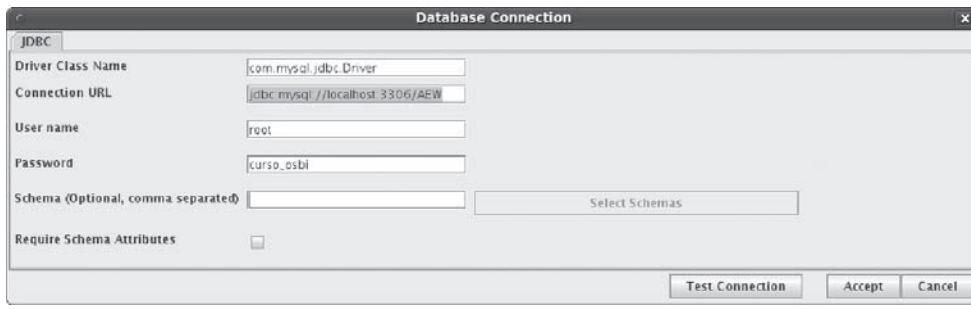

Schema Workbench
File Edit View Tools Windows Help
Schema
MDX Query
JDBC Explorer


Schema Workbench
File Edit View Tools Windows Help
JDBC Explorer – MySQL jdbc:mysql://localhost:3306/AEW
All Schemas
Default Schema
d_cliente_remoto
id_cliente_remoto – INTEGER(10)
ip – VARCHAR(15)
d_fecha
d_navegador
d_protocolo
d_recurso
d_referente
d_resultado
d_so
h_visita

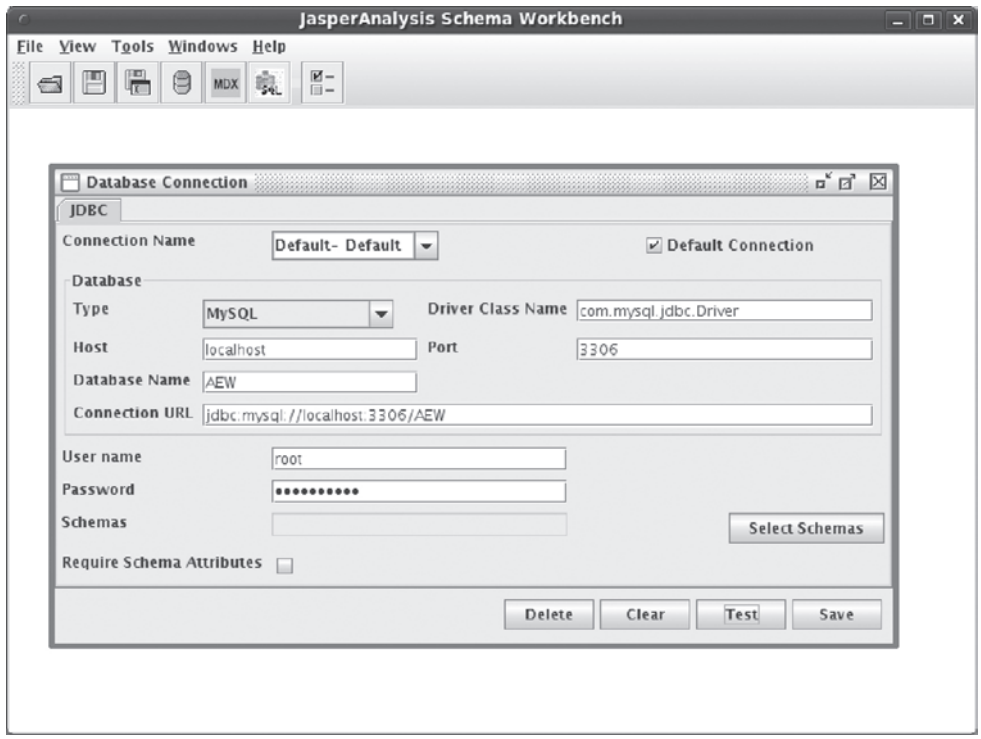
Los pasos en el proceso de creación son los siguientes:

- Creación de una conexión al data warehouse. La herramienta de diseño necesita conocer cuál es la fuente de origen de tablas y datos. Por ello, antes de empezar cualquier diseño es necesario dicha conexión. Una vez creada se guarda en memoria y queda grabada para futuras sesiones. En nuestro caso particular los parámetros de conexión son:
  - Driver Class Name: com.mysql.jdbc.
  - Driver Connection URL: jdbc:mysql://localhost:3306/AEW.
  - User Name: root.
  - Password: curso_osbi.

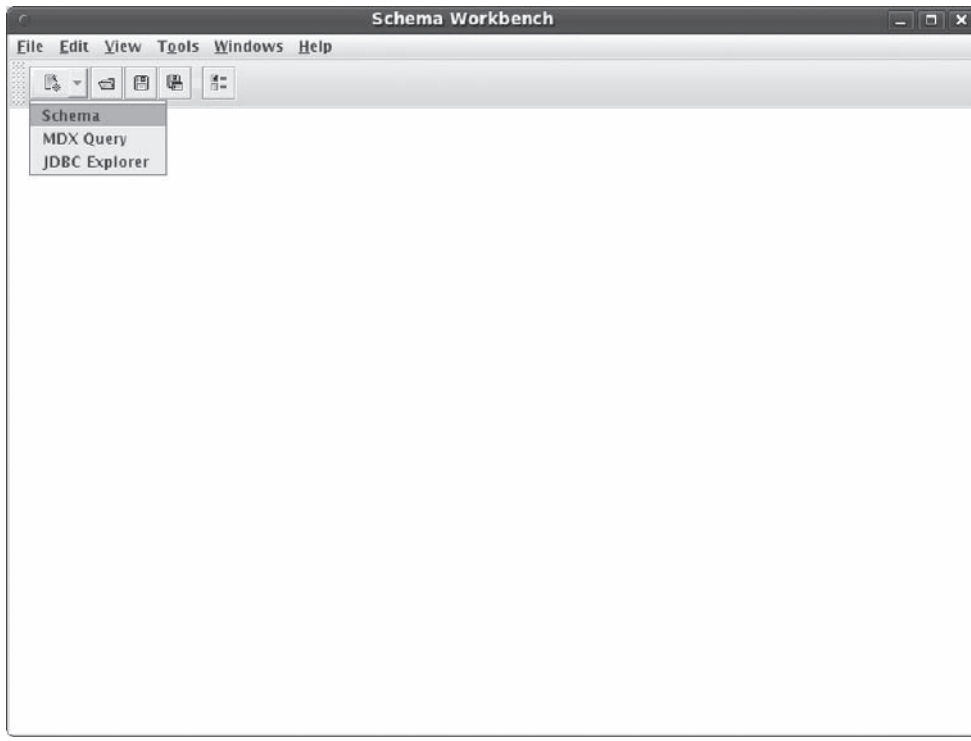
Es necesario recordar que en caso de que la base de datos fuera diferente, estos parámetros serían diferentes. En tal caso también sería necesario comprobar la inclusión del plugin jdbc en la herramienta.



En el caso de la herramienta de JasperSoft, este menú es ligeramente diferente, si bien las opciones a cumplimentar son las mismas.

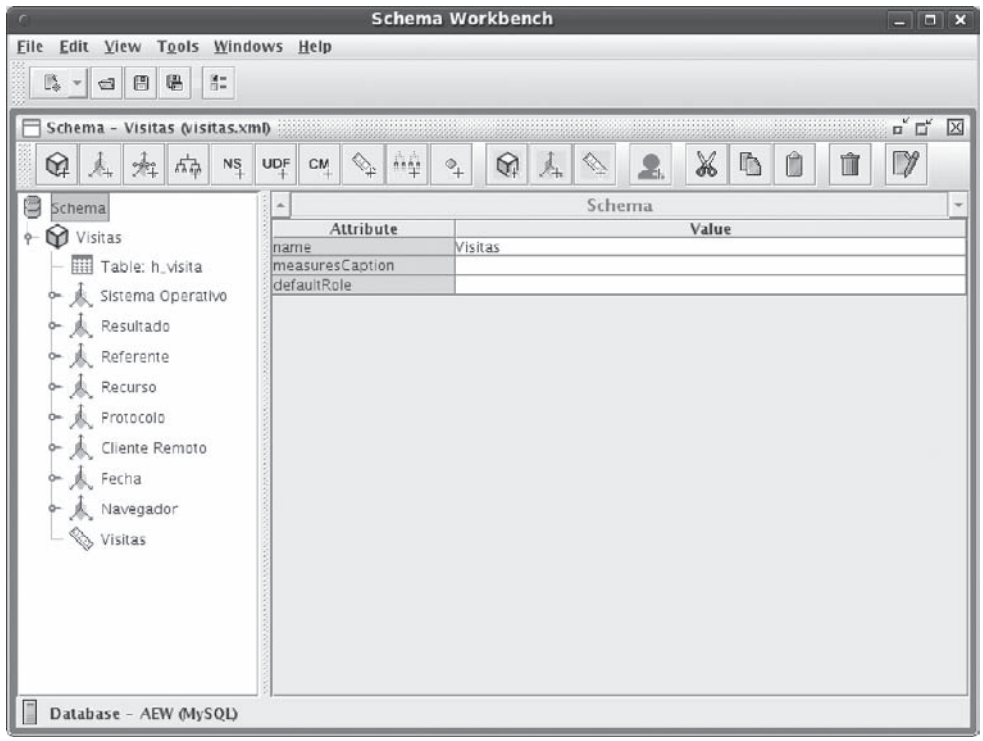


- Una vez creada la conexión podemos crear nuestro primer esquema. Los pasos son: crear un esquema, uno o varios cubos, una o varias tablas de hecho, una o varias dimensiones y una o varias métricas.

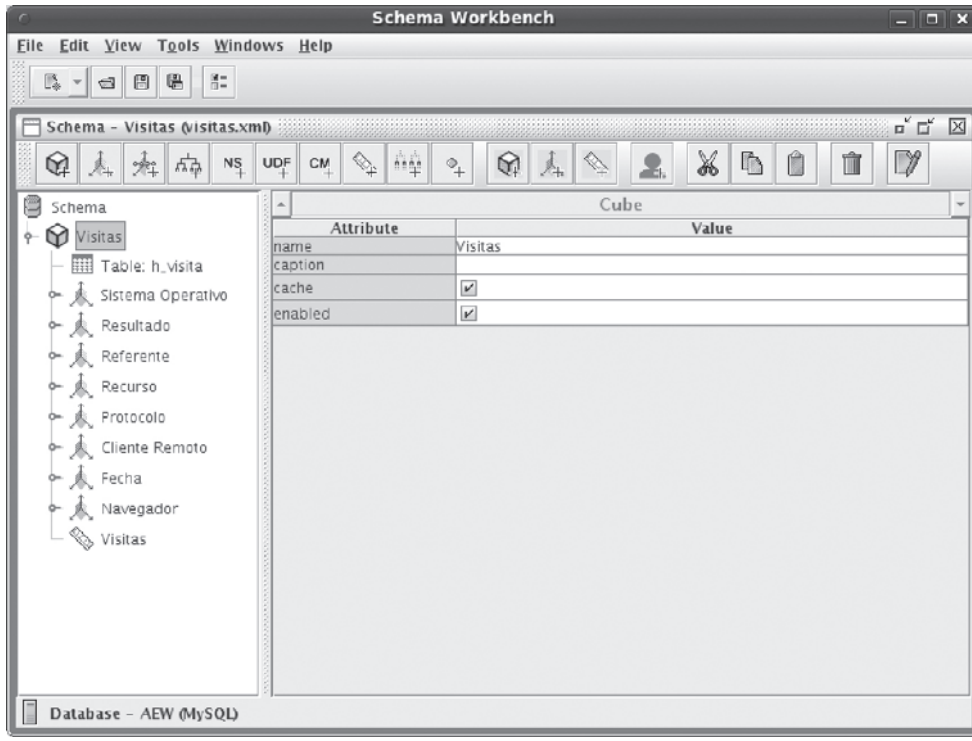


Para entender el proceso analizaremos un esquema creado completamente.

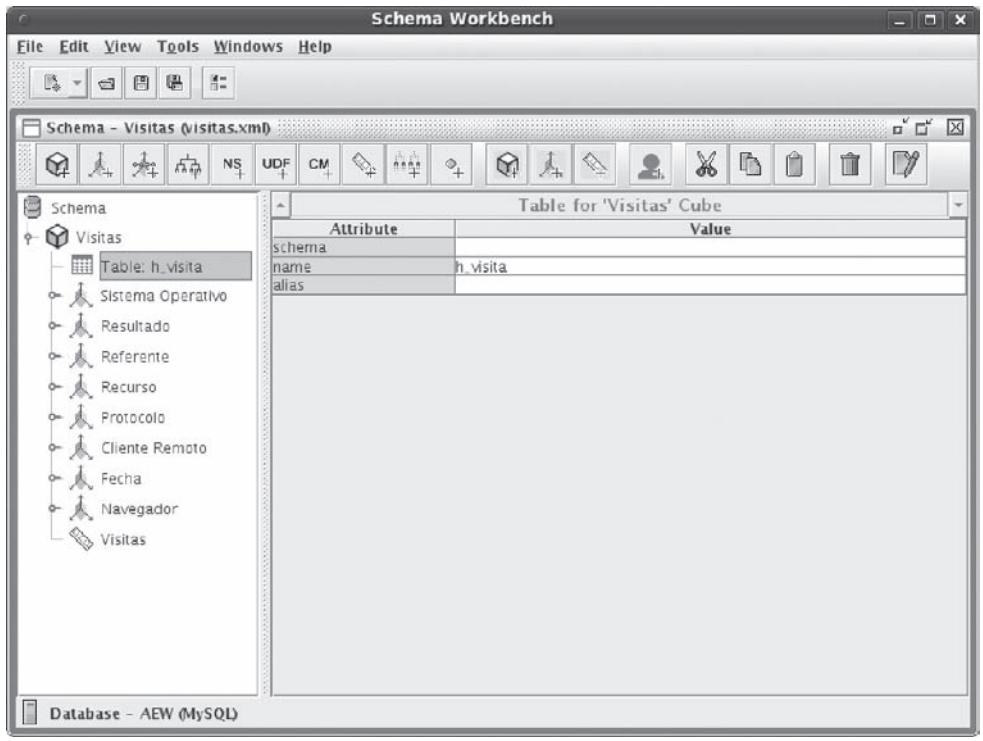
- Primero completamos el esquema introduciendo el nombre del esquema. En nuestro caso, por ejemplo, visitas. En caso de que se vaya creado un esquema que contiene diversos cubos, la recomendación sería nombrarlo con el nombre del proyecto, AEW.


Schema Workbench
File Edit View Tools Windows Help
Schema - Visitas (visitas.xml)
NS
UDF
CM
Schema
Visitas
Table: h_visita
Sistema Operativo
Resultado
Referente
Recurso
Protocolo
Cliente Remoto
Fecha
Navegador
Visitas
Schema
Attribute
Value
name
Visitas
measuresCaption
defaultRole
Database - AEW (MySQL)

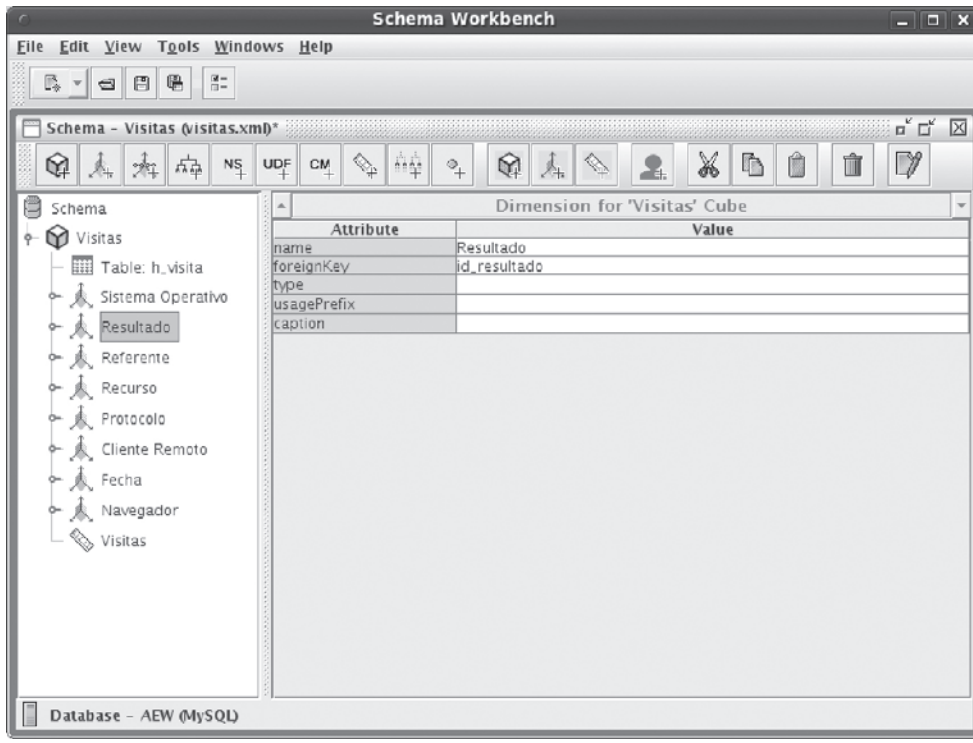
- Creamos un cubo. Debemos definir el nombre y activar las opciones de enabled y caché. Esta última opción es importante dado que indica al motor Mondrian que las consultas que se hagan deben guardarse en la caché.



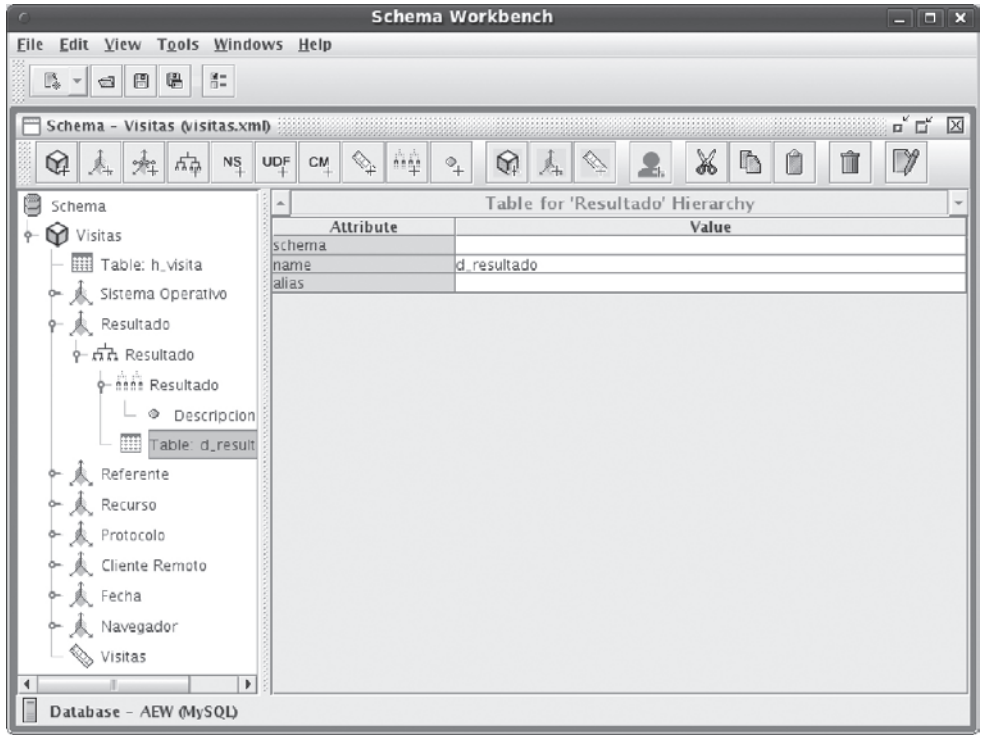
- Todo cubo necesita de una tabla de hecho. En nuestro caso, la de visitas. En el campo name, elegimos la tabla del data warehouse que tiene el rol de tabla de hecho.



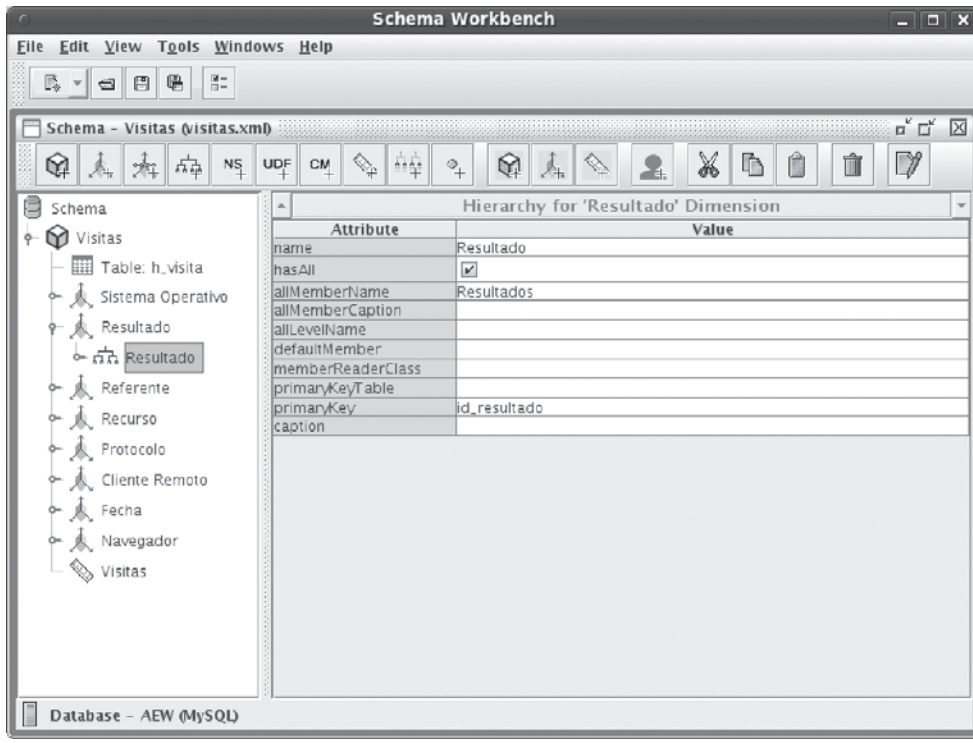
- Un cubo necesita de al menos una dimensión. Analizamos el caso de creación de la dimensión resultado. Definimos su nombre y la correspondiente clave foránea (foreignKey).



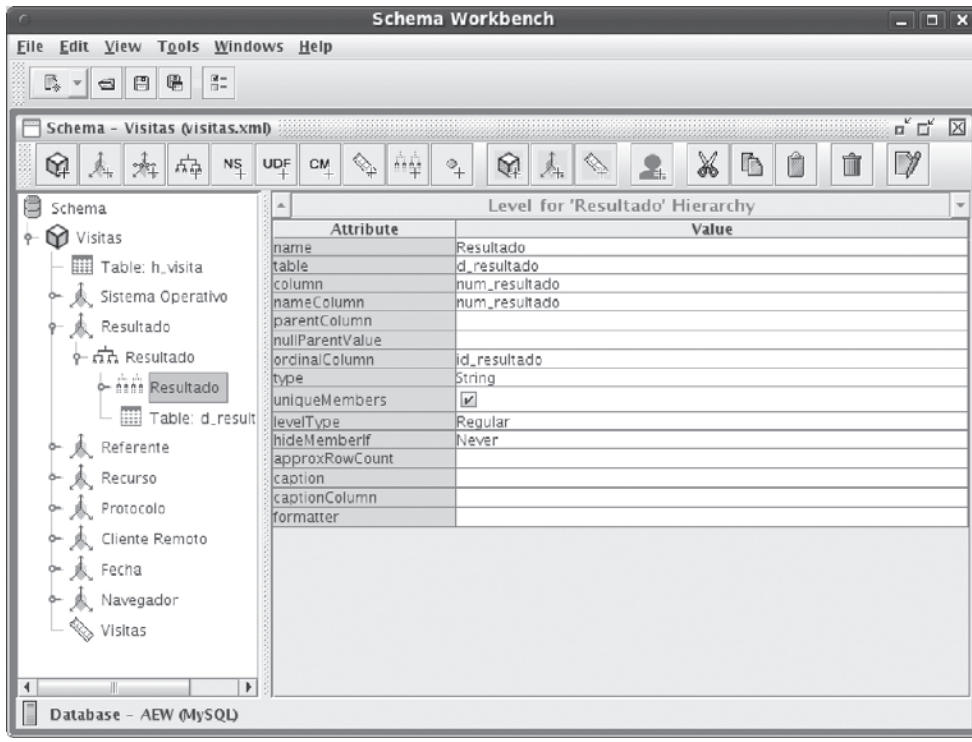
- Es necesario definir la tabla que contiene la información de la dimensión. En nuestro caso, d_resultado.



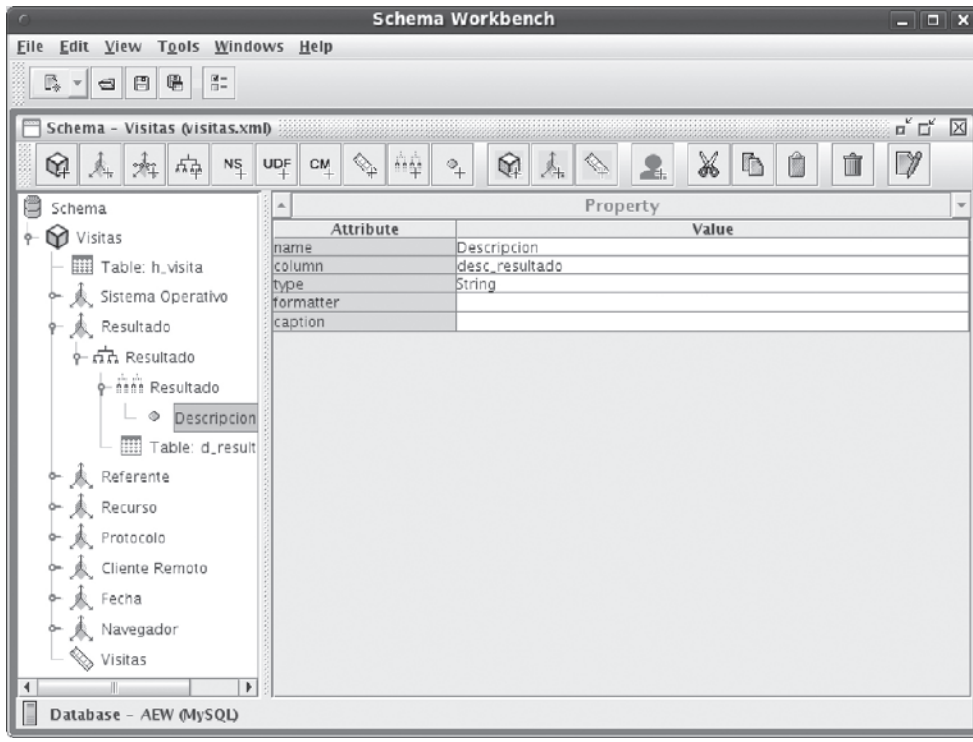
- Toda dimensión tiene una o varias jerarquías. Para definir cada jerarquía es necesario definir su nombre, indicar si acumula los valores (hasAll), el nombre de dicha acumulación y, finalmente, su clave primaria.



- Toda jerarquía necesita de como mínimo un nivel que la componga. Para definir correctamente el nivel debemos especificar el nombre del nivel, la tabla donde está la información, la columna que contiene dicha información, cómo se ordenan los resultados, la tipología del valor, la tipología de nivel, si los valores son únicos y si es necesario ocultar el nivel en algún caso.

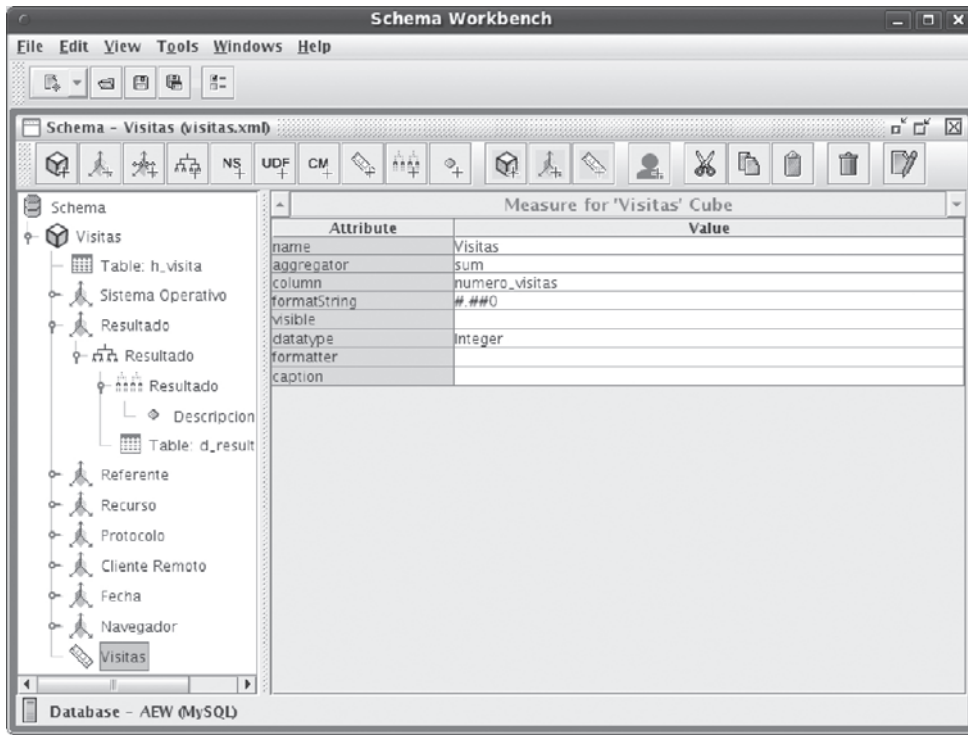


- Algunos miembros contienen información adicional que podemos mostrar como propiedades. En este caso, para resultado podemos mostrar la descripción del valor. Añadimos una propiedad y definimos el nombre, la columna que contiene la información y la tipología del valor.



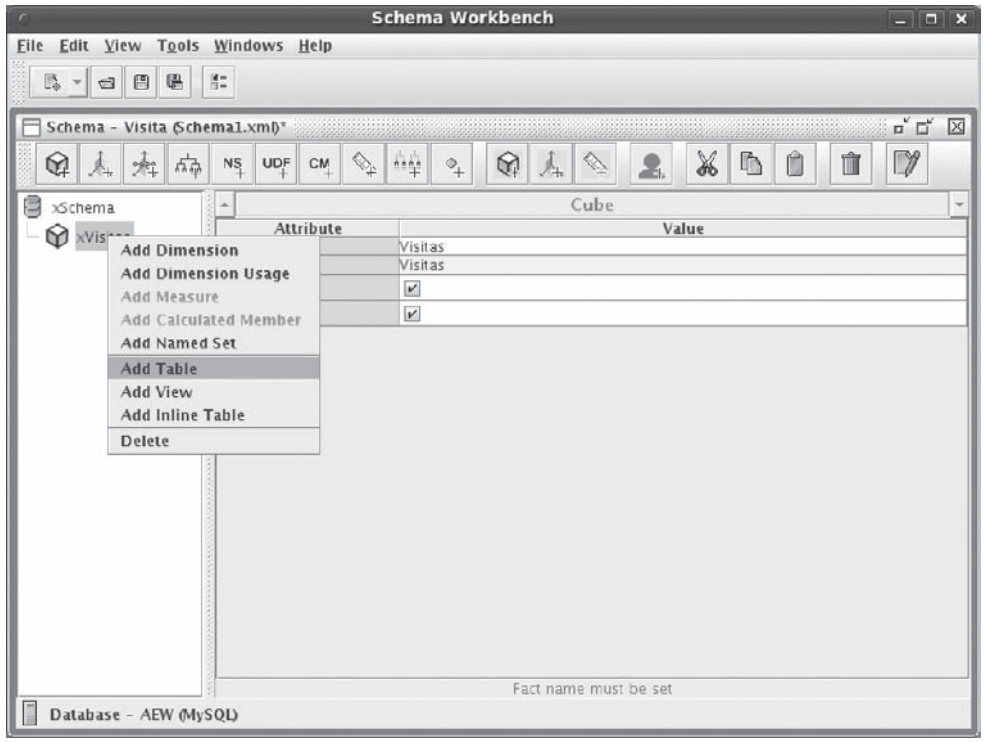
- El resto de dimensiones se definen de la misma forma.

- Una vez definidas las dimensiones, es necesario definir las métricas. En este caso, sólo definimos las visitas.

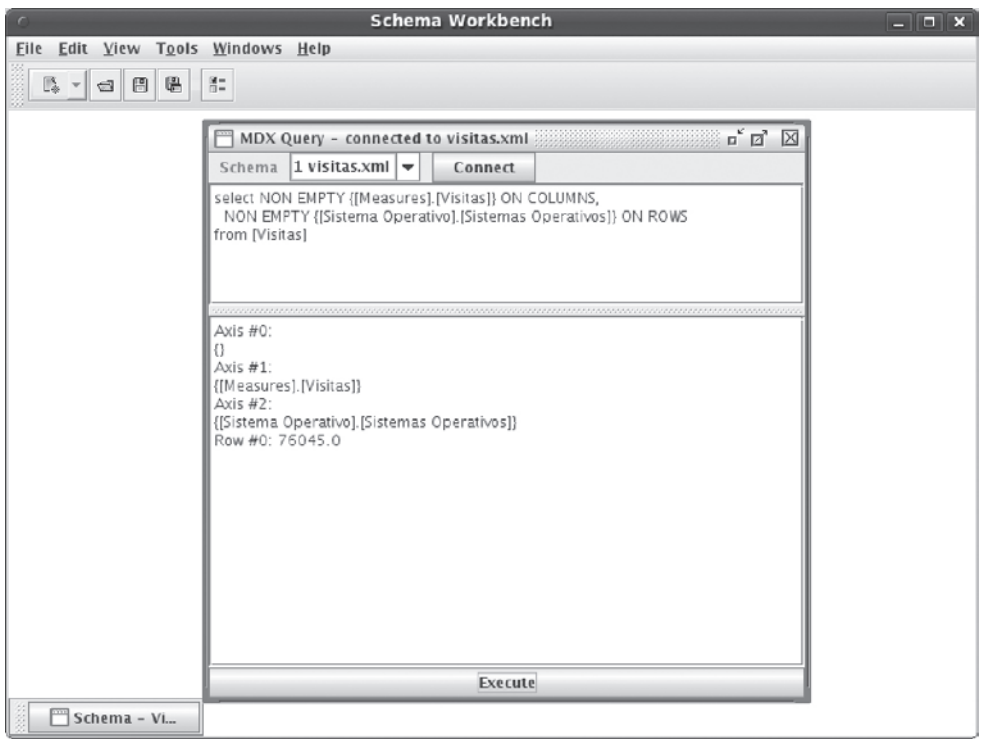


- Finalmente guardamos el esquema creado.

- La herramienta de desarrollo comprueba que el esquema que vamos creando está bien definido (en caso negativo, en la parte inferior aparece un mensaje en rojo).



