

Diseño e implementación de una base de datos relacional para la gestión de la red de bibliotecas.

- Memoria

F. Jorge Blanca Sánchez
Ingeniería en Informática

Consultor: Josep Vallverdú Vergé

13 - Junio - 2010

Resumen

El presente proyecto pretende ser una síntesis de los conocimientos adquiridos a lo largo de los últimos años como alumno de la UOC aplicados en la simulación de un caso práctico real de un proyecto de implementación de una base de datos.

El foco en el cual me he centrado en los últimos años de estudio ha sido el de las bases de datos. Han sido especialmente valiosos para este proyecto los contenidos de las asignaturas de *Modelos Multidimensionales*, *Minería de Datos*, *Comercio Electrónico*, *Metodología y Gestión de Proyectos Informáticos* y, por su puesto, las asignaturas cursadas en primer ciclo: *Bases de Datos I* y *Bases de Datos II*.

El proyecto se inicia por la gestión y organización del mismo. Se plantean los requerimientos funcionales declarados por el usuario final y un plazo de entrega. Se determina de manera concreta el alcance del proyecto así como un calendario con los distintos hitos que deben cumplirse durante su ejecución. Esta es una parte muy importante del proyecto, pues, en caso de desacuerdo, tanto lo que está escrito, como la forma en que está escrito, así como, lo que no está escrito puede llegar a ser fundamental para que un tercero o, una autoridad judicial, pueda dar o quitar razones ante un conflicto interpretativo.

A continuación se hace referencia a la metodología empleada para un correcto seguimiento del proyecto de cara a garantizar su cumplimiento en las fechas pactadas. Se seguirá una metodología en cascada y se declara los hitos intermedios y un calendario que sirva para pronosticar los posibles desvíos que puedan originarse.

Finalmente, se desarrolla la parte técnica del proyecto desglosándose en dos apartados fundamentales: el desarrollo del sistema operacional y el desarrollo del sistema analítico y su correspondiente valoración económica.

Dedicatoria

A mi mujer.

Tabla de contenido

1.	Introducción.....	7
1.1.	Contexto	7
1.2.	Objetivos.....	9
1.3.	Enfoque y método seguido.....	10
1.3.1.	Metodología de desarrollo.....	10
1.3.2.	Seguimiento del proyecto	10
1.3.3.	Hitos principales.	10
1.4.	Planificación	11
1.4.1.	Instalación y configuración del software.....	11
1.4.2.	Formación en Cognos.....	11
1.4.3.	Análisis de la estructura de datos.	11
1.4.4.	Diseño de la nueva base de datos.....	11
1.4.5.	Creación de procedimientos para acceso a datos.....	11
1.4.6.	Construcción, carga y almacén de datos.	11
1.4.7.	Obtención de informes y gráficos.....	12
1.4.8.	Documentación final.	12
1.4.9.	Calendario	13
1.5.	Productos obtenidos.....	15
2.	Diseño conceptual. Modelo E-R	18
2.1.	Identificación de entidades.....	18
2.2.	Diagrama Entidad Relación.....	20
3.	Diseño lógico (modelo relacional)	21
3.1.	Definición de tablas y campos.....	21
3.2.	Esquema de las tablas de la Base de Datos.....	27
3.3.	Controles de integridad	28
3.4.	Índices	28
3.5.	Procedimientos.....	28
3.6.	Diseño del sistema de actividad (log).....	32
4.	Implementación.....	33
4.1.	Plataforma de trabajo	33
4.2.	Aspectos generales de la implementación.....	33
4.2.1.	Código estructurado	33
4.2.2.	Tipos de variable: %type	33
4.2.3.	Parámetros de los procedimientos.....	33
4.2.4.	Campos auto-incrementales	34
4.2.5.	Control de excepciones	34
4.2.6.	Valores del resultado de salida de los procedimientos: <i>pRC</i>	34
4.2.7.	Sistema de registro LOG	34
4.2.8.	Guiones	35
4.3.	Guión de creación de tabla	37

4.4.	Guión de creación de procedimientos.....	38
4.5.	Pruebas	39
5.	Diseño analítico.....	42
5.1.	Flujo de trabajo con las herramientas Cognos.....	43
5.2.	Conceptos de modelado dimensional	44
5.3.	Requerimientos	44
5.4.	Generación de informes. Recomendación de documentos	45
5.5.	Diagramas del modelo conceptual	48
5.6.	Modificaciones en el modelo operacional	52
5.7.	Preparación del plan de pruebas.	52
5.8.	Proceso de integración y transformación. ETL	53
5.9.	Implementación con Cognos.....	54
5.9.1.	Creación de catálogos con Impromptu.....	54
5.9.2.	Definición de informes con Impromptu.....	56
5.9.3.	Definición de cubos OLAP con Transformer	59
5.9.4.	Representación de los datos con PowerPlay.....	61
6.	Valoración económica.....	68
	<i>Recursos Técnicos</i>	68
	<i>Otros recursos</i>	69
	<i>Presupuesto Total</i>	69
7.	Conclusiones.....	70
	Bibliografía.....	71

Tabla de ilustraciones

Diagrama de Gantt	14
Diagrama Entidad Relación.....	20
Diagrama lógico.....	27
Primeras 50 filas de la tabla TBLOG.....	39
Detalle de las tablas después de su creación.....	40
Flujo de trabajo con las herramientas Cognos.....	43
Informe de Impromptu	47
Diagrama conceptual 1	48
Diagrama conceptual 2	49
Diagrama conceptual 3	50
Diagrama conceptual 4	51
Definición del catálogo Impromptu.....	54
Definición del catálogo Impromptu.....	55
Definición Query Impromptu.....	56
Definición Query Impromptu.....	57
Definición Query Impromptu.....	58
Definición PowerCube Transformer	59
Definición PowerCube Transformer	60
Tabla 1 PowerPlay Préstamos.....	61
Tabla 2 PowerPlay Préstamos.....	62
Tabla 3 PowerPlay Préstamos.....	62
Gráfico 1 PowerPlay Préstamos	63
Tabla 1 PowerPlay Autores.....	64
Gráfico 1 PowerPlay Autores	64
Tabla 1 PowerPlay Afluencia	65
Gráfico 1 PowerPlay Afluencia.....	65
Tabla 1 PowerPlay Vida de Documentos.....	66
Gráfico 1 PowerPlay Vida de Documentos	66
Tabla 1 PowerPlay Estado Documentos.....	67
Gráfico 1 PowerPlay Estado Documentos	67

1. Introducción

1.1. Contexto

AllTogether, empresa dedicada a la construcción y distribución de software, nos propone un proyecto de implementación de una base de datos para una red de bibliotecas y nos expone su situación actual.

La empresa AllTogether se propone realizar una aplicación enfocada en tres áreas: inventario de recursos gestionados por la biblioteca, registro de usuarios, y por último préstamos y devoluciones de recursos. Tras un par de jornadas de conversaciones con los técnicos, se recogen las siguientes especificaciones:

La biblioteca gestiona recursos bibliográficos, audiovisuales y electrónicos materializados sobre cualquier tipo de soporte. Debemos saber cuál es el material que da soporte a cada recurso, por ejemplo, libros, cintas de audio, películas, elementos multimedia, mapas, videos y otros.

Por otra parte, los recursos de la biblioteca se dividen en colecciones (Novela, Ficción, Ensayo, Juvenil, Periódicos, Vídeos). Que un recurso pertenezca a una colección determina el período máximo de préstamo y la penalización por cada día de retraso en su devolución.

Los recursos bibliográficos constituyen una de las entidades principales de la aplicación. Como hemos visto, cada recurso pertenece a una colección y es de un soporte de material específico. Cualquier recurso tiene los siguientes atributos: título, autores, declaración de responsabilidad de los contenidos.

Los ejemplares de los recursos bibliográficos constituyen los objetos de préstamo de la biblioteca. Uno de los libros de "War and Peace" de la biblioteca es un ejemplar del recurso "War and Peace" en soporte "Libro" de la colección "Novela", de la misma manera que un compact disc de "Thriller" es una copia del recurso del mismo nombre en formato CD. Cada ejemplar se identifica por un código de barras único que lo diferencia de cualquier otro ejemplar de la biblioteca.

Un ejemplar puede estar en diferentes estados (Disponible, Prestado, Perdido, Dañado, Pedido, etc.). En función de su estado será posible su préstamo. Debe guardarse un historial de todos los estados por los que ha pasado un ejemplar a lo largo de su historia.

Los socios son los usuarios de la biblioteca. Cada miembro se identifica de manera única por un código de barras que está dibujado en su carnet de socio. De cada miembro se dispone de sus datos personales: nombre, dirección, teléfono, dirección electrónica.

La biblioteca clasifica sus socios en diferentes perfiles. Por lo general, estos perfiles son "Juvenil" y "Adulto", aunque como es lógico, de una biblioteca a otra puede haber más perfiles de clasificación. Cada perfil tiene un límite de penalizaciones permitidas. Al alcanzar ese límite el socio pierde cualquier posibilidad de solicitar ningún recurso en préstamo.

En base al perfil de clasificación del socio de la biblioteca y al material que da soporte a un recurso se establece el tiempo límite de préstamo y el número de renovaciones permitidos. Por ejemplo, cualquier vídeo solicitado por un socio juvenil se podrá ser prestado por un periodo máximo de 10 días sin posibilidad de renovación.

Cuando no queda disponible ningún ejemplar de un recurso determinado y éste es solicitado en préstamo, es posible realizar una reserva sobre él. Cuando el ejemplar ha sido reservado, el usuario que lo tiene en préstamo pierde la posibilidad de renovarlo y se ve en la obligación de devolverlo en la fecha pactada.

Se deben registrar los pagos de cuotas o penalizaciones que realicen los socios por medio de los distintos tipos de transacción aceptados por la biblioteca (efectivo, tarjeta de crédito, etc.) anotando el importe, la fecha, la descripción y el tipo de transacción.

Por último, se mantendrá un fichero con los empleados de la biblioteca que deben manejar esta aplicación informática. En este registro contendrá los códigos de usuario y contraseña para autenticarse en el aplicativo. Además, tendrá la información para autorizar en los distintos de la aplicación en los que el empleado podrá actuar. Por otra parte, cuando un empleado haga una anotación en las tablas de recursos, socios, usuarios o pagos debe quedar reflejado.

La aplicación está pensada para una red de bibliotecas. Por cada biblioteca se guarda: el nombre, una imagen con el logo, el horario, el teléfono y la dirección web de internet. Los ejemplares de los recursos bibliográficos están asignados a una biblioteca. Sin embargo los socios lo son de la red de bibliotecas y pueden usar los servicios de cualquiera de ellas.

AllTogether nos plantea que su personal informático está especializado en servidores de aplicaciones IBM Websphere Application Server y Apache Tomcat. Es decir, son expertos conocedores Java. Con respecto a las bases de datos, observan que la mayoría de sus clientes utilizan Oracle, y por las prestaciones que presta deciden utilizar Oracle.

Además el nuevo cliente le ha solicitado que sea posible realizar un la explotación de un almacén de datos para el análisis estadístico de los datos contenidos en sus ficheros. Por ejemplo desean saber las fechas de más solicitud de préstamos por parte del público, vida de documentos, pérdidas de libros, etc.

1.2. Objetivos

El objetivo del proyecto es implementar una base de datos para que pueda ser integrada en un aplicativo de gestión de una red de bibliotecas que está desarrollando la empresa AllTogether, SL.

Aparte de satisfacer los requerimientos de nuestro contratante, la gestión del acceso a la información se realizará en su totalidad por medio de procedimientos almacenados de base de datos, esto facilitará la integración de la base de datos al aplicativo desarrollado por nuestro cliente.

Además, para poder auditar posibles conflictos con AllTogheter o con sus cliente, dejaremos una huella de todas las operaciones que se realicen en la base de datos de forma que quede registrado en un fichero de diario o 'log' cada uno de los accesos de escritura que se realice a la base de datos.

Otro objetivo es realizar un sistema capaz de obtener de forma eficiente de aquellos elementos valiosos de toma de decisiones. Es decir hacer un uso adecuado de los datos para obtener información útil para el soporte de la toma de decisiones, lo cua resulta difícil lograr con los sistemas operacionales.

Las partes en que se divide el proyecto son:

- Análisis de los requerimientos, modelo conceptual y modelo lógico relacional.
- Crear el modelo físico para la nueva base de datos en Oracle.
- Planificación de un plan de pruebas que garantice el correcto funcionamiento de los procedimientos realizados.
- Construcción y carga de un almacén de datos (Datawarehouse). Creación de informes de análisis de los datos. Para ello se utilizarán las herramientas Cognos v 7.1 de IBM.

Análisis de riesgos

Fecha de entrega: La entrega final del proyecto es 13 de junio de 2010, lo cual significa que cualquier desviación o imprevisto se deberá resolver sabiendo que la fecha de entrega es inamovible.

Título	Enunciado	Entrega
PAC 1 (PT)	25/02/2010	21/03/2010
PAC 2	25/02/2010	18/04/2010
PAC 3	25/02/2010	16/05/2010
Entrega final (memoria+producto+presentación)	25/02/2010	13/06/2010

1.3. Enfoque y método seguido

1.3.1. Metodología de desarrollo.

Por las características del presente proyecto donde los requerimientos están perfectamente definidos y con apenas ningún factor de riesgo, me inclino por seguir una metodología en cascada.

Esta metodología consiste en un proceso secuencial en el que los pasos de desarrollo son vistos hacia abajo (como una cascada de agua) a través de las fases de análisis de requerimientos, diseño implementación, pruebas e integración.

El proyecto se divide en dos fases: una de análisis, diseño e implementación del sistema operacional y otra para el sistema analítico.

1.3.2. Seguimiento del proyecto

Un comité de seguimiento del proyecto será responsable de de la gestión del proyecto y garantizará la coherencia global. Estará compuesto por el director de proyecto de nuestra empresa y el responsable de desarrollo de AllTogether. Se reunirán quincenalmente, aunque se puede convocar con carácter de urgencia a petición de cualquiera de los componentes.

El proyecto se ha desarrollado sin problemas siguiendo la planificación inicial.

A la fecha prevista, 13 junio, se hace entrega de la última entrega con la que se da por finalizado el presente proyecto.

No se han producido incidentes que merezcan comentarse.

1.3.3. Hitos principales.

- **Inicio del proyecto**
- **Entrega del plan de proyecto.**
El presente documento.
- **Informe final de implementación del sistema operacional**
- **Informe final de implementación del sistema analítico (datawarehouse)**
- **Entrega final del proyecto**

1.4. Planificación

Las actividades que se van a realizar en este proyecto son las siguientes:

1.4.1. Instalación y configuración del software.

Oracle Express v10.2.0.1 Suministrado por UOC.

<http://www.oracle.com/us/products/database/index.htm>

Versión reducida del fsmodo y divulgsdo gestor de base de datos Oracle.

Oracle SQL Developer v1.0.0.15 Suministrado por UOC.

http://www.oracle.com/technology/products/database/sql_developer/index.html

SQL Developer es una herramienta libre y totalmente gráfica para el desarrollo de bases de datos. Puede explorar objetos de base de datos, ejecutar sentencias SQL y secuencias de comandos SQL, y editar y depurar PL / SQL.

Cognos 7.1. Suministrado por UOC <http://www-01.ibm.com/software/data/cognos/>

IBM Cognos es de una herramienta Business Intelligence líder en el mercado junto con Business Objects (SAP) y Hyperion (Oracle).

1.4.2. Formación en Cognos.

Cognos <http://www-01.ibm.com/support/docview.wss?uid=swg27014591>

<https://merlin.state.ms.us/Merlin/MERLINAn.nsf/ae447c6ec3083152862568aa00549d72/6b96ab2fca9c6a41862574730058be83?OpenDocument>

Cognos. Este es un producto Business Intelligence (BI) que proporciona una gama completa de capacidades de reporting, análisis, indicadores, cuadros de mando, gestión de eventos de negocio, además de integración de dadtos en una única estructura.

La propia herramienta ya incorpora abundante documentación en forma de ficheros PDF. Consiste en guías de referencia de los distintos productos (Impromptu, Transformer y PowerPlay)

1.4.3. Análisis de la estructura de datos.

Realizaremos un análisis de la base de datos que se va a implementar a partir de las especificaciones obtenidas.

1.4.4. Diseño de la nueva base de datos.

Esta tarea incluye la generación de un script de generación de la base de datos y de todos los objetos necesarios.

1.4.5. Creación de procedimientos para acceso a datos.

En esta apartado se realizarán todos los pasos necesarios para el acceso a los datos, tanto altas, bajas como modificaciones de datos. Se crearán los procedimientos almacenados, disparadores, registros de login y todos los elementos requeridos para mantener los datos según las especificaciones de nuestro contratante. Se realizará un script que sirva para realizar un plan de pruebas que sirva para validar el correcto funcionamiento de los objetos de base de datos creados.

1.4.6. Construcción, carga y almacén de datos.

En esta tarea se describirá el modelo multidimensional conceptual que sirva de base para el diseño y construcción del sistema de información. Este modelo estará compuesto por dimensiones de análisis (y los atributos correspondientes) y hechos (conjunto de indicadores con sus dimensiones relacionadas)

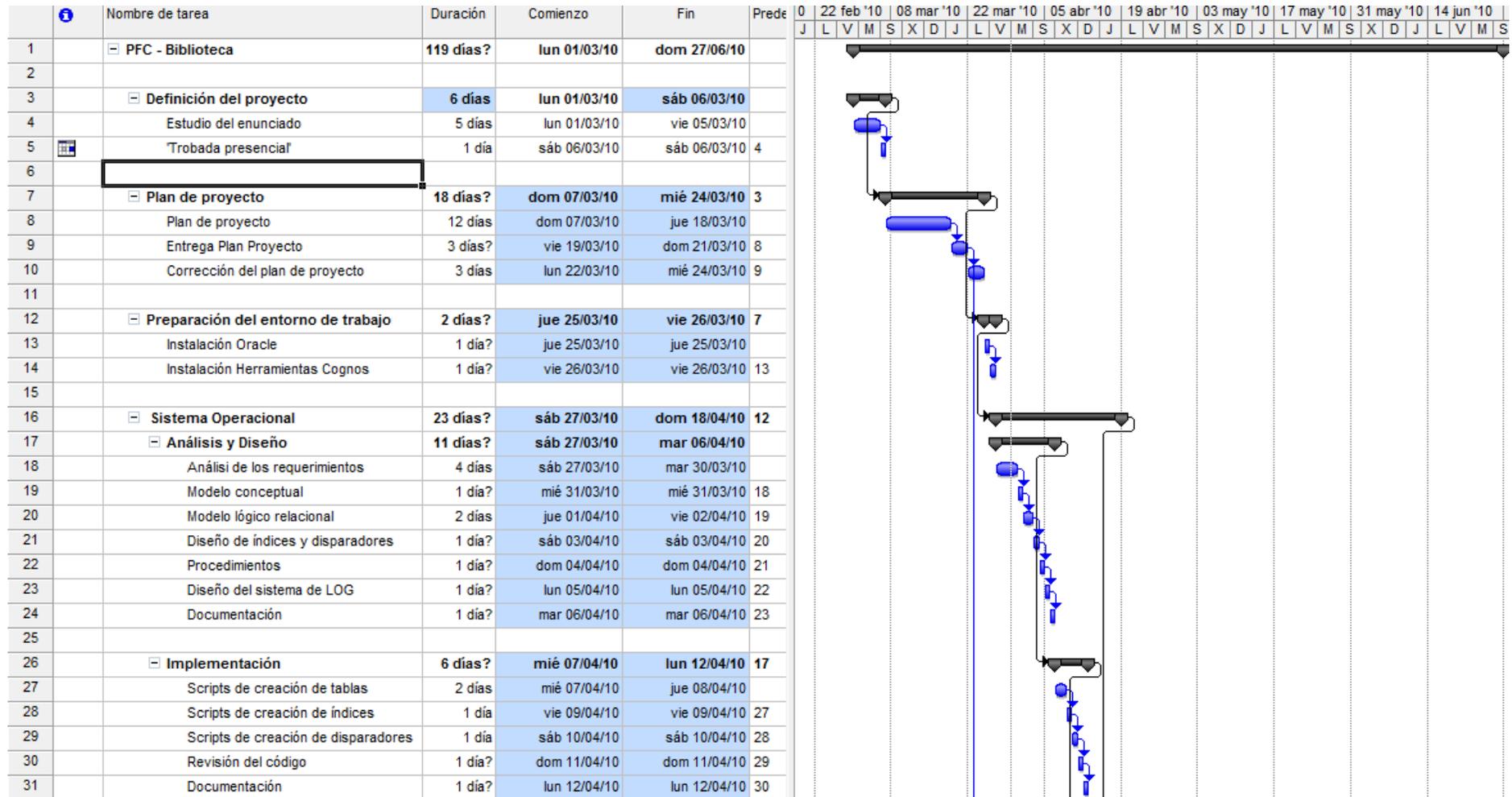
1.4.7. Obtención de informes y gráficos.

Se crearán los informes de análisis según hipotéticos requerimientos de usuarios, puesto que finalmente serán ellos los que realizarán los suyos propios. Para ello utilizaré las tres herramientas Cognos mencionadas anteriormente (Impromptu, Transformer y PowerPlay).

1.4.8. Documentación final.

Esta tarea consiste en la unificación de la documentación realizada durante el desarrollo del proyecto y obtener la memoria. También se realizará una presentación de diapositivas en formato PowerPoint.

1.4.9. Calendario



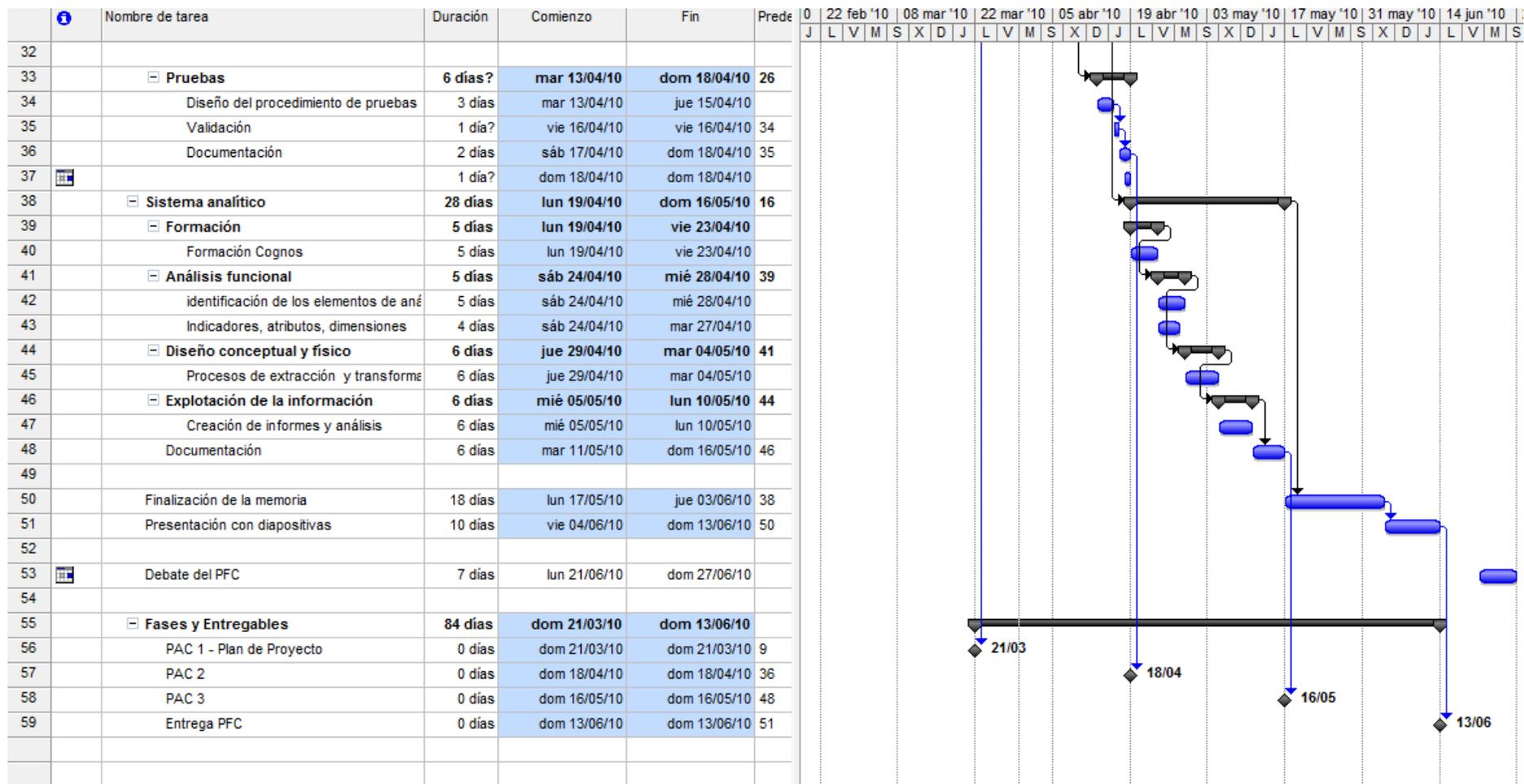


Ilustración 1: Diagrama de Gantt

1.5. Productos obtenidos

GUIÓN 0. Borrado de los objetos de la base de datos.

Este gui3n borra todos los objetos definidos en este proyecto. No borra el usuario.

GUIÓN 1. Creaci3n del usuario administrador.

Este gui3n debe ejecutarse con un usuario administrador y nos crear3 un usuario y un esquema donde ubicaremos todos los objetos de nuestra base de datos.

GUIÓN 2. Creaci3n de secuencias, tablas e 3ndices

Este gui3n se ejecutar3 con el usuario creado en el p3rrafo anterior (UOC) el cual tiene permisos totales sobre la base de datos.

Las funciones que realiza este script son las siguientes:

- Creaci3n de secuencias
- Creaci3n de tablas e 3ndices
- Inserci3n de registros

GUIÓN 3. Creaci3n de funciones y procedimientos.

Este gui3n se ejecutar3 con el usuario creado en el p3rrafo anterior (UOC) el cual tiene permisos totales sobre la base de datos.

Las funciones que realiza este script son las siguientes:

- Inserci3n de registros
- Creaci3n de funciones y procedimientos de car3cter general
- Creaci3n de procedimientos espec3ficos de esta aplicaci3n

GUIÓN 4. Juego de pruebas

Contiene las llamadas a los procedimientos que hemos creado.

Inicialmente se hacen llamadas con par3metros correctos de cara a llenar de datos nuestra base de datos.

Una vez introducidos los datos correctamente se comprobar3 que las funciones fallan de la manera que se espera cuando le pasamos incorrectamente los par3metros. Este procedimiento va escribiendo l3neas de comentario por una salida est3ndar.

GUIÓN 0-ETL-drop.SQL. Eliminaci3n de tablas de hechos y dimensiones

Este procedimiento borra todas las tablas de hechos y dimensiones y las correspondientes secuencias.

GUIÓN 6. ETL

El gui3n *GUIÓN6-ETL.sql* ejecuta el proceso de borrado y creaci3n de las tablas de dimensiones para el sistema anal3tico. Asimismo, este procedimiento tambi3n realiza la transformaci3n de los datos partiendo del sistema operacional, cumplimentando las tablas para su posterior proceso en el sistema multidimensional.

La primera vez que se ejecuta este procedimiento, se deben comentar las instrucciones DROP de tablas si se quiere evitar que den error, dado que no existen.

Ficheros Cognos

Los ficheros que vienen a continuación son objetos de los productos Cognos. Para que funcionen correctamente debe existir una definición de conexión ODBC denominada '**Biblioteca**' con la base de datos creada en los pasos anteriores.

Por otra parte, estos ficheros de configuración se relacionan entre ellos y deben estar ubicados en el directorio: **C:\Users\Jordi\Cognos**

FICHEROS *.CAT (PFC-ETL-Ejemplar, PFC-ETL-Prestamo, PFC-Operacional)

Contienen la definición del catálogo.

FICHEROS *.IMR

(PFC-Recomendacion, PFC-1-Documentos, PFC-2-Autores, PFC-3-Ejemplares)

Contienen los proyectos Impromptu.

FICHEROS *.IQD (PFC-1-Documentos, PFC-2-Autores, PFC-3-Ejemplares)

Ficheros exportados desde Impromptu en formato *Query Definition File*.

FICHEROS *.PYI

(PFC-1-Documentación, PFC-2-Autor, PFC-3-Ejemplar)

Contienen los proyectos Transformer.

FICHEROS *.MDC

(PFC-1-Documentación, PFC-2-Autor, PFC-3-Ejemplar)

Ficheros de tipo PowerCube generados desde Transformer.

FICHEROS *.PPR

(PFC-1-Documentación, PFC-2-Autor, PFC-3-Ejemplar)

Contienen los proyectos PowerPlay.

MODELO OPERACIONAL

2. Diseño conceptual. Modelo E-R

2.1. Identificación de entidades

Según la descripción de los requerimientos vistos anteriormente, ya se pueden identificar las entidades. A modo informativo, se describen los campos de la entidad para ampliar la información y ofrecer una idea más clara de la entidad. Más adelante, en la definición de tablas y campos, se indicará en detalle los campos definitivos en cada entidad.

Biblioteca	Identifica cada una de las bibliotecas de la red de bibliotecas. Nombre, dirección, teléfono, horario, URL de Internet, logo.
Personal	Datos de los empleados de las bibliotecas que pueden interactuar con el aplicativo. Datos personales, código de usuario, función hash de la contraseña, autorizaciones a las distintas partes de la aplicación: administración, gestión del catálogo, gestión de préstamos, solicitud de informes.
Colección	Los recursos que gestionan las bibliotecas se gestionan agrupándolos en colecciones. Esta agrupación es tanto administrativa como espacial en el recinto de la propia biblioteca. A continuación se relacionan algunas colecciones posibles: Ficción, Novela, Historia, Juvenil, Software de ordenador, Vídeos y DVDs, Periódicos, Compact Discs, etc. Dependiendo de la colección a la que pertenece un recurso de la biblioteca, se establece el máximo de días de préstamo y la multa diaria por retraso en la devolución.
Material	La biblioteca gestiona productos en diferentes soportes o materiales, por ejemplo: libros, revistas, mapas, vídeos y DVDs, CDs de audio, CDs de software. En función del material se establece el límite teórico de préstamos de un recurso y el límite de renovaciones de préstamo.
Recurso	Esta es una entidad intangible de los recursos de la biblioteca. Las instancias de los recursos son la materialización en libros, CDs, DVDs y corresponden a otra entidad. Cada recurso se identifica por un título, autor, editorial, ISBN, etc. Cada recurso pertenece a una colección y se materializa en un soporte determinado.
Ejemplar	Cada uno de los recursos de la biblioteca se materializa en los ejemplares de la biblioteca. Es el objeto del préstamo. Se caracteriza por un código de barras interno que es único por cada ejemplar. Cada ejemplar está asociado a una única biblioteca.
Socio	Los socios no están relacionados a una biblioteca sino que lo son de cualquiera de las bibliotecas de la red. A los socios se les suministra un carnet con un código de barras que lo identifica. De los socios se registra sus datos personales, teléfono, email.
TipoSocio	Los socios se clasifican por lo general en dos categorías, aunque en el futuro podrían ser más. Estas categorías son Juvenil y Adulto. En función de la categoría se establece el número máximo de multas permitidas hasta proceder a la inhabilitación de préstamos.

Limite	En función del tipo de socio y del material en que se soporta el recurso se establecen un número máximo de préstamos y un número máximo de renovaciones. De esta forma, por ejemplo, se puede no permitir que se haga una renovación de un préstamo a socios juveniles que pidan vídeos en préstamo.
Estado	Información sobre la situación en que se encuentra un recurso: Prestado, Devuelto, Dañado, Perdido,
HistoricoEstado	Establece el estado de un recurso durante un intervalo de fechas.
Préstamo	Esta entidad se corresponde al compromiso entre los usuarios de la biblioteca y los recursos que son objeto de préstamo, es decir, es el acto por el cual la biblioteca deja en préstamo uno de sus recursos a uno de sus socios. Los principales atributos de este compromiso son la fecha del préstamo y la fecha de devolución. Para que se pueda prestar un recurso, éste debe estar en estado disponible, no estar en Reserva por otro socio y que el socio esté habilitado para poderlo solicitar. Al vencimiento del periodo de préstamo, el usuario podrá pedir una renovación que se le podrá conceder si no sobrepasa el límite de renovaciones permitido y si el recurso no ha sido reservado por otro socio. En esta entidad se registra la fecha prevista de devolución y el número de renovaciones realizadas por el usuario sobre este recurso.
Transacción	Conjunto de actividades que supongan una transacción económica entre socios y la biblioteca como pago de cuotas, multas, cuotas o servicios.
Pagos	Representa un pago realizado por un socio, en una fecha determinada, por un importe concreto.

2.2. Diagrama Entidad Relación

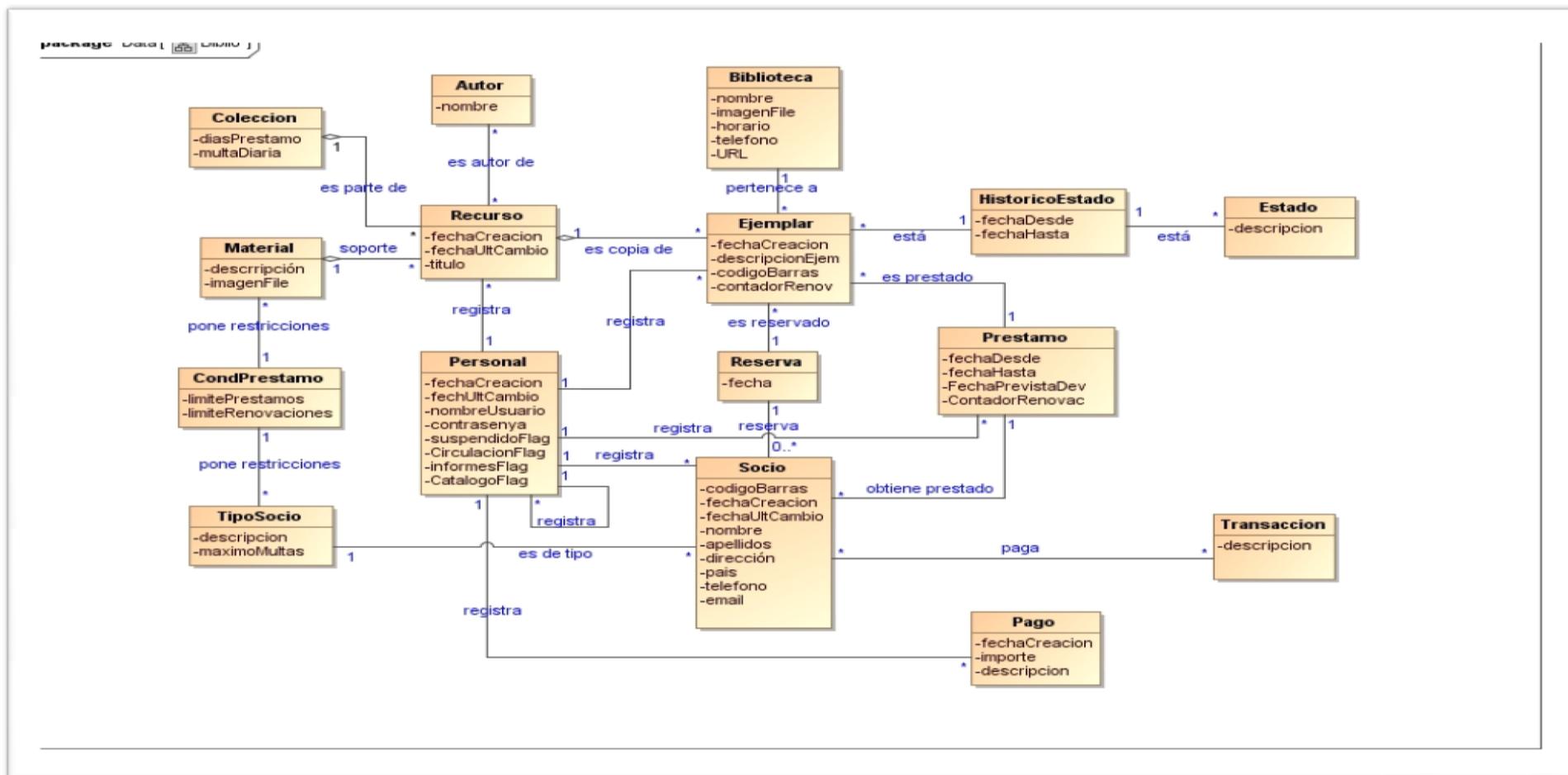


Ilustración 2: Diagrama Entidad Relación

3. Diseño lógico (modelo relacional)

Una vez definido el modelo conceptual estamos en condiciones de pasar al modelo relacional.

3.1. Definición de tablas y campos

TABLA Biblioteca

Descripción: Datos de cada una de las bibliotecas que componen la red de bibliotecas

- ***bibliotecald*** : [Clave Primaria] ; Campo entero auto-incremental, clave artificial para cada una de las bibliotecas.
- ***nombre***: Campo de texto. Nombre de la biblioteca.
- ***direccion*** : Campo de texto. Dirección de la biblioteca.
- ***telefono*** : Campo de texto. Teléfono.
- ***correoElectronico***: Campo de texto. Dirección de correo electrónico.
- ***URL***: Campo de texto; Dirección oficial de la web de la biblioteca.

TABLA Coleccion

Descripción: Distribución temática en que se distribuyen los recursos de una biblioteca.

- ***coleccionId*** : [Clave Primaria] ; Campo entero auto-incremental, clave artificial para cada una de las colecciones.
- ***descripcion***: Campo de texto. Descripción de la colección.
- ***diasPrestamo***: Campo numérico entero. Establece el número de días que un recurso perteneciente a esta colección puede ser prestado a un socio.
- ***multaDiaria***: Campo numérico decimal. Indica la multa por día de retraso en la devolución de un recurso prestado.

TABLA Material

Descripción: Identifica el soporte en que está realizado cada uno de los recursos de la biblioteca, por ejemplo, libros, revistas, mapas, DVD, CD.

- ***materialId***: [Clave Primaria]. Campo entero auto-incremental, clave artificial para cada uno de los materiales.
- ***descripcion***: Campo de texto. Descripción del material.
- ***imagenFile***: Campo de texto. URL de un recurso de red que contiene una imagen tipo jpg que identifica visualmente un material determinado.

TABLA Empleado

Descripción: Representa los empleados que pueden acceder al aplicativo.

- ***empleadold***: [Clave Primaria]; Campo entero auto-incremental, clave artificial para cada uno de los empleados.
- ***codigoUsuario***: [Clave Única]. Campo de texto. Código de usuario del empleado para acceder al sistema informático.
- ***contrasena***: Campo de texto. Función hash de la contraseña del empleado.
- ***apellidos***: Campo de texto. Apellidos del empleado.
- ***nombre***: Campo de texto. Nombre del empleado.

- **bibliotecald:** [Clave Foránea, tabla Biblioteca]. Representa la biblioteca a la que está asignado el empleado.
- **fechaAlta:** Campo de fecha. Indica la fecha de alta del empleado en la aplicación
- **fechaBaja:** Campo de fecha. Indica la fecha de baja del empleado en la aplicación
- **suspendidoFlag:** Campo de texto. Campo binario [S/N] que deshabilita/habilita al empleado a interactuar con la aplicación
- **administradorFlag:** Campo de texto. Campo binario [S/N] que habilita/deshabilita al empleado a interactuar como administrador con la aplicación, por ejemplo, alta de empleados.
- **catalogoFlag:** Campo de texto. Campo binario [S/N] que habilita/deshabilita al empleado a interactuar con la parte de la aplicación correspondiente al catálogo.
- **prestamoFlag:** Campo de texto. Campo binario [S/N] que habilita/deshabilita al empleado a interactuar con la parte de la aplicación correspondiente a la gestión de préstamos.
- **informeFlag:** Campo de texto. Campo binario [S/N] que habilita/deshabilita al empleado a interactuar con la parte de la aplicación correspondiente a informes.
- **ultimaModFecha:** Campo de fecha. Indica la fecha en que se produjo la última modificación de los datos de este empleado
- **ultimaModEmpl:** [Clave Foránea, tabla Empleado]. Indica el empleado que realizó la última modificación de los datos de este empleado. Es una clave recursiva.