- Podemos comprobar que el diseño que hemos realizado funciona correctamente y responde a nuestras preguntas. Para ello usamos el MDX query. Si la conexión a base de datos se ha definido correctamente y el esquema OLAP está bien definido, se conectará con éxito. Para poder comprobar el funcionamiento es necesario escribir una consulta MDX. El lenguaje MDX es similar a SQL pero mucho más complejo. Vamos a considerar una consulta sencilla en la que ponemos las medidas en la columna y la dimensión sistema operativo en las filas.



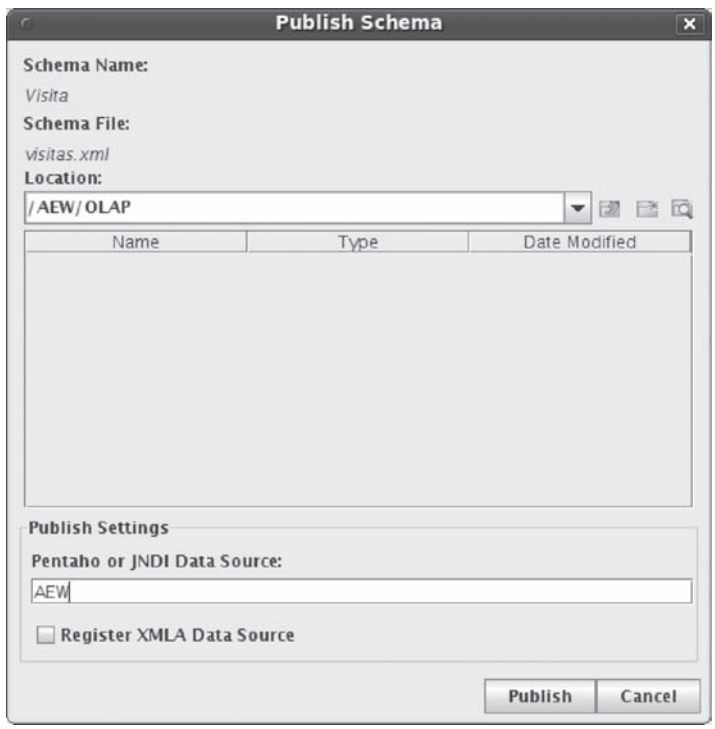
El resultado son 76.045 visitas.

3.2. Publicación de un esquema OLAP en Pentaho Server

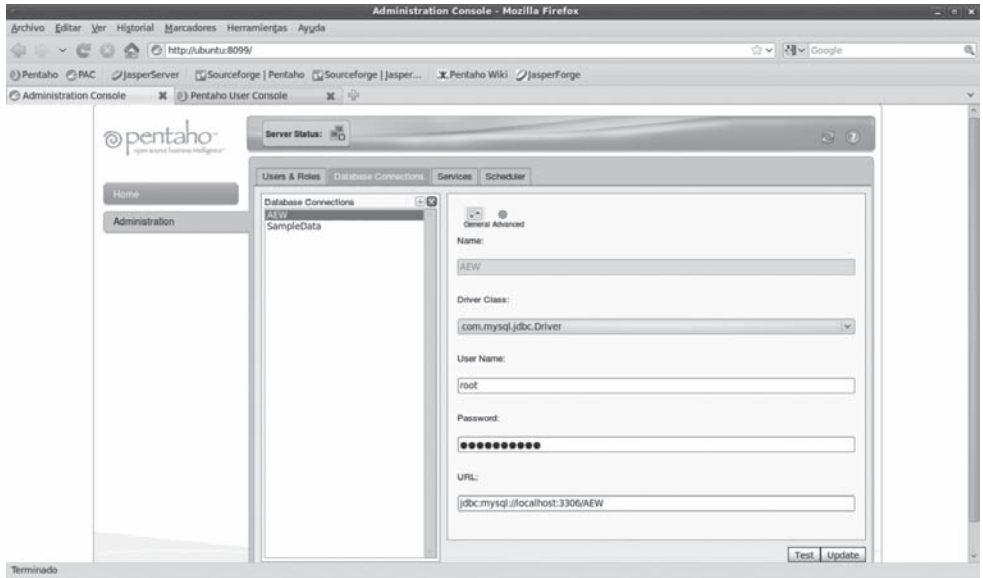
La publicación del esquema OLAP en Pentaho Server se puede hacer directamente en Pentaho Schema Workbench. Seguimos el siguiente proceso:

- Nos conectamos al repositorio. Previamente tenemos que haber definido una contraseña de publicación.

- Definimos la carpeta en pentaho-solutions donde se guardará el esquema que JDNI usará (debe estar previamente definida a través de la consola de administración de Pentaho).

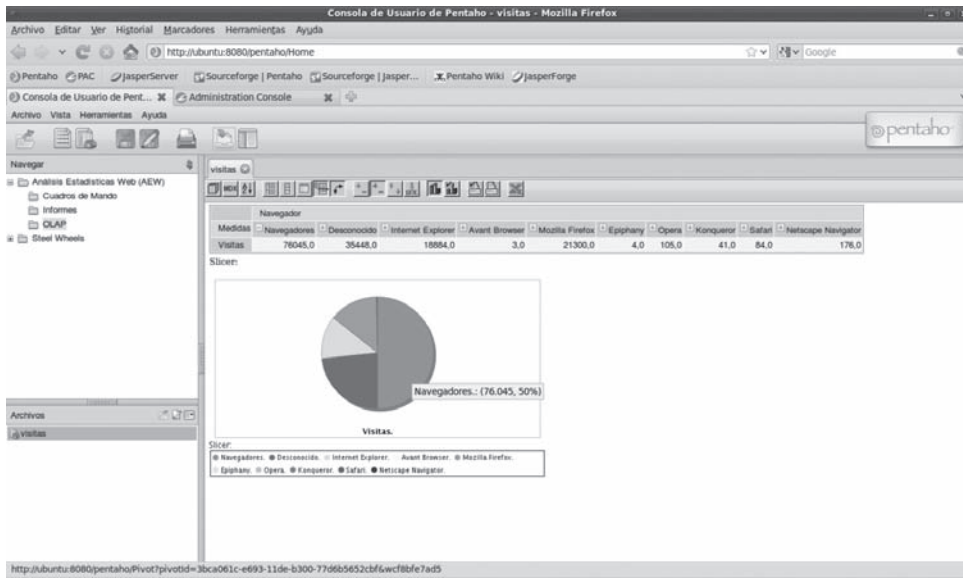


- En la PAC (Pentaho Administration Console) podemos editar las conexiones a base de datos.

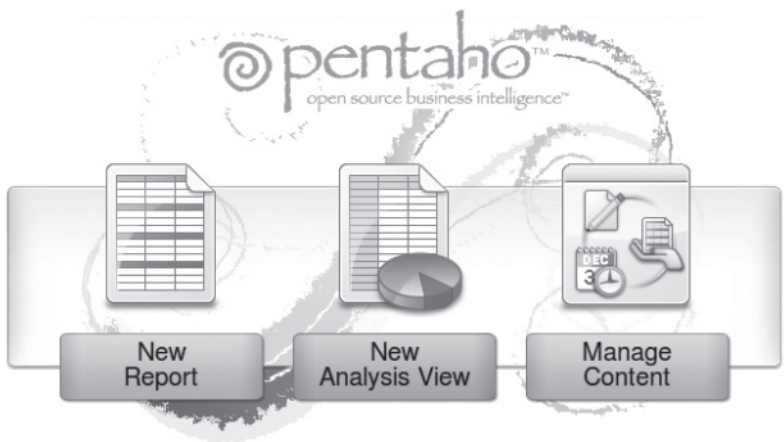


- Si todo está definido de forma correcta, la publicación finalizará con éxito.

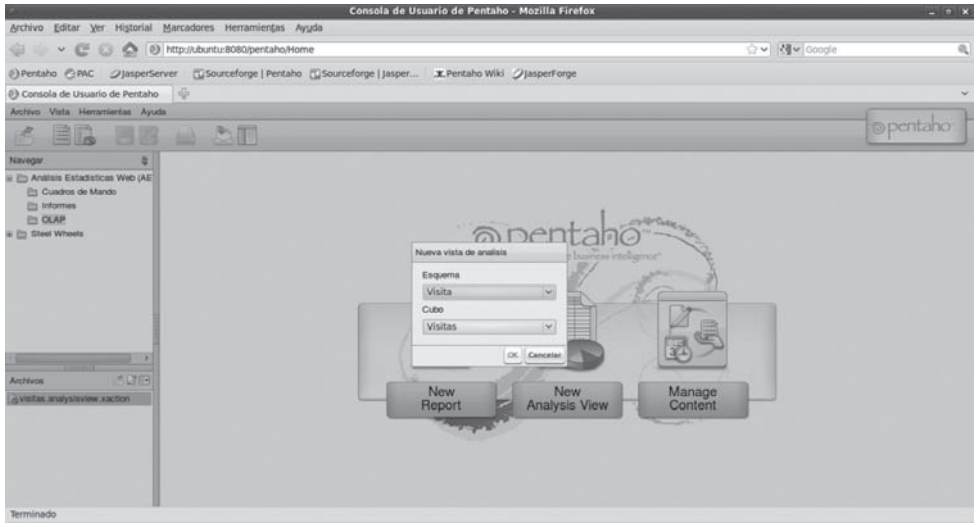
- Finalmente podemos comprobar que ha sido publicado correctamente entrando en el servidor de Pentaho.



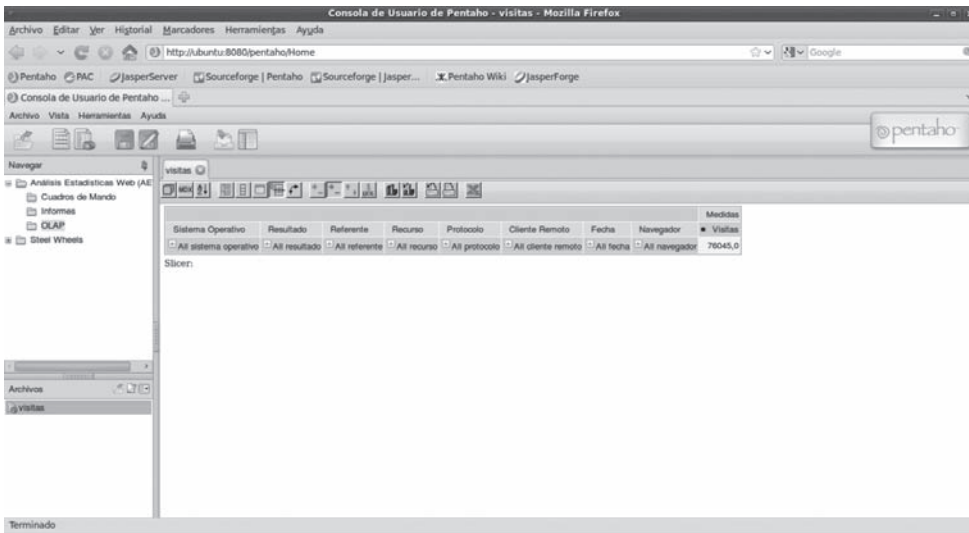
- Podemos crear una nueva consulta a través de la interfaz web de Pentaho. En la pantalla inicial seleccionamos new analysis view.



- Escogemos nuestro esquema y nuestro cubo visitas.



- Aparecerá la navegación por defecto que a partir del menú superior podremos refinar la información mostrada. Una vez terminemos, guardamos la consulta con el menú superior, y escogemos dónde se guarda nuestra vista OLAP.



4. Anexo 1: MDX

MDX es un lenguaje de consultas OLAP creado en 1997 por Microsoft. No es un estándar pero diversos fabricantes lo han adoptado como el estándar de hecho.

Tiene similitudes con el lenguaje SQL, si bien incluye funciones y fórmulas especiales orientadas al análisis de estructuras jerarquizadas que presentan relaciones entre los diferentes miembros de las dimensiones.

Sintaxis de MDX

La sintaxis de MDX es compleja; la mejor manera de ejemplificarla es a través de un caso determinado. Imaginemos un cubo de ventas con las siguientes dimensiones:

- Temporal de las ventas con niveles de año y mes.
- Productos vendidos con niveles de familia de productos y productos.
- Medidas: importe de las ventas y unidades vendidas.

Para obtener, por ejemplo, el importe de las ventas para el año 2008 para la familia de productos lácteos, la consulta seria:

```
SELECT
{[medidas].[importe ventas]}
on columns,
{[tiempo].[2008]}
on rows FROM [cubo ventas] WHERE ([Familia].[lácteos])
```

Es posible observar que la estructura general de la consulta es de la forma SELECT... FROM... WHERE...:

- En el select se especifica el conjunto de elementos que queremos visualizar que debe especificarse si se devuelve en columnas o filas.
- En el from, el cubo de donde se extrae la información.
- En el where, las condiciones de filtrado.
- { } permite crear listas de elementos en las selecciones.
- [ ] encapsulan elementos de las dimensiones y niveles.

Funciones de MDX

MDX incluye múltiples funciones para realizar consultas a través de la jerarquía existente en el esquema OLAP. Podemos destacar entre ellas:

- CurrentMember: permite acceder al miembro actual.
- Children: permite acceder a todos los hijos de una jerarquía.
- prevMember: permite acceder al miembro anterior de la dimensión.
- CrossJoin(conjunto_a,conjunto_b): obtiene el producto cartesiano de dos conjuntos de datos.
- BottomCount(conjunto_datos,N): obtiene un número determinado de elementos de un conjunto, empezando por abajo, opcionalmente ordenado.
- BottomSum(conjunto_datos,N,S): obtiene de un conjunto ordenado los N elementos cuyo total es como mínimo el especificado (S).
- Except(conjunto_a,conjunto_b): obtiene la diferencia entre dos conjuntos.
- AVG COUNT VARIANCE VARIANCE y todas las funciones trigonométricas (seno, coseno, tangente, etc.).
- PeriodsToDate: devuelve un conjunto de miembros del mismo nivel que un miembro determinado, empezando por el primer miembro del mismo nivel y acabando con el miembro en cuestión, de acuerdo con la restricción del nivel especificado en la dimensión de tiempo.
- WTD(<Miembro>): devuelve los miembros de la misma semana del miembro especificado.
- MTD(<Miembro>): devuelve los miembros del mismo mes del miembro especificado.
- QTY(<Miembro>): devuelve los miembros del mismo trimestre del miembro especificado.
- YTD(<Miembro>): devuelve los miembros del mismo año del miembro especificado.
- ParallelPeriod: devuelve un miembro de un periodo anterior en la misma posición relativa que el miembro especificado.

Miembros calculados en MDX

Una de las funcionalidades más potentes que ofrece el lenguaje MDX es la posibilidad de realizar cálculos complejos tanto dinámicos (en función de los datos que se están analizando en ese momento) como estáticos. Los cubos mul-

tidimensionales trabajan con medidas (del inglés *measures*) y con miembros calculados (*calculated members*).

Las medidas son las métricas de la tabla de hechos a las que se aplica una función de agregación (count, distinct count, sum, max, avg, etc.).

Un miembro calculado es una métrica que tiene como valor el resultado de la aplicación de una fórmula que puede utilizar todos los elementos disponibles en un cubo, así como otras funciones de MDX disponibles. Estas fórmulas admiten desde operaciones matemáticas hasta condiciones semafóricas pasando por operadores de condiciones.

5. Glosario

DOLAP	Desktop On-Line Analytical Processing
EPL	Eclipse Public License
HOLAP	Hybrid On-Line Analytical Processing
JDBC	Java Database Connection
JNDI	Java Naming and Directory Interface
JSP	Java Server Page
MDX	Multidimensional eXpressions
MOLAP	Multidimensional On-Line Analytical Processing
MTD	Month To Day
OLAP	On-Line Analytical Processing
OSBI	Open Source Business Intelligence
PAT	Pentaho Analysis Tool
PSW	Pentaho Schema Workbench
QTY	Quarter To Day
ROLAP	Relational On-Line Analytical Processing
SaaS	Software as a Service
SQL	Structured Query Language
XML/A	XML for Analysis
WTD	Week To Day
YTD	Year To Day

6. Bibliografía

BOUMAN, R., y VAN DONGEN, J. (2009). *Pentaho® Solutions: Business Intelligence and Data Warehousing with Pentaho® and MySQL*. Indianapolis: Wiley Publishing.

MENDACK, S. (2008). *OLAP without Cubes: Data Analysis in non-cube Systems*. Hoboken: VDM Verlag.

SCHRADER, M., y otros (2009). *Oracle Essbase & Oracle OLAP: The Guide to Oracle's Multidimensional Solutions*. Nueva York: McGraw-Hill.

THOMSEN, E. (2002). *OLAP Solutions: Building Multidimensional Information Systems*. Hoboken: John Wiley & Sons.

WEBB, C., y otros (2006). *MDX Solutions: With Microsoft SQL Server Analysis Services 2005 and Hyperion Essbase*. Hoboken: John Wiley & Sons.

WREMBEL, R. (2006). *Datawarehouses and OLAP: Concepts, Architectures and Solutions*. Hersbey: IGI Globals.

Capítulo V

Diseño de informes

El punto de entrada tradicional para una herramienta de inteligencia de negocio en el contexto de una organización es la necesidad de informes operacionales.

A lo largo de la vida de una empresa, la cantidad de datos que se generan por su actividad de negocio crece de forma exponencial, y esa información se guarda tanto en las bases de datos de las aplicaciones de negocio como en ficheros en múltiples formatos.

Es necesario generar y distribuir informes para conocer el estado del negocio y poder tomar decisiones a todos los niveles: operativo, táctico y estratégico.

El primer enfoque es modificar las aplicaciones de negocio para que las mismas puedan generar los informes. Frecuentemente el impacto en las aplicaciones es considerable, y afecta tanto el rendimiento de los informes como de las operaciones que soporta la aplicación.

Es en ese momento cuando se busca una solución que permita generar informes sin impactar en el rendimiento de las aplicaciones de negocio.

Es necesario comentar que:

- Las herramientas de informes existen desde hace mucho tiempo y, por ello mismo, son soluciones maduras que permiten cubrir las necesidades de los usuarios finales en lo que se refiere a informes.
- Cada fabricante soporta la creación de todo tipo de informes; en función del enfoque, la dependencia de los usuarios finales respecto el departamento IT puede ser diferente.
- Las fuentes de origen de los informes son varias, desde el propio data warehouse, OLAP, metadatos u ODS.

- Las últimas tendencias en informes son incorporar mayores capacidades de visualización y proporcionar mayor libertad a los usuarios finales y funcionalidades para embeber informes dinámicos en PDF o PPT.

El objetivo de este capítulo es presentar los elementos de informe, criterios de realización y un ejemplo a través de un caso práctico.

1. Informes e inteligencia de negocio

Las herramientas de informes (o también llamadas de reporting) permiten responder principalmente a la pregunta de ¿qué pasó? Dado que ésa es la primera pregunta que se formulan los usuarios de negocio, todas las soluciones de Business Intelligence del mercado incluyen un motor de reporting.

Definamos primero qué es un informe:

> Un informe es un documento a través del cual se presentan los resultados de uno o varios procesos de negocio. Suele contener texto acompañado de elementos como tablas o gráficos para agilizar la compresión de la información presentada.

Los informes están destinados a usuarios de negocio que tienen la necesidad de conocer la información consolidada y agregada para la toma de decisiones.

Ahora podemos definir formalmente las herramientas de reporting:

> Se entiende por plataforma de reporting aquellas soluciones que permiten diseñar y gestionar (distribuir, planificar y administrar) informes en el contexto de una organización o en una de sus áreas.

1.1. Tipos de informes

Existen diferentes tipos de informes en función de la interacción ofrecida al usuario final y la independencia respecto al departamento TI:

- **Estáticos:** tienen un formato preestablecido inamovible.
- **Paramétricos:** presentan parámetros de entrada y permiten múltiples consultas.
- **Ad-hoc:** son creados por el usuario final a partir de la capa de metadatos que permite usar el lenguaje de negocio propio.

1.2. Elementos de un informe

Principalmente un informe puede estar formado por:

- Texto: que describe el estado del proceso de negocio o proporciona las descripciones necesarias para entender el resto de elementos del informe.
- Tablas: este elemento tiene forma de matriz y permite presentar una gran cantidad de información.
- Gráficos: este elemento persigue el objetivo de mostrar información con un alto impacto visual que sirva para obtener información agregada o sumarizada con mucha más rapidez que a través de tablas.
- Mapas: este elemento permite mostrar información geolocalizada.
- Métricas: que permiten conocer cuantitativamente el estado de un proceso de negocio.
- Alertas visuales y automáticas: consiste en avisos del cambio de estado de información que pueden estar formadas por elementos gráficos como fechas o colores resultados y que deben estar automatizadas en función de reglas de negocio encapsuladas en el cuadro de mando.

1.3. Tipos de métricas

Los informes incluyen métricas de negocio. Es por ello necesario definir los diferentes tipos de medidas existentes basadas en el tipo de información de recopilan así como la funcionalidad asociada:

- Métricas: valores que recogen el proceso de una actividad o los resultados de la misma. Estas medidas proceden del resultado de la actividad de negocio.
  - Métricas de realización de actividad (leading): miden la realización de una actividad. Por ejemplo, la participación de una persona en un evento.
  - Métricas de resultado de una actividad (lagging): recogen los resultados de una actividad. Por ejemplo, la cantidad de puntos de un jugador en un partido.
- Indicadores clave: entendemos por este concepto, valores correspondientes que hay que alcanzar, y que suponen el grado de asunción de los objetivos. Estas medidas proporcionan información sobre el rendimiento de una actividad o sobre la consecución de una meta.
  - Key Performance Indicator (KPI): indicadores clave de rendimiento. Más allá de la eficacia, se definen unos valores que nos explican en qué rango óptimo de rendimiento nos deberíamos situar al alcanzar los objetivos. Son métricas del proceso. Por ejemplo, la ratio de crecimiento de altas en un servicio.
  - Key Goal Indicator (KGI): indicadores de metas. Definen mediciones para informar a la dirección general si un proceso TIC ha alcanzado sus requisitos de negocio, y se expresan por lo general en términos de criterios de información. Si consideramos el KPI anterior, sería marcar un valor objetivo de crecimiento del servicio que se pretende alcanzar, por ejemplo, un 2%.

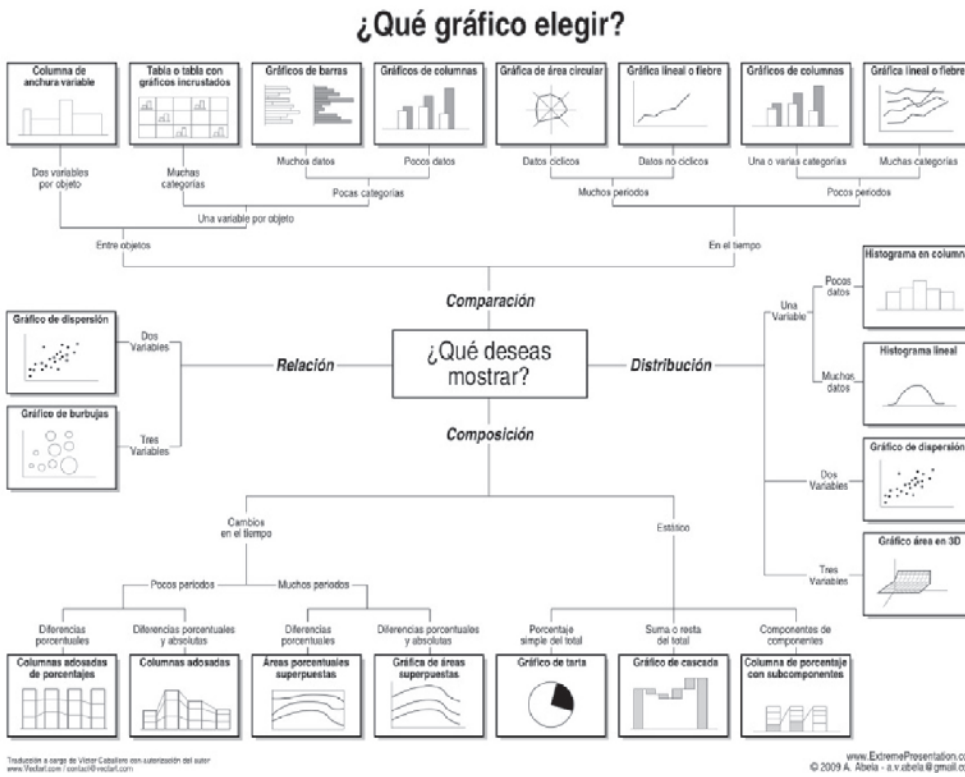
Debemos distinguir que:

- Existen también indicadores de desempeño. Los indicadores clave de desempeño (son, en definitiva, KPI) definen mediciones que determinan cómo se está desempeñando el proceso de TI para alcanzar la meta. Son los indicadores principales que indican si será factible lograr una meta o no, y son buenos indicadores de las capacidades, prácticas y habilidades.

- Los indicadores de metas de bajo nivel se convierten en indicadores de desempeño para los niveles altos.

1.4. Tipos de gráficos

En el proceso de confección de un cuadro de mandos, uno de los puntos más complicados es la selección del tipo de gráfico. El siguiente diagrama recoge los criterios que se suelen utilizar para escoger qué gráficos utilizar en un informe.



Fuente: *http://www.extremepresentation.com/*

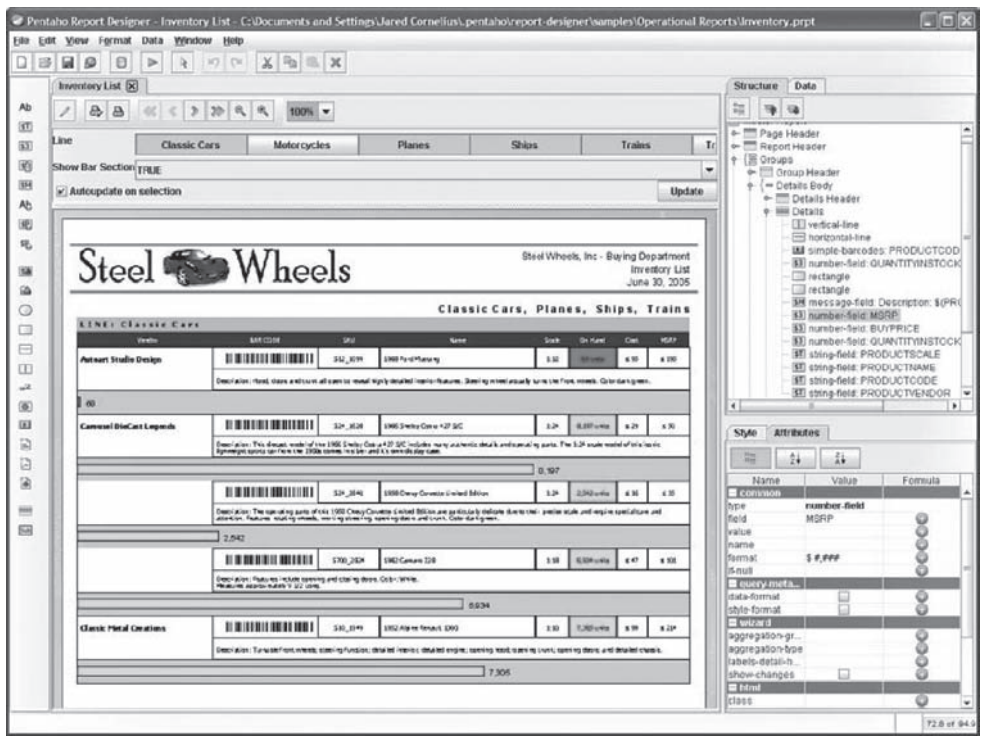
2. Informes en el contexto de Pentaho

Pentaho Reporting es el motor de reporting de Pentaho que está integrado en la suite. El proyecto se inició bajo el nombre JFreeReports en 2002. El origen del mismo se remonta a un desarrollo de David Gilbert, creador de JFreeChart, para cubrir necesidades de renderizado de informes. Pronto, se unió Thomas Morgner al proyecto erigiéndose en el principal desarrollador.

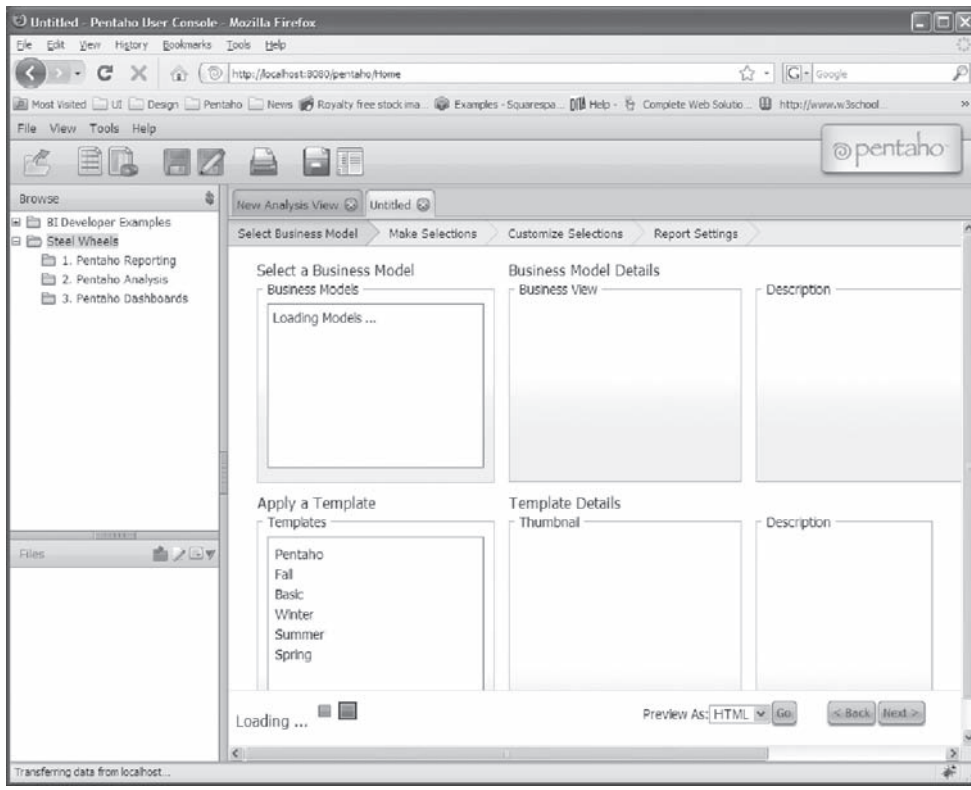
En el 2006, Pentaho adquirió el proyecto y Thomas entró a formar parte de la compañía.

En la actualidad se halla en la versión 3.5 y ha sido completamente rediseñado para incluir múltiples funcionalidades. Existen tres herramientas de desarrollo:

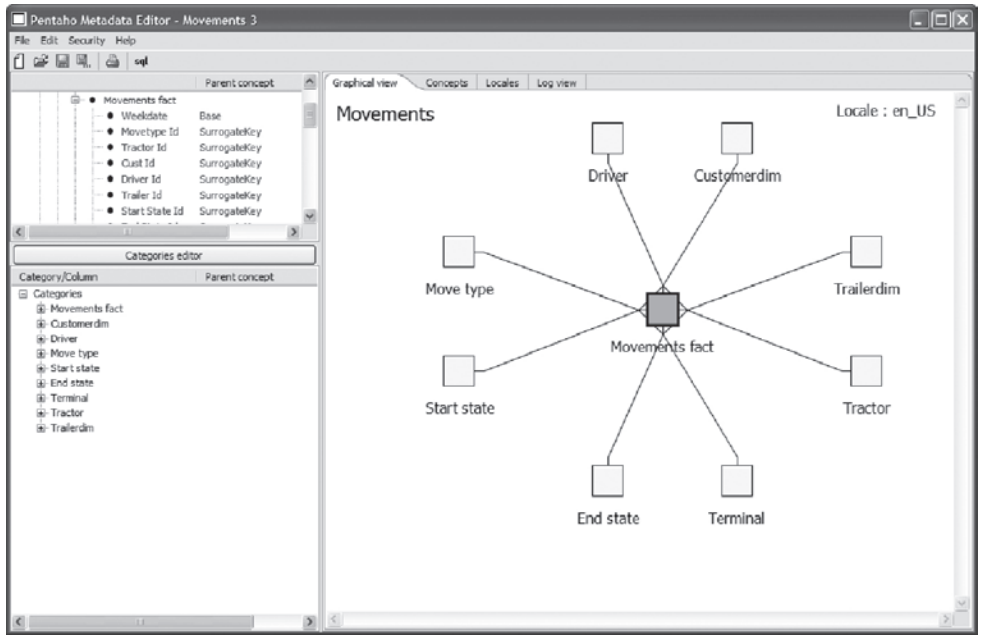
- Pentaho Report Designer: un editor basado en eclipse con prestaciones profesionales de customización de informes destinado a desarrolladores.



- WAQR (Web Ad-hoc Query Reporting): editor web que permite construir informes a los usuarios finales. Está basado en templates creados con Pentaho Report Designer y extrae la información de la capa de metadatos.



- Pentaho Metadata: permite crear una capa de metadatos basada en la información consolidada en el data warehouse.



Resumiendo:

	Community	Enterprise
Motor generador de informes (ejecución de informes)	Pentaho Reporting (JFreeReports)	
Herramientas de desarrollo (diseño de informes)	Pentaho Report Designer, WAQR, Pentaho Metadata	

Tratamos, a continuación, estas herramientas.

2.1. Pentaho Reporting

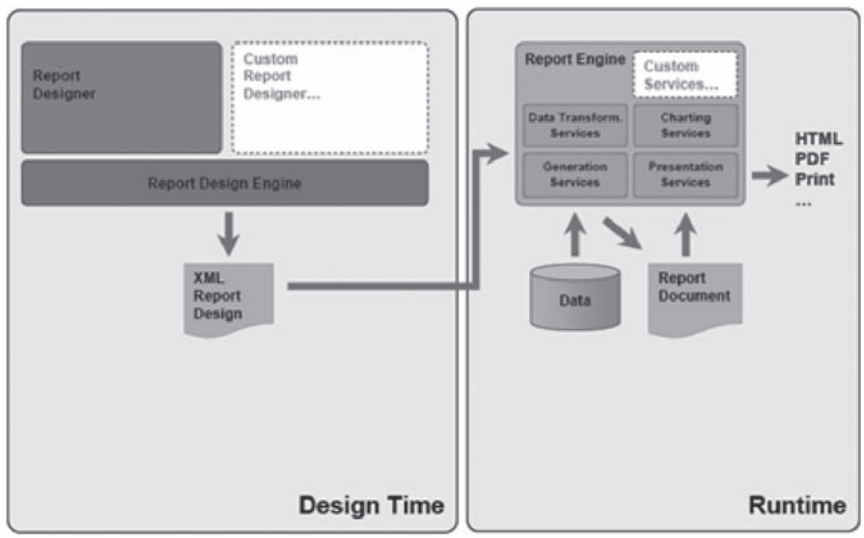
Es un motor de generación de informes basado en Pentaho Report Designer. Su única funcionalidad es la de renderizar informes generados con la herramienta de diseño.

A través de la suite de Pentaho, presenta capacidades de:

- Distribución de emails.
- Combinación del informe en un proceso complejo. Por ejemplo, ejecutar un proceso ETL para recoger información, pasarla al informe, ejecutar el informe y enviarlo vía correo.

2.2. Pentaho Report Designer

Pentaho Report Designer es un editor programado en Java que encapsula la lógica de un informe en un fichero XML (con extensión). El flujo de creación de informes está representado por el siguiente esquema:



Es una herramienta de diseño de informes basados en múltiples fuentes de origen. Dichos informes pueden ser consultados a través del servidor de Pentaho o, de ser necesario, distribuidos por correo a múltiples usuarios.

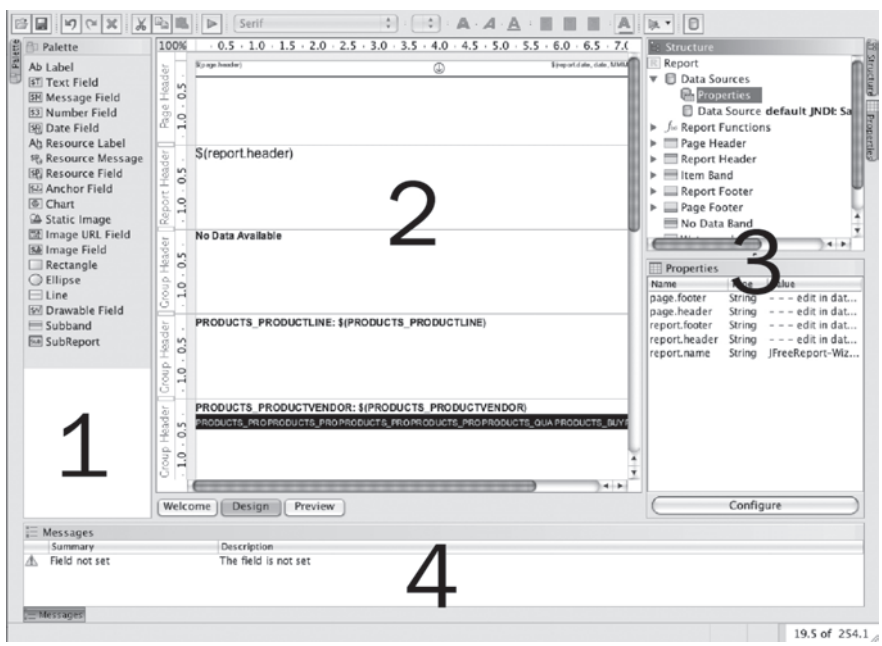
Las principales características de Pentaho Report Designer:

- Generales: previsualizador, editor gráfico, creación basada en bandas, drag & drop...
- Elementos que se pueden integrar en los informes: Chart, List, Table, Dynamic CrossTabs, Text, Dynamic Text, Image, Label, subreports, barcodes/sparklines, drill-back, crosstabs y elementos flash.

- Formato de salida de gráficos: PNG, JPG, SVG, EPS, PDF.
- Propiedades de los informes: Paginación, Templates, subreports, javascrpt scripting, hyperlinks...
- Tipos de informes: estáticos, paramétricos, ad-hoc.
- Formato de salida: PDF, HMTL, EXCEL y RTF.
- Distribución: bursting, email, web service.
- Fuentes de datos: JDBC, ODBC, Reflection, Hibernate, Kettle, Mondrian, OLAP4J, Pentaho Metadata, Scripted data.
- Existencia de un API de programación.
- Funcionalidades avanzadas de creación de mensajes de correo electrónico.
- Integración de un wizard para la creación de informes.

Esta herramienta sigue los despliegues por área tradicionales de las herramientas de diseño:

1) Paleta de elementos.
2) Área de trabajo (cada informe está formado por diferentes bandas, cabecera, agrupación, cuerpo, pie de página...).
3) Propiedades de los elementos.
4) Zona de errores y logs.



2.3. WAQR

Es un plugin existente en Pentaho Server cuyo objetivo es permitir crear informes a los usuarios finales a partir de la información encapsulada en la capa de metadatos de Pentaho.

Su funcionamiento consiste en un wizard de diversos pasos sencillos (como elegir el conjunto de datos para el informe y el template, elegir los elementos que salen en el informe y cómo se agrupan, elegir su formato y elegir el formato de salida de los elementos) que guían al usuario a crear un informe sin la necesidad de depender del departamento IT.

Actualmente sólo permite crear listados, es decir, que no permite usar gráficos.

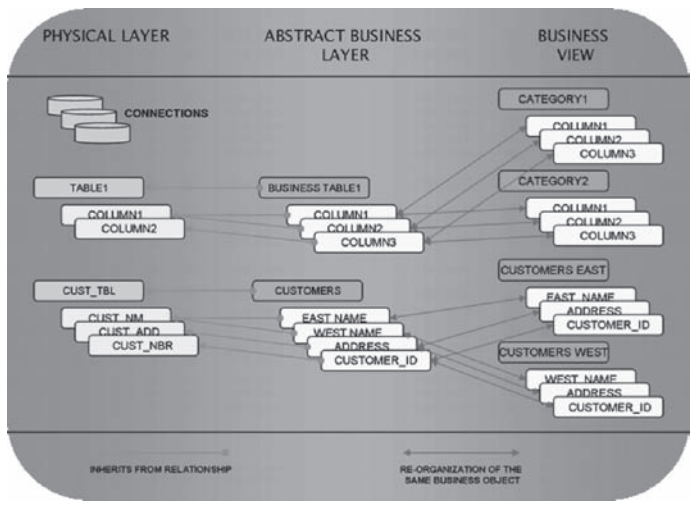
2.4. Pentaho Metadata

Pentaho incluye en su suite una capa de metadatos basada en el estándar del mercado CWM (Common Warehouse Metamodel).

Este proyecto está auspiciado por Matt Casters, creador de Pentaho Data Integration.

En el contexto de la inteligencia de negocio, la existencia de este elemento ofrece diversos beneficios:

• Independencia de los elementos de negocio del data warehouse.



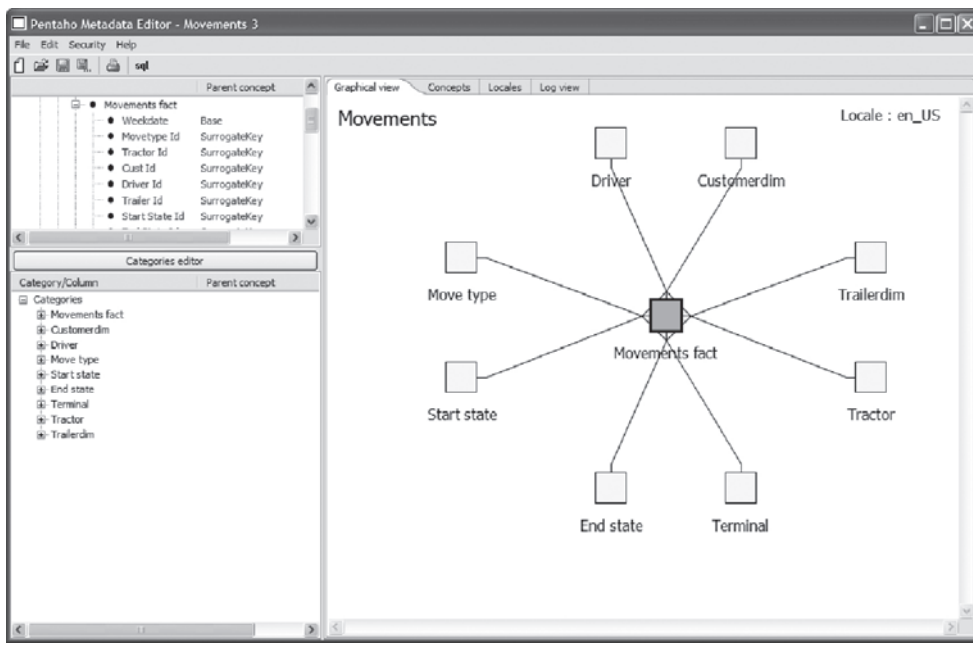
- Ofrecer la información del data warehouse a los usuarios finales mediante su lenguaje de negocio.
- Encapsular funcionalidades propias de diferentes departamentos en la capa de negocio: color, fuentes y descripción diferentes, etc.
- Soporte de la internacionalización de elementos.
- Definir la seguridad.

En el caso concreto de Pentaho, sólo las herramientas de reporting están explotando actualmente las capacidades de la capa de metadatos. Pero se espera que se extienda a otros elementos de análisis.

Esta herramienta de diseño permite crear vista de negocio. Estas vistas de negocio deben referenciar a una tabla del data warehouse. A partir de dicha referencia se crean las relaciones que estarán disponibles al usuario final en función de las necesidades que han sido identificadas.

El área de trabajo se divide en dos partes:

- Esquema jerárquico de elementos que incluye las conexiones y las vistas de negocio.
- Esquema gráfico de la vista de negocio.



3. Caso práctico

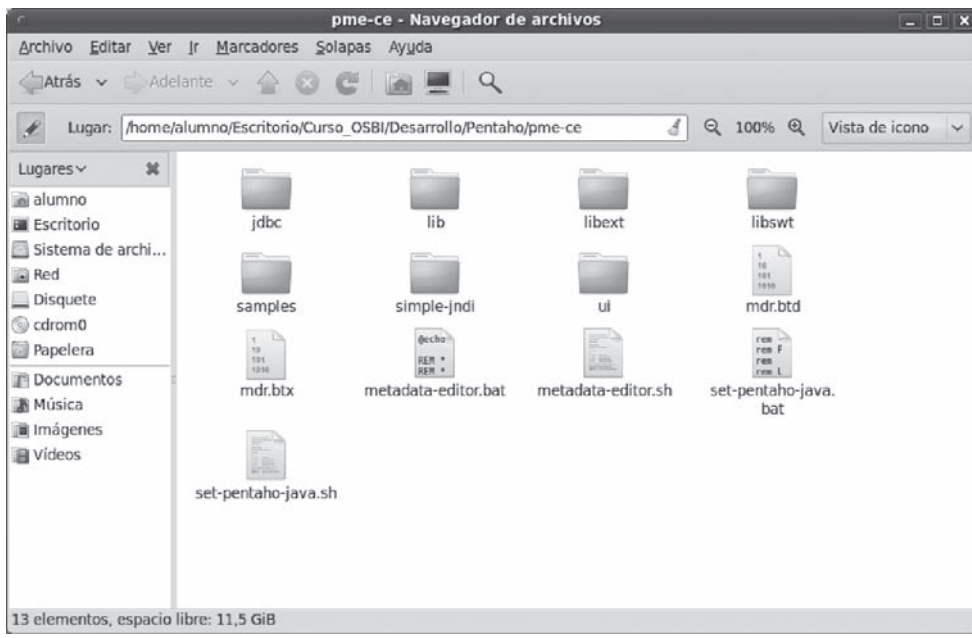
En el contexto de Pentaho existen diferentes opciones para publicar informes:

- Mediante la capa de metadatos.
- Mediante el wizard incluido en Pentaho Report Designer.
- Creando un informe desde cero con Pentaho Report Designer.

3.1. Diseño de la capa de metadatos en Pentaho

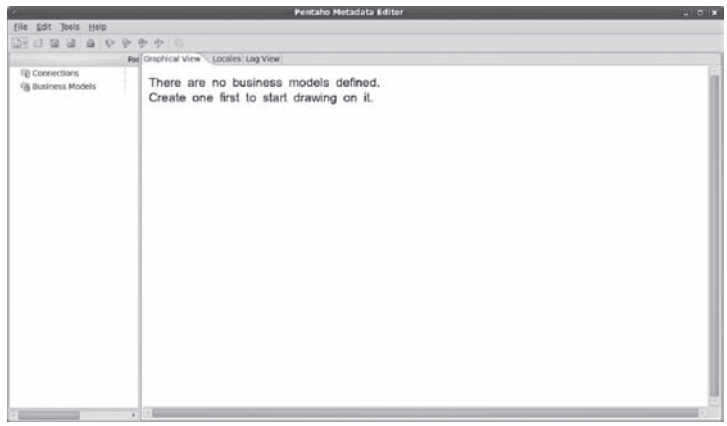
Para poder crear informes basados en la capa de metadatos, es necesario primero crear una capa de metadatos. Esta capa es una abstracción entre el data warehouse y los informes cuyo beneficio principal es proporcionar al usuario final la capacidad de crearse sus propios informes sin depender del departamento IT.

La herramienta Pentaho Metadata Editor es la que nos permite crear esta capa; se inicia a partir del fichero metadata-editor.sh.



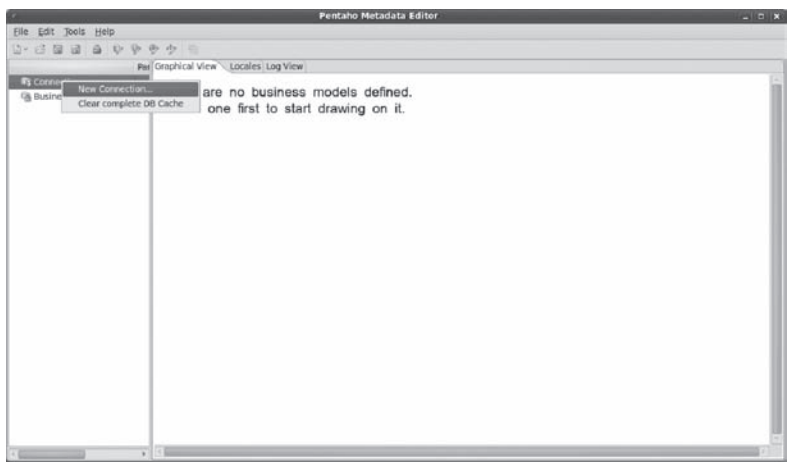
Cuando se inicia la herramienta no existe ningún elemento creado. Para crear una capa de metadatos es necesario:

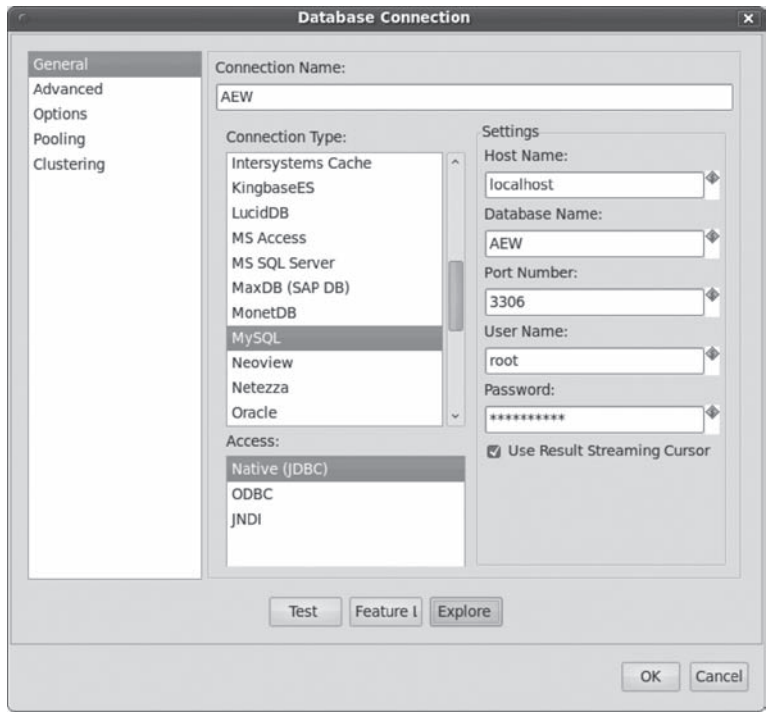
- Crear una conexión al data warehouse.
- Crear un modelo de negocio.
- Crear las relaciones del modelo de negocio.
- Crear las categorías.

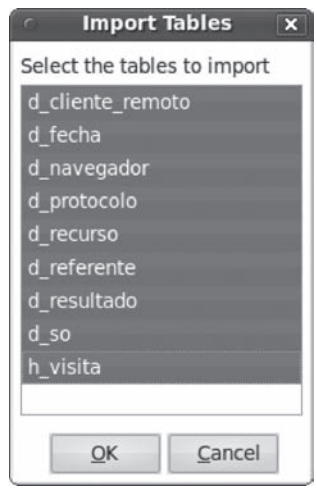


Procedemos a explicar el proceso.

- Creamos una nueva conexión y completamos los parámetros de conexión. Una vez conectados correctamente, seleccionamos las tablas que deben formar parte de nuestro modelo. Se han definido por lo tanto las tablas físicas de la capa de metadatos.

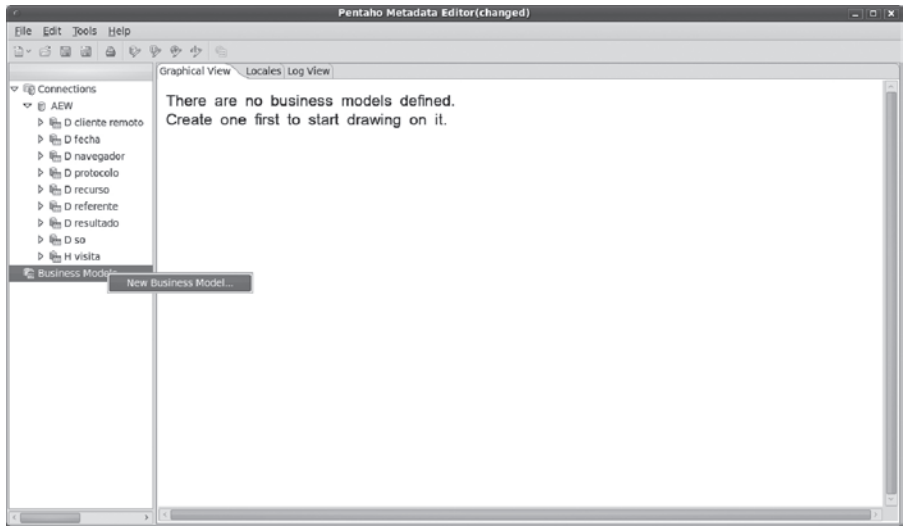


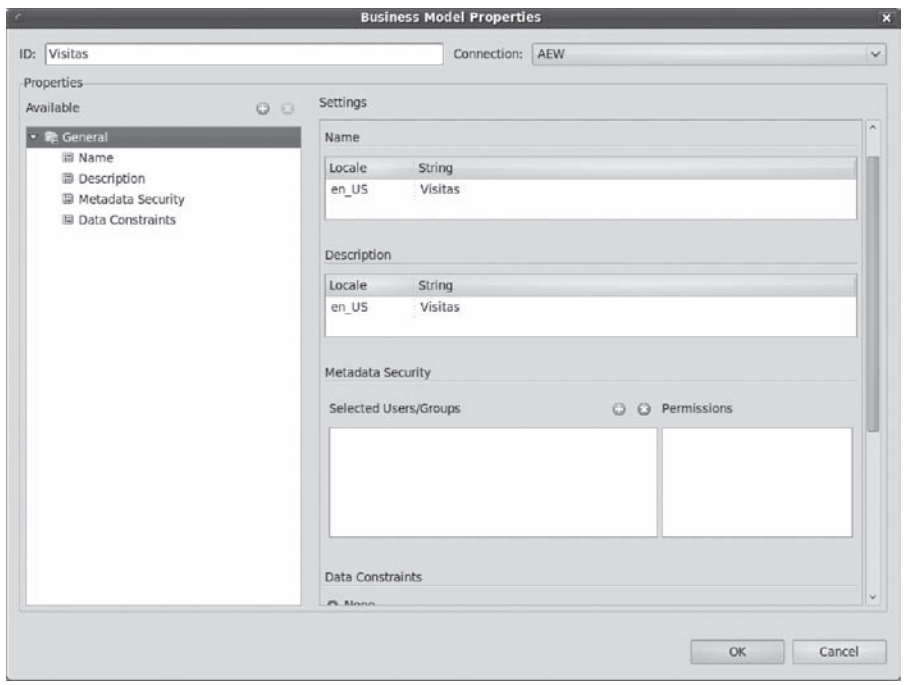




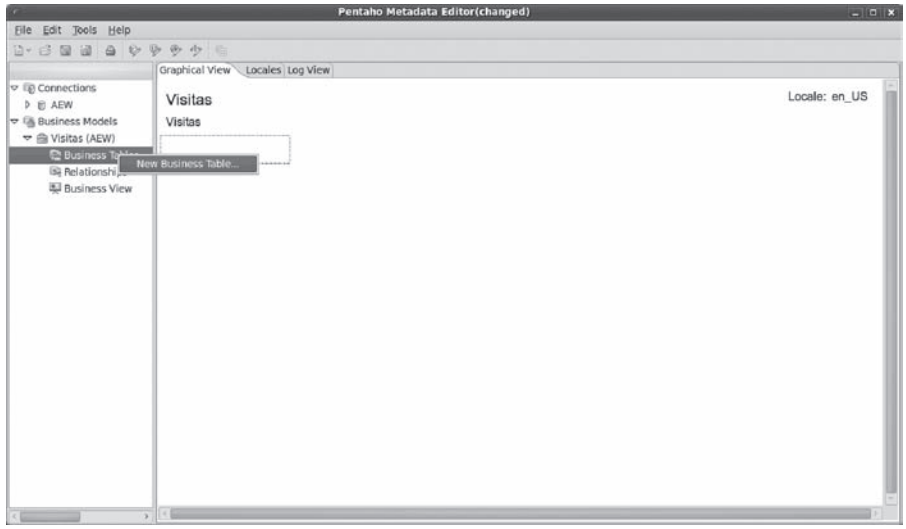
- El hecho de tener las tablas a nivel de conexión no supone que sean consideradas realmente en la capa de datos. Simplemente que están disponibles para el uso. Por lo tanto, es necesario mapear las tablas físicas con su

equivalente en el modelo de negocio usando lenguaje de negocio para el usuario final.

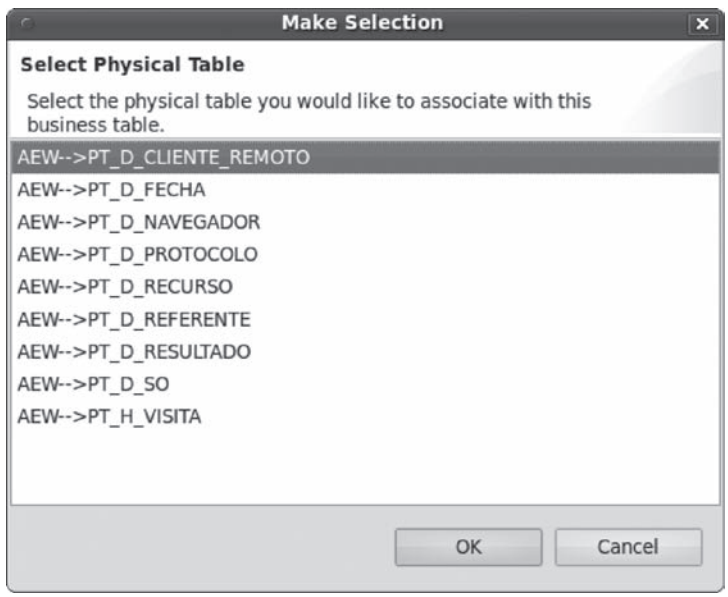




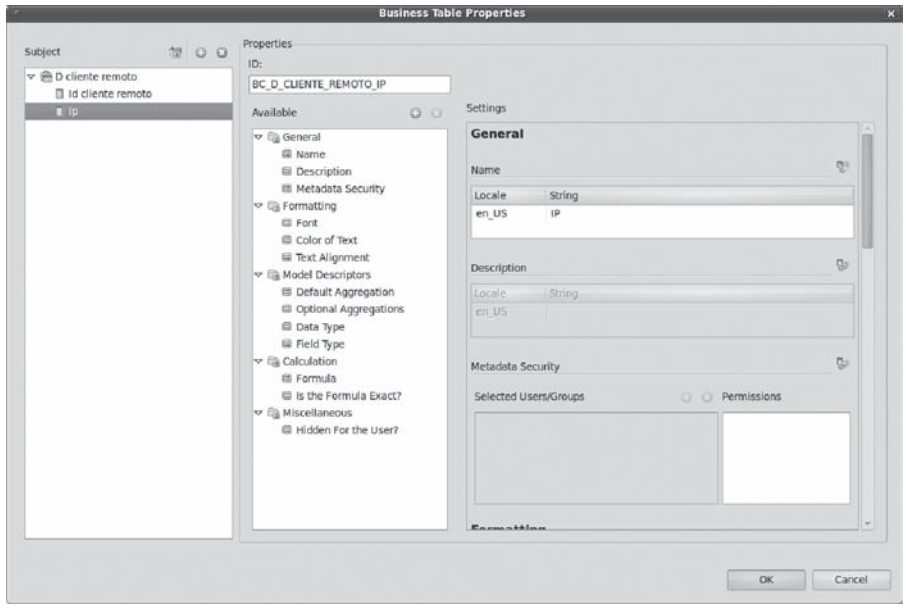
- Ahora es necesario crear las tablas de negocio. Esto consiste en el proceso de mapear las tablas físicas de la conexión con los nombres de negocio a usar tanto en el nombre de la tabla como en los atributos de la misma.



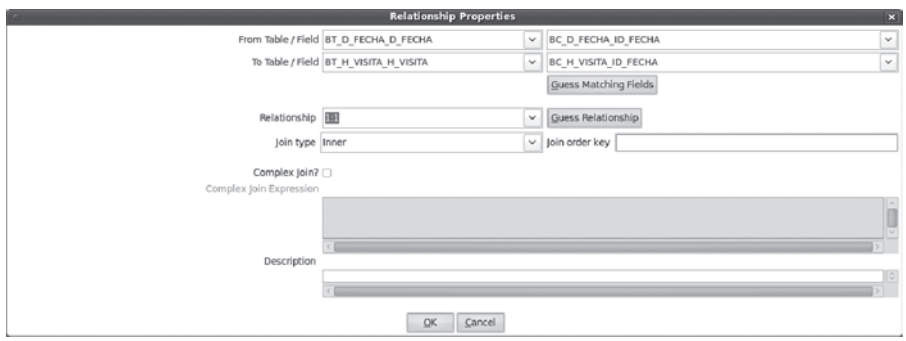
- Seleccionamos la tabla de cliente remoto.



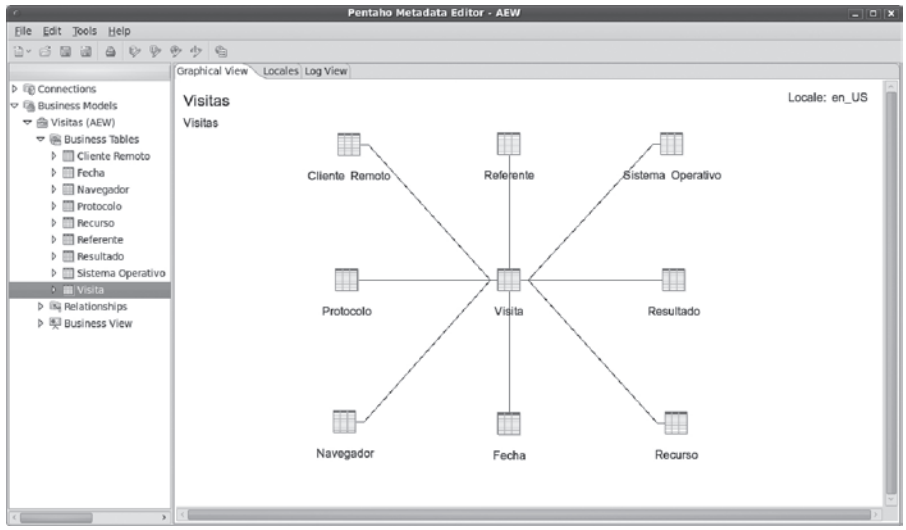
- Damos el nombre adecuado tanto a la tabla como a sus campos. Podemos definir otras características, pero para nuestro ejemplo nos centramos sólo en el nombre.



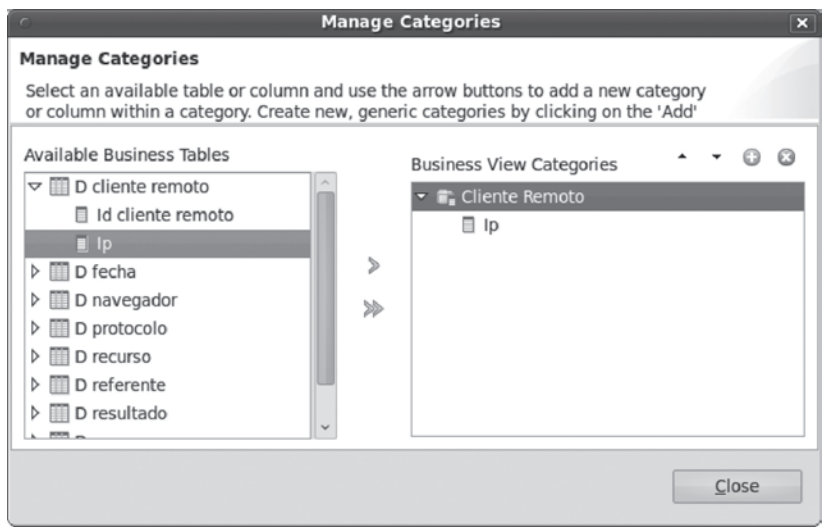
- Este proceso se debe realizar para todas las tablas que han sido exportadas del modelo, y posteriormente hay que crear las relaciones de negocio entre ellas para que el modelo esté bien definido.



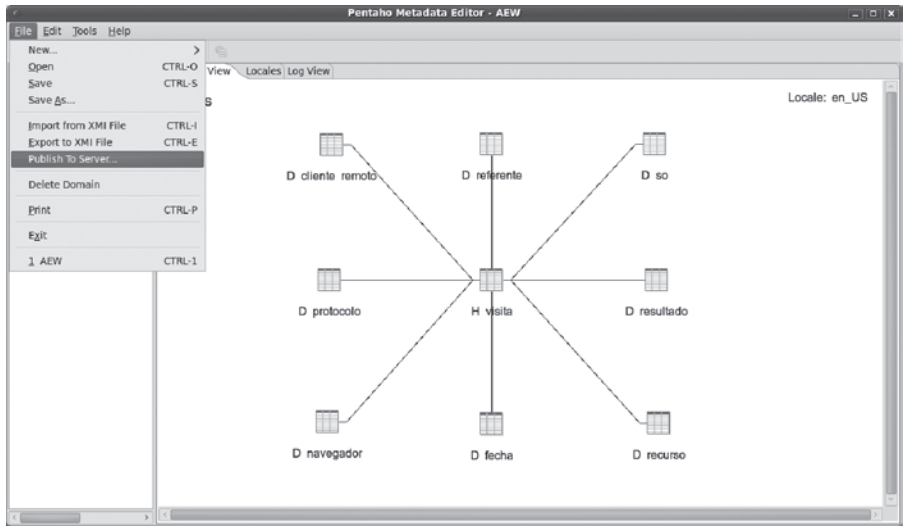
- El resultado final de crear todas las tablas en la vista de negocio y sus relaciones es el siguiente:



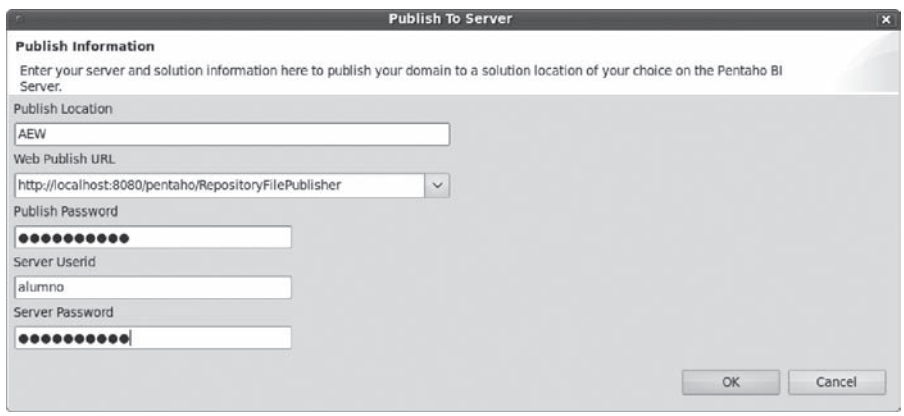
- Sin embargo, tener el modelo de negocio y sus relaciones no es suficiente. Es necesario definir las categorías disponibles en la capa de metadatos al usuario final. En el ejemplo, el usuario sólo podrá acceder y usar el atributo IP de cliente remoto.

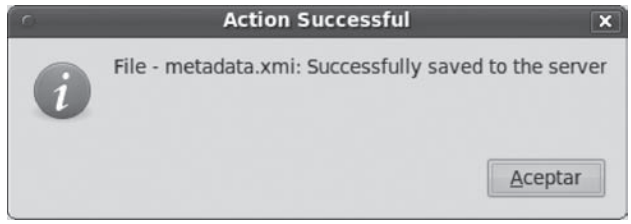


- Una vez terminado el proceso, guardamos el fichero y publicamos la capa de metadatos. Es necesario que el servidor de Pentaho esté en funcionamiento y previamente esté configurada la contraseña de publicación.



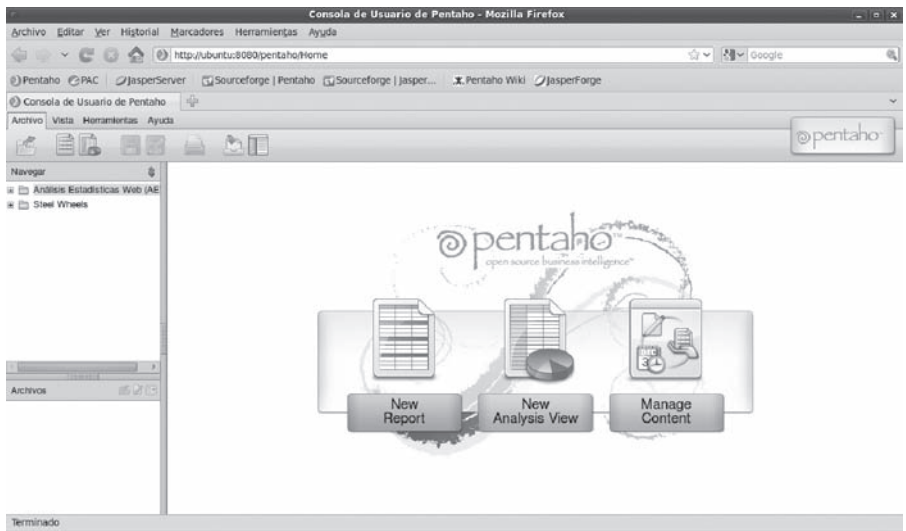
- Completamos los parámetros (en este caso las contraseñas son la misma: curso_osbi).