TABLA Autor

Descripción: Representa los autores de la obras contenidas en las bibliotecas.

- **autorId:** [Clave Primaria]; Campo entero auto-incremental, clave artificial para cada uno de los autores.
- **nombre:** Campo de texto. Nombre completo del autor.

TABLA Recurso

Descripción: Representa las obras contenidas en la red de bibliotecas, pero no su materialización en las distintas copias de libros, Vídeos, CDs, etc., las cuales están representadas por otra entidad.

- **recursold:** [Clave Primaria]. Campo entero auto-incremental, clave artificial para cada uno de los recursos.
- **título:** Campo de texto. Título de la obra
- **subtítulo:** Campo de texto. Subtítulo de la obra
- **materialId:** [Clave Foránea, tabla Material]. Indica el soporte en que se materializa este recurso (libro, CD, vídeo).
- **fechaAlta:** Campo de fecha. Indica la fecha de alta del recurso en la aplicación.
- **fechaBaja:** Campo de fecha. Indica la fecha de baja del recurso en la aplicación.
- **ultimaModFecha:** Campo de fecha. Indica la fecha en que se produjo la última modificación de los datos de este recurso.
- **ultimaModEmpl:** [Clave Foránea, tabla Empleado]. Indica el empleado que realizó la última modificación de los datos de este recurso.

TABLA Autor

Descripción: Representa los autores de las obras contenidas en las bibliotecas.

- **autorId:** [Clave Primaria]; Campo entero auto-incremental, clave artificial para cada uno de los autores.
- **nombre:** Campo de texto. Nombre completo del autor.

TABLA RecursoAutor

Descripción: tabla auxiliar para representar la relación *N a N* entre autores y recursos.

- **recursold:** [Clave Primaria] [Clave Foránea, tabla Recurso]; Identificador del recurso
- **autorId:** [Clave Primaria] [Clave Foránea, tabla Autor]; Identificador del autor

TABLA Ejemplar

Descripción: Representa cada uno de los objetos susceptibles de préstamo de la biblioteca: libros, películas, revistas.

- **ejemplarId:** [Clave Primaria]. Campo entero auto-incremental, clave artificial para cada uno de los ejemplares.
- **bibliotecald:** [Clave Foránea, tabla Biblioteca]. Identificador de la biblioteca propietaria de este ejemplar.
- **recursold:** [Clave Foránea, tabla Recurso]. Identificador del recurso, por ejemplo si se trata de un ejemplar del libro "*Guerra y Paz*", este campo contiene el identificador de este recurso.
- **descripcion:** Campo de texto. Cualquier comentario sobre este ejemplar.
- **códigoBarras:** Campo numérico entero. Código de barras adherido al ejemplar que lo identifica de forma única.
- **ContadorRenovac:** Campo numérico entero. Contador de renovaciones que ha tenido el ejemplar.
- **fechaAlta:** Campo de fecha. Indica la fecha de alta de ejemplar en la aplicación
- **fechaBaja:** Campo de fecha. Indica la fecha de baja del ejemplar en la aplicación
- **ultimaModFecha:** Campo de fecha. Indica la fecha en que se produjo la última modificación de los datos de este ejemplar.
- **ultimaModEmpl:** [Clave Foránea, tabla Empleado]. Indica el empleado que realizó la última modificación de los datos de este ejemplar.

TABLA Estado

Descripción: Representa los distintos estados en los que se puede encontrar un ejemplar de la biblioteca: prestado, perdido, disponible, dañado, pedido a editorial, etc.

- **estadold:** [Clave Primaria]; Campo de texto, abreviación de los distintos estados.
- **descripcion:** Campo de texto. Descripción del estado

TABLA HistoricoEstado

Descripción: distintos estados por los que ha pasado un ejemplar.

- ***ejemplarId:*** [Clave Primaria] [Clave Foránea, tabla Ejemplar]; Identificador del ejemplar
- ***estadold:*** [Clave Primaria] [Clave Foránea, tabla Estado]; Identificador del estado
- ***fechaDesde:*** [Clave Primaria]. Campo Fecha. Representa la fecha en que el ejemplar pasa a este estado
- ***fechaHasta:*** Campo Fecha. Representa la fecha en que el ejemplar deja de estar en este estado. Si esta a nulos quiere decir que es el estado actual.

TABLA Tipo

Descripción: Representa la tipología de socios de la biblioteca. En la actualidad se distinguen los tipos Juvenil y Adulto, pero no se descarta que puedan haber más en el futuro.

- ***tipold:*** [Clave Primaria]; Campo de texto, abreviación de los distintos tipos de socio.
- ***descripcion:*** Campo de texto. Descripción de la tipología de los socios.
- ***maximoMultas:*** Campo numérico entero. Número máximo de multas aplicables a un socio, una vez sobrepasado no se permiten más préstamos.

TABLA Socio

Descripción: Representa los socios de la red de bibliotecas. Debe considerarse que los socios lo son de la red de bibliotecas y no de una determinada.

- ***sociold:*** [Clave Primaria]; Campo entero auto-incremental, clave artificial para cada uno de los socios.
- ***tipold:*** [Clave Primaria] [Clave Foránea, tabla Tipo]; Identificador de tipo de socio
- ***códigoBarras:*** Campo numérico entero. Código de barras impreso en el carnet de socio de la biblioteca que lo identifica de forma única.
- ***apellidos:*** Campo de texto. Apellidos del socio.
- ***nombre:*** Campo de texto. Nombre del socio.
- ***telefono:*** Campo de texto. Teléfono particular.
- ***correoElectronico:*** Campo de texto. Correo electrónico
- ***fechaAlta:*** Campo de fecha. Indica la fecha de alta del empleado en la aplicación
- ***fechaBaja:*** Campo de fecha. Indica la fecha de baja del empleado en la aplicación
- ***ultimaModFecha:*** Campo de fecha. Indica la fecha en que se produjo la última modificación de los datos de este socio.
- ***ultimaModEmpl:*** [Clave Foránea, tabla Empleado]. Indica el empleado que realizó la última modificación de los datos de este socio.

TABLA Prestamo

Descripción: Acción por la que un recurso se deja en régimen de préstamo a un socio de la red de bibliotecas.

- **ejemplarId:** [Clave Primaria] [Clave Foránea, tabla Ejemplar]; Identificador del ejemplar
- **socioid:** [Clave Primaria] [Clave Foránea, tabla Socio]; Identificador del socio
- **fechaDesde:** [Clave Primaria]. Campo Fecha. Representa la fecha en que el ejemplar es prestado
- **fechaHasta:** Campo Fecha. Representa la fecha en que el ejemplar es devuelto. Si se encuentra a nulos quiere decir que está prestado.
- **fechaPrevistaDevol:** Campo Fecha. Fecha prevista de devolución considerando el máximo número de días de préstamo
- **contadorRenovac:** Campo Numérico entero. Número de renovaciones del préstamo que se han realizado.

TABLA Reserva

Descripción: Acción por la que se reserva un recurso que actualmente está en estado prestado. Al vencimiento del préstamo no será posible la renovación por el actual usuario si es que existe una reserva sobre él.

- **ejemplarId:** [Clave Primaria] [Clave Foránea, tabla Ejeemplar]; Identificador del ejemplar
- **socioid:** [Clave Primaria] [Clave Foránea, tabla Socio]; Identificador del socio
- **fecha:** [Clave Primaria]. Campo Fecha. Representa la fecha en que el ejemplar es reservado

TABLA Limite

Descripción: En función de la tipología de socio y del soporte material de los recursos se establecen unos límites en cuanto a préstamos y renovaciones.

- **materialId:** [Clave Primaria]. [Clave Foránea, tabla Material]. Indica el soporte en que se materializa este recurso (libro, CD, vídeo, ...)
- **tipoid:** [Clave Primaria] [Clave Foránea, tabla Tipo]; Identificador de tipo de socio
- **limitePrestamos:** Campo numérico entero. Representa el límite de préstamos que un socio de un tipo determinado puede realizar sobre un recurso de un soporte material determinado.
- **limiteRenovaciones:** Campo numérico entero. Representa el límite de renovaciones que un socio de un tipo determinado puede realizar sobre un recurso de un soporte material determinado.

TABLA Transaccion

Descripción: Representa los tipos de transacciones económicas que se realizan entre socios y la biblioteca.

- **trasaccionId:** [Clave Primaria]; Campo de texto, abreviación de los distintos tipos de transacción económica.
- **descripcion:** Campo de texto. Descripción de las distintas transacciones

TABLA Pago

Descripción: Acción por la que se realiza un pago por un tipo de transacción concreto, por ejemplo cuotas, multas, traducciones, etc.

- **pagold:** [Clave Primaria]; Campo entero auto-incremental, clave artificial para cada uno de los pagos.
- **sociold:** [Clave Foránea, tabla Socio]; Identificador del socio
- **trasaccionld:** [Clave Foránea, tabla Socio]; Identificador de tipo de transacción.
- **descripción:** Campo de texto. Descripción del pago.
- **importe:** Campo numérico decimal. Importe del pago.
- **fecha:** Campo de fecha. Fecha del pago
- **ultimaModFecha:** Campo de fecha. Indica la fecha en que se produjo la última modificación de los datos de este pago.
- **ultimaModEmpl:** [Clave Foránea, tabla Empleado]. Indica el empleado que realizó la última modificación de los datos de este pago.

3.2. Esquema de las tablas de la Base de Datos

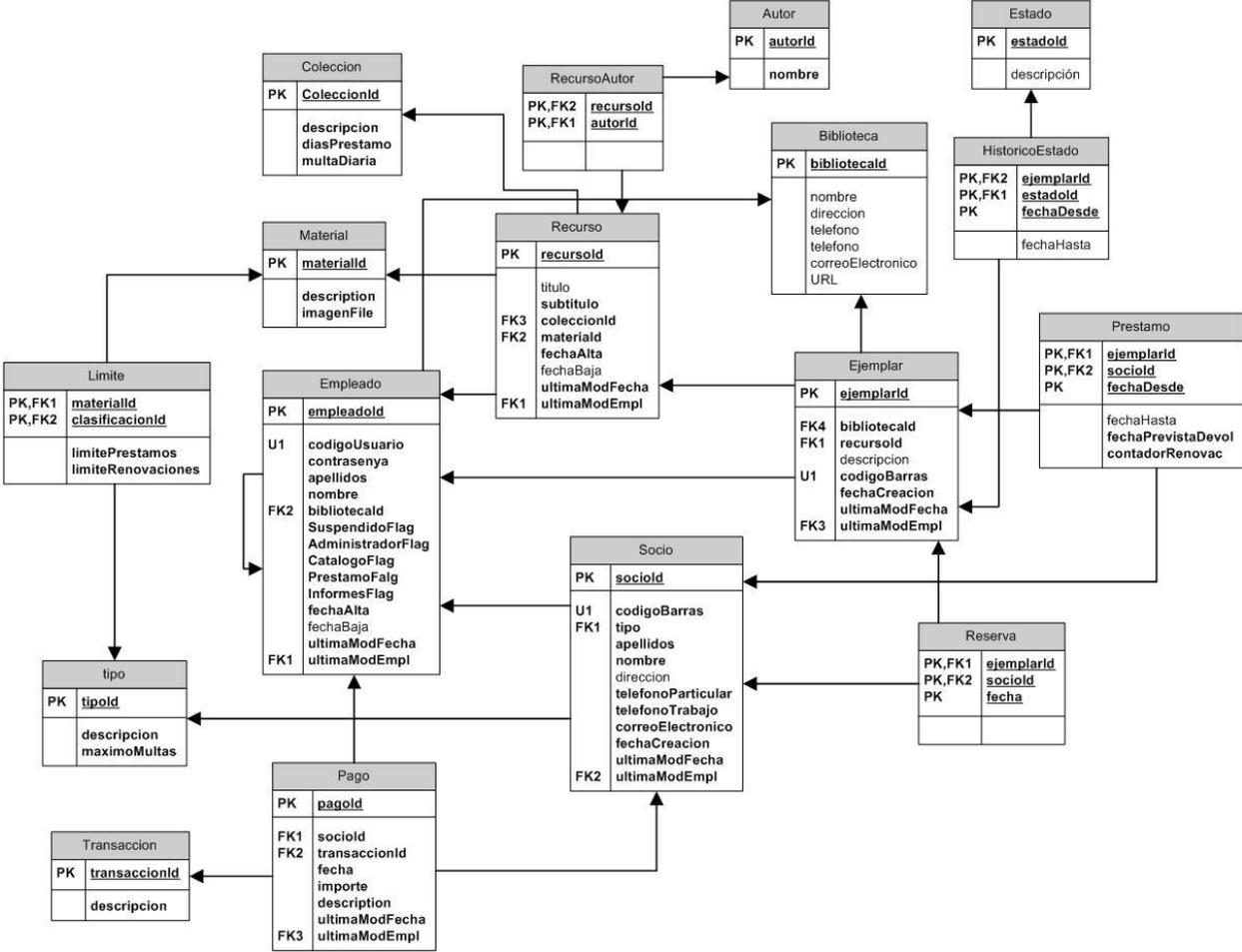


Ilustración 3: Diagrama lógico

3.3. Controles de integridad

En los apartados anteriores se han indicado algunos controles de integridad en aquellos campos que se han marcado como Claves primarias y Claves foráneas.

El hecho de que un campo sea clave primaria implica que:

- No acepta valores nulos.
- No permite duplicados

El hecho de que un campo sea clave foránea implica que:

No puede existir si no existe en la tabla a la que hace referencia la clave foránea.

Con estos mecanismos estamos asegurando parte de la integridad de los datos. El SGBD controla automáticamente la integridad de nuestros datos.

Existen otras restricciones que no quedan cubiertas por estos mecanismos, y que deberemos comprobar con comandos CHECK al definir las tablas, y o bien por comprobaciones que deberán realizarse en el aplicativo.

3.4. Índices

Nos interesará tener índices de todos los campos que hacemos servir para relacionar tablas entre sí, de manera que las consultas relacionadas sean lo más rápidas posible.

Después valoraremos el resto de campos. La ventaja de indexar es que las consultas en estos campos serán más rápidas, sin embargo, se penalizan inserciones, supresiones y modificaciones y además ocupan espacio adicional.

Los campos seleccionados para ser indizados son los siguientes:

- Los campos de clave primaria que lo son por su propia naturaleza.
- Los campos de clave foránea que haremos servir para relacionar tablas.

También se indizarán los campos correspondientes a 'Código de Barras' para garantizar la unicidad de estos campos.

3.5. Procedimientos

Nuestro sistema permitirá libremente el acceso en modo lectura a las tablas de datos para el software, sin embargo todos los accesos de escritura deberán realizarse por medio de los procedimientos almacenados. Se van a crear todos los procedimientos almacenados necesarios, y serán estos procedimientos los responsables de mantener la consistencia de los datos.

Lo importante es proporcionar los procedimientos básicos con un código robusto de control de errores. Las aplicaciones podrán preparar procedimientos más complejos partiendo de los procedimientos aportados en este proyecto.

En definitiva, en este proyecto se van a realizar los procedimientos necesarios para inserciones, modificaciones y supresiones en tablas.

Comportamiento común de todos los procedimientos almacenados.

Todos los procedimientos tienen un parámetro de salida denominado RC (Return Code) que siempre será el primer parámetro y servirá para identificar si la ejecución del procedimiento ha sido correcta a no.

Todos los procedimientos tienen un segundo parámetro denominado Usuario. Este será siempre el segundo parámetro e indica cuál es el usuario que lo está ejecutando. Esta persona deberá ser un empleado dado de alta en el fichero de empleados. Para la realización de esta comprobación se creará una función, de forma que si varía el criterio de validación, la modificación quede repercutida inmediatamente en todo el aplicativo.

Para todos los procedimientos en los que se añada un dato, se comprobará antes de insertar el nuevo valor que no exista en la base de datos. Si existe devolveremos un error.

En los procedimientos de alta / modificación de empleados, recursos, ejemplares y pagos, se actualizan los campos ultimaModFecha y ultimaModEmpl con la fecha del sistema y usuario que llama al procedimiento.

Definición de los procedimientos

A continuación se muestran los procedimientos que se proporcionan como interface con el sistema.

AltaBiblioteca (RC, USR, nombre, dirección, telefono, correoElectronico, URL)

Añadir biblioteca

Devuelve el identificador bibliotecald de la biblioteca creada.

AltaEmpleado (RC, USR, codigoUsuario, contrasena, apellidos, nombre, bibliotecald, suspendidoFlag, administradorFlag, catalogoFlag, PrestamoFlag, InformeFlag)

Añadir empleado

Devuelve el identificador empleadold del empleado creado.

Los campos fechaAlta, ultimaModFecha y ultimaModEmpl se rellenan automáticamente

AltaColeccion (RC, USR, descripción, díasPrestamo, multaDiaria)

Añadir colección

Devuelve el identificador coleccionId de la colección creada.

AltaMaterial (RC, USR, descripción, imagenFile)

Añadir material

Devuelve el identificador materialId del material creado.

AltaAutor (RC, USR, nombre)

Añadir autor

Devuelve el identificador autorId del autor creado.

AltaRecurso (RC, USR, titulo, subtítulo, coleccionId, materialId)

Añadir recurso

Devuelve el identificador recursold del recurso creado.

Los campos fechaAlta, ultimaModFecha y ultimaModEmpl se rellenan automáticamente

AltaRecursoAutor (RC, USR, recursold, aurorld)

Añadir recurso-autor

AltaTipoSocio (RC, USR, descripción, maximoMultas)

Añadir tipo de socio

Devuelve el identificador tipold del tipo de socio creado.

AltaSocio (RC, USR, codigoBarras, tipold, apellidos, nombre, dirección, teléfono, correoElectronico, fechaNacimiento)

Añadir socio

Devuelve el identificador sociold del socio creado.

Los campos fechaAlta, ultimaModFecha y ultimaModEmpl se rellenan automáticamente

El campo fechaBaja se deja a nulos

AltaLimite (RC, USR, materialId, tipold, limitePrestamos, limiteRenovaciones)

Añadir límite

AltaEjemplar (RC, USR, bibliotecald, recursold, descripción, codigoBarras)

Añadir ejemplar

Devuelve el identificador ejemplarId del ejemplar creado.

Los campos fechaAlta, ultimaModFecha y ultimaModEmpl se rellenan automáticamente

El campo fechaBaja se deja a nulos

AltaEstado (RC, USR, descripcion)

Añadir estado

Devuelve el identificador estadold del estado creado.

AltaHistoricoEstado (RC, USR, ejemplarId, estadold, fechaDesde)

Añadir estado

AltaReserva (RC, USR, ejemplarId, sociold, fecha)

Añadir reserva

AltaPrestamo (RC, USR, ejemplarId, sociold, fechaDesde, fechaHasta, fechaPrevistaDevolucion, contadorRenovac)

Añadir préstamo

AltaTransaccion (RC, USR, descripción)

Añadir transacción

Devuelve el identificador transaccionId de la transacción creada.

AltaPago (RC, USR, sociold, transacciónId, fecha, importe, descripción)

Añadir pago

Devuelve el identificador pagold del pago creado.

Los campos fechaAlta, ultimaModFecha y ultimaModEmpl se rellenan automáticamente

ModificaBiblioteca (RC, USR, bliotecald, nombre, dirección, telefono, correoElectronico, URL)

Modificar biblioteca

Permite modificar nombre, dirección, teléfono, correo electrónico y URL.

Los campos ultimaModFecha y ultimaModEmpl se rellenan automáticamente

ModificaEmpleado (RC, USR, empleadold, contrasena, apellidos, nombre, bibliotecald, suspendidoFlag, administradorFlag, catalogoFlag, PrestamoFlag, InformeFlag, fechaBaja)

Modificar empleado

Permite modificar contraseña, nombre, identificador de biblioteca, y los flags de suspendido, administración, catálogo, préstamo e informes.