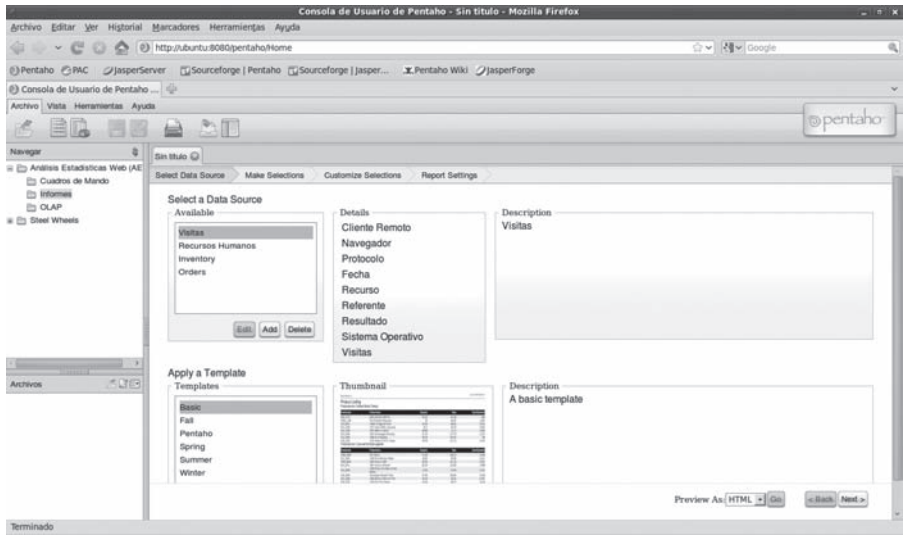


3.2. Diseño de un informe basado en la capa de metadatos en Pentaho

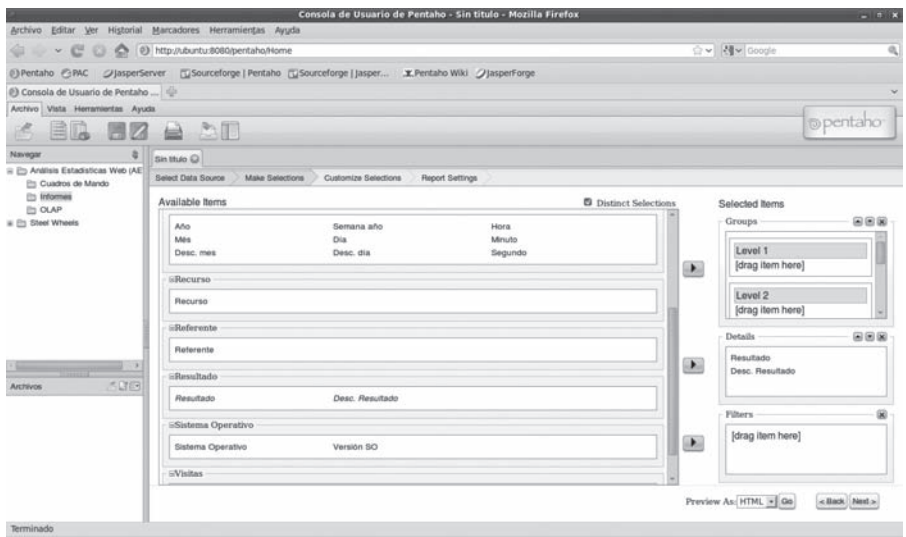
- Entramos en Pentaho y pulsamos new report.



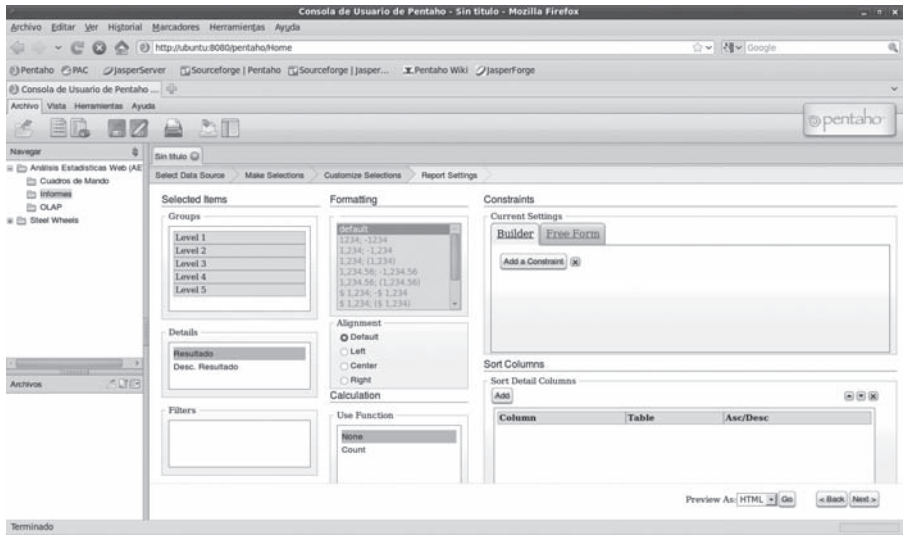
- Seleccionamos la información con la que trabajaremos, en este caso visitas y uno de los templates existentes.



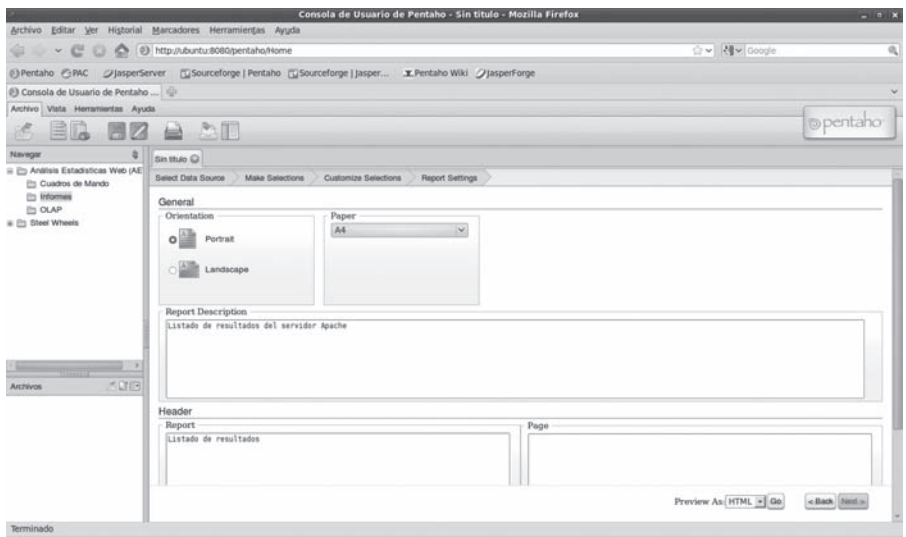
- Seleccionamos los elementos del informe respecto a las categorías creadas anteriormente. En este caso, resultado y su descripción.



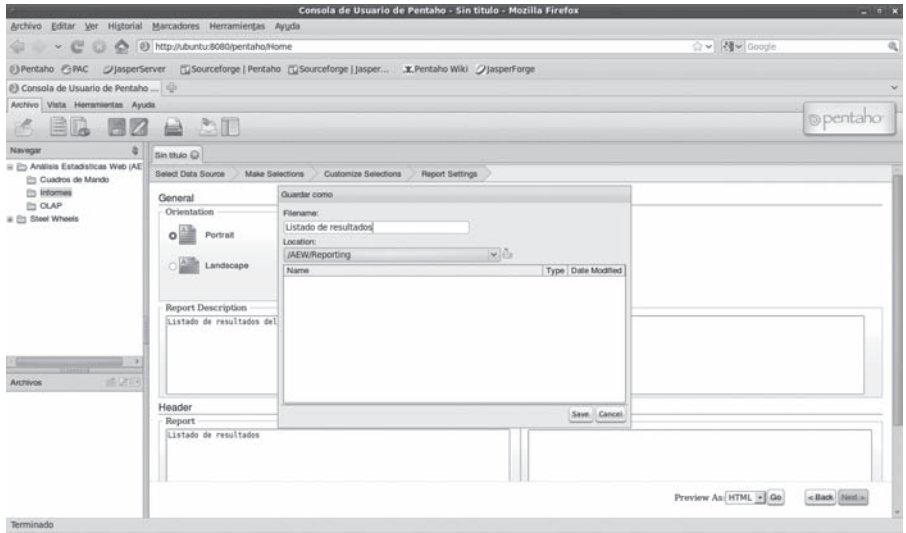
- En caso de que sea necesario, damos el formato a cada campo.



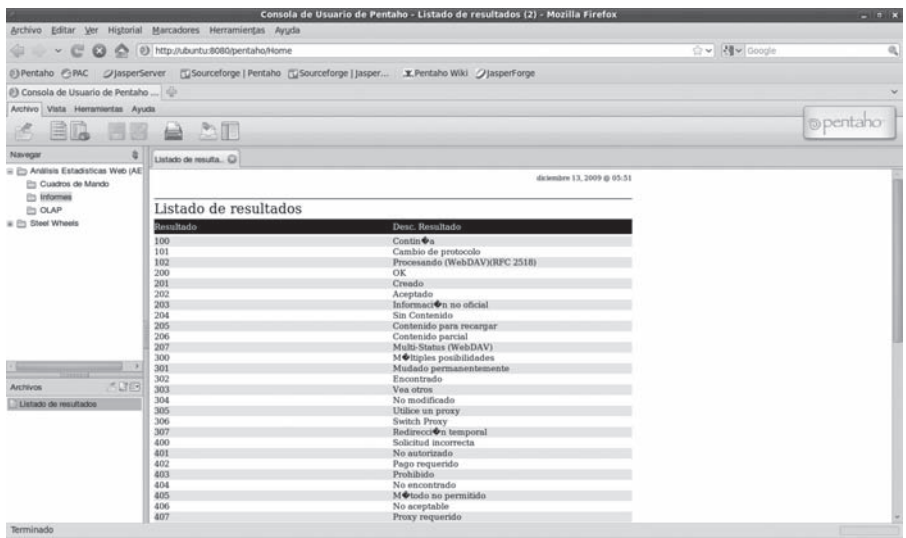
- Se configuran los parámetros del informe.



- Guardamos el informe resultante.



- Ejecutamos el informe para ver el resultado.

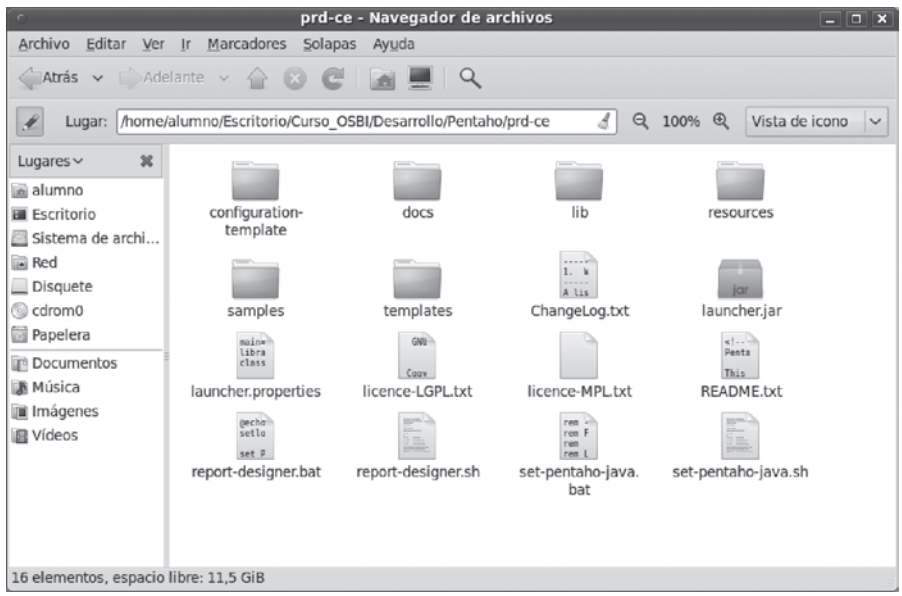


3.3. Diseño de un informe mediante el wizard en Pentaho

Pentaho Report Designer (PRD) ofrecer dos posibles formas para crear informes:

- Mediante el wizard.
- En formato libre.

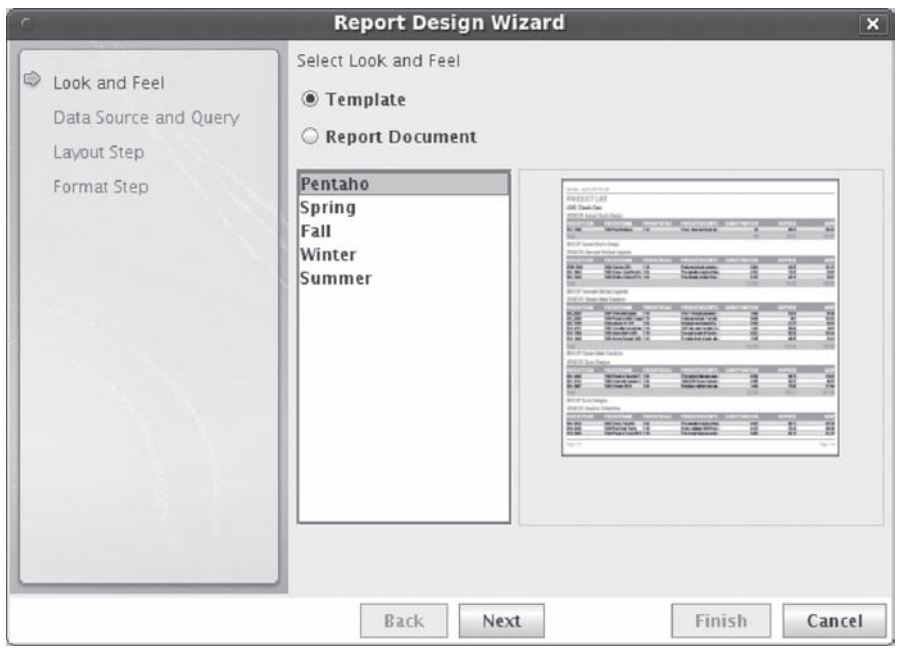
Iniciamos el PRD mediante el fichero report-designer.sh.



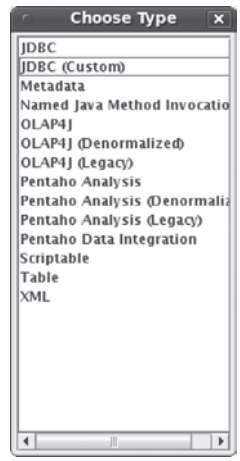
Elegimos crear un informe mediante el wizard.

La herramienta nos guía por el proceso de creación del informe:

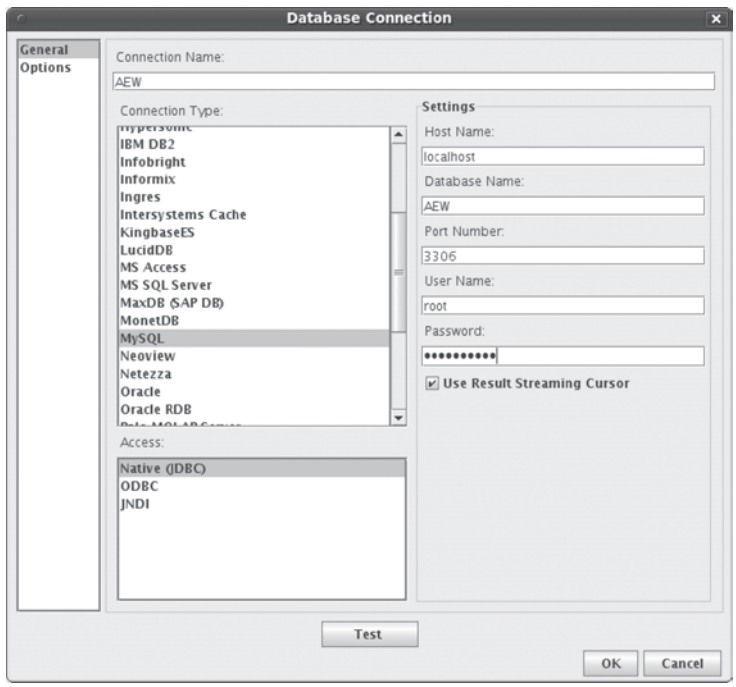
• Elegimos uno de los templates disponibles.

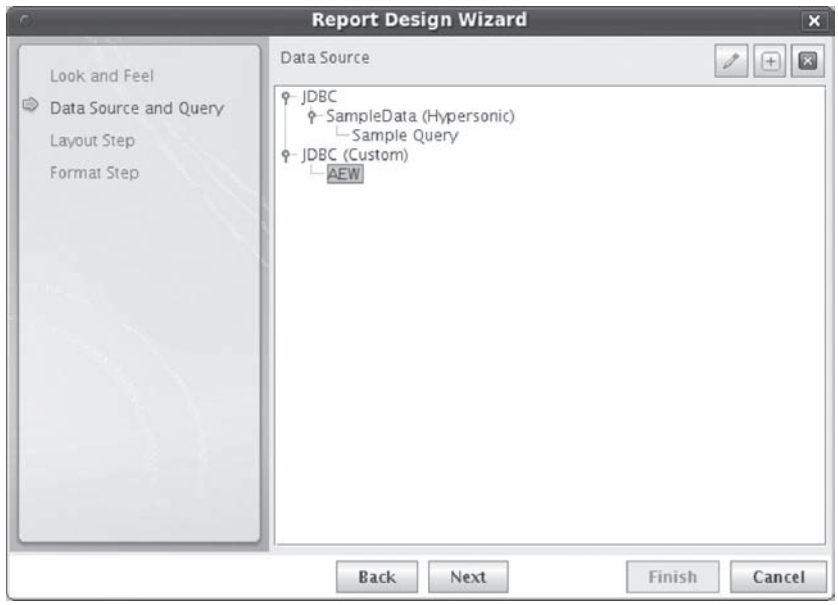


- Es necesario definir el conjunto de datos respecto al cual se realizará el informe. Están soportadas múltiples entradas de datos (desde la capa de metadatos hasta base de datos pasando por OLAP). Escogemos JDBC.

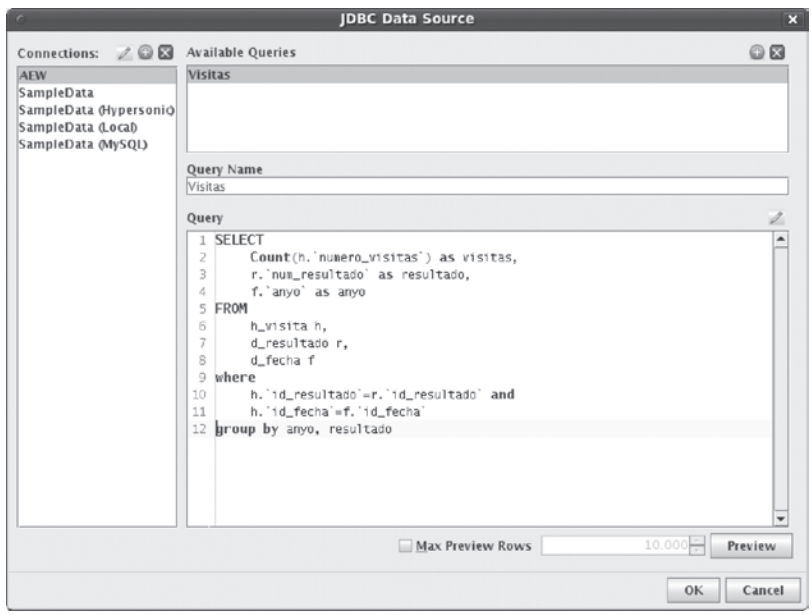


- Definimos la conexión.

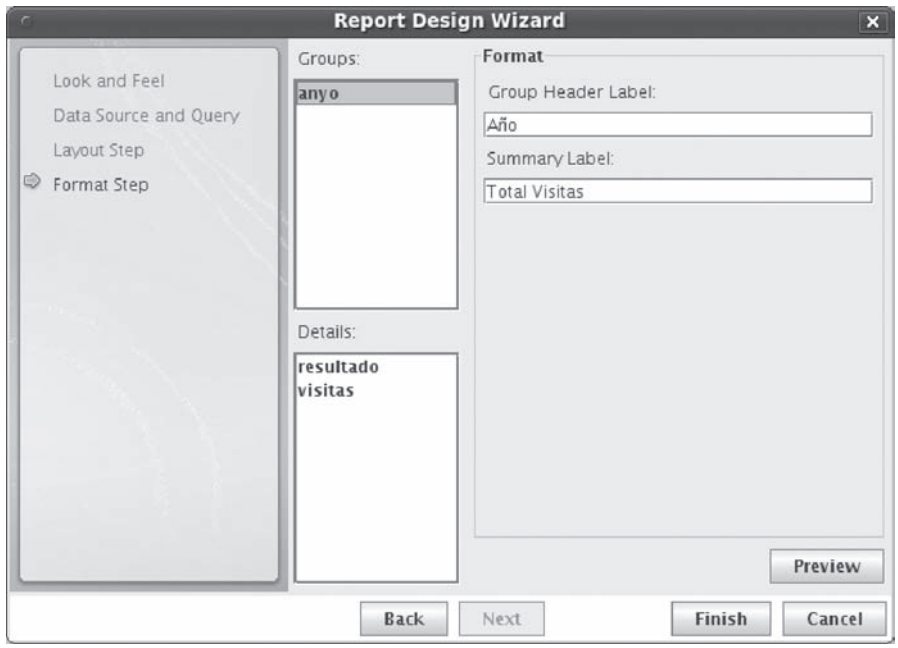


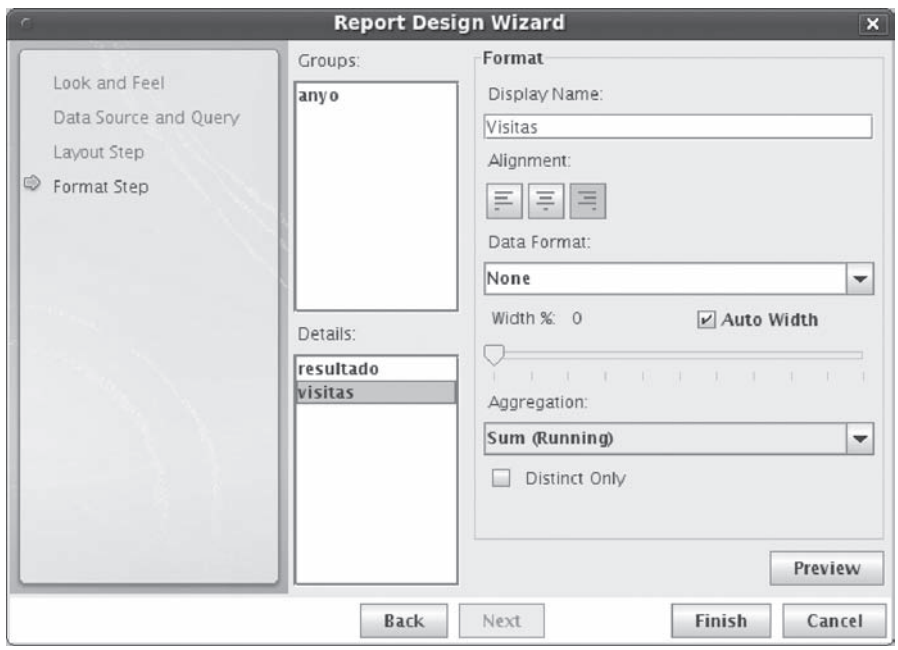


- Definimos el data set, es decir, el conjunto de datos que será usado en nuestro informe. En este caso, creamos una consulta para recuperar las visitas agrupadas por resultado y año.

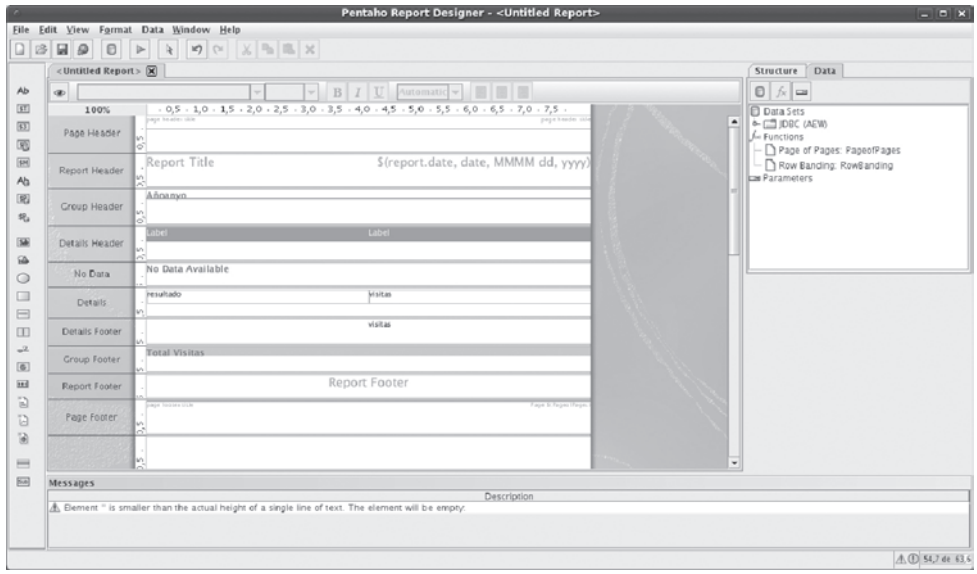


- Formateamos los datos para encajarlos en el template escogido.

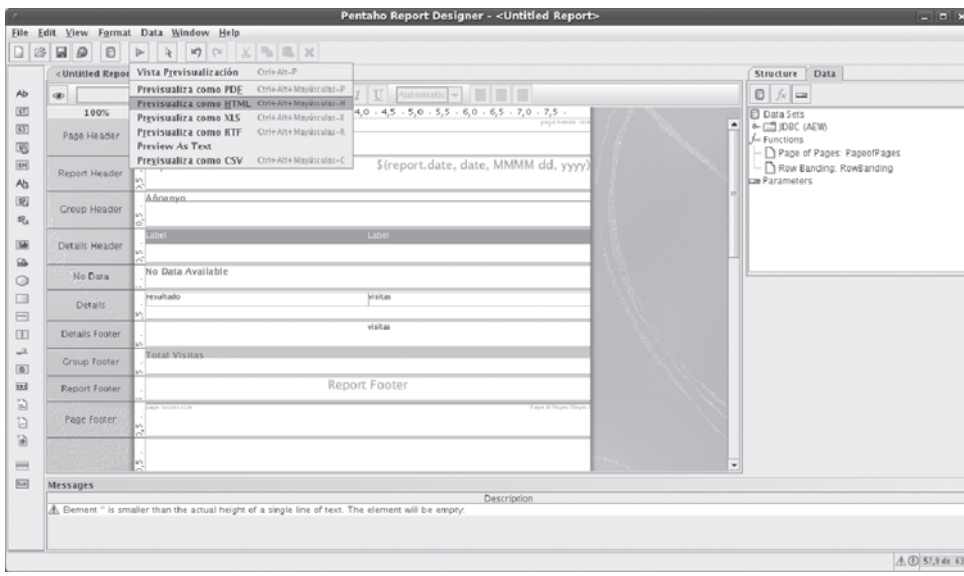




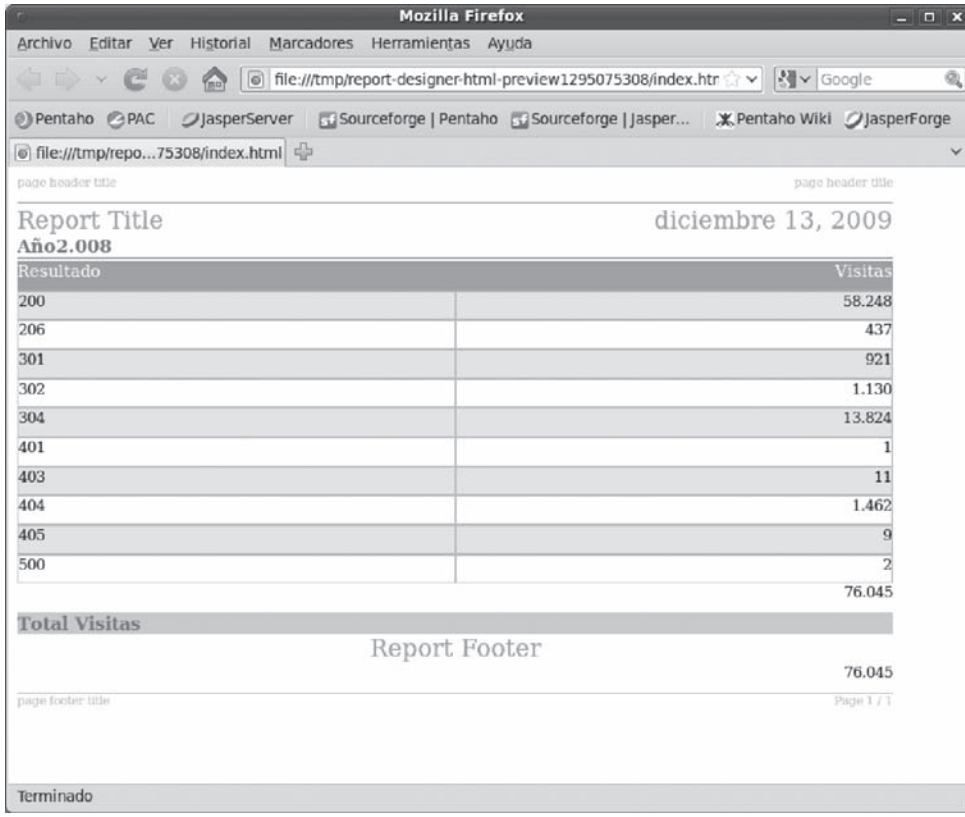
- El resultado es un informe completamente formateado al pulsar finish.



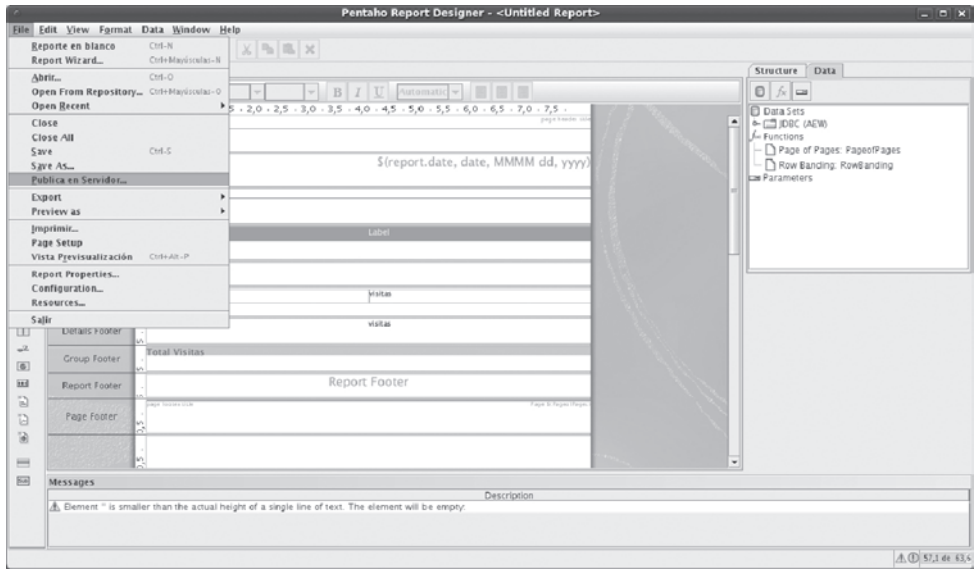
- Podemos elegir el formato de previsualización antes de su publicación.



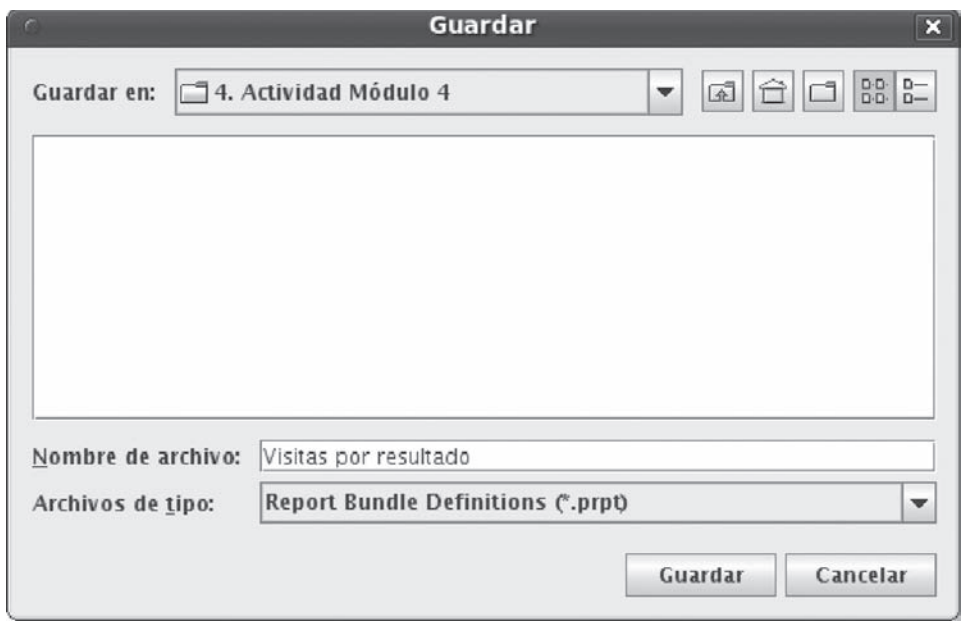
- El resultado en este caso es un informe en formato HTML.



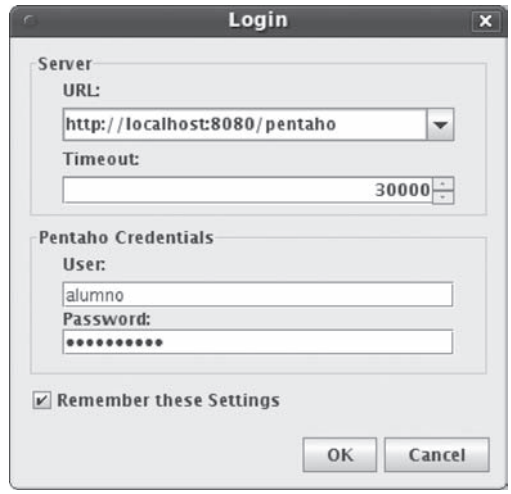
- Para publicar el informe, primero nos pedirá guardarlo.

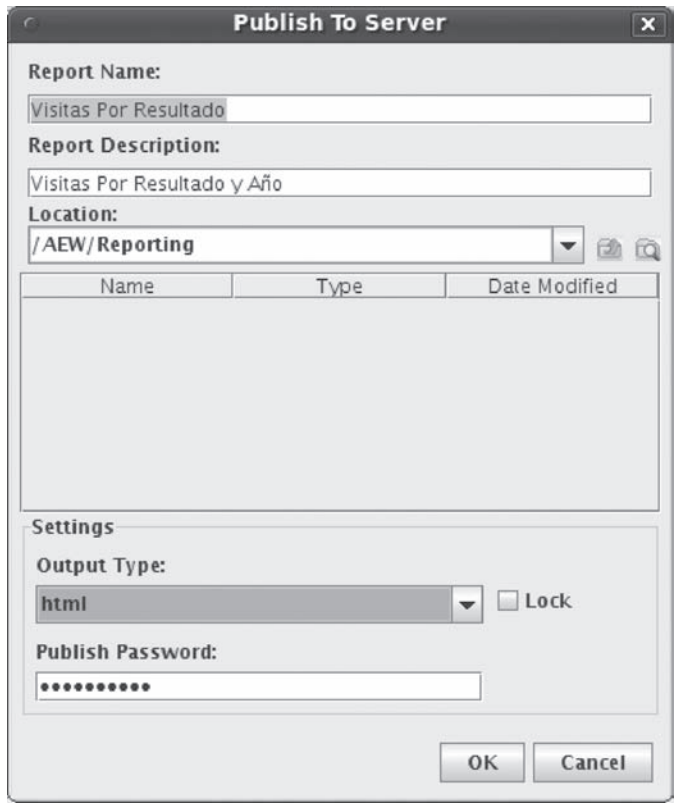


- La extensión del informe en Pentaho (desde la versión 3.5) es prpt.