Los campos ultimaModFecha y ultimaModEmpl se rellenan automáticamente

ModificaColeccion (RC, USR, coleccionId, descripción, díasPrestamo, multaDiaria)

Modificar colección

Permite modificar descripción, los días de préstamo, la multa diaria.

ModificaMaterial (RC, USR, material_id, descripción, imagenFile)

Modificar material

Permite modificar descripción, y la dirección del fichero imagen.

ModificaAutor (RC, USR, autorId, nombre)

Modificar autor

Permite modificar el nombre del autor.

ModificaRecurso (RC, USR, recursoid, titulo, subtítulo, coleccionId, materialId, fechaBaja)

Modificar recurso

Permite modificar el título, subtítulo, identificadores de colección y material y fecha de baja.

Los campos ultimaModFecha y ultimaModEmpl se rellenan automáticamente

ModificaTipoSocio (RC, USR, tipoid, descripción, maximoMultas)

Modificar tipo de socio

Permite modificar el nombre del autor.

ModificaSocio (RC, USR, socioid, tipoid, apellidos, nombre, direccion, telefono, correoElectronico, fechaNacimiento, fechaBaja)

Modificar socio

Permite modificar el tipo de socio, nombre y apellidos, dirección, correo electrónico, fecha de nacimiento y fecha de baja del socio.

Los campos ultimaModFecha y ultimaModEmpl se rellenan automáticamente

ModificaLimite (RC, USR, materialId, tipoid, limitePrestamos, limiteRenovaciones)

Añadir límite

Permite modificar los límites de préstamos y renovaciones.

ModificaEjemplar (RC, USR, ejemplarId, bibliotecald, descripción, fechaBaja)

Modificar ejemplar

Permite modificar el código de biblioteca, la descripción y la fecha de baja

Los campos ultimaModFecha y ultimaModEmpl se rellenan automáticamente

ModificaEstado (RC, USR, estadId, descripción)

Modifica estado

Permite modificar la descripción del estado

ModificaPrestamo (RC, USR, ejemplarId, socioid, fechaDesde, fechaHasta, fechaPrevistaDevolucion, contadorRenovac)

Modificar préstamo

Permite modificar la fecha hasta, la fecha provista para devolución, y el contador de renovaciones.

ModificaTransaccion (RC, USR, transaccionId, descripción)

Modificar transacción

Permite modificar la descripción de las transacciones

ModificaPago (RC, USR, socioid, transaccionId, fecha, importe, descripción)

Modificar pago

Permite modificar el importe y descripción de un pago

Los campos ultimaModFecha y ultimaModEmpl se rellenan automáticamente

EliminaBiblioteca (RC, USR, bibliotecId)

EliminaEmpleado (RC, USR, empleadold)
EliminaColeccion (RC, USR, coleccionId)
EliminaMaterial (RC, USR, materialId)
EliminaAutor (RC, USR, autorId)
EliminaRecurso (RC, USR, recursold)
EliminaRecursoAutor (RC, USR, recursold, aurorId)
EliminaTipoSocio (RC, USR, tipold)
ElimiaSocio (RC, USR, sociold)
EliminaLimite (RC, USR, materialId, tipold)
EliminaEjemplar (RC, USR, ejemplarId)
EliminaEstado (RC, USR, estadold)
EliminaHistoricoEstado (RC, USR, ejemplarId, estadold, fechaDesde)
EliminaReserva (RC, USR, ejemplarId, sociold, fecha)
EliminaPrestamo (RC, USR, ejemplarId, sociold, fechaDesde)
EliminaTransaccion (RC, USR, transaccionId)
EliminaPago (RC, USR, pagold)

3.6. Diseño del sistema de actividad (log)

El sistema tiene una tabla (TBLOG) destinada a guardar las llamadas que se han hecho a los procedimientos. La utilización de los procedimientos relacionados en el párrafo anterior debe ser la única forma de actualización de los datos. Los procedimientos grabarán en la tabla TBLOG los movimientos realizados de forma que servirá de diario de la base de datos.

Con esta tabla debemos ser capaces de detectar cualquier problema y auditar los movimientos realizados. Debemos ser capaces de detectar y reproducir cualquier error puesto que tenemos un histórico de todas las llamadas realizadas y podríamos reproducir la situación.

Por cada llamada que se haga a los procedimientos guardaremos una fila con la siguiente información:

- Hora de la ejecución
- Usuario que ejecuta el procedimiento
- Nombre del procedimiento almacenado
- Valor de los parámetros de entrada
- Valor de los parámetros de salida
- Resultado de la ejecución RC.
- Traza
- Código de error Oracle
- Descripción Oracle del error

4. Implementación

4.1. Plataforma de trabajo

La plataforma de trabajo sobre la que proyecta este proyecto se basa en un procesador Xeon X5550 a 2,66 GHz con 4 núcleos, una controladora RAID PERC H700 con 512MB de cache y 4 GB de memoria DDR3 a 1066MHz.

Está previsto poder trabajar con dos sistemas operativos: Windows Server 2008 SP2, Standard Edition y con Novell/SuSE Linux Enterprise Server. De salida saldrá certificado con la opción Windows dejándose como mejora una implementación en Linux. El gestor de base de datos que se instala en la máquina es Oracle Database 11g Standard Edition One.

Para poder editar y desarrollar la aplicación utilizaré las herramientas de cliente de Oracle: SQL en línea de comandos y el cliente web de desarrollo. Además, también resulta una buena herramienta de desarrollo el producto Oracle SQL Developer.

4.2. Aspectos generales de la implementación

En este punto se declaran los criterios que se han seguido para la implementación del proyecto.

4.2.1. Código estructurado

Se ha dado importancia a hacer un código fácilmente legible de cara a minimizar el impacto en posibles modificaciones futuras.

Para ello se procede la siguiente metodología

- Indentación del código.
- Se ha respetado el criterio mayúsculas/minúsculas generalizado en Java. Aunque Oracle no es sensible a este criterio.
- Se han separado las consultas SQL en varias líneas

4.2.2. Tipos de variable: %type

Al definir variables en los procedimientos se ha utilizado esta característica de Oracle que nos permite asignar a una variable el mismo tipo que pueda tener otra variable determinada. Este sistema da la ventaja que si se debe modificar un tipo de variable de una tabla, automáticamente cambiará en todas las variables de procedimientos que se hayan creado con este criterio.

4.2.3. Parámetros de los procedimientos

El nombre de los parámetros de los procedimientos va con el prefijo 'p' minúscula y la siguiente letra en mayúscula.

Los dos primeros campos de todos los procedimientos son los mismos:

- pRC: es el código de retorno de la salida del procedimiento.
- pUSR: es el código de empleado de la persona que hace la llamada al procedimiento.

4.2.4. Campos auto-incrementales

En algunas tablas, el campo utilizado como clave principal es una clave ficticia, de tipo numérico que se incrementa automáticamente a cada inserción de una fila. Oracle no incorpora este tipo de datos auto-incrementales, sin embargo contiene un objeto llamado secuencia que dispone de un método que permite la generación de un nuevo número cada vez que se inserta un registro, de forma que siempre se generan números diferentes.

4.2.5. Control de excepciones

En cada procedimiento se capturan distintos tipos de excepciones, las primeras son las que genera el propio procedimiento cuando se detecta un error en los datos que nos pasa la aplicación por parámetro, el resto son excepciones que controla el propio Oracle.

Cuando se captura una excepción, se llama a la función *retornaRC* pasándole como parámetro la variable *SqlErrm*, que contiene el texto que Oracle nos devuelve como explicación del error.

Actualmente esta función no tiene contenido, por lo que se informa con el mensaje que devuelve Oracle, pero en próximas versiones se podrán personalizar estos mensajes de error y en lugar de dar un mensaje del tipo '*Restricción única PK_ejemplar violada*', dará el mensaje personalizado '*Se intenta dar de alta un ejemplar que ya existe*'. Para los casos que no tengamos controlados, se informará de la excepción tal como la da Oracle, sin hacerle ninguna modificación.

En caso de detección de error, siempre se ejecuta un *rollback* y se elimina cualquier cambio que se haya hecho en la base de datos.

4.2.6. Valores del resultado de salida de los procedimientos: *pRC*

Todos los procedimientos tienen un primer parámetro común denominado *pRC*. Este parámetro es de sólo salida y se denomina *pRC*. Los valores que puede tomar comienzan siempre por uno de estos dos prefijos: OK, ERROR.

OK → Se devuelve solo OK e indica que la ejecución ha sido correcta.

ERROR-CODIGO-Descripción → Se ha producido algún error y no se ha hecho efectivo ningún cambio en la base de datos.

4.2.7. Sistema de registro LOG

Tal como se indica en las especificaciones, se ha de implementar una tabla de registro de la actividad (tabla TBLOG), que almacena un registro por cada llamada que se hace a un procedimiento.

El motivo de esta tabla es de guardar toda la actividad que se ha producido en nuestras tablas, puesto que todas las inserciones, modificaciones y eliminaciones se realizarán por medio de los procedimientos suministrados.

En esta tabla debemos reproducir las condiciones que han llevado al sistema a un estado determinado, para poder analizar en caso de un error concreto qué es lo que ha pasado para que el sistema haya quedado en ese estado, o haya dado ese error.

En esa tabla se almacena:

Código	→ Autonumérico
Hora	→ Fecha y hora del momento del registro
Empleado	→ Código de usuario que ejecuta la acción
Procedimiento	→ Nombre del procedimiento ejecutado
Parámetros entrada	→ Valores de los parámetros de entrada
Parámetros salida	→ Valores de los parámetros de salida
Resultado	→ Valor del parámetro pRC
Traza	→ Traza dentro del procedimiento
Código Error	→ Código de error de Oracle
Descripción Error	→ descripción de error de Oracle

4.2.8. Guiones

El código generado se guarda en ficheros de texto que se podrán copiar posteriormente al editor Oracle SQL Developer para su ejecución. Los guiones generados se anexan a esta memoria.

GUION 0. Borrado de los objetos de la base de datos.

Este guión borra todos los objetos definidos en este proyecto. No borra el usuario.

GUION 1. Creación del usuario administrador.

Este guión debe ejecutarse con un usuario administrador y nos creará un usuario y un esquema donde ubicaremos todos los objetos de nuestra base de datos.

GUION 2. Creación de secuencias, tablas e índices

Este guión se ejecutará con el usuario creado en el párrafo anterior (UOC) el cual tiene permisos totales sobre la base de datos.

Las funciones que realiza este script son las siguientes:

- Creación de secuencias
- Creación de tablas e índices
- Inserción de registros

GUION 3. Creación de funciones y procedimientos.

Este guión se ejecutará con el usuario creado en el párrafo anterior (UOC) el cual tiene permisos totales sobre la base de datos.

Las funciones que realiza este script son las siguientes:

- Inserción de registros
- Creación de funciones y procedimientos de carácter general
- Creación de procedimientos específicos de esta aplicación

GUION 4. Juego de pruebas

Contiene las llamadas a los procedimientos que hemos creado.

Inicialmente se hacen llamadas con parámetros correctos de cara a llenar de datos nuestra base de datos.

Una vez introducidos los datos correctamente se comprobará que las funciones fallan de la manera que se espera cuando le pasamos incorrectamente los parámetros. Este procedimiento va escribiendo líneas de comentario por una salida estándar.

GUIÓN 5. Eliminación de todos los registros de la base de datos

Este procedimiento borra todos los registros grabados en el GUIÓN 4 y las distintas pruebas que se hayan podido realizar.

4.3. Guión de creación de tabla

A continuación se muestra lista de tablas creadas que forman parte de este proyecto. El detalle de cada una de ellas puede verse en el apartado de diseño o bien en el propio guión de creación de tablas.

Biblioteca(*), Empleado(*), TBLOG(*), Coleccion(*), Material(*), Recurso(*), Autor(*), RecursoAutor, Ejemplar(*), Tipo(*), Socio(*), Estado(*), HistEstado, Prestamo, Reserva, Limite, Transaccion(*), Pago(*).

Las tablas marcadas con (*) requieren de la creación de una secuencia por ser de clave autonumérica.

A continuación veremos la definición de un par de tablas contenidas dentro de Guión2. En ellas se puede ver cómo se establece la integridad referencial tal como se definió en el apartado de diseño lógico.

```
CREATE TABLE Prestamo (
  ejemplarId      INTEGER      CONSTRAINT NN_pre_ejemplar NOT NULL
                                     CONSTRAINT FK_pre_eje_eje REFERENCES Ejemplar(ejemplarId),
  socioId         INTEGER      CONSTRAINT NN_pre_socio   NOT NULL
                                     CONSTRAINT FK_pre_soc_soc REFERENCES Socio(socioId),
  fechaDesde      DATE         CONSTRAINT NN_pre_fechaD  NOT NULL,
  fechahasta      DATE,
  fechaPrevistaDev DATE        CONSTRAINT NN_pre_fechaPrev NOT NULL,
  contadorRenovac INTEGER      CONSTRAINT NN_pre_renovac NOT NULL,
  CONSTRAINT PK_prestamo PRIMARY KEY (ejemplarId, socioId, fechaDesde)
);
```

```
CREATE TABLE Socio (
  socioId          INTEGER      CONSTRAINT PK_socio       PRIMARY KEY,
  codigoBarras     CHAR(8)      CONSTRAINT NN_soc_barras NOT NULL,
  tipoId           INTEGER      CONSTRAINT NN_tip_tipo   NOT NULL
                                     CONSTRAINT FK_soc_tip_tip REFERENCES Tipo(tipoId),
  apellidos        VARCHAR2(100) CONSTRAINT NN_soc_apellidos NOT NULL,
  nombre           VARCHAR2(100),
  telefono         VARCHAR2(16) CONSTRAINT NN_soc_direccion NOT NULL,
  correoElectronico VARCHAR2(100) CONSTRAINT NN_soc_email NOT NULL,
  fechaAlta        DATE         CONSTRAINT NN_soc_fechaAl NOT NULL,
  fechaBaja        DATE,
  ultimaModFecha   DATE        CONSTRAINT NN_soc_ultModF NOT NULL,
  ultimaModEmpl    INTEGER      CONSTRAINT NN_soc_ultModE NOT NULL
                                     CONSTRAINT FK_soc_emp_emp REFERENCES Empleado(empleadoId),
  CONSTRAINT UN_Socio UNIQUE(codigoBarras)
);
```

4.4. Guión de creación de procedimientos

El código de generación de objetos se incluye en un anexo a este documento. Concretamente el de los procedimientos es especialmente largo.

Todos los procedimientos tienen una estructura análoga, por lo que cualquiera de ellos puede servir de ejemplo para poder analizar su comportamiento.

A continuación se muestra el código del procedimiento almacenado de AltaEjemplar:

```
CREATE PROCEDURE AltaEjemplar
    (pRC OUT VARCHAR2, pUSR Empleado.codigoUsuario%type,
    pBibliotecaId Ejemplar.bibliotecaId%type, pRecursoId Ejemplar.recursoId%type,
    pDescripcion Ejemplar.descripcion%type, pCodigoBarras Ejemplar.codigoBarras%type,
    pFechaAlta Ejemplar.fechaAlta%type, pFechaBaja Ejemplar.fechaBaja%type,
    pEjemplarId OUT Ejemplar.ejemplarId%type
    ) is
    parametrosIn TBLOG.paramEntrada%type;
    parametrosOut TBLOG.paramSalida%type;
    vtrace INTEGER;
    ej_usuario EXCEPTION;
    ej_yaExiste EXCEPTION;
    contador INTEGER;
    procedimiento VARCHAR(30);
    iEmpleadoId INTEGER;
begin
    vtrace:=0;
    procedimiento:='AltaEjemplar';
    parametrosIn:=pBibliotecaId||'P'||pRecursoId||'P'||pDescripcion||'P'||pCodigoBarras||'P'||
        pFechaAlta||'P'||pFechaBaja;
    -- Compruebo que el usuario pueda realizar esta acción
    IF (NOT esEmpleado(pUSR)) THEN
        vtrace:=10;
        -- El usuario no tiene permiso
        RAISE ej_usuario;
    END IF;
    vtrace:=20;
    Compruebo si ya existe un registro con este nombre
    IF (contador <> 0) THEN
        vtrace:=30;
        -- Ya existe un registro con este nombre
        raise ej_yaExiste;
    ELSE
        vtrace:=40;
        SELECT sEjemplar.NEXTVAL
        INTO contador
        FROM DUAL;
        SELECT empleadoId
        INTO iEmpleadoId
        FROM Empleado
        WHERE codigoUsuario = pUSR;
        vtrace:=50;
        -- OK, se añade en la base de datos
        INSERT INTO Ejemplar
        VALUES (contador, pBibliotecaId, pRecursoId, pDescripcion, pCodigoBarras, pFechaAlta,
            pFechaBaja, sysdate, iEmpleadoId);
        pEjemplarId:= contador;
    END IF;
    vtrace:=1000;
    pRC:='OK';
    parametrosOut:=pEjemplarId;
    registrarlog (pUSR, procedimiento, parametrosIN, parametrosOut, pRC, vtrace, null, null);
    commit;

    EXCEPTION
    WHEN ej_usuario THEN
        rollback;
        pRC:=retornaRC('INT_User');
        parametrosOut:=pEjemplarId;
        registrarlog (pUSR, procedimiento, parametrosIN, parametrosOut, pRC, vtrace, null, null);
        commit;

    WHEN ej_YaExiste THEN
        rollback;
        pRC:=retornaRC('INT_YaExiste');
        parametrosOut:=pEjemplarId;
        registrarlog (pUSR, procedimiento, parametrosIN, parametrosOut, pRC, vtrace, null, null);
        commit;

    WHEN OTHERS THEN
        rollback;
        pRC:=retornaRC(SqlErrm);
        parametrosOut:=pEjemplarId;
        registrarlog (pUSR, procedimiento, parametrosIN, parametrosOut, pRC, vtrace, null, null);
        commit;

end AltaEjemplar;
```

4.5. Pruebas

En la documentación anexa se incluye un juego de pruebas de todas las funciones que se han implementado.

El juego de pruebas está pensado para ser ejecutado con las tablas vacías

Primeras 50 filas de la tabla TBLOG después de proceder al juego de prueba.

TBLOG ID	HORA	CODIGO USUARIO	PROCEDIMIENTO	PARAMETROS ENTRADA	PARAMETROS SALIDA	RESUL TADO	TRAZA
1	18/04/2010 23:09	admin	AltaBiblioteca	Biblioteca CataluñaAv. Diagonal 200, BarcelonaP933009090PbibliotecaDiagonal@UOC.EDUPbibliotecaDiagonal.UOC.edu	Biblioteca CataluñaAv. Diagonal 200, BarcelonaP933009090PbibliotecaDiagonal@UOC.EDUPbibliotecaDiagonal.UOC.edu2	OK	1000
2	18/04/2010 23:09	admin	ModificaBiblioteca	2PBiblioteca CatalunyaAv. Diagonal 200, BarcelonaP93311111PbibliotecaDiagonal@UOC.EDUPbibliotecaDiagonal.UOC.edu	2PBiblioteca CatalunyaAv. Diagonal 200, BarcelonaP93311111PbibliotecaDiagonal@UOC.EDUPbibliotecaDiagonal.UOC.edu	OK	1000
3	18/04/2010 23:09	admin	EliminaBiblioteca			2 OK	1000
4	18/04/2010 23:09	admin	AltaEmpleado	userJBSUserJBSBlanca SánchezPF. JorgeP1PNPNPNPN		2 OK	1000
5	18/04/2010 23:09	admin	ModificaEmpleado	2PBlanca SánchezPF. JorgeP1PYYPYPYP10-12-2010	2PBlanca SánchezPF. JorgeP1PYYPYPYP10-12-2010	OK	1000
6	18/04/2010 23:09	admin	EliminaEmpleado			2 OK	1000
7	18/04/2010 23:09	admin	AltaColeccion	Adult FictionP21P,5		1 OK	1000
8	18/04/2010 23:09	admin	AltaColeccion	Adult NonfictionP21P,25		2 OK	1000
9	18/04/2010 23:09	admin	AltaColeccion	CassettesP7P,75		3 OK	1000
10	18/04/2010 23:09	admin	AltaColeccion	Compact DiscsP7P,5		4 OK	1000
11	18/04/2010 23:09	admin	AltaColeccion	Computer SoftwareP7P1,15		5 OK	1000
12	18/04/2010 23:09	admin	AltaColeccion	Easy ReadersP21P,45		6 OK	1000
13	18/04/2010 23:09	admin	AltaColeccion	Juvenile FictionP21P,25		7 OK	1000
14	18/04/2010 23:09	admin	AltaColeccion	Juvenile NonfictionP21P,25		8 OK	1000
15	18/04/2010 23:09	admin	AltaColeccion	New BooksP14P,3		9 OK	1000
16	18/04/2010 23:09	admin	AltaColeccion	Videos and DVDsP3P1		10 OK	1000
17	18/04/2010 23:09	admin	AltaColeccion	ReferenciaP3P1		11 OK	1000
18	18/04/2010 23:09	admin	ModificaColeccion	11PVideos and DVDsP5P1,5		11 OK	1000
19	18/04/2010 23:09	admin	EliminaColeccion			11 OK	1000
20	18/04/2010 23:09	admin	AltaMaterial	audio tapesPtape.gif		1 OK	1000
21	18/04/2010 23:09	admin	AltaMaterial	bookPbook.gif		2 OK	1000
22	18/04/2010 23:09	admin	AltaMaterial	cd audioPcd.gif		3 OK	1000
23	18/04/2010 23:09	admin	AltaMaterial	cd computerPcd.gif		4 OK	1000
24	18/04/2010 23:09	admin	AltaMaterial	equipmentPcase.gif		5 OK	1000
25	18/04/2010 23:09	admin	AltaMaterial	mapsPmap.gif		6 OK	1000
26	18/04/2010 23:09	admin	AltaMaterial	video/dvdPcamera.gif		7 OK	1000
27	18/04/2010 23:09	admin	AltaMaterial	pruebaPprueba.gif		8 OK	1000
28	18/04/2010 23:09	admin	ModificaMaterial	8PMiMaterialPmaterial.gif	8PMiMaterialPmaterial.gif	OK	1000
29	18/04/2010 23:09	admin	EliminaMaterial			8 OK	1000
30	18/04/2010 23:09	admin	AltaRecurso	UNIX in a NutshellPA Desktop Quick Reference for System V and Solaris 2.0P2P2P18/04/10P		1 OK	1000
31	18/04/2010 23:09	admin	AltaRecurso	Java in a NutshellPA Desktop Quick ReferenceP4P2P18/04/10P		2 OK	1000
32	18/04/2010 23:09	admin	AltaRecurso	Design PatternsPElements of Reusable Object-Oriented SoftwareP6P2P18/04/10P		3 OK	1000
33	18/04/2010 23:09	admin	AltaRecurso	Applying UML and PatternsPAN Introduction to Object Oriented Analysis and DesignP7P2P18/04/10P		4 OK	1000
34	18/04/2010 23:09	admin	AltaRecurso	Beverly Cleary, Illustrated by Louis DarlingP1P6P18/04/10P		5 OK	1000
35	18/04/2010 23:09	admin	AltaRecurso	Beverly Cleary, Illustrated by Louis DarlingP3P7P18/04/10P		6 OK	1000
36	18/04/2010 23:09	admin	AltaRecurso	PC World (Periodical):2003P3P World Communications, Inc.P4P6P18/04/10P		7 OK	1000
37	18/04/2010 23:09	admin	AltaRecurso	U.S. News and World Report (Periodical):2003P3P U.S. News and World ReportP5P7P18/04/10P		8 OK	1000
38	18/04/2010 23:09	admin	AltaRecurso	yyyPzzzP5P7P18/04/10P		9 OK	1000
39	18/04/2010 23:09	admin	ModificaRecurso	9P111P222P5P7P18/04/10P18/04/10P	9P111P222P5P7P18/04/10P18/04/10P1	OK	1000
40	18/04/2010 23:09	admin	EliminaRecurso			9 OK	1000
41	18/04/2010 23:09	admin	AltaAutor	Gilly,Daniel		1 OK	1000
42	18/04/2010 23:09	admin	AltaAutor	David Lanagan		2 OK	1000
43	18/04/2010 23:09	admin	AltaAutor	Flanagan, David		3 OK	1000
44	18/04/2010 23:09	admin	AltaAutor	Erich Gamma		4 OK	1000
45	18/04/2010 23:09	admin	AltaAutor	Richard Helm		5 OK	1000
46	18/04/2010 23:09	admin	AltaAutor	Ralph Johnson		6 OK	1000
47	18/04/2010 23:09	admin	AltaAutor	John Vlissides		7 OK	1000
48	18/04/2010 23:09	admin	AltaAutor	Craig Larman		8 OK	1000
49	18/04/2010 23:09	admin	AltaAutor	Beverly Cleary		9 OK	1000
50	18/04/2010 23:09	admin	AltaAutor	Henry Huggins		10 OK	1000

Ilustración 4 Primeras 50 filas de la tabla TBLOG

Detalle de las tablas después de su creación (Guión 2)

The screenshot displays the Oracle SQL Developer interface. On the left, a tree view shows the database structure under 'DBConnection1', with the 'EMPLEADO' table selected. The main window shows the 'Columns' tab for the 'EMPLEADO' table, displaying a list of columns with their respective data types, nullability, and other attributes.

Column Name	Data Type	Nullable	Data Default	COLUMN ID	Primary Key	COMMENTS
EMPLEADOID	NUMBER	No		1	1	
CODIGOUSUARIO	CHAR(8 Bytes)	No		2		
CONTRASENYA	CHAR(10 Bytes)	No		3		
APELLIDOS	VARCHAR2(100 Bytes)	No		4		
NOMBRE	VARCHAR2(100 Bytes)	Yes		5		
BIBLIOTECAID	NUMBER	No		6		
SUSPENDIDOFLAG	CHAR(1 Bytes)	No		7		
ADMINISTRADORFLAG	CHAR(1 Bytes)	No		8		
CATALOGOFLAG	CHAR(1 Bytes)	No		9		
PRESTAMOFLAG	CHAR(1 Bytes)	No		10		
INFORMESFLAG	CHAR(1 Bytes)	No		11		
FECHAALTA	DATE	No		12		
FECHABAJA	DATE	Yes		13		
ULTIMAMODFECHA	DATE	No		14		
ULTIMAMODEMPL	NUMBER	No		15		

Ilustración 5 Detalle de las tablas después de su creación

MODELO ANALÍTICO

5. Diseño analítico

Según las especificaciones, a parte de los datos básicos para el funcionamiento de las bibliotecas, se quiere ofrecer alguna funcionalidad adicional como por ejemplo, recomendar documentos a los usuarios en función del tipo de documentos cogidos previamente. También se desea recoger información sobre los autores o documentos más exitosos, fechas con más afluencia de usuarios, vida de los documentos, pérdidas de libros, etc.

Para ello se diseñará un almacén de datos que sintetice los datos recogidos en nuestro sistema operacional ubicado en la base de datos comentada en capítulos anteriores. Los datos de las bibliotecas, libros, préstamos, autores, etc. serán explotados con una herramienta Business Intelligence. La seleccionada es Cognos 7.1 de IBM, herramienta líder del mercado junto con Business Objects (SAP), Hyperion (Oracle) y AlphaBlox (IBM).

Para la creación de estos informes es necesaria la creación de un almacén de datos 'Data Warehouse' que pueda dar las funcionalidades a los requerimientos solicitados.

El producto final incluirá la automatización del proceso de recogida y manipulación de datos. El almacenaje de los datos se materializará en un modelo físico almacenado en el mismo SGBD con el que hemos desarrollado el modelo operacional y su posterior explotación se realizará con los módulos Impromptu, Transformer y PowerPlay de Cognos.

PowerPlay Transformer, es el componente de Cognos que se utiliza para definir la estructura de cubos OLAP y construir los cubos de análisis multidimensional, los denominados PowerCubos. Un PowerCubo contiene datos calculados y resumidos, organizados en dimensiones y medidas, los cuales pueden ser vistos y analizados con el cliente PowerPlay.

La construcción de un modelo de PowerPlay implica designar columnas en los archivos de origen que en el modelo representarán las medidas, dimensiones y niveles.

Origen de datos - es la lista de los orígenes de datos conectados. Cada fuente puede ser ampliada y lista todas las columnas disponibles. Los usuarios pueden inspeccionarlo utilizando el visor de fuentes de datos. Las columnas de una fuente de datos se pueden arrastrar y crear las dimensiones y medidas correspondientes.

Mapa de dimensiones gestiona las dimensiones y niveles jerárquicos en el modelo. Permite crear nuevas dimensiones y modificar las existentes por medio de una interface gráfica. Las dimensiones y niveles subyacentes se crean moviendo las columnas de los 'orígenes de datos' a la columna de dimensión correspondiente.

5.1. Flujo de trabajo con las herramientas Cognos

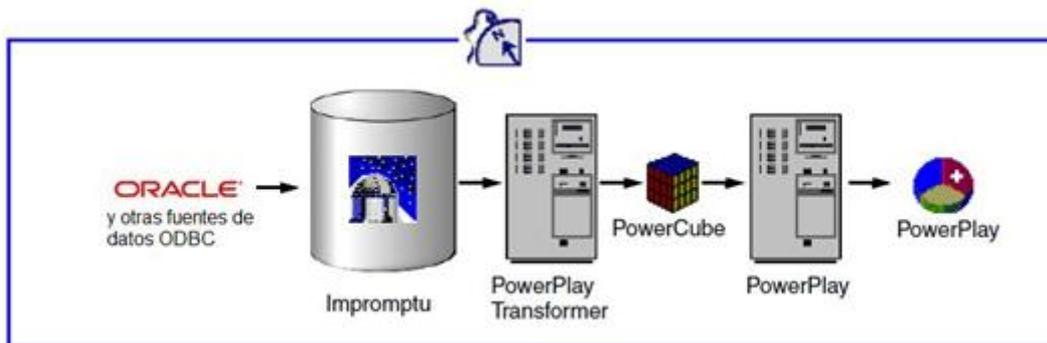


Ilustración 6 Flujo de trabajo con las herramientas Cognos

Cognos Transformer es la herramienta de modelación utilizada para construir modelos multidimensionales de una fuente de datos.

EL rango de fuentes de datos que incluye Transformer incluye:

- Ficheros 'Impromptu query definition' (.iqd), lo cuales pueden realizar consultas sobre servidores de bases de datos
- Ficheros ASCII delimitados (.asc) o ficheros CSV.
- Ficheros de texto de campos de longitud fija.
- Bases de datos locales, incluyendo Microsoft Access,
- Hojas de cálculo incluyendo Microsoft Excel y Lotus 1-2-3

Para obtener los datos de Oracle, utilizaré ficheros IQD (Impromptu Query Definiton)

Cognos Impromptu

Por este motivo, el proceso empieza con la herramienta Cognos Impromptu. En este producto se crea un catálogo a partir de un origen de Datos ODBC, del cual se seleccionan las tablas requeridas para nuestro proyecto. Se definen las relaciones entre tablas por medio de la opción 'JOIN'. El objetivo de esta herramienta es producir informes de forma fácil y orientada a usuarios finales (no informáticos). Nuestro objetivo será definir un informe básico que genere un informe con la información requerida en nuestro modelo conceptual y salvarlo como un fichero IQD.

Cognos Transformer

A continuación, un wizard nos permite importar los datos de nuestro fichero IQD como origen de datos de este aplicativo. La herramienta genera un modelo multidimensional preliminar, sin embargo conviene refinarlo retocando dimensiones, medidas y niveles según el modelo conceptual. A partir del modelo Transformer ya se puede crear un PowerCube.

PowerCube ya es un cubo multidimensional generado que puede ser utilizado con la tercera herramienta: Cognos PowerPlay.

Cognos PowerPlay

PowerPlay ofrece capacidad total para explorar cubos y generar informes gráficos a partir de los PowerCubos obtenidos en el apartado anterior.

5.2. Conceptos de modelado dimensional

Las dimensiones normales y de medida se utilizan para habilitar una presentación OLAP de metadatos, replegar y profundizar funciones OLAP. Se utilizan grupos de esquema en estrella (un hecho con varias dimensiones) si desea utilizar un análisis con un origen de datos relacional.

Al crear un modelo, se recomienda que las dimensiones normales y las dimensiones de medida se creen según un modelo relacional en el que se hayan aplicado los conceptos de esquema en estrella.

Dimensiones normales

Una dimensión normal contiene información clave del negocio y descriptiva, y su función es organizar la información en una jerarquía, desde el nivel más alto del nivel de detalles al más bajo. Normalmente, tiene varios niveles, cada uno de los cuales requiere una clave y un título. Si no tiene una clave única para su nivel, se recomienda crear una en un cálculo. Por ejemplo, una dimensión de producto puede contener los niveles Biblioteca, Material y Recurso organizados en una sola jerarquía llamada Biblioteca. Otro ejemplo es una dimensión de tiempo, que tiene los niveles Año, Trimestre y Mes.

Las dimensiones normales de modelo se basan en el origen de datos o en objetos de consulta modelo que ya se han definido en el modelo. Debe definir una clave de negocio y un título de tipo de cadena para cada nivel.

Dimensiones de medida

Las dimensiones de medida representan los datos cuantitativos descritos por las dimensiones normales. Una dimensión de medida, denominada de muchas otras formas en los diversos productos OLAP, es simplemente el objeto que contiene los datos de hechos.

Una dimensión de medida es una colección de hechos. Se puede crear una dimensión de medida para uno o varios objetos de consulta que tengan una relación sólida entre sí.

5.3. Requerimientos

- 1.- Recomendación de documentos al usuario en función de los tipos de documentos cogidos previamente.
- 2.- Se desea estudiar los recursos solicitados en préstamo con mayor afluencia, por biblioteca, colección y soporte material, de esta forma conocer las preferencias de los usuarios por tipo, edad y sexo.
- 3.- Se desea estudiar los autores solicitados con el mismo criterio que el apartado anterior.
- 4.- Estadísticas de afluencia de usuarios a las bibliotecas.
- 5.- Estadísticas sobre la vida de los documentos.
- 6.- Estadísticas sobre el estado de los documentos (perdidos, prestados, dañados)

5.4. Generación de informes. Recomendación de documentos

Según las especificaciones, a parte de los datos básicos para el funcionamiento de las bibliotecas, se quiere ofrecer alguna funcionalidad adicional como por ejemplo, recomendar documentos a los usuarios en función del tipo de documentos cogidos previamente.

Según las conversaciones realizadas con nuestro interlocutor con la empresa responsable de la totalidad del sistema, AllTogether SL, se acuerda que el criterio para recomendar un libro es el siguiente:

Dado un socio S:

- 1 Obtener los dos últimos recursos (libros) solicitados por S: L1 y L2
- 2 Obtener los socios que han solicitado alguna vez los libros L1 o L2
- 3 Obtener la lista de libros que en alguna ocasión han solicitado los socios obtenidos en el paso anterior.
- 4 Estos libros deben estar agrupados por título del recurso y ordenados en descendente según el contador de la agrupación.
- 5 Se seleccionan los 3 primeros recursos para recomendación y que no hayan sido solicitados previamente por el usuario S.

El algoritmo se traduce en recomendar la lectura, visión o audición de un recurso según las dos últimas solicitudes del usuario al que se pretende recomendar, en base a las preferencias del colectivo de usuarios de la red de bibliotecas.

Para ello se obtienen todas las solicitudes de préstamo de los socios que han leído alguno de los dos últimos documentos solicitados por el usuario. Se agrupan todos los libros y se seleccionan los tres recursos que han sido solicitados más veces. De esta lista se han excluido los recursos que ya han sido solicitados por el usuario en el pasado.

A continuación se presenta la consulta a la base de datos para obtener este resultado. Esta consulta se puede dar como un procedimiento almacenado de la base de datos en el que la entrada es el código de usuario y la salida es un cursor con los títulos seleccionados. Sin embargo, para esta consulta he preferido incluirla como un informe de Impromptu que en cualquier caso podría ser accedida a través de las API's del producto desde el aplicativo.

Para ello, en Impromptu se ha definido un catálogo con la base de datos Oracle del operacional y se han definido todos los Joins para acceder a los datos. Este catálogo se presenta en un entregable llamado: *PFC-Operacional.cat*.

A continuación se crea un nuevo informe (*Report*) con este catálogo y se incluye la consulta que realiza la recomendación. Se configura la presentación del informe y se graba como *PFC-Recomendacion.imr*.

La generación de estos ficheros se describe con mayor detalle en los apartados "7.9.1 Crear un catálogo con Impromptu" y "7.9.2. Definición de informes con Impromptu"

Consulta de recomendación de recursos

```
SELECT Q5.SOCIOID, Q5.APELLIDOS, Q5.NOMBRE, Q0.RECURSOID, Q1.TITULO, Q6.DESCRIPCION,
       Q2.CODIGOBARRAS, Q4.DESCRIPCION
FROM RECURSO Q1, EJEMPLAR Q2, HISTESTADO Q3, ESTADO Q4, SOCIO Q5, COLECCION Q6,
(
  SELECT DISTINCT RECURSO.RECURSOID,
                  Count(PRESTAMO.EJEMPLARID) AS CuentaDeEJEMPLARID,
                  row_number()over (order by Count(PRESTAMO.EJEMPLARID) DESC) rn2
FROM (PRESTAMO INNER JOIN
      (EJEMPLAR INNER JOIN RECURSO ON EJEMPLAR.RECURSOID = RECURSO.RECURSOID)
      ON PRESTAMO.EJEMPLARID = EJEMPLAR.EJEMPLARID)
      INNER JOIN SOCIO ON PRESTAMO.SOCIOID = SOCIO.SOCIOID
WHERE SOCIO.SOCIOID in
(
  SELECT DISTINCT SOCIO.SOCIOID
FROM (PRESTAMO INNER JOIN
      (EJEMPLAR INNER JOIN RECURSO
        ON EJEMPLAR.RECURSOID = RECURSO.RECURSOID)
      ON PRESTAMO.EJEMPLARID = EJEMPLAR.EJEMPLARID)
      INNER JOIN SOCIO
        ON PRESTAMO.SOCIOID = SOCIO.SOCIOID
WHERE RECURSO.RECURSOID in
(
  SELECT RECURSOID
FROM
(
  SELECT DISTINCT R.RECURSOID, P.FECHADESDE,
                  row_number()over (order by P.FECHADESDE DESC) rn
FROM PRESTAMO P INNER JOIN
      (EJEMPLAR E INNER JOIN RECURSO R
        ON E.RECURSOID = R.RECURSOID)
      ON P.EJEMPLARID = E.EJEMPLARID
WHERE P.SOCIOID=?Socio?
order by P.FECHADESDE desc
)
)
WHERE rn <= 2
)
)
AND RECURSO.RECURSOID NOT IN
(
  SELECT DISTINCT R.RECURSOID
FROM PRESTAMO P INNER JOIN
      (EJEMPLAR E INNER JOIN
        RECURSO R ON E.RECURSOID = R.RECURSOID)
      ON P.EJEMPLARID = E.EJEMPLARID
WHERE P.SOCIOID=?Socio?
)
)
GROUP BY RECURSO.RECURSOID
order by CuentaDeEJEMPLARID desc
) Q0
WHERE Q0.RECURSOID = Q1.RECURSOID AND
       Q1.RECURSOID = Q2.RECURSOID AND
       Q2.EJEMPLARID = Q3.EJEMPLARID AND
       Q3.ESTADOID = Q4.ESTADOID AND
       Q3.FECHAHASTA IS NULL AND
       Q5.SOCIOID = ?Socio? AND
       Q1.COLECCIONID = Q6.COLECCIONID AND
       rn2 <=3
ORDER BY Q0.RECURSOID, Q2.EJEMPLARID
```

RECOMENDACIÓN DE RECURSOS**Usuario:** Jorge Blanca Sanchez

4

TITULO	COLECCION	EJEMPLAR	ESTADO
Freaky Friday	Easy Readers	8100404	checked out
		8100405	checked in
		8100406	checked out
		8100407	on hold
		8100408	on hold
		8100409	checked out
		8100410	checked out
		8100411	checked out
		8100412	checked in
		8100413	display area
		8100414	display area
8100415	on order		
The Indian in the Cupboard	Easy Readers	8100557	checked in
		8100558	checked out
		8100559	checked out
		8100560	checked out
		8100561	checked in
		8100562	checked out
		8100563	checked out
		8100564	checked out
8100565	checked out		
8100566	display area		
Little Women	Juvenile Fiction	8100634	checked out
		8100635	checked out
		8100636	checked in
		8100637	checked in
		8100638	shelving cart
		8100639	checked out
		8100640	checked out
		8100641	shelving cart
		8100642	display area
8100643	checked out		

Ilustración 7: Informe de Impromptu

El informe muestra los Títulos seleccionados para recomendar al usuario *Jorge Blanca Sánchez*, así como la colección a la que pertenecen, el código de barras de los ejemplares y el estado en que se encuentran.