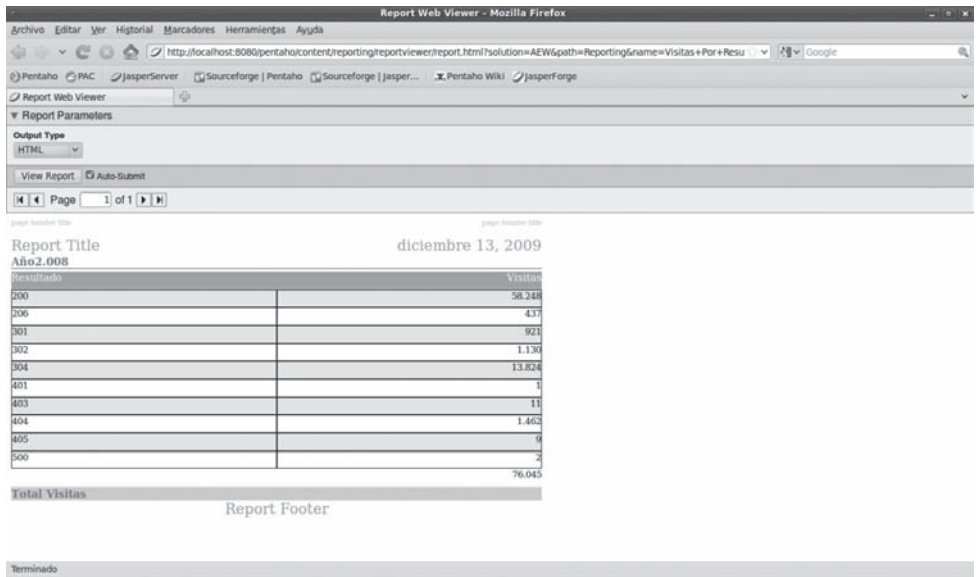


- Introducimos los parámetros de publicación.





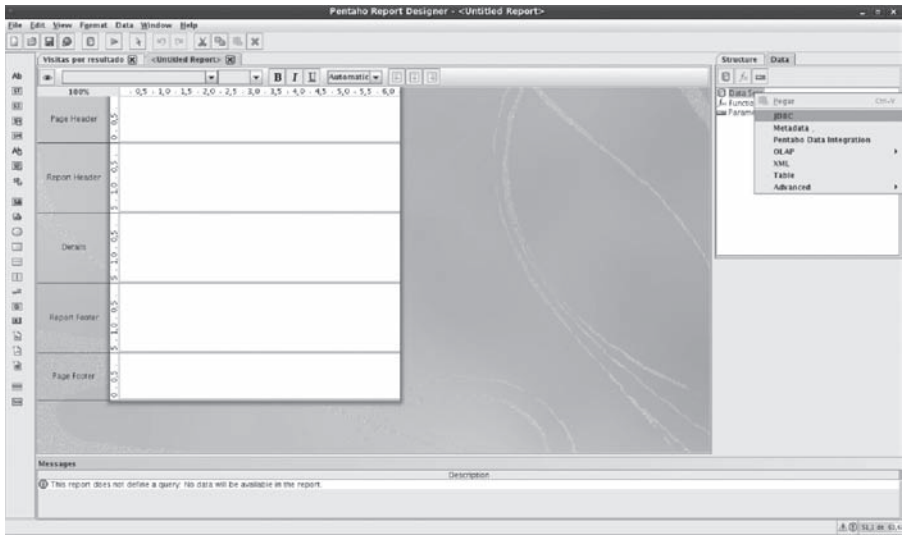
- Finalmente podemos consultar el informe vía el servidor de Pentaho. Dado que no hemos fijado la salida del informe, será uno de los parámetros por defecto.

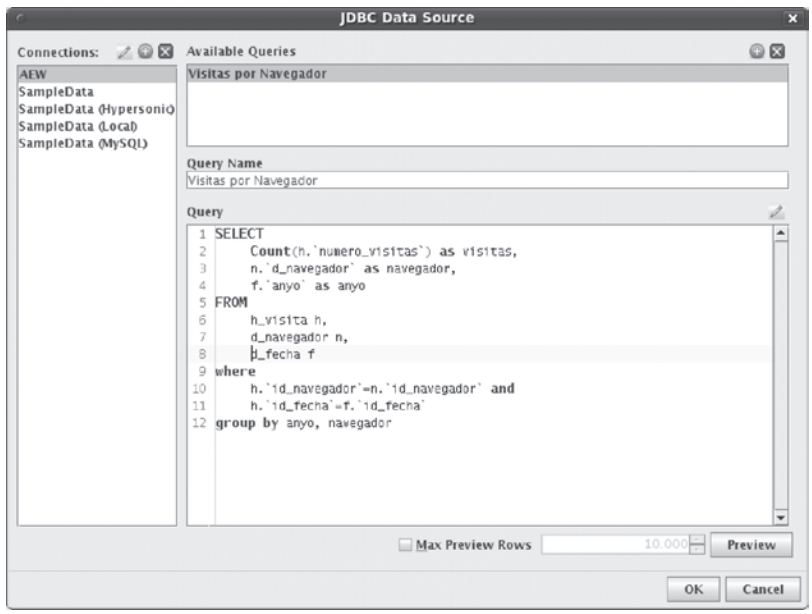


3.4. Diseño de un informe mediante Pentaho Report Designer

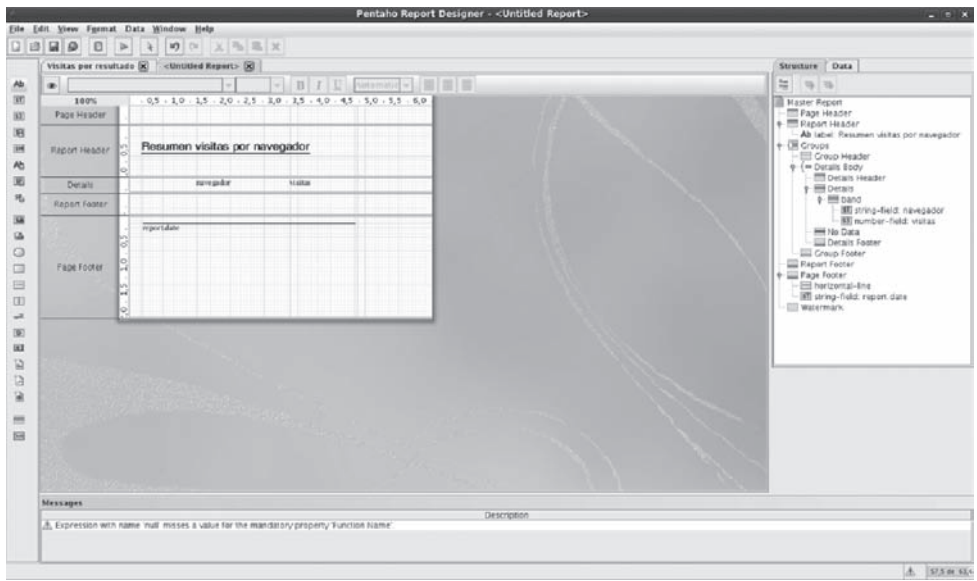
Diseñar un informe desde cero con Pentaho Report Designer (PRD) es seguir el siguiente proceso:

- Crear el data set.

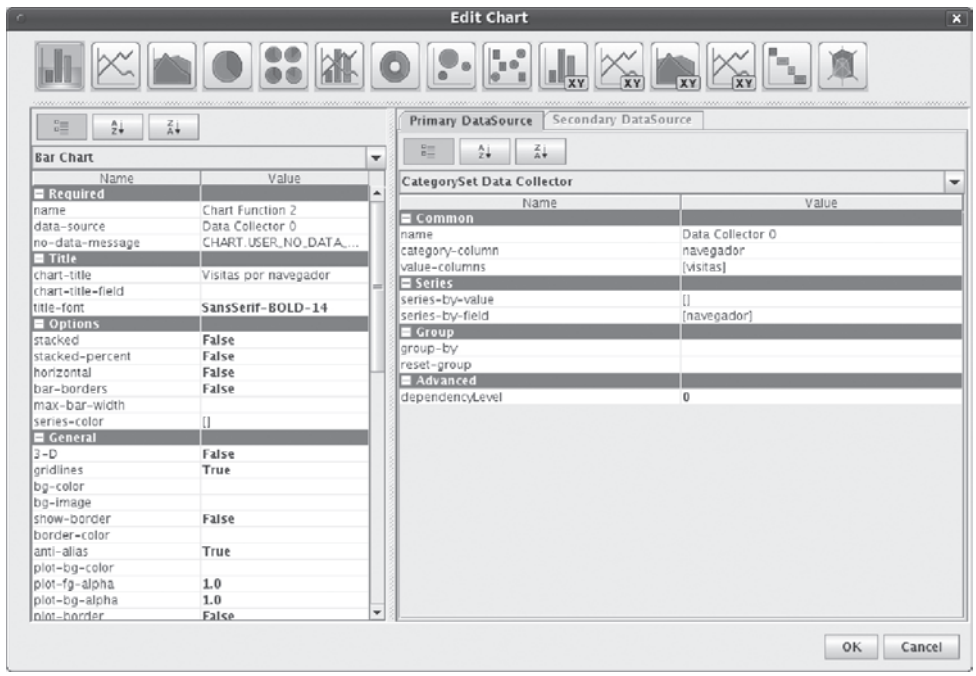




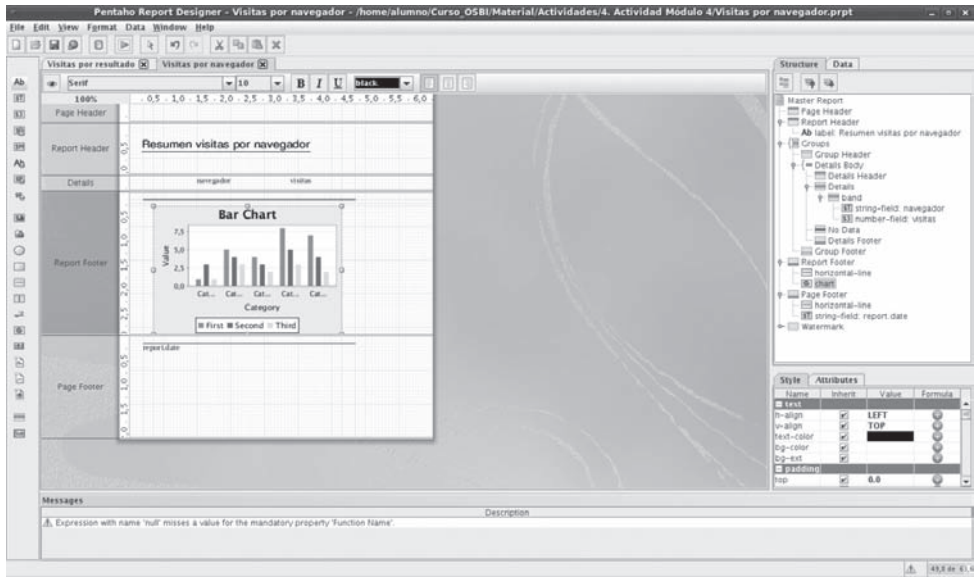
- Incluir los elementos del informe mediante drag & drop. Hemos incluido:
- Un label con el título del informe en report header.
- En detalles, una band que incluye los campos de la base de datos navegador y visitas (que nos proporcionará el número de visitas por navegador).
- En el page footer, la fecha de generación y una línea horizontal para diferenciarlo de la parte de detalles.



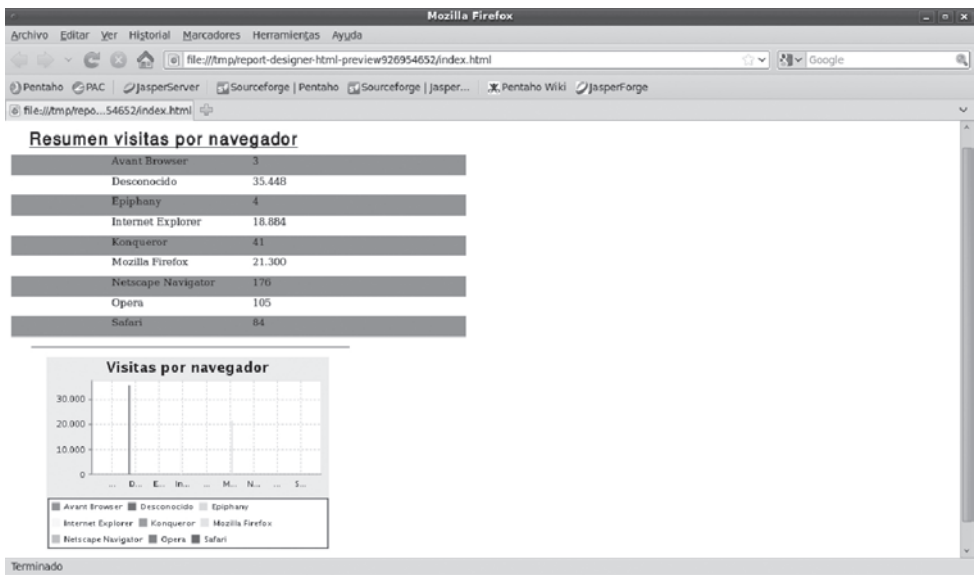
- Para enriquecer este informe respecto al anteriormente creado, incluimos un gráfico (completamos chart-title, category-column, value-columns y series-by-field como mínimo).



- De esta forma tendremos el siguiente informe:



- Finalmente, al publicarlo vemos nuestro informe con gráfico.



4. Glosario

CSV	Comma Separated Value
CWM	Common Warehouse Metamodel
EJB	Enterprise JavaBeans
EPS	Encapsulated PostScript
HTML	HyperText Markup Language
JDBC	Java Database Connection
MDX	Multidimensional eXpressions
ODBC	Open Database Connectivity
ODS	Operational Data Store
OLAP	On-Line Analytical Processing
PDF	Portable Document Format
PNG	Portable Network Graphics
POJO	Plain Old Java Object
PSV	Pipe Separated Value
RTF	Rich Format Text
SQL	Structured Query Language
SSV	Semi-colon Separated Value
SVG	Scalable Vector Graphics
TSV	Tabular Separated Value
XML	eXtensible Markup Language
WAQR	Web Ad-hoc Query Reporting

5. Bibliografía

BOUMAN, R. y VAN DONGEN, J. (2009). *Pentaho® Solutions: Business Intelligence and Data Warehousing with Pentaho® and MySQL*. Indianapolis: Wiley Publishing.

DANCIU, T., y CHIRITA, L. (2007). *The Definitive Guide to JasperReports*. Nueva York: Apress.

GORMAN, W. (2009). *Pentaho Reporting 3.5 for Java Developers*. Birmingham: Packt Publishing.

Capítulo VI

Diseño de cuadros de mando

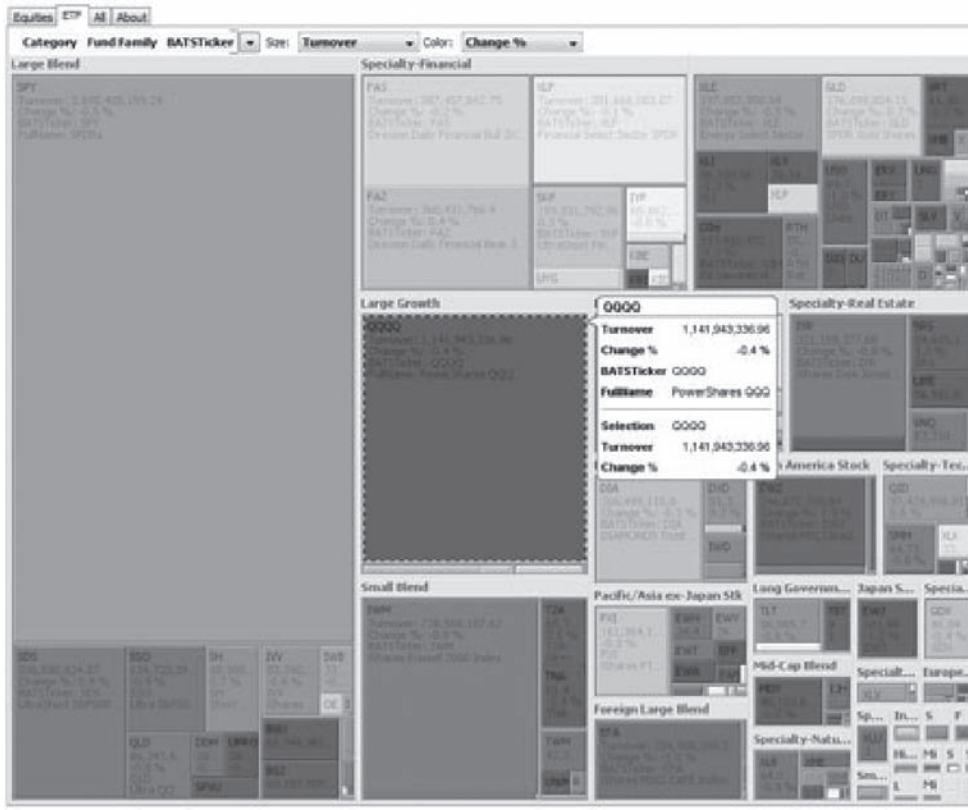
Tanto los informes como los OLAP son herramientas que proporcionan información a los usuarios finales. La gran cantidad de información que normalmente incluyen estas herramientas puede hacerlas inadecuadas para usuarios que necesiten tomar decisiones de forma rápida a partir de ellas.

El cuadro de mando proviene del concepto francés *tableau du bord*, y permite mostrar información consolidada a alto nivel. Se focaliza en:

- Presentar una cantidad reducida de aspectos de negocio.
- Uso mayoritario de elementos gráficos.
- Inclusión de elementos interactivos para potenciar el análisis en profundidad y la comprensión de la información consultada.

Los cuadros de mando son una herramienta muy popular dado que permiten entender muy rápidamente la situación de negocio y son muy atractivos visualmente. Por ello, todas las soluciones del mercado los incluyen. La oferta se diferencia principalmente en el nivel de madurez del proceso de creación del cuadro de mando, en las opciones disponibles de visualización, y en la capacidad de trabajar con flujos continuos de datos y el reflejo de dichos cambios en tiempo real.

Una de las últimas tendencias, en el contexto de cuadros de mando, es la inclusión de elementos gráficos que permiten el análisis de grandes cantidades de información. Es lo que se conoce como visual analytics. No todas las herramientas del mercado incluyen esta tendencia: sólo se encuentra en algunos productos innovadores. El siguiente gráfico es un tree map.



El objetivo de este capítulo es presentar el concepto de cuadro de mando y ejemplificarlo mediante Pentaho.

1. Cuadro de mando como herramienta de monitorización

Un cuadro de mando permite monitorizar los procesos de negocio dado que muestra información crítica a través de elementos gráficos de fácil comprensión. Este tipo de herramientas, cuya periodicidad de refresco suele ser cercana al tiempo real, son de gran utilidad para todos aquellos usuarios encargados de tomar decisiones diariamente.

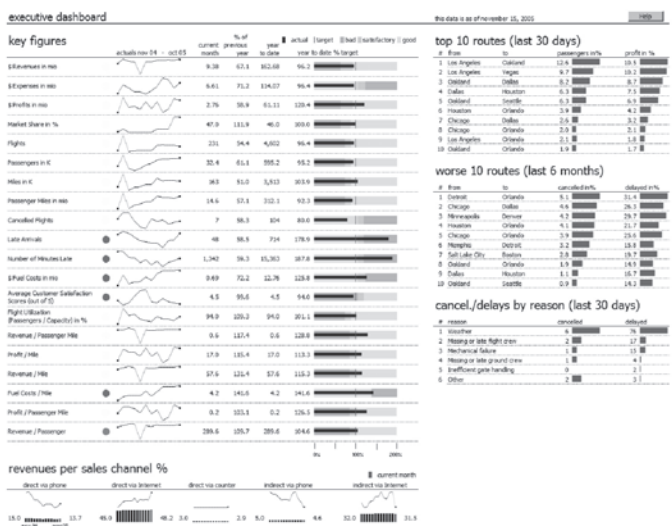
Estos sistemas pueden encontrarse integrados en suites de Business Intelligence o ser simplemente aplicaciones independientes.

Es necesario, antes de continuar, introducir una definición formal de cuadro de mando:

> Se entiende por **cuadro de mando** o dashboard al sistema que informa de la evolución de los parámetros fundamentales de negocio de una organización o de un área del mismo.

La información que se presenta en un cuadro de mando se caracteriza por:

- Usar diferentes elementos (gráficos, tablas, alertas...).
- Combinar los elementos de forma uniforme.
- Basar la información presentada en indicadores clave de negocio.
- Presentar las tendencias de negocio para propiciar la toma de decisiones.



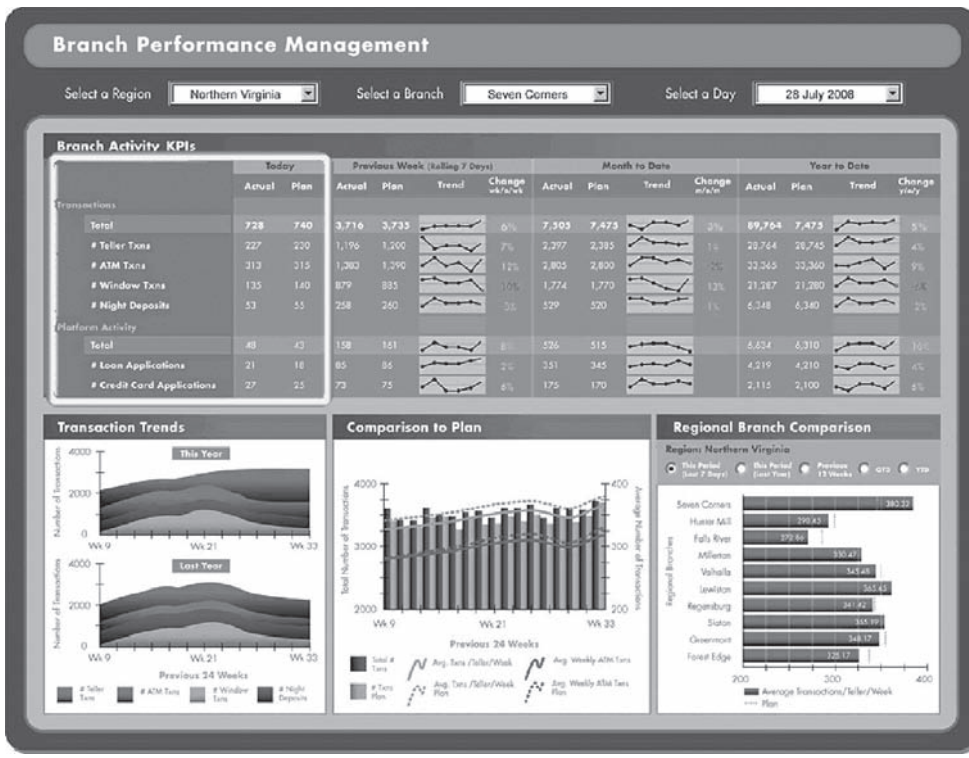
La tipología de usuarios que necesitan de estas herramientas son:

- Alta dirección.
- Gerentes que deben monitorizar procesos de negocio.

1.1. Elementos de un cuadro de mando

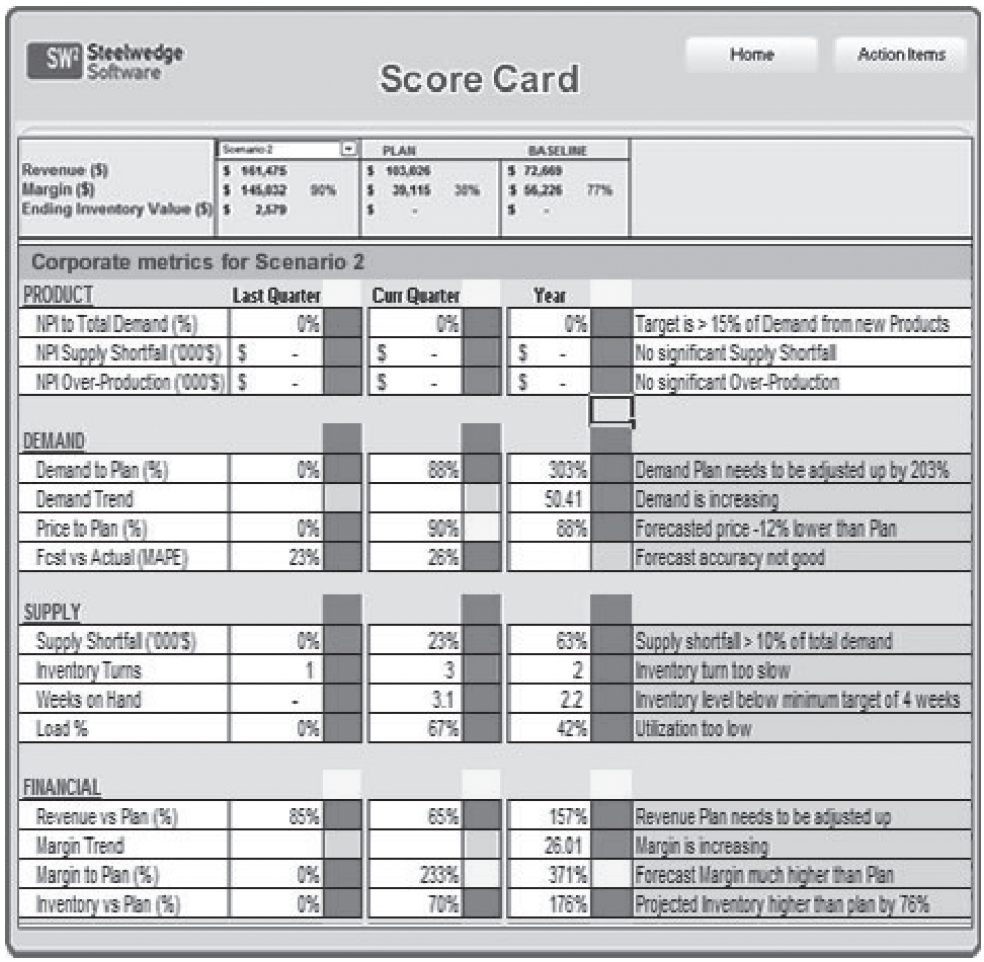
Un cuadro de mando está formado principalmente por diversos elementos combinados:

- Tabla: tiene forma de matriz y permite presentar una gran cantidad de información. La tabla puede ser estática, dinámica, o incluso un análisis OLAP. Se persigue con este elemento presentar información de forma estructurada al usuario final.



- Métricas: valores que recogen el proceso de una actividad o los resultados de la misma. Estas medidas proceden del resultado de la actividad de negocio. Como ya sabemos, existen diferentes tipos de métricas. En un cuadro de mando, se suelen usar KPI.

- Listas: comúnmente formadas por KPI. En caso de que el cuadro de mando sólo esté formado por este tipo de elemento, se denomina scorecard.



Steelwedge Software

Score Card

Home | Action Items

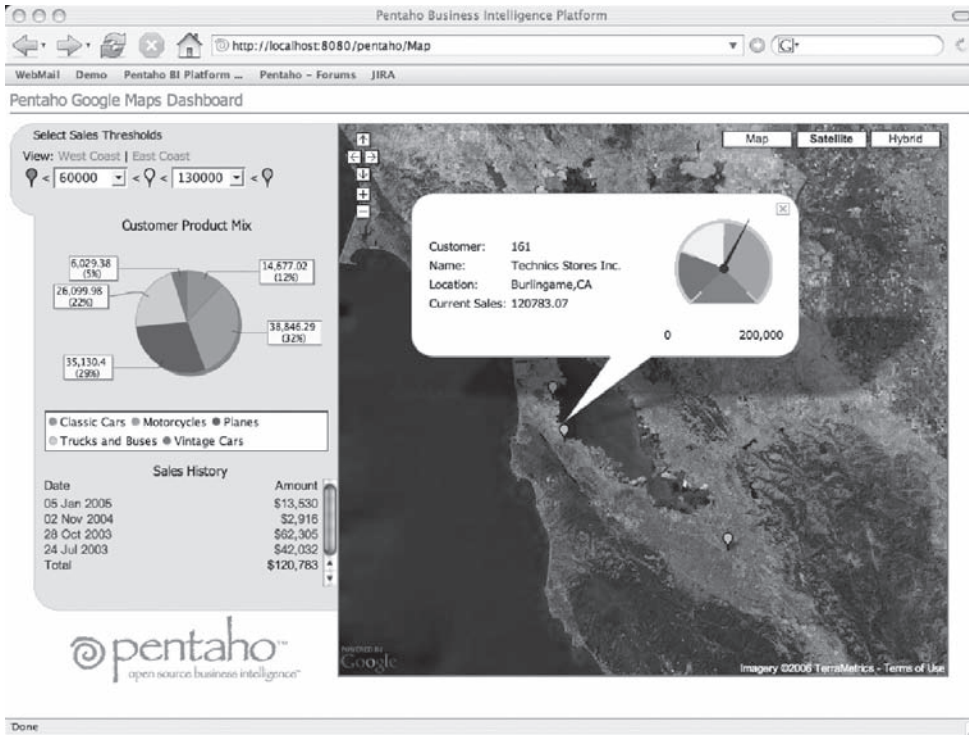
	Scenario 2		PLAN		BASELINE	
Revenue ($)	$ 161,475		$ 163,826		$ 72,669	
Margin ($)	$ 145,832	90%	$ 39,115	38%	$ 56,226	77%
Ending Inventory Value ($)	$ 2,579		$ -		$ -	

Corporate metrics for Scenario 2

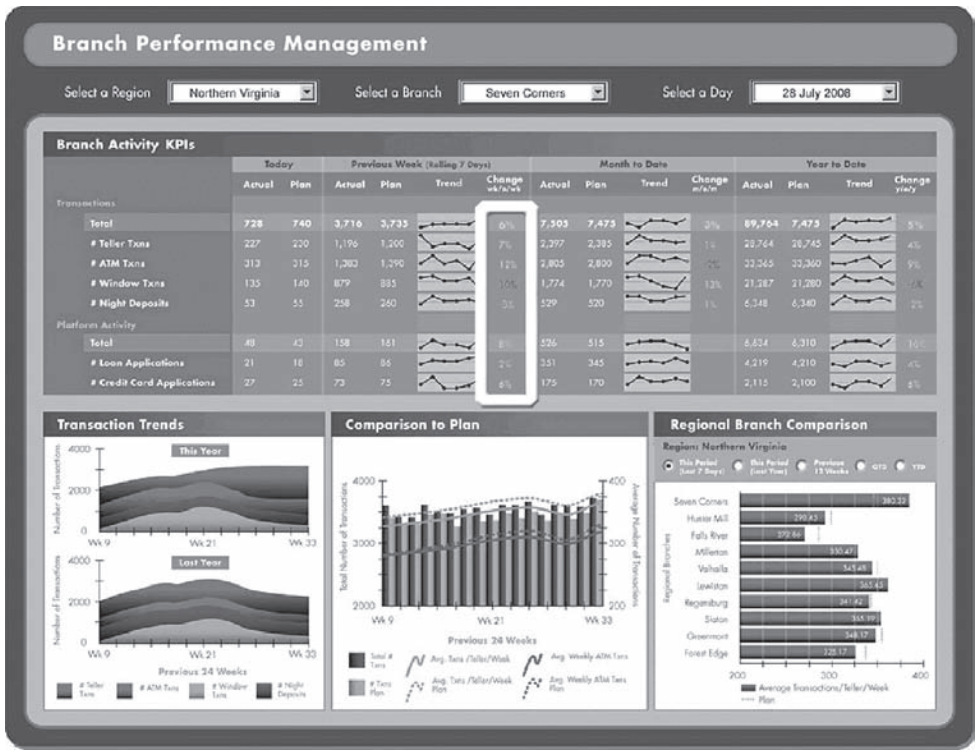
PRODUCT	Last Quarter		Curr Quarter		Year		
NPI to Total Demand (%)	0%		0%		0%		Target is > 15% of Demand from new Products
NPI Supply Shortfall ('000'$)	$ -		$ -		$ -		No significant Supply Shortfall
NPI Over-Production ('000'$)	$ -		$ -		$ -		No significant Over-Production
DEMAND							
Demand to Plan (%)	0%		88%		303%		Demand Plan needs to be adjusted up by 203%
Demand Trend					50.41		Demand is increasing
Price to Plan (%)	0%		90%		88%		Forecasted price -12% lower than Plan
Fcst vs Actual (MAPE)	23%		26%				Forecast accuracy not good
SUPPLY							
Supply Shortfall ('000'$)	0%		23%		63%		Supply shortfall > 10% of total demand
Inventory Turns	1		3		2		Inventory turn too slow
Weeks on Hand	-		3.1		2.2		Inventory level below minimum target of 4 weeks
Load %	0%		67%		42%		Utilization too low
FINANCIAL							
Revenue vs Plan (%)	85%		65%		157%		Revenue Plan needs to be adjusted up
Margin Trend					26.01		Margin is increasing
Margin to Plan (%)	0%		233%		371%		Forecast Margin much higher than Plan
Inventory vs Plan (%)	0%		70%		176%		Projected Inventory higher than plan by 76%

- Gráficos: este elemento persigue el objetivo de mostrar información con un alto impacto visual que sirva para obtener información agregada o sumarizada con mucha más rapidez que a través de tablas. El gráfico puede estar formado por la superposición de diferentes tipos de visualización.

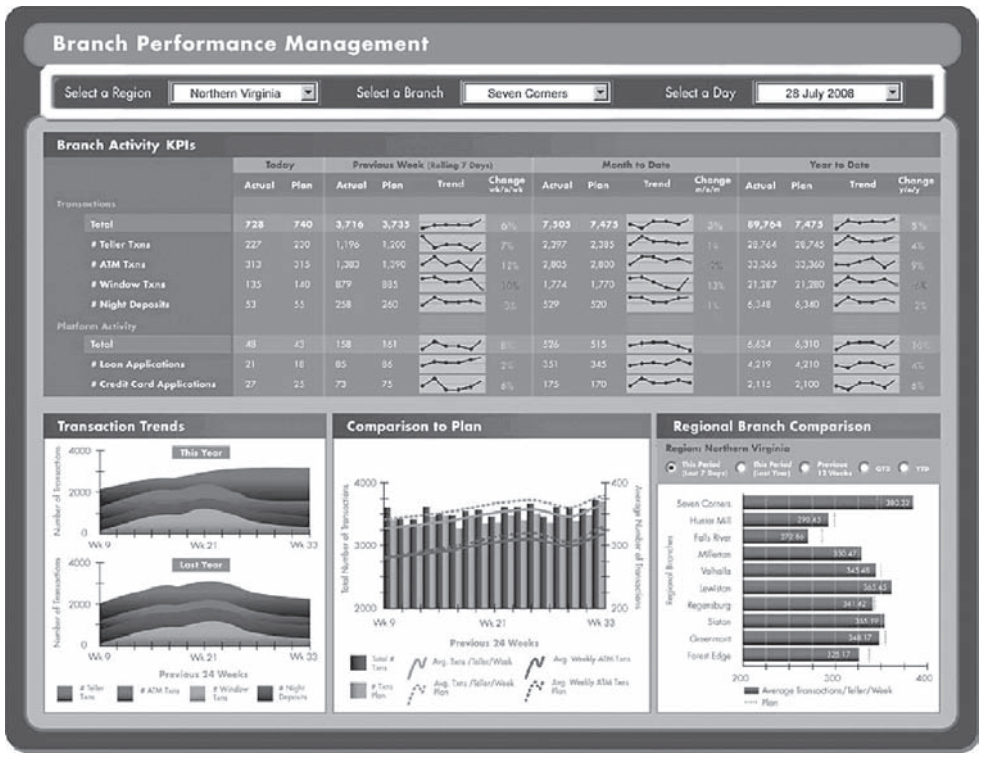
- Mapas: este elemento permite mostrar información geolocalizada. No toda la información es susceptible de estar en este tipo de formato. Se combina con otros elementos para presentar el detalle de la información.



- Alertas visuales y automáticas: consisten en alertas que informan del cambio de estado de información. Pueden estar formadas por elementos gráficos como fechas o colores resultados, y deben estar automatizadas en función de reglas de negocio encapsuladas en el cuadro de mando.



- Menús de navegación: que facilitan al usuario final realizar operaciones con los elementos del cuadro de mando.



El cuadro de mando, por lo tanto, comparte la mayoría de los elementos de los informes.

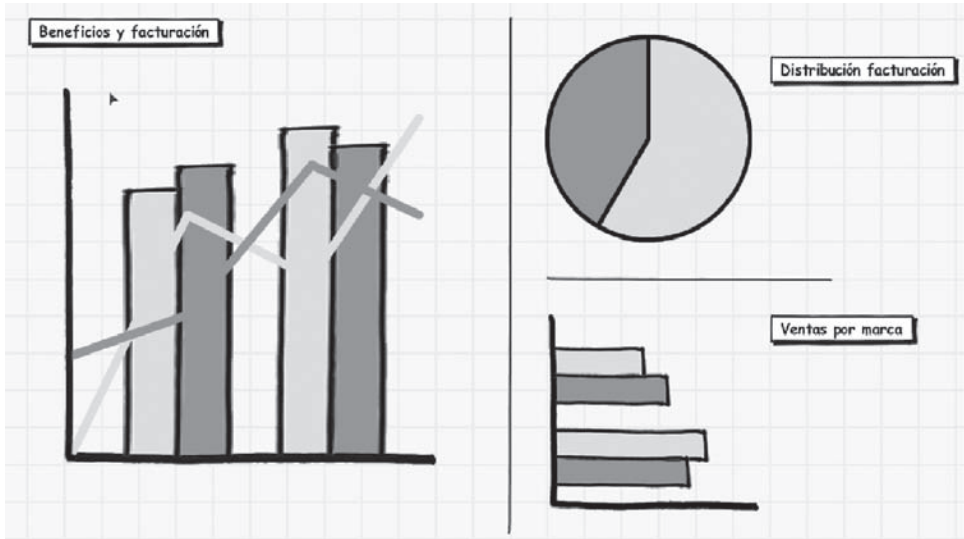
1.2. Proceso de creación de un cuadro de mando

El proceso de crear un cuadro de mando es un proceso iterativo que combina diversos pasos:

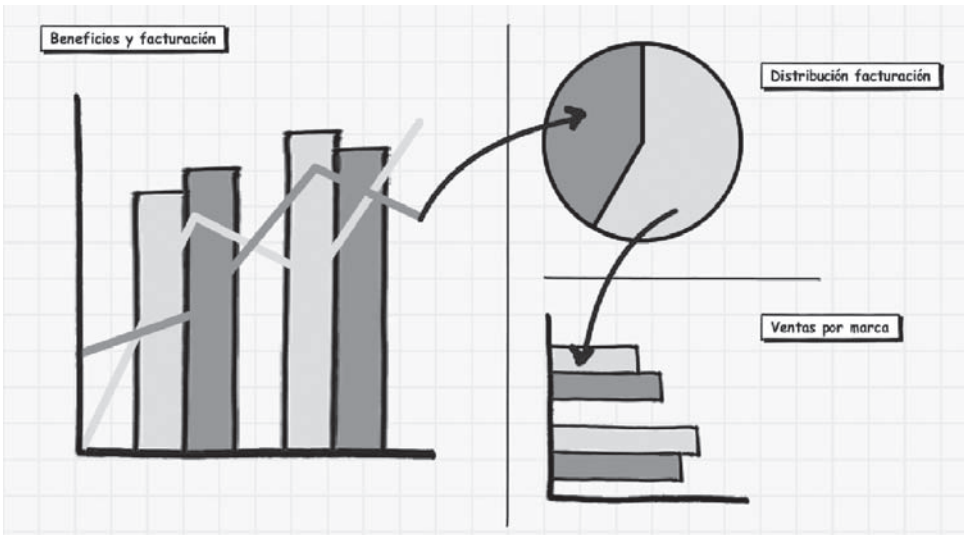
1) Elegir los datos a mostrar. El punto de partida es elegir qué datos se van a mostrar en el cuadro de mando. En este punto es necesario tener en cuenta las necesidades del usuario final.

2) Elegir el formato de presentación. A partir la información a mostrar y las necesidades del cliente, es posible determinar qué tipo de elemento de un cuadro de mando es el más adecuado. Se recomienda realizar un boceto.

3) Combinar datos y presentarlos conjuntamente. Una vez tenemos los diferentes elementos, se realiza un boceto con todos ellos.



4) Planificar la interactividad del usuario.



5) Implementar el cuadro de mandos. En este punto entra la herramienta seleccionada. Incluye los siguientes pasos:

- Conseguir los datos y formatearlos para conseguir los KPI.
- Formatear los elementos del cuadro de mando en función de las capacidades de la solución escogida.

1.3. Dashboard vs. Balanced ScoreCard

Frecuentemente se confunde el cuadro de mando o dashboard con el cuadro de mando integral o balanced scorecard. La razón es la similitud de los nombres.

Se entiende por **balanced scorecard** al método de planificación estratégica basado en métricas y procesos ideado por los profesores Kaplan y Norton, que relaciona factores medibles de procesos con la consecución de objetivos estratégicos.

La teoría del balanced scorecard surgió en los años noventa como respuesta ante la necesidad de analizar las organizaciones desde un punto de vista diferente al financiero, que se estaba quedando obsoleto. El objetivo era establecer un nuevo modelo de medidas que permitiera conocer mejor las organizaciones.

Para ello, el instituto Nolan Norton patrocinó un estudio de un año en el que participaron varias compañías de múltiples sectores y cuyo objetivo era definir un scorecard corporativo. De dicho estudio surgió el concepto de balanced scorecard, que organizaba indicadores clave de negocio en cuatro grandes grupos o perspectivas: financiera, cliente, interna e innovación y aprendizaje.

Balanced refleja que los indicadores tratan de ser un equilibrio entre los objetivos a corto y largo plazo, entre las medidas financieras y las no financieras, entre los indicadores de retraso o liderazgo, y entre las perspectivas internas y externas.

Por ello, el balanced scorecard permite traducir la estrategia de la empresa en un conjunto comprensible de medidas de rendimiento que proporcionen el marco de medida estratégica y de sistema de gestión.

Un cuadro de mando integral está formado por los siguientes elementos:

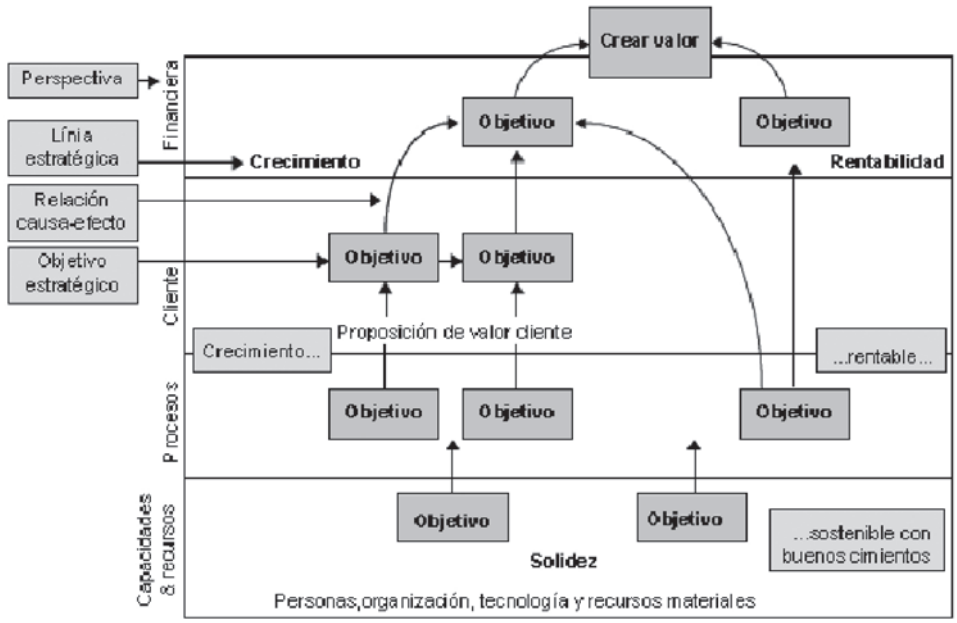
- **Perspectiva:** punto de vista respecto al cual se monitoriza el negocio. Según esta metodología, toda empresa tiene cuatro perspectivas: financiera, de cliente, de procesos y de aprendizaje y crecimiento, si bien puede extenderse o reducirse en número de perspectivas. Vamos a detallar las perspectivas clásicas:
  - Financiera: permite medir las consecuencias económicas de las acciones tomadas en la organización. Incorpora la visión de los accionistas y mide la creación de valor de la empresa.
  - Cliente: refleja el posicionamiento de la empresa en el mercado o en los segmentos de mercado donde quiere competir.
  - Interna: pretende explicar las variables internas consideradas como críticas, así como definir la cadena de valor generado por los procesos internos de la empresa.
  - Aprendizaje y crecimiento: identifica la infraestructura que la organización debe construir para crear crecimiento y valor a largo plazo.
- **Objetivos:** a cumplir en cada una de las perspectivas.
- **Líneas estratégicas:** engloban los objetivos que siguen una relación de causalidad.
- **Indicadores:** son principalmente KPI.
- **Relaciones causa-efecto:** permiten comprender cómo la consecución de un objetivo participa en otro.
- **Planes de acción:** acciones que se realizan para la consecución de un objetivo.
- **Pesos relativos:** importancia de un objetivo dentro de una perspectiva o de una línea estratégica.
- **Matriz de impacto:** permite dirimir cómo un plan de acción afecta a los objetivos y en la medida en que lo hace.

El proceso de construcción de un cuadro de mando integral es:

- Definir las perspectivas de negocio. Frecuentemente, las perspectivas clásicas son suficientes para representar la estrategia.
- Definir para cada perspectiva los objetivos estratégicos.
- Definir para cada objetivo, planes de acción para conseguir dichos objetivos.
- Definir indicadores para monitorizar la consecución de los objetivos.
- Definir las relaciones de causalidad entre los objetivos.

- Identificar las líneas estratégicas a las que pertenecen los objetivos estratégicos.

Este proceso se estructura a través de un mapa estratégico que podemos ver en la siguiente figura:



Un punto importante a destacar es que un balanced scorecard debe ser flexible y ágil, por lo que la recopilación de información debe llevarse a cabo de forma rápida, sencilla y en un tiempo oportuno para que las acciones que se deriven puedan tomarse de forma eficaz.

La implantación de un cuadro de mando integral proporciona los siguientes beneficios:

- Define y clarifica la estrategia.
- Suministra una imagen del futuro mostrando el camino que conduce a él.
- Comunica la estrategia a toda la organización.
- Permite alinear los objetivos personales con los departamentales.
- Facilita la vinculación entre el corto y el largo plazo.
- Permite formular con claridad y sencillez las variables más importantes objeto de control.

- Constituye un instrumento de gestión.
- Facilita el consenso en toda la empresa gracias a su capacidad de explicitar un modelo de negocio y traducirlo en indicadores.
- Se puede utilizar para comunicar los planes de la empresa, aunar los esfuerzos en una sola dirección y evitar la dispersión. En este caso, el CMI actúa como un sistema de control por excepción.
- Permite detectar de forma automática desviaciones en el plan estratégico u operativo, e incluso indagar en los datos operativos de la compañía hasta descubrir la causa original que dio lugar a esas desviaciones.

Por lo tanto, las diferencias entre un cuadro de mando y un cuadro de mando integral son:

	Cuadro de mando	**Cuadro de mando integral**
Objetivo	Monitorizar un área de negocio y tomar decisiones operativas y/o tácticas.	Definir la estrategia de una organización y enlazarla con la operativa a través de planes de acción.
Elementos	Tablas, gráficos, listas, alertas, menús, mapas...	Perspectivas, objetivos, indicadores, metas...

2. Cuadro de mando en el contexto de Pentaho

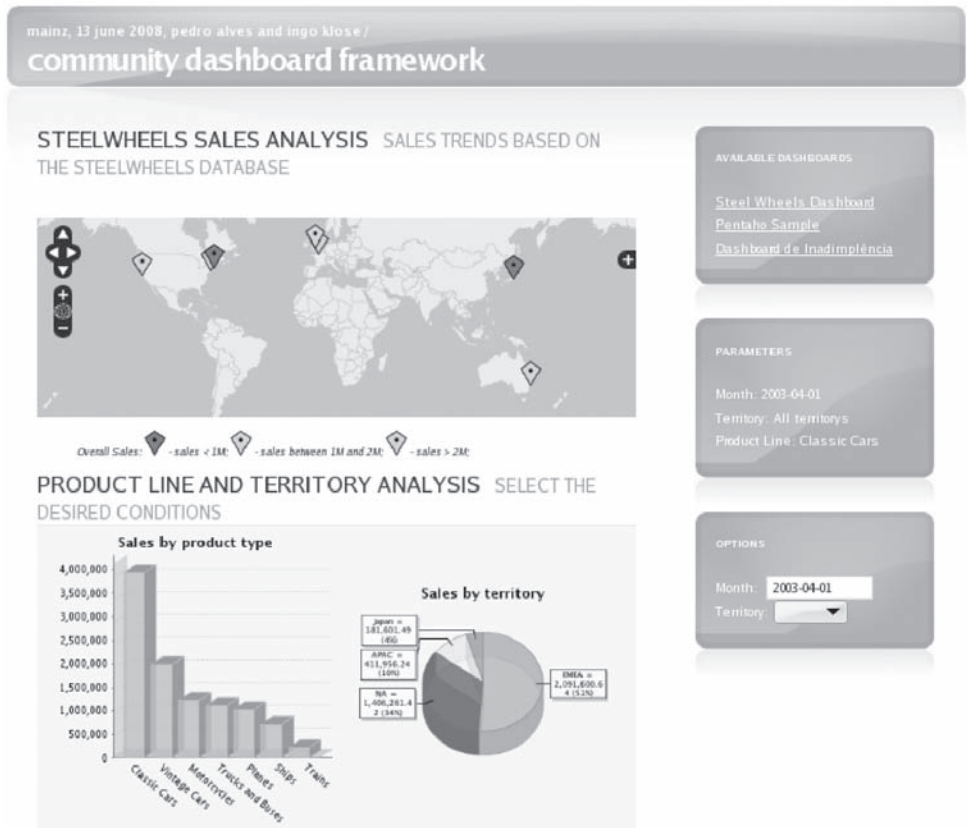
Pentaho soporta cuadros basados en JSP, HTML y JavaScript, pero actualmente no dispone de una herramienta específica para la creación de cuadros de mando. Esto significa que el proceso de desarrollo de un cuadro de mandos se convierte en un proceso artesanal. Es, por lo tanto, necesario que un desarrollador web participe en la creación de un cuadro de mando.

Para paliar dicho inconveniente, y a razón de la importancia de este tipo de herramientas, la comunidad ha creado Community Dashboard Framework (CDF).

2.1. Community Dashboard Framework

CDF es un plugin creado por la comunidad para agilizar la creación de cuadros de mando. Si bien el proceso aún requiere de un desarrollador web, se ha facilitado la integración de cuadros de mando en Pentaho.

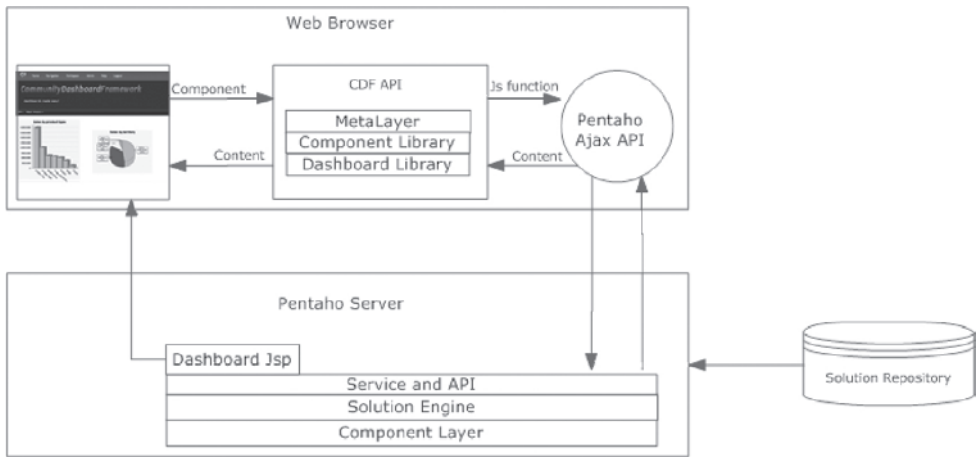
Permite construir cuadros basados, como el de la figura, en CSS y plantillas, y que incluyen elementos de Pentaho.



Este plugin está incluido por defecto en Pentaho Server. El plugin ofrece una arquitectura que permite minimizar la dificultad de uso de las librerías AJAX y el API de Pentaho para la confección de cuadros de mando.

La arquitectura del plugin permite comunicar diferentes capas y API de Pentaho de forma trasparente para el usuario que sólo debe preocuparse por

crear el esqueleto del cuadro de mando y sus elementos, como podemos ver a continuación.



Existe una herramienta[1] web de diseño de cuadros de mando basada en CDF llamada CDF-DE, pero aún está en una fase temprana y requiere conocer la estructura de un cuadro de mando creado con CDF para poder usarse correctamente, por lo que actualmente el diseño de un cuadro de mando se realiza mediante Pentaho Design Studio, la herramienta que permite crear acciones de Pentaho.

También es necesario comentar que existe otro proyecto, llamado CDA[2] (Community Data Access), que consiste en crear una capa de acceso basada en llamadas URL de diferentes fuentes de datos: SQL, MDX, Metadatos, Kettle..., e incluso composiciones proporcionando diferentes formatos de salida: JSON, XML, CSV, XLS, HTML.

El beneficio de CDA es crear una capa independiente para ser usada por CDF y CDF-DE, o, en el futuro, por otros proyectos.

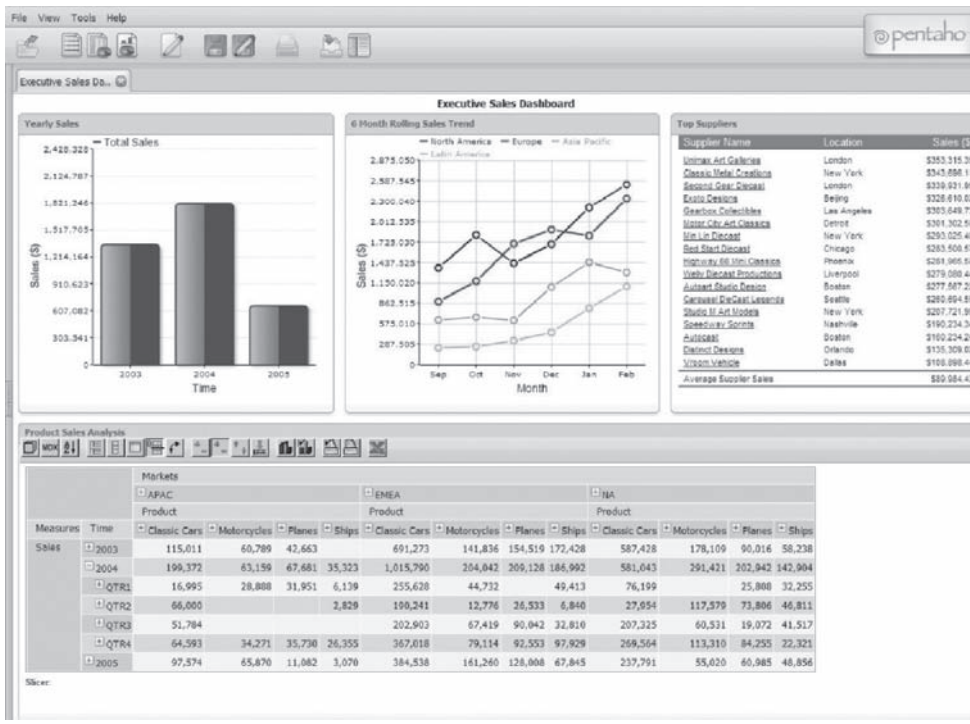
1. Esta herramienta está disponible en Google code: http://code.google.com/p/cdf-de/.
2. Esta herramienta está disponible en Google code: http://code.google.com/p/pentaho-cda/.

2.2. Pentaho Dashboard Designer

A partir de CDF, Pentaho ha habilitado un plugin para la versión profesional que permite crear un cuadro de mando de forma sencilla.

Permite:

- Crear y guardar un cuadro de mando.
- Usar elementos preexistentes (informes, gráficos, OLAP) como elementos.
- Usar plantillas preexistentes de cuadros de mando.



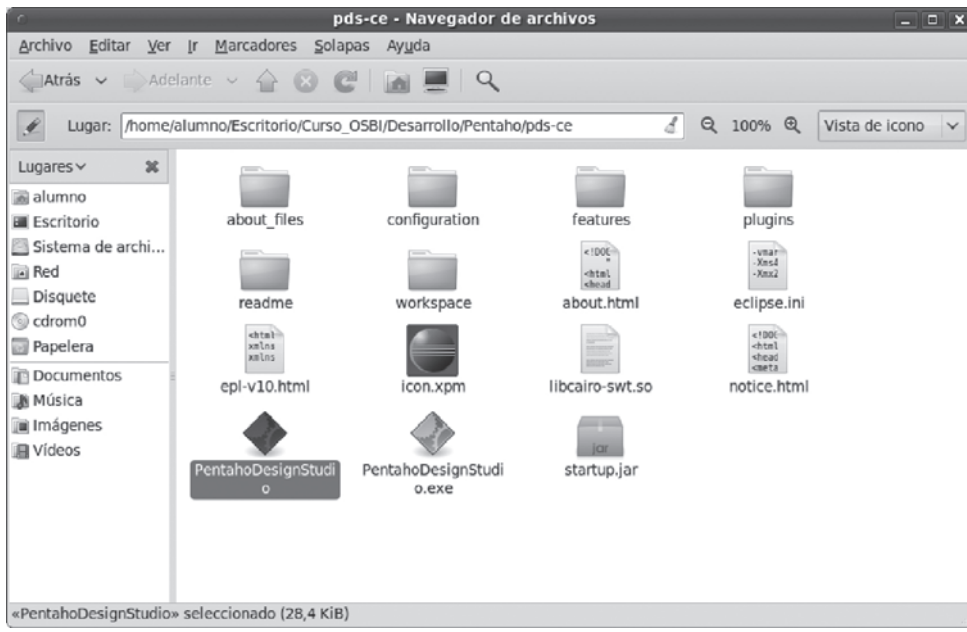
3. Caso práctico

3.1. Cuadro de mando mediante CDF

Pentaho Server incluye CDF, framework[3] que simplifica la creación de un dashboard. Es necesario remarcar que para crear un cuadro de mandos en Pentaho, las tecnologías usadas son: HTML, CSS, JavaScript y las acciones de Pentaho.

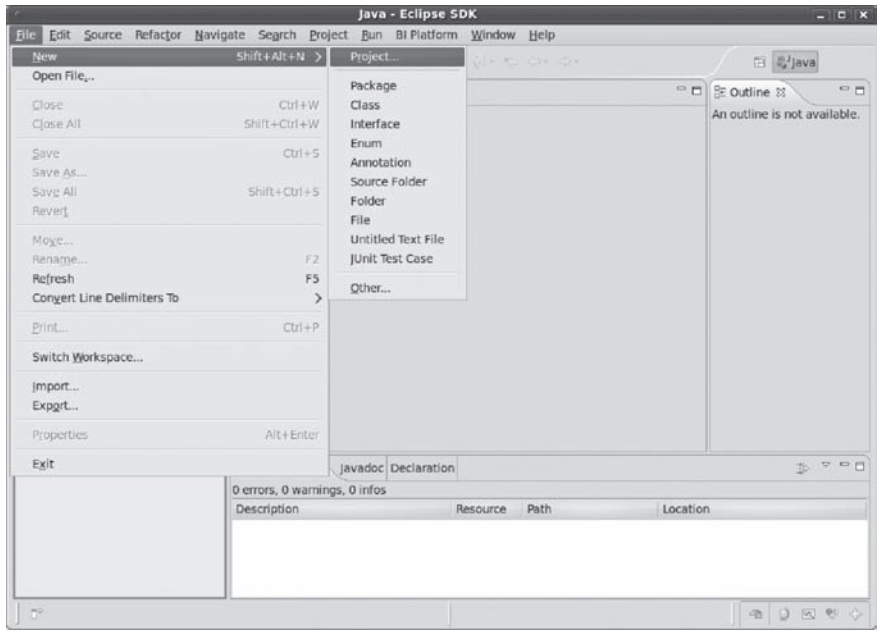
Los pasos son los siguientes:

- Iniciamos Pentaho Design Studio.

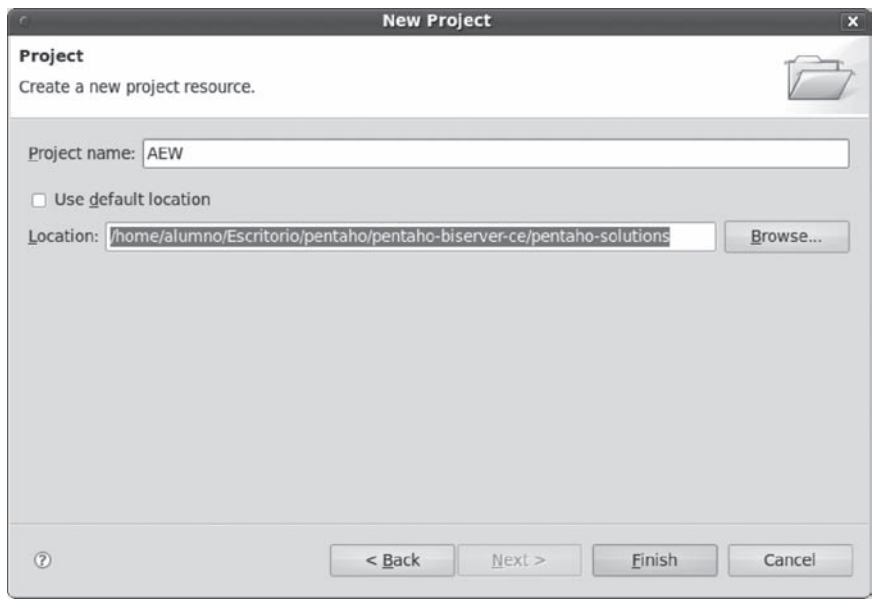


- Esta herramienta está basada en Eclipse, y por lo tanto es necesario crear un proyecto.

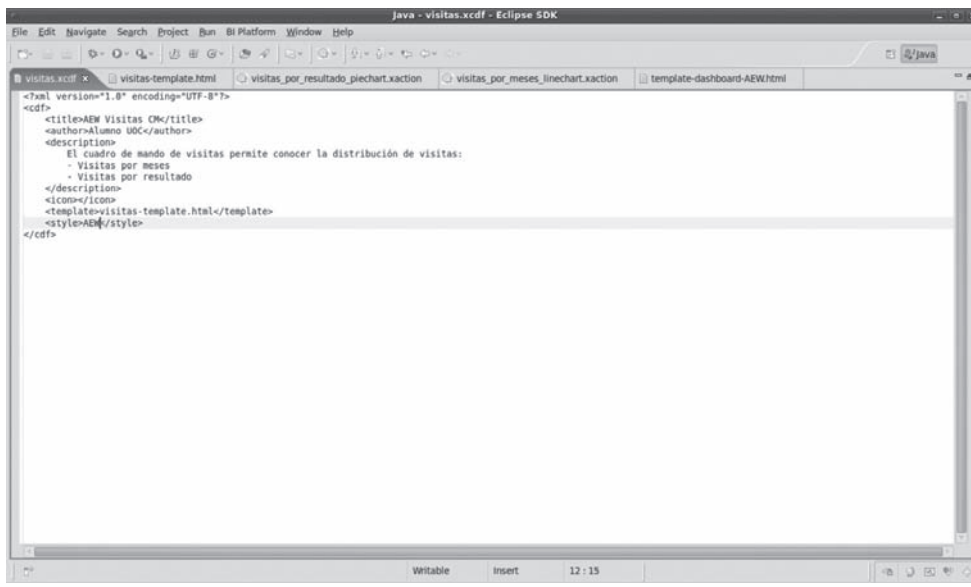
3. Para saber más, consultar: http://wiki.pentaho.com/display/COM/Community+Dashboard+Framework. Otro ejemplo interesante de construcción de un cuadro de mando: http://b-e-o.blogspot.com/2009/09/dynamically-creating-pentaho-cdf_9214.html.



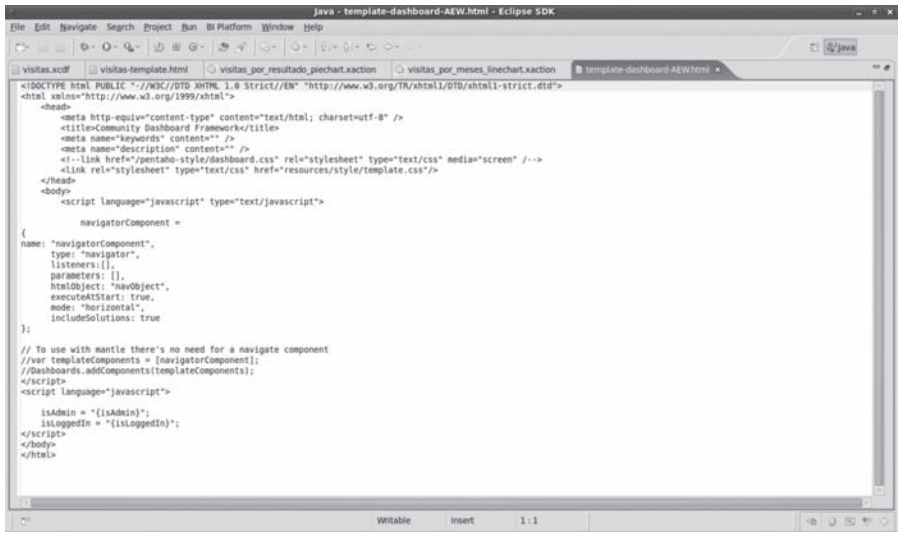
- Mapearemos el proyecto a la carpeta pentaho-solutions. Llamaremos el proyecto AEW.



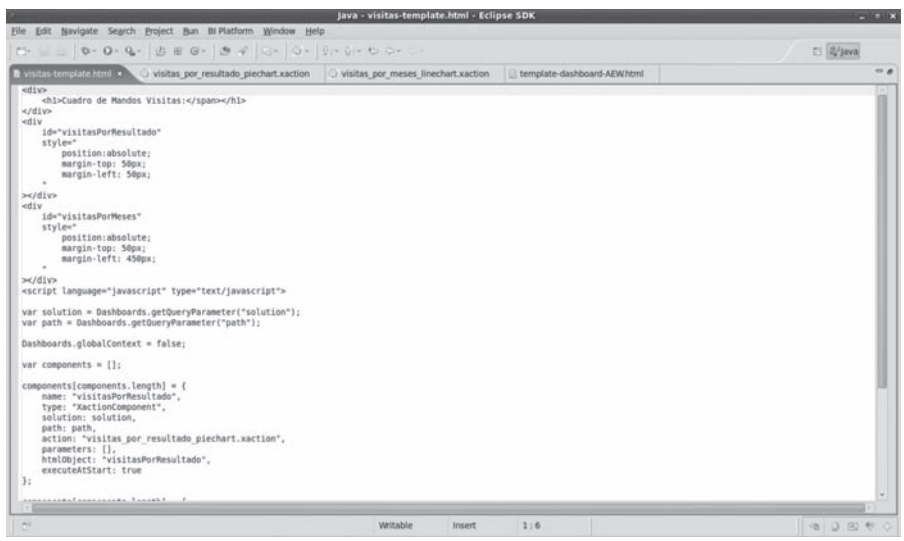
- Un cuadro de mando en Pentaho se compone de diferentes elementos que deben ser creados para su correcto funcionamiento. El hecho de estar dividido en diferentes elementos proporciona gran versatilidad.
- Todos los ficheros de nuestro cuadro de mando –excepto template-dashboard-AEW.html, que debe estar con el resto de templates situado en pentaho-solutions/system/pentaho-cdf– se sitúan en la carpeta donde estamos construyendo nuestra solución.
- Fichero con visitas.xcdf, que es el fichero que permite indicar la plantilla en la que se basa el cuadro de mando (por ejemplo, si debe incluir elementos comunes básicos), CSS (que recogerá en un contexto de producción el formato de la organización), el título, etc.



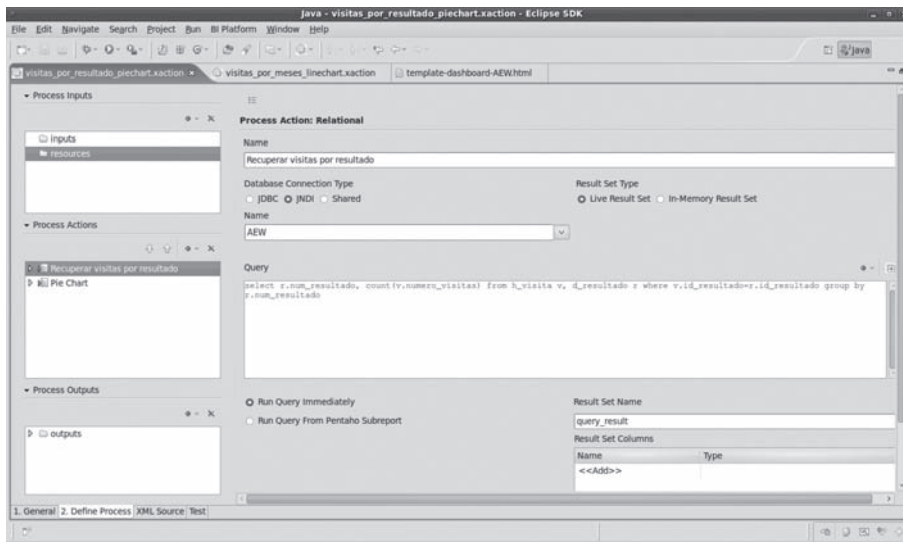
- Fichero con template-dashboard-AEW.html, que define la estructura base del cuadro de mando. Por ejemplo, bandas superiores e inferiores, logos, etc. En nuestro caso, consideramos una plantilla sin contenido exceptuando el contenido por defecto.