5.5. Diagramas del modelo conceptual

Para el resto de requerimientos estadísticos se utilizará un almacén de datos diseñado siguiendo un modelo multidimensional, lo cual mejorará el rendimiento debido a la utilización de técnicas específicas para el almacenamiento de datos.

El diseño conceptual tiene por objetivo la construcción de una descripción abstracta y completa del problema. Comienza con el análisis de requerimientos de los usuarios y de reglas de negocio, y finaliza con la construcción de un esquema conceptual expresado en términos de un modelo conceptual.

En una primera fase se seleccionan los objetos relevantes para la toma de decisiones, y se especifica el propósito de utilizarlos como dimensiones y/o medidas.

Para el requerimiento “Se desea estudiar los recursos solicitados en préstamo con mayor afluencia, por biblioteca, colección y soporte material, de esta forma conocer las preferencias de los usuarios por tipo, edad y sexo”, el diagrama del modelo conceptual es el siguiente:

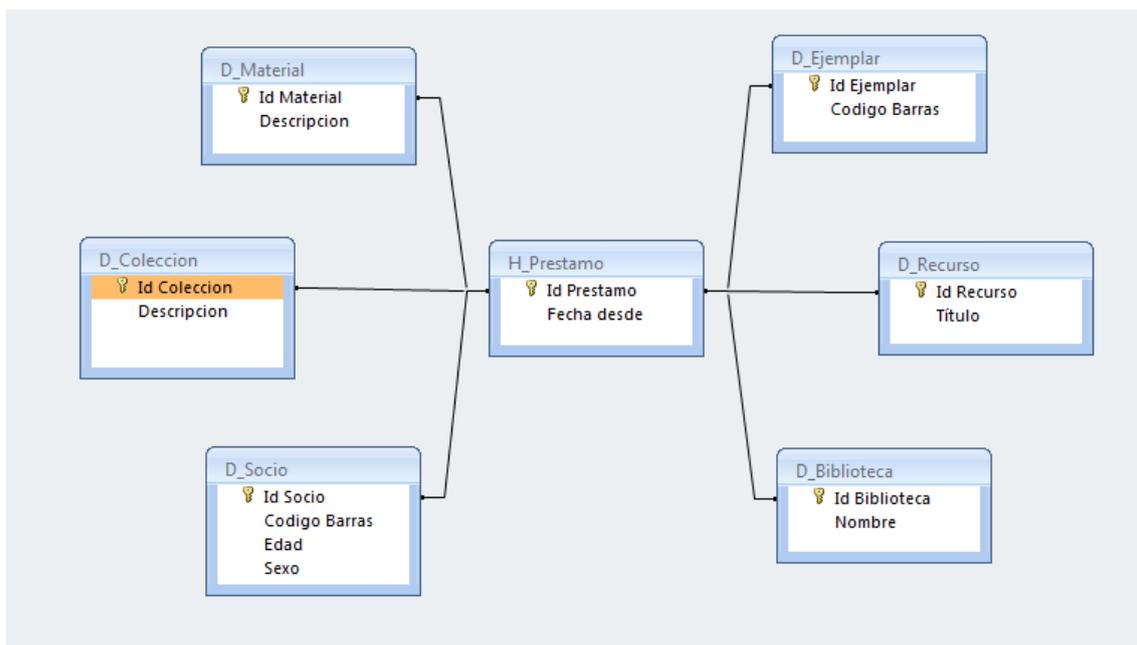


Ilustración 8: Diagrama conceptual 1

Un enfoque basado en requerimientos analiza los requerimientos de los usuarios, y se identifican en ellos los hechos, dimensiones y medidas relevantes. La realidad se modela como un conjunto de cubos multidimensionales, que se obtienen a partir de los hechos, dimensiones y medidas.

Hechos:	Préstamos
Dimensiones:	Recurso, Material, Colección, Biblioteca, Socio, Fecha
Medidas:	Número de préstamos

Atributos de cada dimensión

Establecidas las dimensiones y medidas se procede a determinar cuáles son los niveles de detalle y las jerarquías de agregación. Estos son los puntos en que cada dimensión se puede agrupar por niveles según una cierta jerarquía.

Dimensión Fecha:

Dado que disponemos de la información de por días de las fechas en que se distribuyen los préstamos se presentará la información por años y meses.

Dimensión Socio-Edad.

El concepto es estudiar el comportamiento de los socios subdividiéndolos por tipo (Adulto, Juvenil), por edad y socio.

Dimensión Socio-Sexo.

El concepto es estudiar el comportamiento de los socios subdividiéndolos por tipo (Adulto, Juvenil), por sexo y socio.

Dimensión Material.

Se desglosarán los ejemplares de las bibliotecas desglosándolos en Material, Recurso y Ejemplar.

Dimensión Colección.

Se desglosarán los ejemplares de las bibliotecas desglosándolos en Colección, Recurso y Ejemplar.

Dimensión Biblioteca.

Se dividen los préstamos diferenciándolos en las distintas bibliotecas, colecciones, materiales recursos y ejemplares.

Para el requerimiento “Se desea estudiar autores más solicitados, por biblioteca, colección y soporte material, de esta forma conocer las preferencias de los usuarios por tipo, edad y sexo”, el diagrama del modelo conceptual es el siguiente:

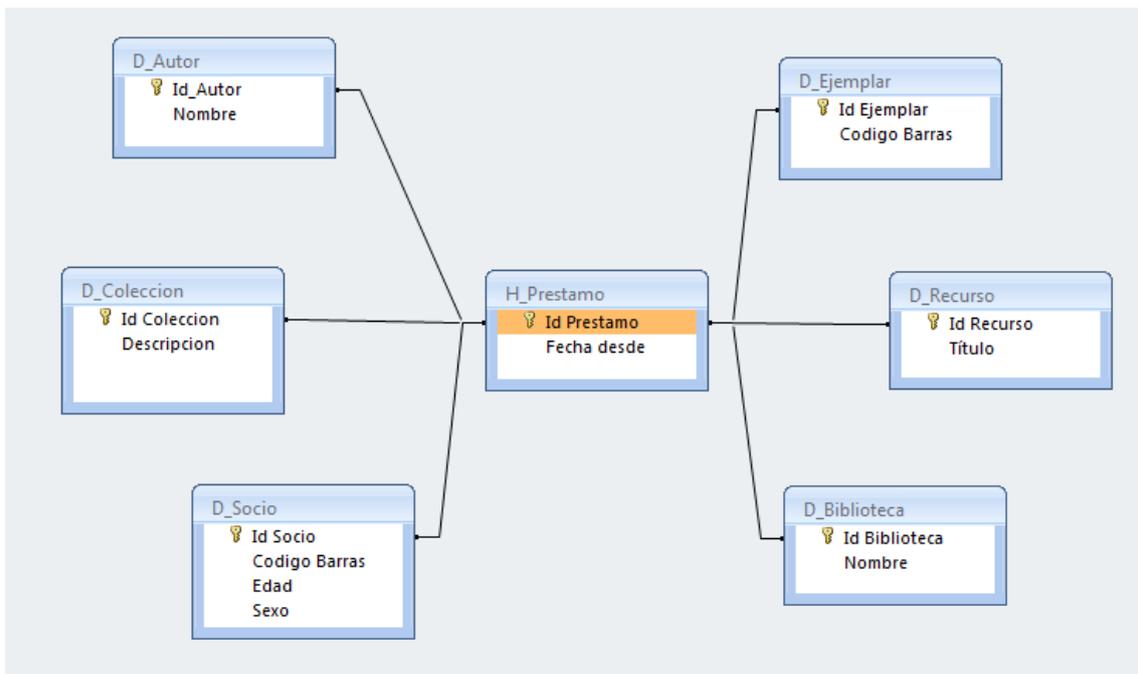


Ilustración 9: Diagrama conceptual 2

Los hechos, dimensiones y medidas, son los siguientes:

Hechos: Préstamos
Dimensiones: Autor, Colección, Biblioteca, Socio, Fecha
Medidas: Número de préstamos

Atributos de cada dimensión

Se procede a determinar cuáles son los niveles

Las dimensiones son las mismas que en el apartado anterior, sólo que en ésta parece la dimensión Autor.

Dimensión Autor.

Se desglosarán los ejemplares de las bibliotecas desglosándolos en Material, Recurso y Ejemplar.

El requerimiento “Estadísticas de afluencia de usuarios a las bibliotecas”, ya se ha recogido en los apartados anteriores puesto que en los préstamos hemos tomado una dimensión fecha y una dimensión socio.

Para el requerimiento “Estadísticas sobre la vida de los documentos”, el diagrama del modelo conceptual es el siguiente:

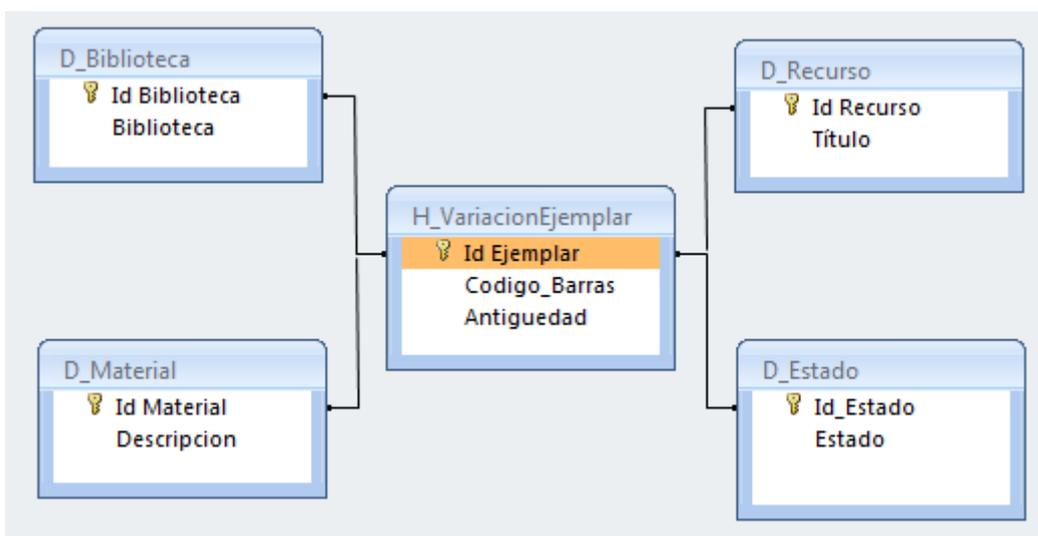


Ilustración 10: Diagrama conceptual 3

Los hechos, dimensiones y medidas, son los siguientes:

Hechos: Vida de documentos
Dimensiones: Biblioteca, Material, Recursos
Medidas: Antigüedad

Atributos de cada dimensión

Los niveles para las distintas dimensiones se llevan dividen en Recursos y Ejemplares.
Dimensión Biblioteca.

Se desglosarán los ejemplares de las bibliotecas desglosándolos en Material, Ejemplar

Dimensión Material.

Se desglosarán los ejemplares de las bibliotecas desglosándolos en Recurso, Ejemplar

Para el requerimiento “Estadísticas sobre el estado de los documentos (perdidos, prestados, dañados)”, el diagrama del modelo conceptual es el siguiente:

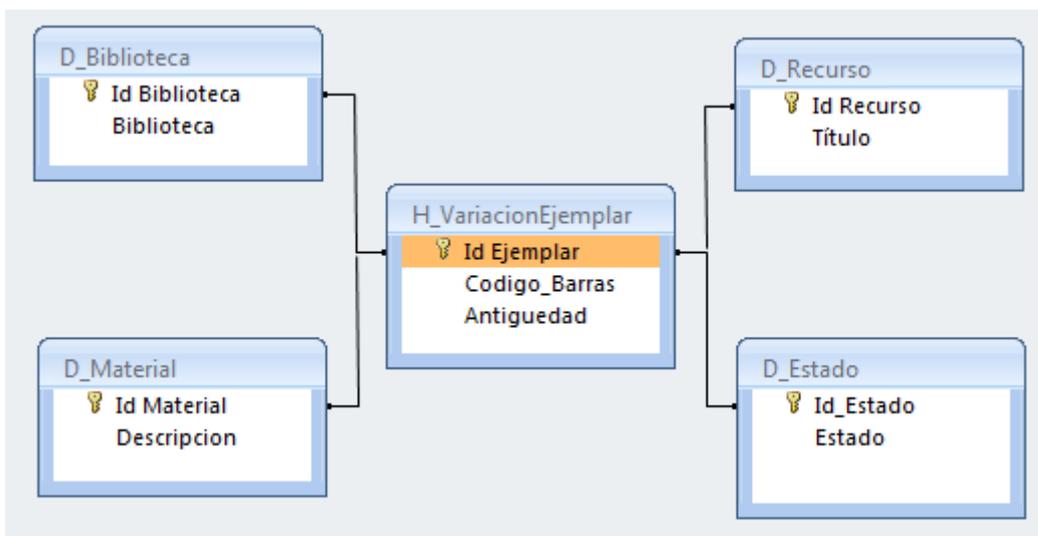


Ilustración 11: Diagrama conceptual 4

Los hechos, dimensiones y medidas, son los siguientes:

Hechos: Estado de documentos
Dimensiones: Biblioteca, Material, Colección
Medidas: Contador

Atributos de cada dimensión

Se procede a determinar cuáles son los niveles

Dimensión Biblioteca.

Se desglosarán los ejemplares de las bibliotecas desglosándolos en Material, Ejemplar

Dimensión Material.

Se desglosarán los ejemplares de las bibliotecas desglosándolos en Recurso, Ejemplar

Dimensión Colección.

Se desglosarán los ejemplares de las bibliotecas desglosándolos en Recurso, Ejemplar

5.6. Modificaciones en el modelo operacional

Como consecuencia de los requerimientos del sistema estadístico se ve la necesidad de recoger algunos datos de nuestro sistema operacional. Concretamente, se requiere obtener más datos de los socios para poder estudiar diferentes perfiles.

Se añaden las columnas fecha de nacimiento y sexo de la tabla SOCIO quedando de la siguiente instrucción DDL:

```
CREATE TABLE Socio (
  socioId          INTEGER          CONSTRAINT PK_socio          PRIMARY KEY,
  codigoBarras    CHAR(8)          CONSTRAINT NN_soc_barras NOT NULL,
  tipoId          INTEGER          CONSTRAINT NN_tip_tipo    NOT NULL,
  apellidos       VARCHAR2(100)    CONSTRAINT FK_soc_tip_tip REFERENCES Tipo(tipoId),
  nombre         VARCHAR2(100),
  telefono        VARCHAR2(16)     CONSTRAINT NN_soc_direccion NOT NULL,
  correoElectronico VARCHAR2(100)  CONSTRAINT NN_soc_email   NOT NULL,
  fechaAlta      DATE              CONSTRAINT NN_soc_fechaAl  NOT NULL,
  fechaBaja      DATE,
  ultimaModFecha DATE              CONSTRAINT NN_soc_ultModF   NOT NULL,
  ultimaModEmpl  INTEGER          CONSTRAINT NN_soc_ultModE   NOT NULL,
  fechaNacto     DATE              CONSTRAINT FK_soc_emp_emp   REFERENCES Empleado(empleadoId),
  sexo          CHAR(1)            CONSTRAINT NN_soc_fecnto  NOT NULL,
  CONSTRAINT UN_Socio UNIQUE(codigoBarras)
);
```

Se modifica el archivo GUION2.SQL con las modificaciones comentadas. También se modifica el archivo GUION3.SQL que contiene los procedimientos almacenados de alta, baja y modificación incluyendo los dos nuevos campos.

5.7. Preparación del plan de pruebas.

Para realizar las pruebas y desarrollar los procesos estadísticos propios del almacenamiento de datos se requiere un completo juego de pruebas.

Se modifica el fichero GUION5.SQL introduciendo los siguientes cambios:

- Se ejecuta el procedimiento AltaRecurso para 150 títulos y subtítulos de recursos, asignándoles aleatoriamente un material y una colección
- Se ejecuta el procedimiento AltaAutor para 118 autores distintos asignando un nombre de autor a cada fila.
- Se ejecuta el procedimiento AltaRecursoAutor para 165 relaciones entre autores y recursos. Se incluye el caso de un recurso con varios autores así como un recurso escrito por varios autores.

- Se ejecuta el procedimiento AltaEjemplar para 1000 ejemplares. A cada ejemplar se le asocia aleatoriamente una biblioteca y un recurso. También se le asigna un número correspondiente al código de barras y una fecha de alta aleatoria dentro del primer trimestre de 2008. Los 400 primeros tienen fecha de baja comprendido en el último trimestre de 2009. Estas fechas se utilizarán para estudiar la vida de los documentos.
- Se ejecuta el procedimiento AltaSocio para 58 socios. A parte de los datos característicos de un socio, se incluye la fecha de nacimiento y el sexo.
- Se ejecuta el procedimiento AltaHistoricoEstado para asignar un estado y una fecha a cada uno de los ejemplares de las bibliotecas.
- Se ejecuta el procedimiento AltaPrestamo para 900 préstamos. Cada una de estas entradas corresponde a la acción de realizar un préstamo de un ejemplar a un socio. Para ello se asigna aleatoriamente el código de usuario y el código de ejemplar, así como la fecha en que se realiza el préstamo.

5.8. Proceso de integración y transformación. ETL

El componente de integración y transformación está formado por un script SQL que se ejecuta en el entorno Oracle. En este proceso se podrían incluir otros aplicativos que corran en paralelo dentro de la gestión de bibliotecas, tales como Contabilidad, Compras, Nómina. El proceso ETL incluiría todos los orígenes de datos y la información contenida en ellos. Sin embargo, en este proyecto se limita a la propia gestión de la biblioteca desarrollada en el apartado de análisis operacional.

A partir de los orígenes de datos se obtienen los datos requeridos por los almacenes de datos. Los almacenes guardan la historia, mientras que el sistema operacional tan solo mantiene la situación actual. Por lo tanto, partiendo de sistemas que mantienen una sola imagen (snapshot) de los datos, obtenemos aquellos que son necesarios para construir su historia.

El guión *GUION6-ETL.sql* ejecuta el proceso de borrado y creación de las tablas de dimensiones para el sistema analítico. Asimismo, este procedimiento también realiza la transformación de los datos partiendo del sistema operacional, cumplimentando las tablas para su posterior proceso en el sistema multidimensional.

La primera vez que se ejecuta este procedimiento, se deben comentar las instrucciones DROP de tablas si se quiere evitar que den error, dado que no existen.

5.9. Implementación con Cognos

5.9.1. Creación de catálogos con Impromptu.

Partiendo del almacén de datos operacional con los pasos descritos en el proceso de integración e transformación, se cargan los datos en COGNOS creando un catálogo importando los datos desde Oracle.

La información que contiene el catálogo es el tipo de acceso ODBC a nuestra base de datos 'Biblioteca' y a las tablas que intervienen en este proceso de Business Intelligence. Además, con la utilidad 'Joins' informaremos sobre los las relaciones entre las tablas.

En la siguiente captura de pantalla se muestran las tablas que intervienen en los requerimientos que giran en torno al Hecho Préstamo así como la definición de los 'Joins'.

El nombre de este catálogo es PFC-ETL_Prestamo.cat y se suministra en la carpeta de entregables.