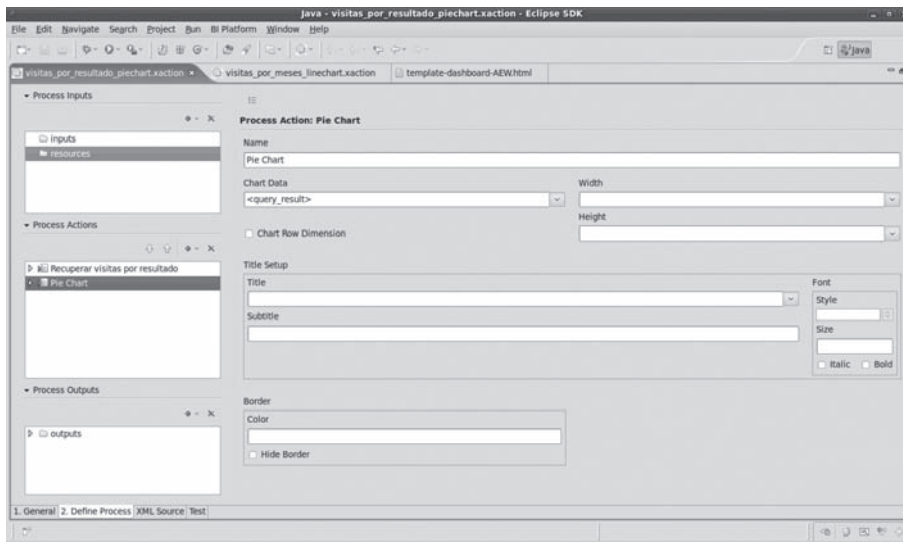


- Fichero con visitas-template.html, que permite determinar el contenido del cuadro de mando. En nuestro caso, el cuadro de mando tendrá un título y dos elementos gráficos. Cada elemento se encapsula en una etiqueta div. En el caso de los elementos gráficos de negocio, también es necesario determinar su posición y el nombre del objeto llamado. En la parte inferior, dentro del script, se indica la xaction (acción de Pentaho) llamada.

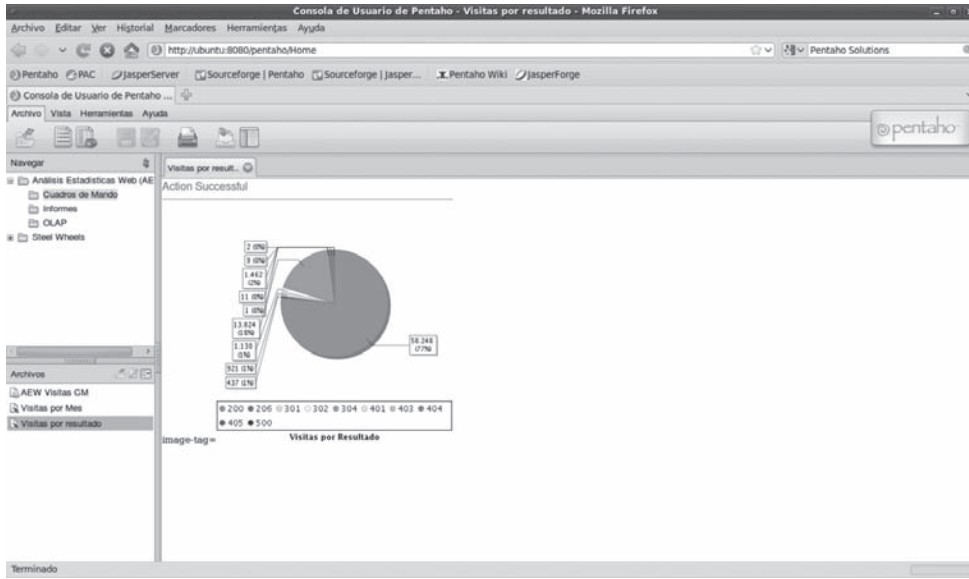


- Fichero con visitas_por_resultado_piechart.xaction, que permite definir el gráfico que se mostrará. Esta acción encapsula dos procesos:
- Petición de los datos a la base de datos (usamos la JNDI definida en Pentaho Admistration Console).
- Se remite el resultado de la consulta al gráfico que se va a generar, en este caso una tarta.
- Dado que este fichero se crea directamente en la carpeta de la solución, ya está publicado y sólo es necesario refrescar la solución.

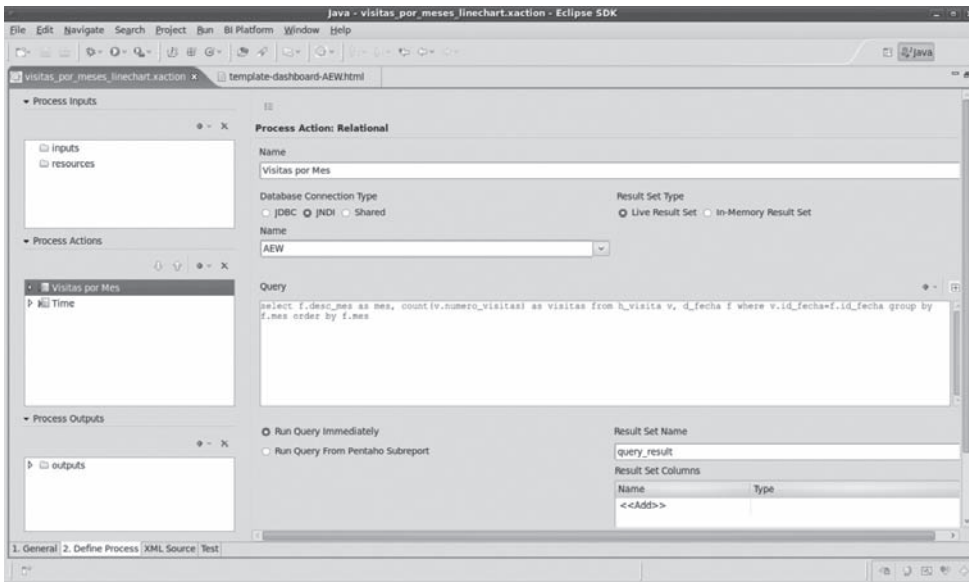




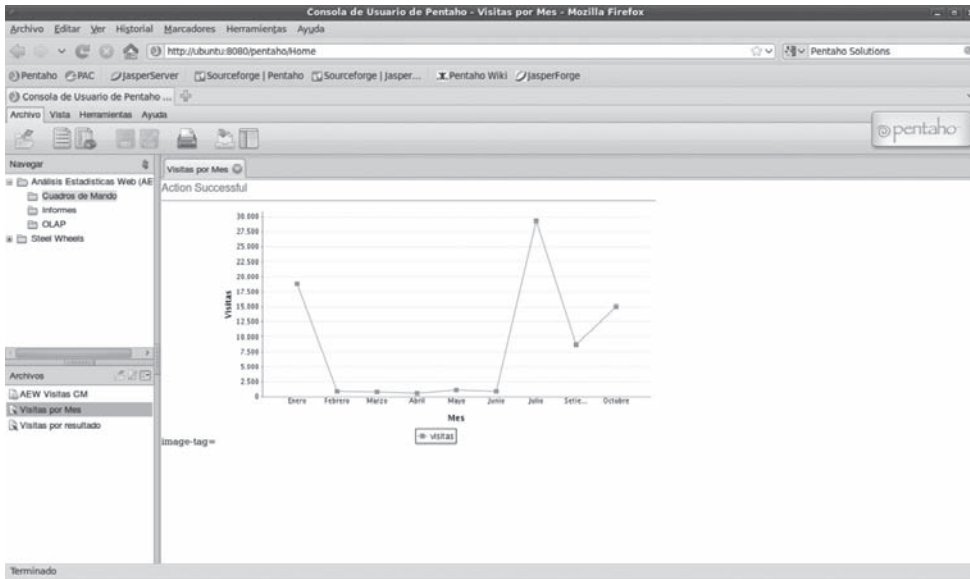
- El resultado es un gráfico que muestra las visitas por resultado.



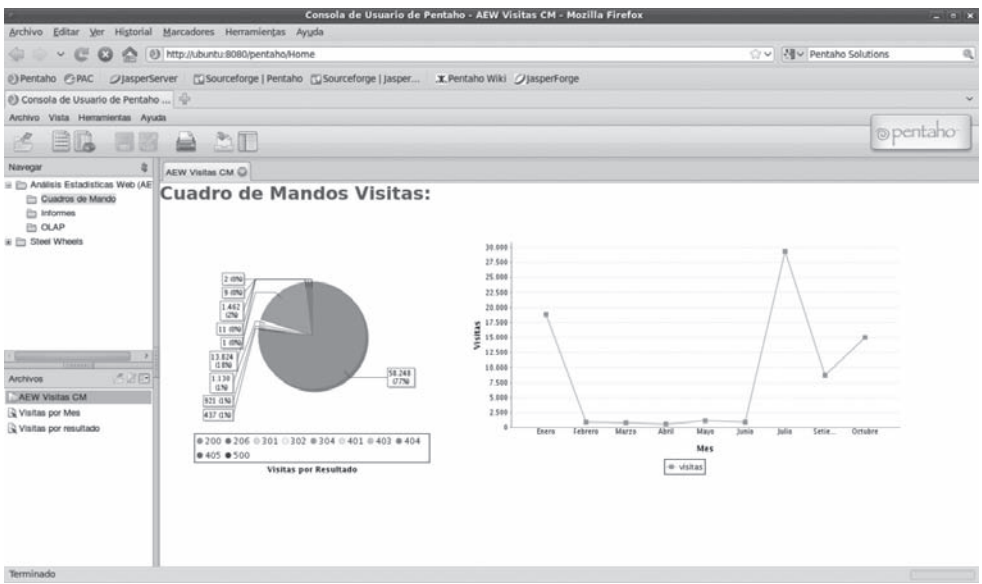
- Lo mismo se realiza para el otro elemento gráfico.



- Cuyo resultado es un gráfico de evolución.



- La combinación de ambos elementos nos proporciona el cuadro de mando de visitas.



4. Anexo 1: Consejos para crear un cuadro de mando

Existen diversas metodologías que pueden emplearse para la creación de un cuadro de mando. Éstas no son excluyentes, sino que la mayor parte de las veces se complementan entre sí. A la hora de su elección, se debe valorar la situación en la que se encuentra inserta la organización y escoger aquella o aquellas que más se adapten a la misma. A continuación se presenta la relación de las más empleadas en el diagnóstico tecnológico:

- Respecto a la disposición de la página:
  - Menos es más: el usuario final tiene una capacidad limitada para analizar información. Es necesario enfatizar la información importante. No es conveniente tener demasiadas vistas por página.
  - Regla de la mano: continuando con lo anterior, se debe limitar a cinco el número de elementos por página.
  - No usar scroll: esto es válido tanto para un cuadro de mandos como para una página web. El usuario avanzado no hará scroll.
  - Disposición en pantalla: existen patrones de atención que pueden ser usados al crear un cuadro de mando.
    - Top-left: primer punto de atención. Lugar natural donde disponer la información más importante.
    - Center: segundo punto de atención. Lugar natural donde disponer la segunda información más importante.
    - Top-right, bottom-left: partes neutrales.
    - Bottom-right: nadie se fija. No poner información relevante.
  - Usar menús fijos.
  - Reducir la cantidad de puntos de navegación por página, para evitar la confusión y la sobreinformación.
  - Concentrarse en la página principal, donde los usuarios centran la mayor parte de su atención.
  - Usar componentes gráficas y destacarlas.
  - Destacar los links con un color significativo (por ejemplo, azul).

- Respecto al contenido:
  - Usar dos decimales y escalar números grandes. Para el usuario es complicado entender números grandes y pequeños.

– Hacer foco en la comprensión de los datos y la información que se transmite, no sólo en la belleza.
  – Recordar que un dato sin contextualizar no significa nada.
  – Tener claro que no todos los gráficos tienen sentido para todo tipo de datos.
  – Usar fuentes claras para la lectura.
  – Usar la misma familia de colores para todos los gráficos.
  – No basarse sólo en colores, dado que hay usuarios que no ven bien los colores.
  – Alinear el texto.
  – Acentuar texto con fondos de color opuesto.
  – Si es posible, usar CSS.

5. Anexo 2: Consideraciones sobre el uso de tablas y gráficos

Las tablas y los gráficos son dos de los principales elementos usados en la creación de un cuadro de mando. Existen diversos consejos para mejorar el uso de estos elementos.

En el momento de considerar tablas como elemento de análisis, se debe tener en cuenta:

- Si las tablas serán estáticas o dinámicas (es decir, si el usuario tendrá capacidad de interactuar con la información y acceder a nuevos datos, aparte de los mostrados por defecto).
- La posibilidad de crear informes asimétricos.[4] De forma que se muestren en un mismo eje una combinación de ciertos valores de varias dimensiones (útil, por ejemplo, para análisis financieros).
- El número de ejes o dimensiones que tendrá la tabla. Un número excesivo puede dificultar la lectura y la comprensión de los datos.
- La posibilidad de crear informes con codificación de color (o semafórica), de forma que se puedan establecer alertas visuales de fácil comprensión que per-

4. Un informe asimétrico es aquel en el que el diseño de filas y columnas no es uniforme.

mitan detectar cambios rápidamente y tomar decisiones de forma más ágil.

- La posibilidad de permitir drill-through,[5] pulsando un valor concreto de nuestra tabla.
- El enlace entre los resultados de tablas y gráficos.
- La posibilidad de usar facilidades de manejo de las tablas como drag & drop (arrastrar y soltar), drill-down (profundizar niveles dentro de una jerarquía), etc.
- Posibilidad de enlazar los elementos de los ejes o de los valores con una URL externa o un fichero (Excel, PDF...).

En el momento de considerar gráficos como elemento de análisis, se debe tener en cuenta que:

- El gráfico debe ajustarse a los datos mostrados, y es necesario usar tipos diferentes para enriquecer el análisis.
- Es muy importante el poder customizar los gráficos, es decir, poder cambiar colores, leyendas, fuentes, etc.
- Es importante poder enlazar dinámicamente tablas y gráficos.
- En los gráficos que aportan mucha información es importante devolver datos cuando el cursor señale un espacio concreto del mismo.
- Pueden ser necesarias funcionalidades como drill hacia otros visualizadores como tablas, otros gráficos, documentos, etc.
- Hay que evitar incluir excesiva información en cada gráfico, pues no resultaría muy útil para el análisis.

5. Drill down significa profundizar niveles dentro de una jerarquía.

6. Glosario

BSC	Balanced ScoreCard
CDF	Community Dashboard Framework
CMI	Cuadro de Mando Integral
DOLAP	Desktop On-Line Analytical Processing
HTML	HyperText Markup Language
JSP	Java Server Page
OLAP	On-Line Analytical Processing

7. Bibliografía

BOUMAN, R., y VAN DONGEN, J. (2009). *Pentaho® Solutions: Business Intelligence and Data Warehousing with Pentaho® and MySQL*. Indianapolis: Wiley Publishing.

ECKERSON, W. (2005). *Performance Dashboards: Measuring, Monitoring and Managing Your Business*. Hoboken: Wiley & Sons.

FEW, S. (2006). *Information Dashboard Design: The Effective Visual Communication of Data*. Sebastopol: O'Reilly Media.

FEW, S. (2009). *Now You See It: Simple Visualization Techniques for Quantitative Analysis*. Sebastopol: O'Reilly Media.

KAPLAN, Robert S., y NORTON, David P. (1996). *The Balance Scorecard: Translating Strategy into Action*. Boston: Harvard Business School Press.

RASSMUSSEN, N., y otros (2009). *Business Dashboards: A Visual Catalog for Design and Deployment*. Hoboken: Wiley Publishing.

Capítulo VII

Tendencias en Business Intelligence

En los capítulos anteriores se han tratado los principales conceptos de la inteligencia de negocio: data warehouse, procesos ETL, análisis OLAP, reporting y cuadros de mando. Los conceptos introducidos permiten iniciarse en las bases consolidadas y maduras de la inteligencia de negocio.

Sin embargo, como ya se ha comentado en el capítulo 1, la inteligencia de negocio aúna múltiples estrategias, tecnologías y metodologías. De la misma forma que la tecnología, la sociedad y los modelos de negocio han evolucionado, la inteligencia de negocio ha cambiado y madurado drásticamente para responder y adaptarse a dichos cambios.

Por otro lado, la madurez del mercado propició en el periodo 2005-2007 una fuerte consolidación del mercado tradicional de la inteligencia de negocio, que en el año 2010 ha vuelto a iniciarse con la consolidación de soluciones de Master Data Management (MDM). Por lo que, en la actualidad, el mercado Business Intelligence se halla segmentado de la siguiente manera:

- Grandes agentes externos que han complementado su portafolio de soluciones para empresas con las soluciones de BI. En este ámbito, las principales marcas son: Oracle, que recientemente ha adquirido Hyperion; SAP, que ha adquirido Business Objects; IBM, que se ha hecho con el control de Cognos.
- Empresas tradicionales del mercado que se mantienen con un portafolio especializado. Como, por ejemplo, Information Builders o Microstrategy.
- Empresas de nicho especializadas en un ámbito concreto de la inteligencia de negocio, como, por ejemplo, data warehouse (Teradata, Netezza, Vertica...), la integración de datos (Informatica, Talend...), análisis visual (Panopticon...), análisis dinámico y flexible (QlikView, Tableau...), etc.

- Empresas open source que cubren todo el stack tradicional de la inteligencia de negocio y ofrecen soluciones con TCO (Total Cost Ownership) reducido.

Aunque pudiera parecer extraño, las sinergias de la consolidación no han provocado que la innovación desaparezca del mercado BI, sino que se ha trasladado de los grandes actores a las empresas pequeñas que basan su competencia en la innovación.

Es necesario destacar que además que el mercado de inteligencia de negocio es uno de los pocos que ha seguido creciendo en el año 2009 aunque de forma más moderada.

Según Gartner, las ventas han crecido un 4,2% respecto a los 9,3 Billones de dólares en 2010.

Este hecho refuerza la idea de la importancia de la inteligencia de negocio en las organizaciones.

A lo largo de este capítulo, hablaremos de diferentes tendencias actuales en este mercado. El objetivo es presentar cómo están evolucionando los sistemas actuales, las necesidades del mercado y las principales tendencias que marcarán el futuro.

1. Factores de evolución

Como ya se ha comentado anteriormente, a pesar de los signos de madurez del mercado Business Intelligence, éste sigue siendo una fuente relativa de crecimiento e innovación. Esto es así debido a que en la inteligencia de negocio confluyen diversos factores. Entre ellos destacamos los siguientes:

1.1. Ubiquitous Computing (computación ubicua)

La computación ubicua se caracteriza principalmente por tres factores: por una proliferación de tecnología embebida en dispositivos de múltiple naturaleza, por la integración de la informática en el ámbito personal (de forma que los ordenadores no se perciban como objetos diferenciados), y por el hecho de que nuestros datos o aplicaciones estén disponibles desde cualquier lugar. Esta tendencia está dando lugar a lo que se conoce como *the internet of things,* donde los objetos son capaces de comunicarse a través de sensores conectados a redes que usan el protocolo de internet. Es decir, el mundo físico se está convirtiendo en un tipo de sistema de información, y estas redes permiten crear nuevos modelos de negocio, mejorar procesos y reducir costes y riesgos. Un punto a destacar es el emergente uso de procesos y de datos contextualizados en el tiempo y el espacio; este tipo de dispositivos se usan de manera natural en todo tipo de situaciones y circunstancias.

Uno de los resultados naturales de esta tendencia es un incremento desproporcionado de la cantidad de datos relacionados con nuestro modelo de negocio. Este incremento produce la necesidad natural, por parte de los usuarios finales, de captar, entender y analizar esta información en tiempo real y a través de mecanismos naturales y sencillos.

1.2. Cloud Computing (computación en la nube)

Cloud Computing es un nuevo paradigma[1] que consiste en ofrecer servicios a través de internet. En los últimos años, este tipo de servicios se ha generalizado entre los principales fabricantes para formar parte de las opciones disponibles de su portafolio de servicios, e incluso en algunos casos para ser la forma predominante o totalitaria de los mismos. Por ejemplo, Google, que ofrece todos sus servicios en la red (desde el buscador hasta aplicaciones para empresas que incluyen correo, editor de documentos, calendario...), Amazon (que ofrece un servicio de almacenamiento), Salesforce (que ofrece un CRM on-demand), Microsoft (que ofrece una plataforma de cloud computing llama-

1. Un paradigma es, desde fines de la década de 1960, un modelo o patrón en cualquier disciplina científica u otro contexto epistemológico.

da Azure) o Abiquo (empresa española que ofrece una solución para gestionar entornos corporativos virtualizados desplegados en la nube).

En la computación en la nube existen diferentes capas:

- **SaaS (Software as a Service):** es la capa externa y es un modelo de despliegue de software en el que una aplicación es alojada como un servicio ofrecido a los clientes. Como ejemplo más claro tenemos el CRM On-Demand[2] de SalesForce.
- **PaaS (Platform as a Service):** es la capa intermedia. Nace a partir del modelo de distribución de aplicaciones SaaS. El modelo PaaS hace que todas las utilidades necesarias para el ciclo de vida completo de construir y distribuir aplicaciones web estén disponibles en internet, sin descargar software o requerir instalación por parte de desarrolladores, responsables de informática o usuarios finales. También es conocido como cloudware. Como ejemplo más claro tenemos Google App Engine.
- **IaaS (Infraestructure as a Service):** es la capa núcleo del servicio. Se refiere al acceso a recursos computacionales que típicamente son poseídos y operados por un proveedor externo, de forma consolidada, en centros de proceso de datos. Los clientes de los servicios de computación en nube compran capacidad computacional on-demand y no se preocupan de la tecnología subyacente usada para conseguir el incremento en capacidad del servidor. Como ejemplo más claro tenemos Amazon Web Services.

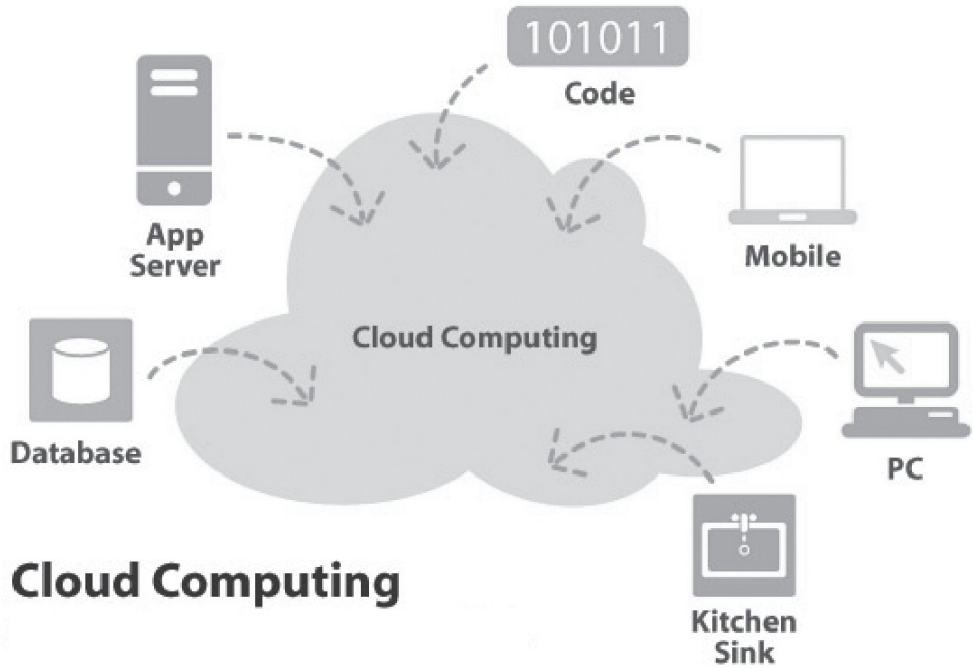
Se considera que Cloud Computing es una evolución natural de ASP[3] dado que actualmente existe el nivel de tecnología adecuado.

Cabe comentar que este servicio puede ser público (manejado por terceros) o privado (manejado por la organización) o híbrido (combinación de los anteriores).

---

2. On-demand: servicio o característica que responde a la necesidad del usuario de gratificación instantánea e inmediata en el uso. En la mayoría de casos la proposición de valor de un servicio on-demand está constituida sobre el hecho de que el usuario o cliente del servicio evita una inversión inicial fuerte y en su lugar participa en un modelo paga-conforme-lo-usas (pay-as-you-go), que habitualmente hace que los servicios on-demand sean más asequibles para los usuarios.

3. ASP (Application Service Provider): es un negocio que ofrece servicios basados en ordenadores a través de una red. El software ofrecido bajo el modelo ASP es llamado en ocasiones Software On-Demand o Software como Servicio (SaaS). El sentido más limitado de este término es ofrecer acceso a una aplicación particular (como facturación médica) utilizando un protocolo estándar como HTTP. No debe confundirse con ASP.NET, el lenguaje de programación de Microsoft.



1.3. Economía de la atención

En los últimos años la cantidad de información, tanto relevante como irrelevante, a la que tienen acceso los usuarios no para de crecer. Se presenta una situación de infoxicación, como apunta Alfons Cornellà, que acuña este término como:

> Exceso informacional, intoxicación informacional, cuando tienes más información de la que humanamente puedes procesar y, como consecuencia, surge la ansiedad (técnicamente, information fatigue syndrome). En inglés el término es *information overload* (sobrecarga informacional).

Por lo que es necesario exigir relevancia a la información, que debe ser obtenida además de forma inmediata. Por poner un ejemplo, Google proporciona respuesta de una búsqueda en apenas unos segundos (o incluso antes) ordenada por relevancia. Por lo que sólo aquella información que sea relevante, rápida y de calidad será capaz de conseguir su cuota de atención.

Este problema de atención puede ser resumido, desde la óptica de Michael H. Goldhaber, de la siguiente manera:

El ancho de banda de datos e información que recibe la gente se está incrementado de forma continua principalmente por dos motivos: la tecnología permite enviar más en menos tiempo y hay más emisores. Podemos entender por banda ancha la cantidad de información que alguien recibe por unidad de tiempo. Por otro lado es posible entender la atención personal como la cantidad de tiempo que una persona puede dedicar a cada información que recibe. De ambos conceptos se deduce que a mayor ancho de banda, menor capacidad de atención personal.

La única solución pasa por centrarse en la información relevante para no caer en cuello de botella sistémico.

1.4. Incremento desproporcionado de datos

La evolución de las tecnologías de la información ha propiciado que la gran mayoría de procesos o bien sean digitales o bien puedan monitorizarse mediante sistemas de información. Por otro lado, estamos en la era de la computación ubicua. Por ello, se genera una gran cantidad de datos que frecuentemente es difícil de comprender, controlar, monitorizar y analizar de forma conjunta. Este incremento de datos guardados es producto de diversos factores, entre los cuales podemos destacar la reducción del precio y el aumento de la capacidad de las componentes de almacenaje de datos así como una evolución de los protocolos y los sistemas de información que son capaces de soportar los procesos de negocio.

Este fenómeno se conoce como Big Data; en él, lo importante no es la gran cantidad de datos que se producen y se guardan, sino lo que se hace con esos datos. Es decir, qué aportan los datos a nuestro modelo de negocio.

1.5. Mercado altamente dinámico y competitivo

El mercado global en el que participan todas las empresas funciona a una velocidad mucho mayor que antaño gracias a las tecnologías de la información y su aceptación en la sociedad. Por ello, empresas, organizaciones e instituciones deben reaccionar mucho más rápidamente a las necesidades de los clientes,

a las acciones de la competencia y a los cambios sociales, tanto positivos como negativos, mediante la incorporación de tecnologías, tanto las que son consideradas una commodity[4] como las que no lo son.

Es decir, teniendo en cuenta la reflexión que hace Nicholas Carr en su famoso artículo "Does IT Matter?", debemos tener presente que la tecnología no es el fin sino el medio a través del cual una empresa consigue ventaja competitiva en el mercado. Esta ventaja se puede obtener de diferentes formas –por ejemplo automatizando procesos, entendiendo mejor el comportamiento de los clientes, tomando mejores decisiones–, todas ellas soportadas por la tecnología.

1.6. Empresa extendida

Las empresas tienen relaciones con partners, proveedores, clientes, competencia e inversores. Todas estas relaciones generan datos de valor a incorporar en la toma de decisiones que permiten comprender de una forma mucho mejor los procesos de negocio. Por ejemplo, conocer los precios y los productos de la competencia y compararlos con los propios nos permite posicionarnos y saber si nuestra empresa es competitiva a nivel de prestaciones y precios.

Este enfoque de considerar que el ámbito de una empresa incluye todos los elementos que interactúan con la misma es lo que se conoce como empresa extendida.

1.7. Democratización de la información

Tanto en el contexto de una organización como en la sociedad, las personas están identificando la necesidad de información de valor para la toma de decisiones del día a día. Es decir, necesitan información de valor no sólo las decisiones estratégicas, sino también las tácticas y operativas. Es por ello que se buscan mecanismos para desplegar procesos que democraticen la información y soporten las acciones de los usuarios de forma no intrusiva.

4. Commodity significa mercancía, y mercancía es todo "lo que se puede vender, comprar o intercambiar por otro bien o cosa". Un commodity es un producto genérico, básico, sin mayor diferenciación entre sus variedades, y destinado a uso comercial.

1.8. Open source

Desde hace bastantes años el open source no es una tendencia emergente, sino que afecta a los procesos de producción de software de forma profunda. Tiene y tendrá, en los años venideros, una presencia importante en todos los sectores, tal y como comenta Gartner:

> En 2012, el 80% del SW comercial incluirá algún componente open source. Incluir componentes open source en los productos para abaratar costes es considerado la mínima estrategia que las compañías pueden llevar a cabo para mantener su ventaja competitiva en 5 años.

Las empresas y organizaciones open source han ido evolucionando para ofrecer una respuesta adecuada a las demandas del mercado teniendo en cuenta los siguientes factores:

- Existen unos costes ocultos en la implantación de software open source. Es necesario contratar o formar personal especializado que dé soporte en la evaluación, en la integración, en la corrección de errores y en el ciclo de vida del producto, y que participe en la comunidad para que conozca la evolución de la aplicación. Tales costes inciden en horas, dinero o ambos.
- Ausencia de soporte y de un roadmap claro y preciso por parte de la organización que desarrolla el producto.
- Ciertas licencias open source no son business-friendly (limitando el uso a ámbitos académicos o organizaciones no comerciales).

Estas empresas cumplen los principios del movimiento open source:

- Abierto: la comunidad tiene libre acceso, uso y participación del código fuente, así como la posibilidad de usar foros para proporcionar feedback.
- Transparencia: la comunidad tiene acceso a roadmap, documentación, defectos y agenda de las milestones.
- Early & Often: la información se publica de manera frecuente, y pronto a través de repositorios públicos (incluyendo el código fuente).

Otros principios que persiguen estas empresas encarados a ofrecer confianza y fiabilidad son:

- Búsqueda de la excelencia a nivel de servicios, tanto cuando hablamos del desarrollo del producto como del trato con el cliente.

- Apuesta por la innovación generando grandes sinergias que pueden derivarse en motor de ideas de negocio.

No todas las organizaciones que desarrollan soluciones open source (como empresas, por ejemplo Pentaho, MySQL –ahora Oracle–, Openbravo, Eclipse, Mozilla) tienen la misma aproximación al modelo de negocio presentado, si bien tienen puntos comunes.

1.9. Nuevos modelos de producción

En el seno de las organizaciones han aparecido nuevos modelos de trabajo. A destacar el siguiente: la cultura open source en las organizaciones puede ser incluida dentro del fenómeno denominado Commons-Based Peer Production (CBPP), acuñado por Yochai Benkler. Este término define una nueva forma dentro del contexto de la producción de información o bienes culturales. A grandes rasgos (y desde la perspectiva de la teoría económica), los individuos trabajan de manera más eficiente porque eligen qué tareas realizar en base a sus propias preferencias y habilidades en un ambiente de colaboración, donde los resultados de la producción son puestos en el dominio público. En otras palabras, la principal baza del CBPP (a diferencia de otros mecanismos de producción "propietarios") es la "autoselección" como mecanismo de asignación de los recursos relacionados con el talento y la creatividad humana. Dado que quienes mejor conocen sus aptitudes y habilidades son los propios individuos, ellos deciden en qué tareas participar. Es más, al quedar sus contribuciones en el dominio público se evita la pérdida de eficiencia que supone la "exclusividad de derechos" impuesta por las licencias típicamente utilizadas en otros contextos productivos más tradicionales.

Las principales características que permiten distinguir las organizaciones que utilizan el CBPP son:

- La estructura interna suele ser aplanada, con jerarquías bastante diluidas, lo que permite un flujo rápido de conocimiento y de información. Muchos proyectos open source suelen girar en torno al fundador o fundadores o a las personas que dieron el paso inicial, aunque no es un requisito imprescindible. En su mayoría poseen una junta directiva (Directors Board) y jefes de proyectos (Project Management Committees Chairs ofcers). Estas figuras son las encargadas de tomar las decisiones y de representar a la

organización frente a terceros. Sin embargo, el rasgo distintivo del CBPP y las iniciativas OS es que funcionan como una meritocracia, lo que quiere decir que aquellos individuos que muestren tener mejores capacidades a través de aportaciones significativas son invitados a formar parte de la comunidad de desarrollo. En otras palabras, se adquiere mayor poder de decisión dependiendo de la calidad de su trabajo.

- Se emplean licencias abiertas (aproximadamente existen 59 tipos de licencias diferentes). El factor común en todas las organizaciones es que para la utilización del bien protegido no es necesario pagar royalties ni otro tipo de cargas. El producto final es puesto en el dominio público y cualquiera puede hacer uso de él o copiarlo, y en muchos casos se permite al consumidor modificar y mejorar el producto, con la condición de que también sea puesto en el dominio público.
- La metodología o sistema de producción de los proyectos, generalmente tiene un grado alto de modularidad, granularidad, y es de carácter asincrónico. Esto significa que las diferentes actividades que componen un determinado proceso pueden romperse en pequeñas tareas ejecutables en diferentes momentos y desarrollarse sin requerir una cantidad muy elevada de tiempo o esfuerzo para los individuos.
- Una particularidad muy importante son los incentivos por los que los agentes deciden participar en la producción. A diferencia de los entornos tradicionales, los participantes no están influenciados directamente por razones monetarias. Entre otros, los incentivos que intervienen son: beneficio del uso individual (own-use), complementariedad con otros bienes, señalización en el mercado laboral (motivos profesionales), educación, formación. Y también otros motivos de carácter psicológico, como el simple altruismo, el sentido del deber para con la comunidad, y retos intelectuales (lo que se denomina *learn and fun*).
- Otra característica relevante es el ambiente de colaboración en el que se desarrolla el CBPP. Al ser los propios individuos quienes deciden qué tareas realizar, se realiza un mejor reparto de los incentivos; es lo que Benkler denomina *self-identication*. Para él, esto último representa una ventaja significativa con respecto a los modelos tradicionales de empresa, donde es el supervisor el encargado de decidir qué tareas son llevadas a cabo por quién. De esta manera se soluciona el problema de asimetría de información entre el empleado y el superior que se da en esas situaciones, y se obtiene lo que en teoría de la agencia se denomina *information gains*.

1.10. Social Media

Social Media son medios de comunicación social donde el contenido es creado por los usuarios mediante el uso de tecnologías de fácil uso y acceso a través de tecnologías de edición, publicación e intercambio. Estos medios de comunicación pueden adoptar muchas formas diferentes: foros de internet, blogs, wikis, podcasts, fotos y vídeo. En un medio social se generan conversaciones basadas en un propósito común, y, por lo tanto, la calidad y proyección depende principalmente de las interacciones de las personas y la riqueza del contenido generado. Así, a través de estos canales se genera la oportunidad del uso de la inteligencia colectiva de los usuarios. Entre los social media más destacados están Twitter, Facebook, YouTube o Flickr.