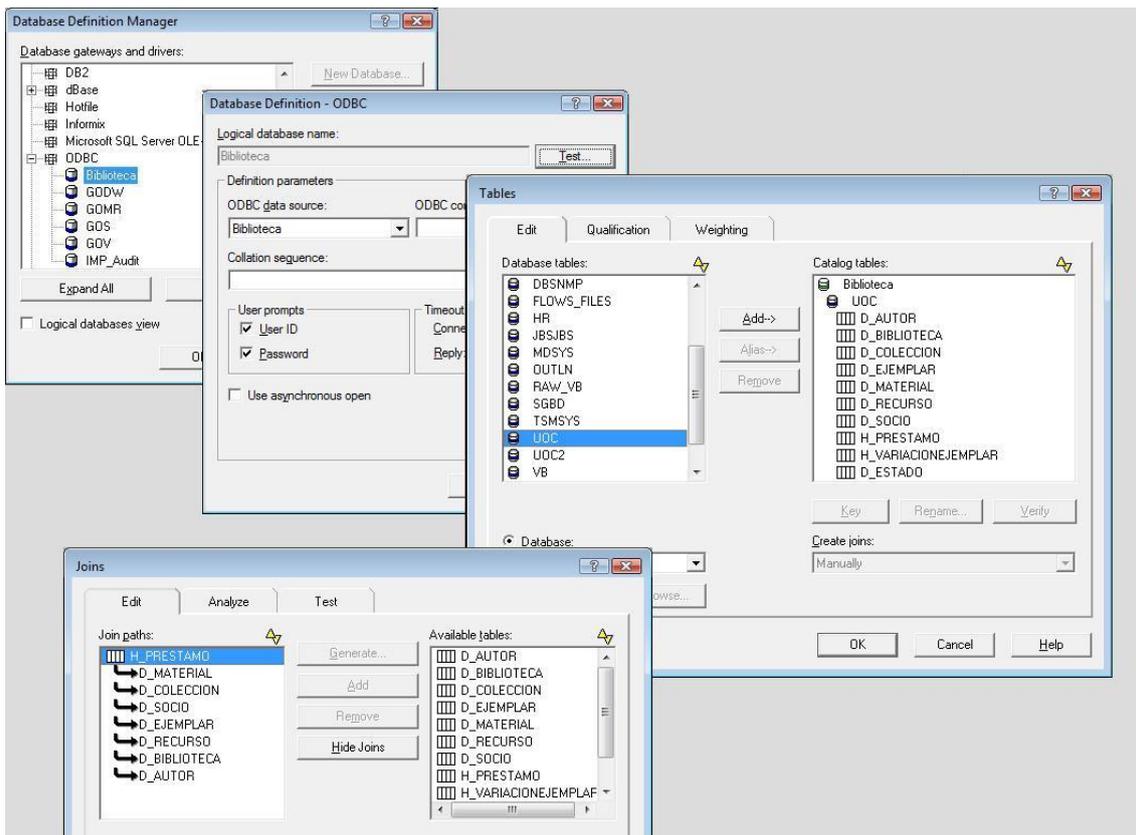


Ilustración 12: Definición del catálogo Impromptu

Para los requerimientos 5 y 6 se requiere un catálogo distinto que gira en torno al hecho H_VariaciónEhemplar, definido en el modelo conceptual. La siguiente captura de pantalla muestra las tablas que intervienen en los requerimientos así como la definición de los 'Joins'.

El nombre de este catálogo es PFC-ETL_Ejemplar.cat y se suministra en la carpeta de entregables.

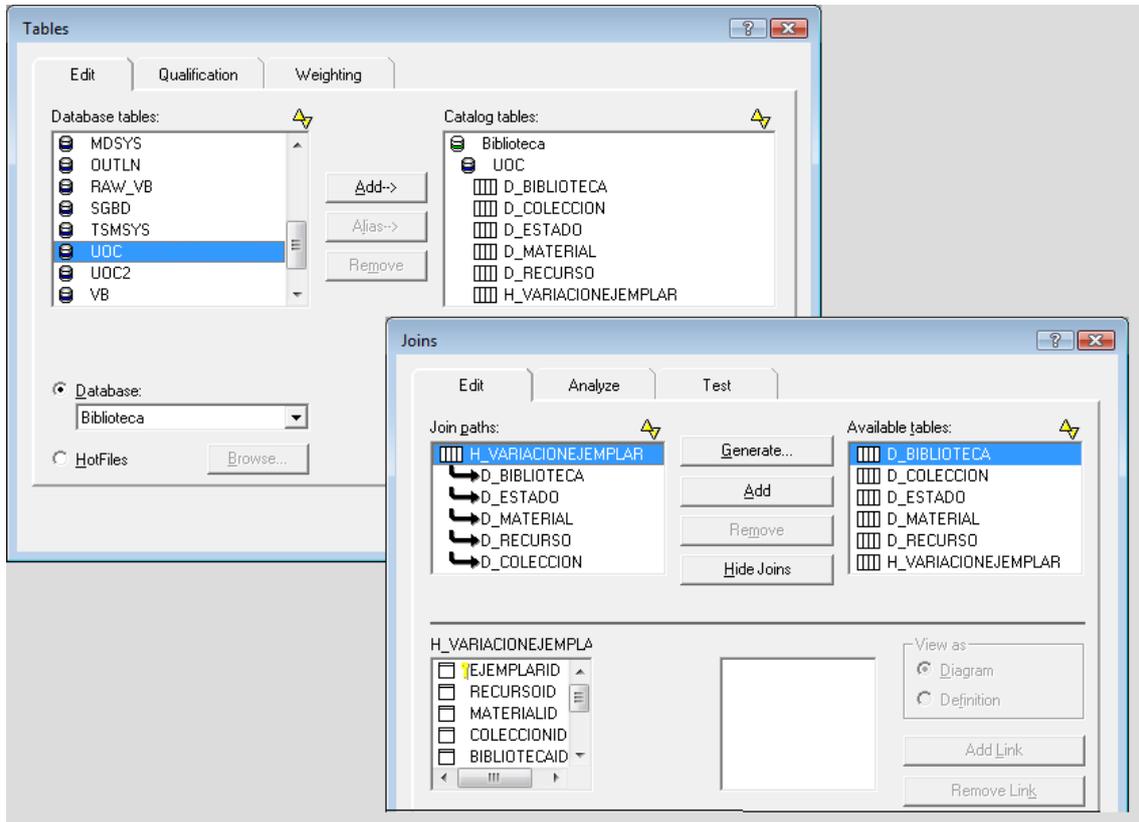


Ilustración 13: Definición del catálogo Impromptu

El nombre de este catálogo es PFC-ETL_Ejemplar.cat y se suministra en la carpeta de entregables.

5.9.2. Definición de informes con Impromptu

Una vez definido el catálogo, ya se está en condiciones de definir un nuevo informe (report). Para ello se puede utilizar un wizard en el que se van definiendo como se obtienen o calculan cada una de las columnas del informe. Además se pueden definir totalizadores o sumadores, aunque éste no es el objetivo, puesto que lo que necesitamos es un informe plano en el que las columnas correspondan a las dimensiones y medidas para ser grabado como un objeto IQD, que será la base del siguiente paso.

Para los requerimientos 2 y 3: “Se desea estudiar los recursos solicitados en préstamo con mayor afluencia, por biblioteca, colección y soporte material, de esta forma conocer las preferencias de los usuarios por tipo, edad y sexo” y “Se desea estudiar los recursos solicitados en préstamo con mayor afluencia, por biblioteca, colección y soporte material, de esta forma conocer las preferencias de los usuarios por tipo, edad y sexo”. La definición del informe es la siguiente:

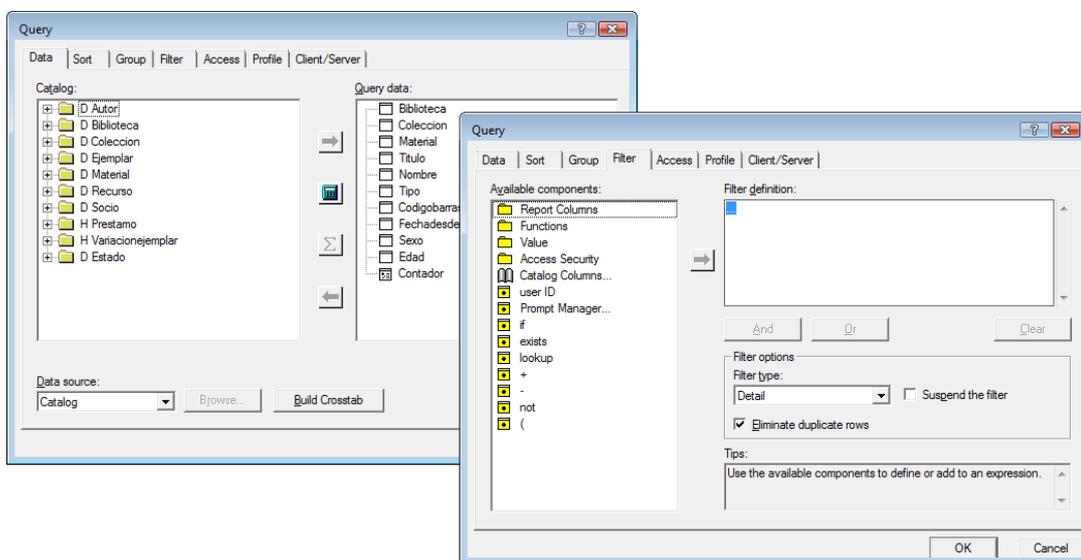


Ilustración 14: Definición Query Impromptu

La consulta generada por es

```
select distinct T1."BIBLIOTECA" "c1", T2."COLECCION" "c2", T3."MATERIAL" "c3",
              T4."TITULO" "c4", T5."TIPO" "c5", T6."CODIGOBARRAS" "c6",
              T7."FECHADESDE" "c7", T5."SEXO" "c8", T5."EDAD" "c9", 1 "c10"
from "UOC"."H_PRESTAMO" T7, "UOC"."D_BIBLIOTECA" T1, "UOC"."D_COLECCION" T2,
     "UOC"."D_MATERIAL" T3, "UOC"."D_RECURSO" T4, "UOC"."D_SOCIO" T5,
     "UOC"."D_EJEMPLAR" T6
where T7."BIBLIOTECAID" = T1."BIBLIOTECAID" and
      T7."COLECCIONID" = T2."COLECCIONID" and
      T7."MATERIALID" = T3."MATERIALID" and
      T7."RECURSOID" = T4."RECURSOID" and
      T7."SOCIOID" = T5."SOCIOID" and
      T7."EJEMPLARID" = T6."EJEMPLARID"
```

El nombre de este informe es *PFC-1-Documentos.imr* y se suministra en el entregable que acompaña a este documento. También se exporta como *PFC-1-Documentos.iqd* para utilizarlo en el siguiente paso

Para el requerimiento 4: “Se desea estudiar los autores solicitados con el mismo criterio que el apartado anterior”, la definición del informe es la siguiente:

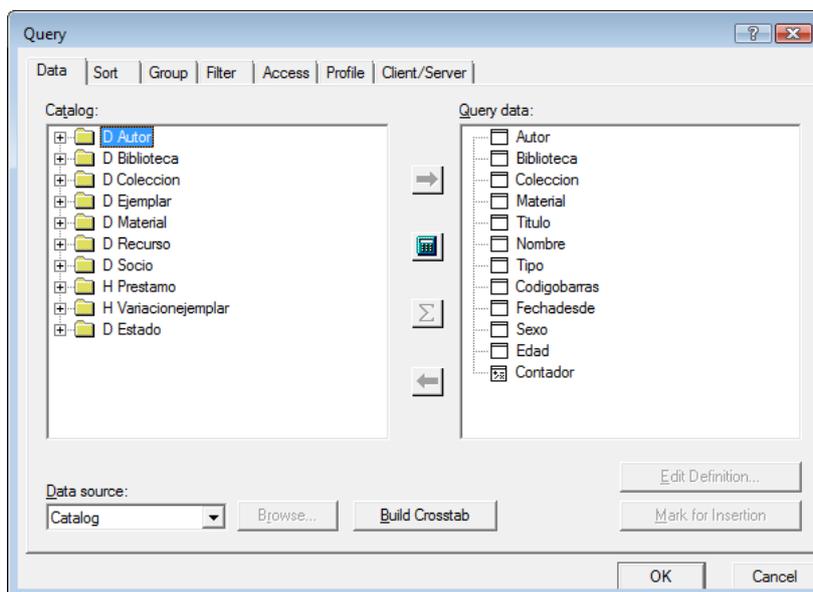


Ilustración 15: Definición Query Impromptu

La consulta generada por el wizard es:

```
select  T1."NOMBRE" "c1" , T2."BIBLIOTECA" "c2" , T3."COLECCION" "c3" ,
        T4."MATERIAL" "c4" , T5."TITULO" "c5" , T6."NOMBRE" "c6" , T6."TIPO" "c7" ,
        T7."CODIGOBARRAS" "c8" , T8."FECHADESDE" "c9" , T6."SEXO" "c10" ,
        T6."EDAD" "c11" , 1 "c12"
from    "UOC"."H_PRESTAMO" T8, "UOC"."D_AUTOR" T1, "UOC"."D_BIBLIOTECA" T2,
        "UOC"."D_COLECCION" T3, "UOC"."D_MATERIAL" T4, "UOC"."D_RECURSO" T5,
        "UOC"."D_SOCIO" T6, "UOC"."D_EJEMPLAR" T7
where   T8."AUTORID" = T1."AUTORID" and
        T8."BIBLIOTECAID" = T2."BIBLIOTECAID" and
        T8."COLECCIONID" = T3."COLECCIONID" and
        T8."MATERIALID" = T4."MATERIALID" and
        T8."RECURSOID" = T5."RECURSOID" and
        T8."SOCIOID" = T6."SOCIOID" and
        T8."EJEMPLARID" = T7."EJEMPLARID"
```

El nombre de este informe es *PFC-2-Autores.imr* y se suministra en el entregable que acompaña a este documento. También se exporta como *PFC-2-Autores.iqd* para utilizarlo en el paso Transformer.

Para los requerimientos 5 y 6: “Estadísticas sobre la vida de los documentos” y “Estadísticas sobre el estado de los documentos (perdidos, prestados, dañados)” la definición del informe es la siguiente:

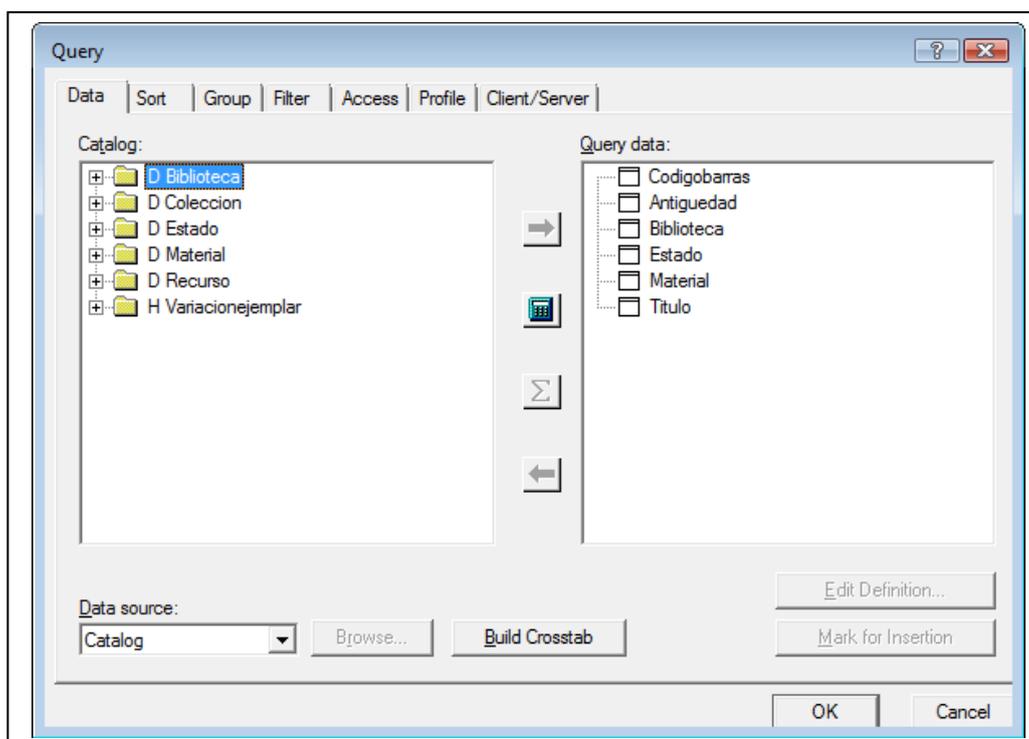


Ilustración 16: Definición Query Impromptu

La consulta generada por el wizard es:

```
select T1."CODIGOBARRAS" "c1", T1."ANTIGUEDAD" "c2", T2."BIBLIOTECA" "c3",
      T3."ESTADO" "c4", T4."MATERIAL" "c5", T5."TITULO" "c6"
from "UOC"."H_VARIACIONEJEMPLAR" T1, "UOC"."D_BIBLIOTECA" T2,
     "UOC"."D_ESTADO" T3, "UOC"."D_MATERIAL" T4, "UOC"."D_RECURSO" T5
where T1."BIBLIOTECAID" = T2."BIBLIOTECAID" and
      T1."ESTADOID" = T3."ESTADOID" and
      T1."MATERIALID" = T4."MATERIALID" and
      T1."RECURSOID" = T5."RECURSOID"
```

El nombre de este informe es *PFC-3-Ejemplares.imr* y se suministra en el entregable que acompaña a este documento. También se exporta como *PFC-3-Ejemplares.iqd* para utilizarlo en el paso Transformer.

5.9.3. Definición de cubos OLAP con Transformer

Transformer es el componente Cognos utilizado para definir datos y estructuras de cubos OLAP en modelos PowerPlay y construir los cubos multidimensionales llamados PowerCubes.

Cuando se crea un nuevo proyecto de Transformer, se presenta un wizard para configurar el nuevo modelo. A continuación nos solicita el tipo de origen de datos y le indicamos que es de tipo 'Impromptu Query Definition' y entramos el fichero creado en el paso anterior.

El propio wizard nos presenta 5 ventanas con un diseño automático del origen de datos, mapa de dimensiones, medidas del modelo Transformer. Generalmente debe ser retocado, puesto que las posibilidades de diseño son muy variadas.

Tal como se aprecia en el dibujo, se definen los 'Dimension Map' según se ha declarado en el modelo conceptual, atendiendo a los requerimientos. Se definen las dimensiones; 'Fecha', 'Biblio-Material', 'Biblio-Coleccion', 'Colección', 'Material', 'Código de barras', 'Socio.Sexo' y 'Socio-Edad' y la Medida 'Contador'.

Con estas definiciones ya se puede crear un PowerCube con la información necesaria para contener el cubo OLAP.

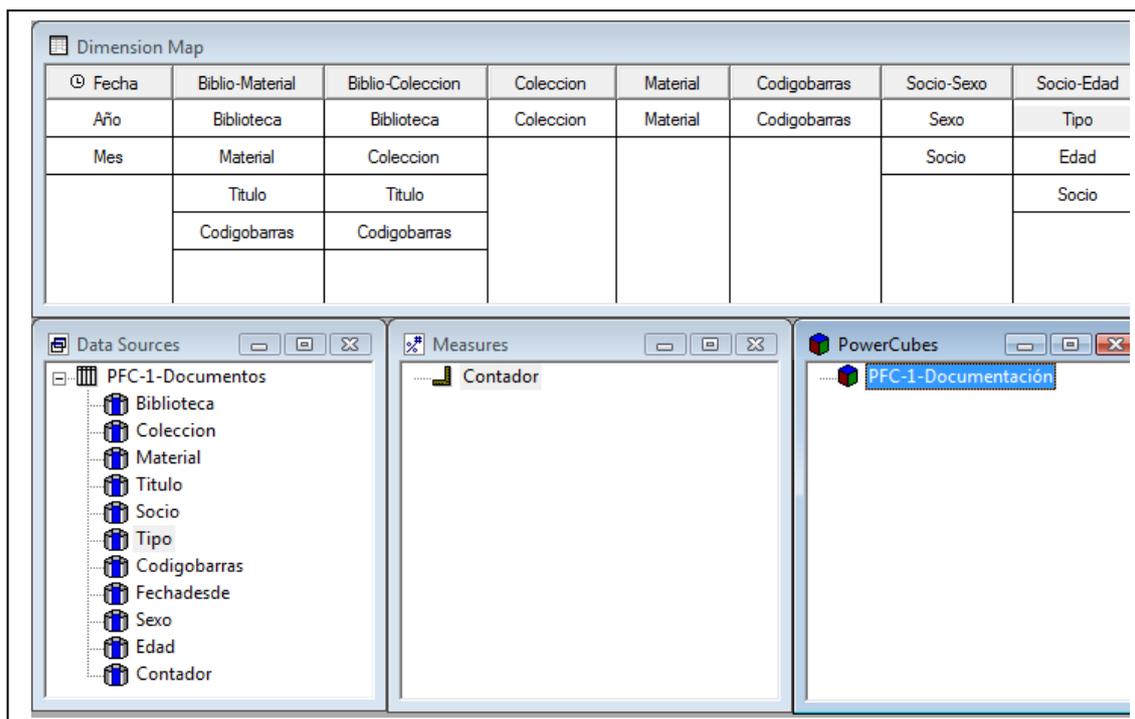


Ilustración 17: Definición PowerCube Transformer

El fichero generado en esta fase Cognos-Transformer se denomina *PFC-1-Documentación.pyi* y el cubo OLAP generado *PFC-1-Documentación.mdc*. Ambos se encuentran en el directorio de Entregables.