1.11. Open Knowledge

Open Knowledge es una nueva filosofía heredera del open source y que tiene como objetivo compartir datos, información y conocimiento. Últimamente han aparecido diferentes organizaciones para dar a conocer esta iniciativa y facilitar un entorno –tanto tecnológico como legal– para compartir información, como por ejemplo The Open Knowledge Foundation (http://www.okfn.org/), The Open Data Foundation (http://www.opendatafoundation.org/), o el portal de proyectos de Open Knowledge llamado KnowledgeForge (http://knowledgeforge.net/).

1.12 Movilidad

El acceso masivo a internet, el uso extendido de las redes sociales y la eclosión de los dispositivos inteligentes está redefiniendo el puesto de trabajo en las organizaciones y, por extensión, cómo se toman las decisiones en el seno de las organizaciones.

La movilidad ha reducido las fronteras del dónde, cuándo y cómo habilitando nuevos lugares de trabajo, la capacidad de acceder a recursos de empresa en cualquier lugar, momento o a través de cualquier dispositivo.

Esto está generando el amanecer de una nueva economía basada en la información en la cada empleado necesita tener:

- La capacidad de tomar decisiones de forma colaborativas (los empleados como red) y automatizadas (mejora de la eficiencia)
- La capacidad de tomar acciones en tiempo real y tener registro de dichas acciones.
- La capacidad de seguir la decisiones de forma ubicua (en linea con la flexibilidad del puesto de trabajo) y continua (monitorización / seguimiento de la decisión efectuada).

2. Tendencias en Business Intelligence

Los factores anteriormente comentados, junto con la evolución de la tecnología subyacente, el refinamiento de la metodología y la madurez de las empresas y los usuarios en lo relativo a sistemas de Business Intelligence ya existentes, nos permiten determinar algunas de las principales tendencias del mercado:

2.1. Business Intelligence Operacional

La necesidad de democratizar la información entre los usuarios de una organización para mejorar la competitividad, ofrece nuevas oportunidades de negocio. Tradicionalmente, las soluciones de inteligencia de negocio han cubierto las necesidades estratégicas y tácticas de las empresas focalizando el servicio en la capa de dirección. La democratización conduce a responder a las necesidades operacionales y consiste en incrustar análisis en los procesos de negocio para poder proporcionar respuestas basadas en información confiable. Esta tendencia es conocida como BI Operacional.

2.2. Gestionar los datos como un activo

Los factores apuntan a un crecimiento desproporcionado de datos. A mayor cantidad de datos, mayores problemas con los mismos. Es por ello que una de

las necesidades básicas es incrementar la calidad de los datos a través de procesos de data quality.

La calidad de datos es sólo una de las iniciativas a tener en cuenta respecto a los datos. Es necesario desplegar políticas de data governance y Master Data Management (MDM).

> Data governance aúna personas, procesos y tecnología para cambiar la forma en que los datos son adquiridos, gestionados, mantenidos, transformados en información, compartidos en el contexto de la organización como conocimiento común, y sistemáticamente obtenidos por la empresa para mejorar la rentabilidad.

Es decir, estamos hablando de una disciplina en la que convergen conceptos como data quality, data management, business process management y risk management.

> MDM consiste en un conjunto de procesos y herramientas que define y gestiona de forma consistente las entidades de datos no transaccionales de una organización. Busca, por lo tanto, asegurar la calidad y la persistencia, recopilar, agregar, identificar y distribuir los datos de forma uniforme en dicho contexto.

MDM se compone de tareas como:

- Identificar las fuentes de origen de los datos.
- Identificar los productores y consumidores de datos maestros, como pueden ser la información de clientes, productos, proveedores...
- Recopilar y analizar metadatos sobre los datos maestros recopilados en el primer paso.
- Determinar los responsables (administradores) de los datos maestros.
- Implementar un programa de data governance (y, de forma consecuente, tener un grupo responsable de dicho programa).
- Desarrollar el modelo de metadatos maestros.
- Escoger una solución o conjunto de soluciones como medio para mejorar la calidad de datos.
- Diseñar la infraestructura necesaria para gestionar los datos maestros.
- Generar y testear los datos maestros.

- Modificar los sistemas consumidores y productores de información para tener un sistema de gestión de datos maestros y para que las aplicaciones que lo necesiten hagan uso de los mismos.
- Implementar un proceso de mantenimiento.

2.3. Una revolución tecnológica

Uno de los principales puntos discutidos en la implantación de soluciones de inteligencia de negocio es la dificultad inherente asociada a estos sistemas en varios aspectos:

- Despliegue.
- Mantenimiento.
- Uso.

Cada vez más, los compradores son capaces de analizar el mercado antes de tomar una decisión de compra ante este tipo de sistemas y, por lo tanto, buscan y demandan una menor complejidad en las soluciones BI.

Esto provoca que los fabricantes estén apostando por desarrollar un portafolio de soluciones que buscan incrementar el ROI (Return On Investment) más rápido asociado a un despliegue, mantenimiento y uso más ágil.

De esta forma nos encontramos:

- Uso de máquinas virtuales o appliances para el despliegue de soluciones de Business Intelligence. Estas soluciones conjugan software (suite BI + base de datos + sistema operativo) y/o hardware optimizado para tareas de inteligencia de negocio. Este tipo de despliegue facilita la gestión y el crecimiento de los sistemas.
- Aparición de BI SaaS. Poco a poco van apareciendo soluciones BI que ofrecen parte o toda su funcionalidad en modalidad SaaS. Aún quedan por resolver ciertos retos, como los relacionados con la integración de datos y la confianza.
- Mejora e inclusión de las soluciones de análisis predictivo (que son una evolución de las herramientas de minería de datos) como uno de los módulos básicos en las suites de inteligencia de negocio.
- Enfoque pragmático respecto a las hojas de cálculo. Uno de los principales problemas en el contexto de las organizaciones es el uso de Excel como herramienta de análisis. Este tipo de herramienta fomenta la aparición de silos de información así como problemas de integridad de datos. Algunos

fabricantes optan por un enfoque en el que gestionan de forma eficiente el uso de Excel como interfaz de trabajo.

- Inclusión de algoritmos avanzados de gestión de grandes cantidades de datos en los motores del data warehouse. Entre los diferentes enfoques tenemos:
  1. Uso de técnicas de consulta en paralelo de estructuras grid.
  2. Uso de bases de datos en columnas.
  3. Uso de motores in-memory combinados con soluciones de compresión de datos para reducir la huella de éstos.
  4. Uso de sistemas de almacenamiento no relacional provenientes de las empresas de social media como LinkedIn, Facebook o incluso Google. Entre estas soluciones tenemos protocolos como Hadoop o Map Reduce. Este tipo de enfoques se combinan con los anteriores para mejorar la escalabilidad.
  5. Uso de sistemas motores Complex Event Processing (CEP) que permiten detectar eventos importantes a nivel de negocio y que permiten enlazar las soluciones de inteligencia de negocio con la operativa a un nivel de eficacia mucho mayor.

2.4. El impacto del Open Source Business Intelligence (OSBI)

Desde hace muchos años existen soluciones de inteligencia de negocio open source, y eso puede parecer sorprendente dado que el eco mediático destacable se sitúa en los últimos cuatro o cinco años.

Actualmente existe un variado panorama en el que destacan las soluciones de compañías como Pentaho, Actuate BIRT, JasperSoft, Talend, Infobright, SpagoBI, Ingres o PALO, por sólo nombrar algunas de las más conocidas y maduras que cubren desde reporting hasta data mining.

Hagamos un ejercicio de recapitulación de los inicios del Open Source Business Intelligence. ¿Dónde nos deberíamos situar? A la mente me vienen dos hitos destacables:

- En 2001, Teodor Danciu inició el desarrollo de JasperReports, la primera solución de reporting open source. Actualmente es uno de los productos que standalone o integrado ofrece JasperSoft.
- En 2003, Julian Hyde inició el desarrollo de Mondrian, el motor ROLAP open source por excelencia. Actualmente forma parte de Pentaho, pero se encuentra presente en la gran mayoría de suites open source de mercado.

Las características comunes a todas estas soluciones son:

- El producto se ofrece en un formato open source.
- Presentan una comunidad que participa activamente en el producto y que incluso guía y lidera partes de su desarrollo.
- Tanto los productos como las empresas detrás de ellos presentan modelos y servicios maduros que pueden proporcionar el mismo nivel de confianza que soluciones comerciales.

Detrás de estas compañías hay profesionales del mundo del Business Intelligence con largas trayectorias, que conocen claramente las necesidades de negocio de los usuarios finales y que se dieron cuenta de que crear productos open source era una magnífica forma de cubrir dichas necesidades.

¿Qué es lo que cambia en un mercado cuando existen soluciones open source? Para poder responder a dicha pregunta, primero es necesario entender qué es lo que puede ofrecer una herramienta OSBI.

- Una solución open source puede identificarse claramente como una herramienta con TCO (Total Cost Ownership) reducido. Claro: si inicialmente dicha solución no cubre las necesidades de los clientes, no hay por qué preocuparse. Pensamiento de la competencia. Pero... ¿qué pasa cuando a lo largo del tiempo dichas soluciones maduran cada vez más hasta que dan respuestas solventes? Pensad, por ejemplo, en el caso Firefox e Internet Explorer. En este caso, las versiones iniciales del Firefox no eran estables ni se diferenciaban de Internet Explorer. Posteriormente, por el hecho de que Firefox cumplía los estándares de W3C y por la introducción de innovaciones como la navegación por pestañas, empezó a ganar cuota del mercado.

Es el lugar donde nos encontramos justo ahora. Y en este punto es cuando en el mercado podemos observar diferentes situaciones:

- Existen otras soluciones con TCO reducido, que principalmente han reducido el precio de sus licencias o bien han surgido con precios ya bajos y competitivos. Por ejemplo, QlikView.
- Algunas empresas ofrecen soluciones gratuitas que presentan una funcionalidad acotada gracias a la modularidad de sus soluciones, mientras que venden módulos y otros servicios.
- Otras empresas han apostado por soluciones innovadoras que no cubren ni las soluciones comerciales ya existentes en el mercado ni las soluciones open source.

Además, la madurez de las propias soluciones se ha conjugado con la situación económica desfavorable para gastos desproporcionados en IT. Esto hace la entrada de las soluciones OSBI en los procesos de evaluación mucho más sencilla, y facilita que sea la solución escogida. Esto último es fácil de comprobar por las últimas y frecuentes noticias por parte de las empresas OSBI y sus partners.

Hemos visto algunos de los impactos resultantes de las sinergias open source así como lo que sucede en el contexto actual. En estos tiempos es natural esperar, por lo tanto, un crecimiento no sólo lineal, sino de alguna magnitud superior para las soluciones del mercado OSBI.

2.5. Una necesidad crítica

Las empresas han identificado que la inteligencia de negocio es una necesidad crítica para la estrategia de negocio. Por ello actualmente están ejerciendo una influencia positiva sobre la tecnología para cubrir sus necesidades. Por ejemplo, en:

- Mejorar la capacidad de análisis.
- Facilitar el análisis de grandes volumetrías de datos mediante técnicas visuales.
- Demandar capacidades de integración de datos corporativos estructurados con todo tipo de datos, ya sean estructurados o no.
- Aparición de soluciones que permitan el desarrollo de elementos de análisis de forma colaborativa.

Dichas necesidades hacen que en el contexto de una organización sea necesario que existan organismos que regulen el proceso de madurez del sistema de inteligencia de negocio; por lo tanto, aparece la necesidad de crear centros de competencia (BICC, Business Intelligence Competency Center). La existencia de este tipo de centros demanda de la participación de especialistas con talento para desplegar dichas iniciativas. De manera que la búsqueda de talento se convierte en una primera necesidad y se establece como prioritario identificar y contratar personal especializado tanto en herramientas a nivel de usuario como en conocimientos.

3. Glosario

ASP	Application Service Provider
BI	Business Intelligence
BICC	Business Intelligence Competency Center
CBPP	Commons-Based Peer Production
CRM	Customer Relationship Management
IaaS	Infraestructure as a Service
MDM	Master Data Management
OLAP	On-Line Analytical Processing
OS	Open Source
PaaS	Platform as a Service
ROI	Return On Investment
SaaS	Software as a Service
SW	Software
TCO	Total Cost Ownership

4. Bibliografía

"Gartner Magic Quadrant for BI Platforms, 2009". Gartner.
"Gartner Magic Quadrant for Data Integration, 2009". Gartner.
"Gartner Magic Quadrant for Data Warehouse, 2009." Gartner.
"Gartner Magic Quadrant for BI Platforms, 2010". Gartner.
"Gartner Magic Quadrant for Data Integration, 2010". Gartner.
"Gartner Magic Quadrant for Data Warehouse, 2010". Gartner.
"The Forrester Wave: BI, Q4 2009". Forrester Research.
"The Forrester Wave: Data Warehouse, Q4 2009". Forrester Research.

Capítulo VIII

Recursos relevantes en Business Intelligence

A continuación, se recopila alguna de las principales fuentes de información para estar al día del mercado Business Intelligence:

1. Portales

Los portales aúnan a los principales expertos del mercado con el objetivo de que ofrezcan su opinión y sus conocimientos a los lectores.

- BeyeNETWORK USA. El principal portal de expertos y noticias de Business Intelligence, donde escriben expertos de la talla de Bill Inmon, Claudia Imhoff o Seth Grimes. URL: http://www.beyenetwork.com.
- BeyeNETWORK Spain. Versión en castellano, lanzada en 2009, del principal portal de expertos y noticias de Business Intelligence. URL: http://www.beyenetwork.es.
- BI-SPAIN. Portal de noticias de Business Intelligence tanto de ámbito nacional como internacional. URL: http://www.bi-spain.com.
- BI-PRACTICES. Iniciativa conjunta entre BeyeNETWORK y TDWI para la recopilación de mejores prácticas por los expertos del sector. URL: http://www.bi-bestpractices.com.
- Dataprix. Portal en español que intenta aunar contenidos propios y de diferentes fuentes en un mismo marco con el objetivo de compartir conocimiento. URL: http://www.dataprix.com.
- AllTops Business Intelligence. Agregador de noticias de Business Intelligence de diferentes fuentes. URL: http://business-intelligence.alltop.com.

- Information Management (DM Review). Portal de Business Intelligence donde escriben expertos de todas las modalidades. URL: http://www.information-management.com.
- SmartData Collective. Comunidad de expertos en Business Intelligence y aplicaciones empresariales auspiciada por Teradata. URL: http://smartdatacollective.com.
- Information Management. Portal anteriormente llamado DM Review que ofrece artículos especializados para expertos. Edita también una publicación digital. URL: http://www.information-management.com.
- Intelligence Enterprise. Portal con artículos especializados de los principales expertos de Business Intelligence en Estados Unidos. URL: http://www.intelligententerprise.com.

2. Comunidades

Las comunidades buscan compartir experiencias, problemas, soluciones y conocimientos comunes. Se han seleccionado las siguientes:

- BeyeCONNECT. Comunidad de expertos que participan en BeyeNETWORK, como Claudia Imhoff, Bill Inmon, etc. URL: http://www.beyeconnect.com.
- Open Business Intelligence. Comunidad creada por TodoBI para el mundo Open Source Business Intelligence. URL: http://www.redopenbi.com.
- BI-LA. Comunidad Business Intelligence para Latinoamérica. URL: http://www.bi-la.com.
- SQL Server Central. Comunidad de desarrolladores de SQL Server. Con grupos dedicados al desarrollo de proyectos de data warehousing. URL: http://www.sqlservercentral.com.
- SQL Server Data Mining. Comunidad de desarrolladores de minería de datos con la solución de Microsoft. URL: http://www.sqlserverdatamining.com/ssdm.
- Data Quality Pro. Comunidad de profesionales interesados en la calidad de datos. URL: http://www.dataqualitypro.com.

3. Blogs

La blogosfera incluye gran cantidad de expertos que comparten su conocimiento a través de interesantes artículos. A destacar:

- TodoBI. Uno de los blogs más antiguos de BI en España. URL: http://www.todobi.blogspot.com.
- Information Management Blog. Blog personal de Josep Curto Díaz sobre Business Intelligence y tecnologías de la información. URL: http://informationmanagement.wordpress.com.
- Blog en BeyeNETWORK de Josep Curto Díaz, experto de canal Open Source Business Intelligence. URL: http://www.beyenetwork.es/blogs/curtodiaz.
- Blog en BeyeNETWORK de Claudia Imhoff, experta de Business Intelligence. URL: http://www.b-eye-network.com/blogs/imhoff.
- Sistemas Decisionales, algo más que Business Intelligence. Blog de Jorge Fernández, director Business Intelligence de ABAST Group. URL: http://sistemasdecisionales.blogspot.com.
- Business Intelligence fácil. Blog de Business Intelligence en castellano. URL: http://www.businessintelligence.info.
- Inteligencia de Negocio. Blog de Rémi Grossat. URL: http://www.intelineg.com.
- SQL Server Sí! Blog de Salvador Ramos, especialista Business Intelligence en tecnologías Microsoft. URL: http://www.sqlserversi.com.
- Richard Lees, experto en Microsoft BI Suite. Su web es muy interesante dado que muestra muchos ejemplos de las tecnologías. URL: http://richardlees.com.au/Pages/RichardsWelcomePage.aspx.
- Peter Thomas, experto en Business Intelligence. URL: http://peterthomas.wordpress.com.
- Business Intelligence News. Blog de Marcus Borba, experto en Business Intelligence que da su opinión sobre noticias del mercado. URL: http://mjfb-books.blogspot.com.
- James Dixon's Blog. Blog de James Dixon, CTO de Pentaho. URL: http://jamesdixon.wordpress.com
- The Open Book on BI. Blog de James Dixon, CEO of JasperSoft. URL: http://openbookonbi.blogspot.com.
- Wayne's World. Blog de Wayne W. Eckerson, director del TDWI. URL: http://tdwi.org/WaynesWorld.

- Julian Hyde. Blog de Julian Hyde, creador de Mondrian. URL: http://julianhyde.blogspot.com.
- Timo Elliot. Blog de Timo Elliot de SAP Business Objects en el que explica las presentaciones de producto que hace. URL: http://timoelliott.com.
- Chris Webb Blog. Blog del autor del libro MDX Solutions. Expero en OLAP. URL: http://cwebbbi.spaces.live.com.
- Visual Business Intelligence. Blog de Stephen Few. URL: http://www.perceptualedge.com/blog.
- El blog de José María Arce, uno de los principales expertos en inteligencia de negocio en España. URL: http://josemariaarce.blogspot.com/

4. Institutos

Existen ciertas entidades que desde hace años establecen guías, cursos y análisis para el desarrollo de proyectos de inteligencia de negocio.

- The Data Warehouse Institute (TDWI). Una de las principales instituciones del sector, especialistas en formaciones que realizan diversos estudios a lo largo del año. Está dirigida por Wayne Eckerson. URL: http://www.tdwi.org.
- The MDM Institute. Fundado con el objetivo de recopilar toda la información respecto MDM en el mercado. URL: http://www.tcdii.com/index.html.
- The Data Governance Institute. Afiliado con BeyeNETWORK, persigue recopilar prácticas de data governance. URL: http://www.datagovernance.com.

5. Másters

En el ámbito de España, cabe destacar:

- Máster en Business Intelligence. Universitat Oberta de Catalunya. Modalidad: Online. Duración: 2 años. Uno de los principales másters en

Business Intelligence de España. URL: http://www.uoc.edu/masters/cat/web/informatica_multimedia_telecomunicacio/business_intelligence.

6. Análisis de mercado

Existen múltiples empresas que se especializan en el análisis del estado del mercado Business Intelligence. Destacamos:

- BI Verdict. Anteriormente llamado OLAP Report, realiza un informe anual analizando todas las herramientas del mercado así como el estado de opinión de despliegue e implementación de proyectos BI en las organizaciones. URL: http://www.bi-verdict.com.
- Gartner. Empresa de análisis de mercado que dedica alguno de sus cuadrantes mágicos así como otros informes a la inteligencia de negocio. URL: http://www.gartner.com.
- Forrester Research. Empresa de análisis de mercado que dedica alguno de sus waves a la inteligencia de negocio. URL: http://www.forrester.com.
- IDC. Empresa de análisis de mercado que dedica una de sus áreas de la inteligencia de mercado y las soluciones analíticas. URL: http://www.idc.com

7. YouTube

YouTube es uno de los principales canales de comunicación y promoción de vídeos. En el contexto del Business Intelligence destacamos:

- Talend Channel: http://www.youtube.com/user/TalendChannel.
- Open Source BI Guru: http://www.youtube.com/cssmgt.
- QlikTech: http://www.youtube.com/user/QlikTech.
- Tableau Software: http://www.youtube.com/user/tableausoftware.
- SAP: http://www.youtube.com/user/saptv.
- Josep Curto: http://www.youtube.com/user/josepcurtodiaz.

8. Facebook

En Facebook, poco a poco, también están apareciendo algunas organizaciones y empresas:

- TDWI: http://www.facebook.com/datawarehouse.
- JasperSoft Corporation: http://www.facebook.com/pages/Jaspersoft-Corporation/78981369547?ref=mf.

9. Slideshare

Slideshare permite compartir presentaciones y documentos. Algunos expertos y organizaciones comparten algunas presentaciones realmente interesantes. Destacamos:

- TDWI: http://www.slideshare.net/tdwi.
- Mark Madsen: http://www.slideshare.net/mrm0.
- Darren Cunningham: http://www.slideshare.net/dcunni07.
- Josep Curto: http://www.slideshare.net/josep.curto.
- Timo Elliot: http://www.slideshare.net/timoelliott.
- Jos van Dongen: http://www.slideshare.net/jvdongen.

10. Twitter

Twitter es uno de los canales de social media que está creciendo de forma más rápida a razón de las interesantes conversaciones que se generan. Destacamos:

Empresas

- Teradata: http://twitter.com/Teradata.
- Jitterbit: http://twitter.com/jitterbit.

- Qlikview: http://twitter.com/QlikView.
- Eobjects: http://twitter.com/eobjects.
- JasperSoft: http://twitter.com/Jaspersoft.
- Pentaho: http://twitter.com/pentaho.
- IBM Cognos: http://twitter.com/ibmcognos.
- TDWI: http://twitter.com/TDWI.
- Tibco Spotfire: http://twitter.com/TibcoSpotfire.
- Infobright: http://twitter.com/infobright.
- Information Management (préviamente DMReview): http://twitter.com/infomgmt.
- Microstrategy: http://twitter.com/microstrategy.

Expertos

- Peter Thomas: http://twitter.com/PeterJThomas.
- Tod: http://twitter.com/TodmeansFox.
- Matt Assay: http://twitter.com/mjasay.
- Marcus Borba: http://twitter.com/marcusborba.
- James Dixon: http://twitter.com/jamespentaho.
- Pedro Alves: http://twitter.com/pmalves.
- Will Gorman: http://twitter.com/wpgorman.
- Matt Casters: http://twitter.com/mattcasters.
- Curt Monash: http://twitter.com/CurtMonash.
- Josep Curto: http://twitter.com/jcurto.
- Lance Walter: http://twitter.com/lancewalter.
- Diego Arenas: http://twitter.com/darenasc.
- Wayne Ekerson: http://twitter.com/weckerson.
- Mark Madsen: http://twitter.com/markmadsen.
- Julian Hyde: http://twitter.com/julianhyde.
- Seth Grimes: http://twitter.com/SethGrimes.
- Richard Hackathorn: http://twitter.com/hackathorn.
- Salvador Ramons: http://twitter.com/salvador_ramos.
- Mervyn Adrian: http://twitter.com/merv.

11. LinkedIn

Existen múltiples grupos en LinkedIn, pero por su relevancia cabe destacar:

- TDWI's Business Intelligence and Data Warehousing Discussion Group: http://www.linkedin.com/groups?home=&gid=45685.
- Gartner Business Intelligence (Xchange): http://www.linkedin.com/groups?home=&gid=1792113.
- Business Intelligence Professionals: http://www.linkedin.com/groups?about=&gid=40057.
- Business Intelligence Group: http://www.linkedin.com/groups?about=&gid=23006.
- Business Intelligence: http://www.linkedin.com/groups?about=&gid=22286.

12. Recursos

Los siguientes recursos resultan de utilidad en el momento de implementar un proyecto de inteligencia de negocio.

- KPI Library. Blog que recopila indicadores clave de negocio (KPI, Key Performance Indicador) por áreas de negocio y temáticas con el objetivo de facilitar la identificación de indicadores de rendimiento en proyectos de Business Intelligence. URL: http://www.kpilibrary.com.
- KDNuggets. Web que recopila toda la información existente sobre minería de datos. URL: http://www.kdnuggets.com.
- Ling Pipe. Web que recopila toda la información existente sobre NPL. URL: http://alias-i.com/lingpipe.
- Thearling. Web que recopila artículos e información sobre minería de datos. URL: http://www.thearling.com.
- Powercenter Templates at Perdue University. Web que recopila templates para procesos ETL con informática powercenter. URL: http://www.itap.purdue.edu/ea/standards/powermart.cfm.

13. Soluciones open source

A nivel de empresas Open Source Business Intelligence destacan las siguientes:

- Pentaho. Una de las soluciones completas líderes del mercado open source que integra ETL, reporting, OLAP, data mining y dashboards. URL: http://www.pentaho.com.
- JasperSoft. Una de las soluciones completas líderes del mercado open source que integra ETL, reporting, OLAP, data mining y dashboards. Comparte motor olap con Pentaho y su herramienta de ETL es la de Talend. URL: http://www.jaspersoft.com.
- LucidDB. Base de datos en columnas open source, óptima para análisis OLAP. URL: http://www.luciddb.org.
- SpagoBI. Una de las soluciones completas líderes del mercado open source que integra ETL, reporting, OLAP, data mining y dashboards. Se diferencia del resto en que sólo existe una versión Community y que es completamente modular. URL: http://www.spagoworld.org/ecm/faces/public/guest/home/solutions/spagobi.
- OpenReports. Solución que se basa en la integración de los tres motores de reporting open source existente y el motor OLAP Mondrian. URL: http://oreports.com.
- BeeProject. Una de las primeras soluciones completas que actualmente es un proyecto abandonado. URL: http://sourceforge.net/projects/bee.
- OpenI. Solución Business Intelligence basada en Mondrian. URL: http://www.openi.org.
- MonetDB. Base de datos en columnas open source, óptima para análisis OLAP. URL: http://monetdb.cwi.nl.
- Ingres. Base de datos relacional de gran escalabilidad y rendimiento. Ofrece appliances con JasperSoft, SpagoBI y Alfresco. URL: http://www.ingres.com.
- Infobright. Motor analítico de gran rendimiento para procesos de data warehousing. Integrada con MySQL. URL: http://www.infobright.com.
- Rapid Miner. Solución de minería de datos madura. Ofrece versión comercial y communitaria. URL: http://rapid-i.com.
- PMML (Predictive Model Markup Languaje). Es una markup language para el diseño de procesos estadísticos y de minería de datos. Usado por la gran mayoría de soluciones del mercado. URL: http://sourceforge.net/projects/pmml.

- Jitterbit. Solución EAI que permite la integración de datos y aplicaciones. URL: http://www.jitterbit.com.
- Teiid. Solución EAI que permite la integración de datos y aplicaciones. URL: http://www.jboss.org/teiid.
- Vainilla/BPM-Conseil. Suite Business Intelligence de origen francés que nace con el objetivo de suplir las carencias de Pentaho y que cubre las principales necesidades de un proyecto de inteligencia de negocio. URL: http://www.bpm-conseil.com.
- DataCleaner. Solución open source para la calidad de datos. URL: http://eobjects.org.
- Palo BI Suite. Solución open source para la gestión de Spreadmarts, planificación y presupuestación basada en un motor MOLAP. URL: http://www.jedox.com
- Octopus. Solución open source para el desarrollo de procesos ETL. URL: http://octopus.objectweb.org.
- Xineo. Solución open source para el desarrollo de procesos ETL. URL: http://sourceforge.net/projects/cb2xml.
- CloverETL. Solución open source para el desarrollo de procesos ETL. URL: http://www.cloveretl.com.
- BabelDoc. Solución open source para manipular flujos de datos. URL: http://sourceforge.net/projects/babeldoc.
- Joost. Solución open source para el desarrollo de procesos ETL sobre ficheros XML. URL: http://joost.sourceforge.net.
- jRubik. Cliente OLAP para Mondrian. URL: http://rubik.sourceforge.net.
- Talend. Versátil y potente solución open source para el desarrollo de procesos ETL que genera scripts en perl o java. También tiene productos de data quality y MDM. URL: http://www.talend.com.
- CB2XML. Solución open source para exportar ficheros XML a Cobol. URL: http://sourceforge.net/projects/cb2xml.
- Transmorpher. Solución open source para construir procesos de ETL en ficheros XLST. URL: http://transmorpher.gforge.inria.fr.
- Apatar. Solución open source para el desarrollo de procesos ETL. URL: http://apatar.com.
- BIRT. Solución completa open source para la creación de informes con capacidades de integración en cualquier aplicación J2EE. Auspiciado por Actuate y la fundación Eclipse. URL: http://www.eclipse.org/birt/phoenix.

- R-project. Solución completa estadística y de data mining proveniente del contexto universitario. Information Builders ha creado un módulo gráfico para incluirla en su paquete de soluciones. URL: http://www.r-project.org.
- Weka. Solución completa de data mining basada en algoritmos de aprendizaje automático procedente del contexto universitario. Ha sido comprada por Pentaho. URL: http://www.cs.waikato.ac.nz/ml/weka.
- KETL. Solución open source para el desarrollo de procesos ETL. URL: http://sourceforge.net/projects/ketl.
- MySQL. Base de datos open source recientemente adquirida por Oracle que puede ser usada en proyectos de Data Warehousing. URL: www.mysql.com.
- PostgreSQL. Base de datos open source que puede ser usada en proyecto de data warehousing. URL: www.postgresql.org.
- EnterpriseDB. Base de datos open source orientada a proyectos de data warehousing basada en PostgreSQL. URL: http://www.enterprisedb.com.
- InfoBright. Base de datos en columnas open source para usar en proyectos de data warehousing que se combina con MySQL como un nuevo motor de análisis. URL: http://www.infobright.com.
- GreenPlum. Base de datos open source especializada en procesos de data warehousing que permite el uso de tecnologías grid como Map Reduce. URL: http://www.greenplum.com.

14. Soluciones propietarias

Destacamos algunas de las principales empresas de inteligencia de negocio:

Suites BI tradicionales

- Information Builders. Plataforma de desarrollo de aplicaciones BI. También tienen una solución de integración sumamente potente con más de 300 conectores. URL: http://www.informationbuilders.com.
- IBM Cognos. IBM ha comprado Cognos y SPSS para incluir en su portafolio de productos una potente solución de inteligencia de negocio. URL: http://www-01.ibm.com/software/data/cognos.

- SAP. Ofrece dos soluciones de BI: Business Objects y Netweaver, que se hallan en un proceso de integración de roadmap. URL: http://wwww.sap.com.
- Microstrategy. Una de las pocas soluciones de BI que no han sido compradas. Destaca por su potente capa de elementos de análisis. No incluye una herramienta de ETL. Actualmente existe una versión gratuita de funcionalidades reducidas. URL: http://www.microstrategy.com.
- Oracle. Oracle ha comprado Hyperion y otras soluciones para tener una suite de productos de BI versátil y completa. URL: http://www.oracle.com.
- Panorama. Una de las empresas tradicionales del sector que frecuentemente hace productos innovadores que son comprados por otras empresas. URL: http://wwww.panorama.com.
- Apesoft. Empresa española que ofrece una suite flexible con un enfoque basado en Excel siguiendo un enfoque pragmático. URL: http://www.apesoft.com.
- Actuate. Solución completa de Business Intelligence propietaria que ofrece su motor de reporting BIRT en versión open source. URL: http://www.actuate.com.

Minería de datos

- SAS. Solución de minería de datos que incluye otros módulos que la convierten en una suite completa. URL: http://www.sas.com.
- Delta Miner. solución que incluye algoritmos de minería de datos en cuadros de mando e informes. URL: http://www.bissantz.com.
- Kxen. Solución de minería de datos orientada como un framework analítico. URL: http://www.kxen.com.

Mobile BI

- PushBI. Empresa orientada a soluciones de movilidad en el entorno de la inteligencia de negocio. URL: http://www.pushbi.com.
- RoamBI. Empresa orientada a soluciones de movilidad para inteligencia de negocio. URL: http://www.roambi.com.

ETL

- Informatica. Empresa con potentes soluciones de integración de datos así como de maestre data management. URL: http://www.informatica.com.
- Expressor. Solución para integración de datos mediante el uso de capa de metadatos. URL: http://www.expressor-software.com.

Data warehouse

- Teradata. Empresa que ofrece solución appliance (hardware + software) para realizar analytics. URL: http://www.teradata.com.
- Paraccel. Empresa que ofrece solución appliance para realizar analytics. URL: http://www.paraccel.com.
- Asterdata. Empresa que ofrece solución appliance para realizar analytics; además incluye el uso de map reduce. ULR: http://www.asterdata.com.
- Vertica. Empresa que ofrece solución appliance para realizar analytics. URL: http://www.vertica.com.
- Netezza. Empresa que ofrece solución appliance para realizar analytics. URL: http://www.netezza.com.
- Kickfire. Empresa que ofrece una appliance basada en open source, columnas y consultas en paralelo. URL: http://www.kickfire.com.
- Kognitio. Solución para data warehousing no basada en hardware que permite consultas masivas en paralelo, instalable en sistemas de IBM, HP u otros. URL: http://www.kognitio.com.
- I-lluminate. Empresa que ofrece una solución de data warehouse basada en la correlación de datos. http://www.i-lluminate.com.
- Dataupia. Ofrece una appliance para la implantación de un data warehouse. URL: http://www.dataupia.com.

Visualización

- Panopticon. Solución in-memory de análisis visual de grandes volúmenes de datos cercano a tiempo real. URL: http://www.panopticon.com.
- QlikView. Solución in-memory basada en AQL que proporciona un desarrollo ágil de informes y cuadros de mandos dinámicos. URL: http://www.qlikview.com.
- Tableau Software. Solución flexible orientada a crear elementos visuales de análisis para usuarios finales. URL: http://www.tableausoftware.com.
- Lyza. Solución orientada al usuario final multiplataforma que potencia el desarrollo colaborativo. URL: http://www.lyzasoft.com.
- Tibco Spotfire. Solución de inteligencia de negocio que destaca por sus capacidades de visualización. URL: http://spotfire.tibco.com.

SaaS

- GoodData. Business Intelligence en modalidad SaaS que hace foco en el aspecto colaborativo. URL: http://www.gooddata.com.

- BIRST. Business Intelligence en modalidad SaaS. URL: http://www.birst.com.
- LiteBI. Business Intelligence en modalidad SaaS basado en open source de origen español. URL: http://www.litebi.com

CEP/streaming de datos

- SQLStream. Motor para realizar streaming de datos cerca de tiempo real. URL: http://www.sqlstream.com.
- Progress Software. Ofrecen un motor de CEP con capacidad para extraer flujos en tiempo real e incorporarlos a un motor de eventos y establecer una monitorización. URL: http://www.progress.com.