El resultado de este paso es un fichero con la extensión mdc (PowerCube).

Para generar el cubo correspondiente al requerimiento de autores más leídos se define el siguiente fichero transformer:

The screenshot shows the PowerCube Transformer interface. At the top is the 'Dimension Map' table. Below it are three panes: 'Data Sources' showing a tree view of dimensions, 'Measures' showing 'Contador', and 'PowerCubes' showing 'PFC-2-Autor'.

Fecha	Biblio-Material	Biblio-Coleccion	Coleccion	Material	Autor	Socio
Año	Biblioteca	Biblioteca	Coleccion	Material	Autor	Edad
Mes	Material	Coleccion				Autor
	Autor	Autor				

Ilustración 18: Definición PowerCube Transformer

Como puede observarse, se definen las dimensiones Fecha, Biblioteca, Colección, Material, Autor y Socio y la medida es el contador de préstamos realizados.

El fichero generado en esta fase se denomina *PFC-2-Autor.pyi* y el cubo OLAP generado *PFC-2-Autor.mdc*. Ambos se encuentran en el directorio de Entregables.

5.9.4. Representación de los datos con PowerPlay

PowerPlay es el último de los componentes para la obtención de informes y gráficos con cubos multidimensionales. Con powerPlay se pueden visualizar los informes y gráficos a partir de los cubos creado en el apartado anterior.

Al abrir el cubo obtenemos información que deseamos analizar en términos de hechos y dimensiones, de manera que la podemos situar en un espacio n-dimensional.

A un toque de click con el ratón del ordenador, se pueden practicar todas las técnicas conocidas en el análisis multidimensional, como:

- el 'slide' consistente en cortar el cubo de manera que se reduzcan el número de dimensiones.
- El 'dice' donde se selecciona un subespacio del cubo original, sin reducir el número de dimensiones, seleccionando un subconjunto de valores en cada una de las dimensiones.
- El roll-up que agrupa celdas del cubo siguiendo una jerarquía de agrupación.
- El drill-down que aumenta el detalle con que se ven los datos. Es la operación inversa al roll-up.
- El drill-across consistente en cambiar el tema del análisis.

Requerimiento 2:

Se desea estudiar los recursos solicitados en préstamo con mayor afluencia, por biblioteca, colección y soporte material, de esta forma conocer las preferencias de los usuarios por tipo, edad y sexo.

Utilizaremos el PowerCube *pfc-1-documentación.mdc* obtenido en la fase Transformer y obtenemos los resultados que a continuación se detallan:

La siguiente captura de pantalla representa los 900 préstamos realizados por las dos bibliotecas (Biblioteca Cataluña y Biblioteca UOC) desglosando por la tipología de socio (juvenil y adulto) por las diferentes colecciones que tiene las bibliotecas.

		Computer Software	Cassettes	Compact Discs	Videos and DVDs	Adult Fiction	Easy Readers	Juvenile Fiction	Juvenile Nonfiction	Adult Nonfiction	New Books	Coleccion
Biblioteca Cataluña	adult	57	57	38	6	13	56	46	39	25	31	368
	juvenile	12	6	9	3	1	16	19	11	6	9	92
	Socio-Edad	69	63	47	9	14	72	65	50	31	40	460
Biblioteca UOC	adult	65	61	34	10	13	62	35	43	24	18	365
	juvenile	11	13	8	1	6	11	10	7	5	3	75
	Socio-Edad	76	74	42	11	19	73	45	50	29	21	440
Biblio-Material	adult	122	118	72	16	26	118	81	82	49	49	733
	juvenile	23	19	17	4	7	27	29	18	11	12	167
	Socio-Edad	145	137	89	20	33	145	110	100	60	61	900

Ilustración 19: Tabla 1 PowerPlay Préstamos

Con una sencilla operación de arrastrar y soltar, se ha cambiado el tema Biblioteca por el de Fecha distribuyéndose los 900 préstamos por los mismos conceptos pero ahora entre los años 2008 y 2009 que son los años en que se realizaron los 900 préstamos.

		Computer Software	Cassettes	Compact Discs	Videos and DVDs	Adult Fiction	Easy Readers	Juvenile Fiction	Juvenile Nonfiction	Adult Nonfiction	New Books	Coleccion
2008	adult	65	64	36	9	12	66	45	39	28	27	391
	juvenile	14	10	7	2	2	13	17	2	5	2	74
	Socio-Edad	79	74	43	11	14	79	62	41	33	29	465
2009	adult	57	54	36	7	14	52	36	43	21	22	342
	juvenile	9	9	10	2	5	14	12	16	6	10	93
	Socio-Edad	66	63	46	9	19	66	48	59	27	32	435
Fecha	adult	122	118	72	16	26	118	81	82	49	49	733
	juvenile	23	19	17	4	7	27	29	18	11	12	167
	Socio-Edad	145	137	89	20	33	145	110	100	60	61	900

Ilustración 20: Tabla 2 PowerPlay Préstamos

Ahora seleccionamos un subespacio correspondiente a los 12 meses del año 2009.

		Computer Software	Cassettes	Compact Discs	Videos and DVDs	Adult Fiction	Easy Readers	Juvenile Fiction	Juvenile Nonfiction	Adult Nonfiction	New Books	Coleccion
2009/Jan	juvenile	1	1	0	0	0	2	2	4	0	2	12
2009/Feb	adult	4	1	4	0	0	6	1	8	1	1	26
	juvenile	2	2	0	0	0	2	1	0	2	1	10
2009/Mar	adult	3	4	7	3	0	6	4	2	1	1	31
	juvenile	1	1	0	0	0	1	0	0	1	0	4
2009/Apr	adult	4	4	1	0	1	6	3	3	1	4	27
	juvenile	1	1	0	0	0	2	1	4	0	0	9
2009/May	adult	6	5	1	0	1	4	6	4	0	1	28
	juvenile	0	1	2	0	2	1	2	2	1	1	12
2009/Jun	adult	4	8	1	2	2	1	4	6	2	2	32
	juvenile	2	1	2	1	0	1	1	0	0	1	9
2009/Jul	adult	6	3	3	1	0	3	5	4	3	1	29
	juvenile	2	0	1	0	1	0	1	0	0	3	8
2009/Aug	adult	6	3	2	0	0	3	3	2	3	0	22
	juvenile	0	1	0	1	1	0	2	0	0	2	7
2009/Sep	adult	3	3	2	0	1	6	1	0	1	4	21
	juvenile	0	0	2	0	0	1	2	2	0	0	7
2009/Oct	adult	3	5	3	0	4	2	1	3	4	0	25
	juvenile	0	0	1	0	1	1	0	1	0	0	4
2009/Nov	adult	4	4	5	0	2	4	3	3	2	2	29
	juvenile	0	1	1	0	0	2	0	2	0	0	6
2009/Dec	adult	4	4	3	0	0	6	1	3	2	3	26
	juvenile	0	0	1	0	0	1	0	1	2	0	5

Ilustración 21: Tabla 3 PowerPlay Préstamos

También se puede disponer de un gráfico tridimensional donde se vean Los distintos títulos de los recursos de la biblioteca repartidos por colecciones y en vertical ver los títulos que han sido solicitados con mayor frecuencia en los dos últimos años.

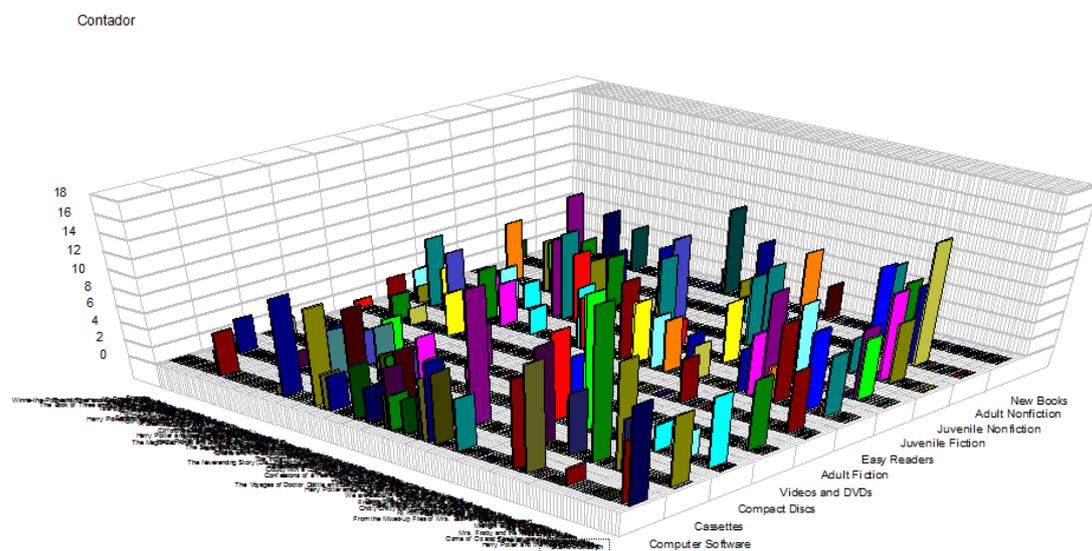


Ilustración 22: Gráfico 1 PowerPlay Préstamos

Requerimiento 3:

Se desea estudiar los autores solicitados con el mismo criterio que el apartado anterior.

Utilizaremos el PowerCube *pfc-2-autor.mdc* obtenidos en la fase Transformer y obtenemos los resultados que a continuación se detallan:

La siguiente captura de pantalla representa los 1001 veces que un autor fue solicitado por medio de un libro del cual es autor. Debe considerarse que un libro puede tener más de un autor. Por este motivo, por 900 préstamos de ejemplares aparecen lass 1001 veces que un autor es leído.

En el siguiente cuadro se representa este reparto por año y colección.

	Compact Discs	Easy Readers	Juvenile Fiction	Computer Software	Cassettes	Adult Nonfiction	New Books	Adult Fiction	Videos and DVDs	Juvenile Nonfiction	Coleccion
2008	53	83	68	86	81	33	30	18	14	47	513
2009	51	68	55	71	73	27	34	22	13	74	488
Fecha	104	151	123	157	154	60	64	40	27	121	1001

Ilustración 23: Tabla 1 PowerPlay Autores

A continuación se muestra un gráfico con los autores que han sido leídos por socios de 20 años. El autor más leído ha sido Thomas M. Dish por un recurso de la colección Equipment y ha sido solicitado 2 veces por este perfil de socio.

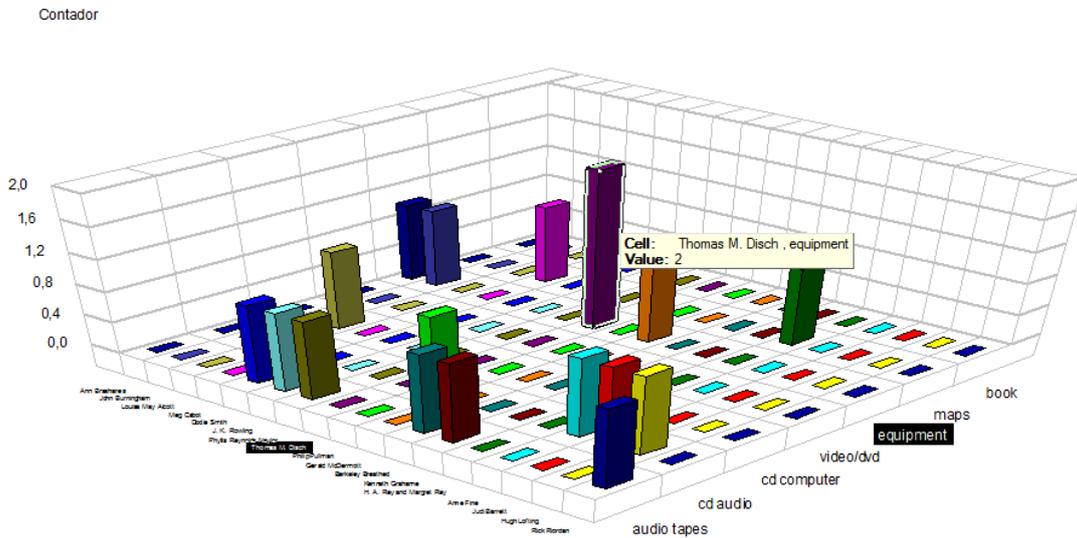


Ilustración 24: Gráfico 1 PowerPlay Autores

Requerimiento 4:

Estadísticas de afluencia de usuarios a las bibliotecas.

Este requerimiento se resuelve con el mismo PowerCube con el que se dio solución al requerimiento 2. Para obtener la afluencia de socios utilizaremos las dimensiones Fecha y Socios.

En la siguiente tabla y gráfico se observan las afluencias del año 2008 medidas en términos de recursos solicitados con los socios desglosados por su tipología.

		adult	juvenile	
2008	2008/Mar	36	4	
	2008/Apr	32	6	
	2008/May	37	2	
	2008/Jun	36	12	
	2008/Jul	32	7	
	2008/Aug	30	5	
	2008/Sep	33	6	
	2008/Oct	29	4	
	2008/Nov	33	10	
	2008/Dec	27	5	
	2009	2009/Jan	46	12
		2009/Feb	26	10
2009/Mar		31	4	
2009/Apr		27	9	
2009/May		28	12	
2009/Jun		32	9	
2009/Jul		29	8	
2009/Aug		22	7	
2009/Sep		21	7	
2009/Oct		25	4	
2009/Nov		29	6	
2009/Dec		26	5	

Ilustración 25: Tabla 1 PowerPlay Afluencia

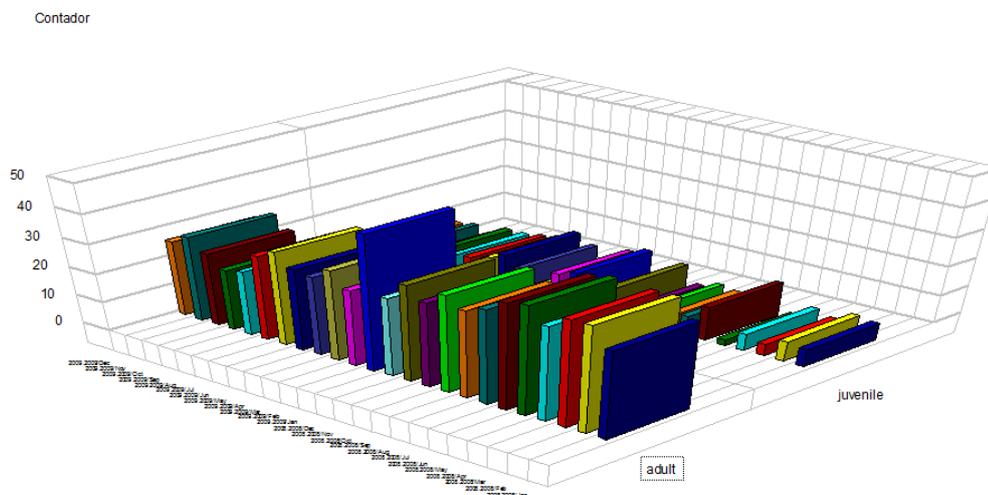


Ilustración 26: Gráfico 1 PowerPlay Afluencia

Requerimiento 5:

Estadísticas sobre la vida de los documentos.

Utilizaremos el PowerCube *pfc-3-ejemplar.mdc* obtenidos en la fase Transformer y obtenemos los resultados que a continuación se detallan:

La siguiente captura de pantalla la antigüedad en meses de los 4 ejemplares del libro 'Harry Potter and the Philosopher's Stone' distinguiendo las dos librerías donde se encuentran.

	8100448	8100449	8100450	8100451
Biblioteca UOC	29	0	0	0
Biblioteca Cata	0	30	31	31

Ilustración 27: Tabla 1 PowerPlay Vida de Documentos

Y una representación gráfica de la misma tabla:

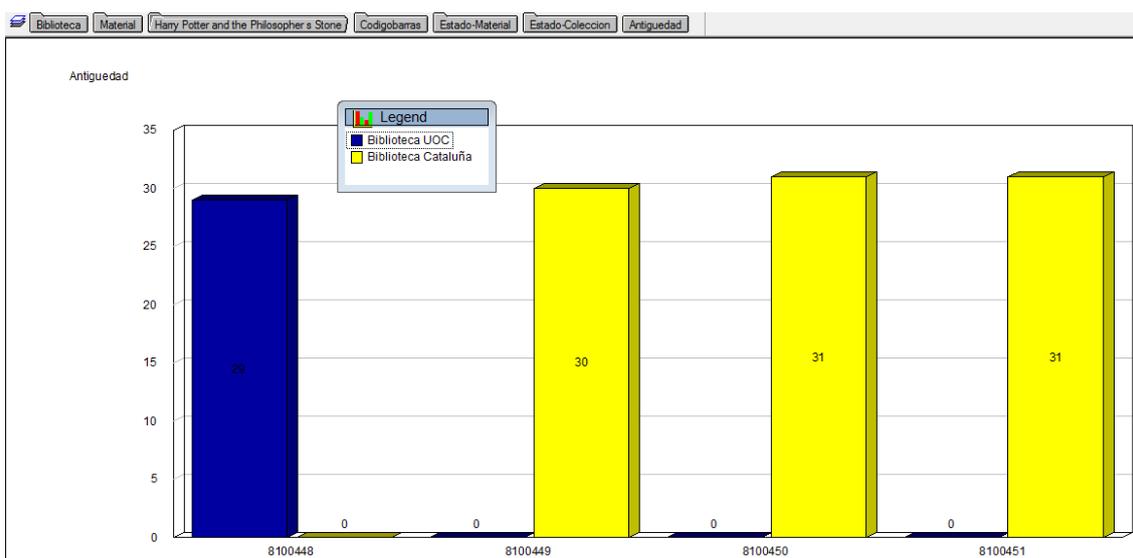


Ilustración 28: Gráfico 1 PowerPlay Vida de Documentos

En el siguiente cuadro se representa este reparto por año y colección.

Requerimiento 6:

Estadísticas sobre el estado de los documentos (perdidos, prestados, dañados)

Utilizaremos de nuevo, el PowerCube *pf3-3-ejemplar.mdc* obtenido en la fase Transformer y obtenemos los resultados que a continuación se detallan:

La siguiente tabla muestra es estado en que se encuentran los ejemplares en función del material en que están confeccionados.

	checked out	display area	shelving cart	checked in	lost	damaged/mending	on hold	on order	on loan	Estado-Material
equipment	60	29	5	20	5	4	4	0	0	127
maps	60	44	1	50	12	8	3	0	0	178
audio tapes	63	34	3	45	10	7	6	3	1	172
book	48	22	3	39	7	3	7	3	3	135
cd audio	55	37	6	49	12	7	5	0	0	171
cd computer	36	28	4	28	4	6	2	1	0	109
video/dvd	36	20	2	34	7	3	4	1	1	108
Material	358	214	24	265	57	38	31	8	5	1000

Ilustración 29: Tabla 1 PowerPlay Estado Documentos

Y el siguiente diagrama circular muestra como se distribuyen los 57 libros perdidos por colecciones.

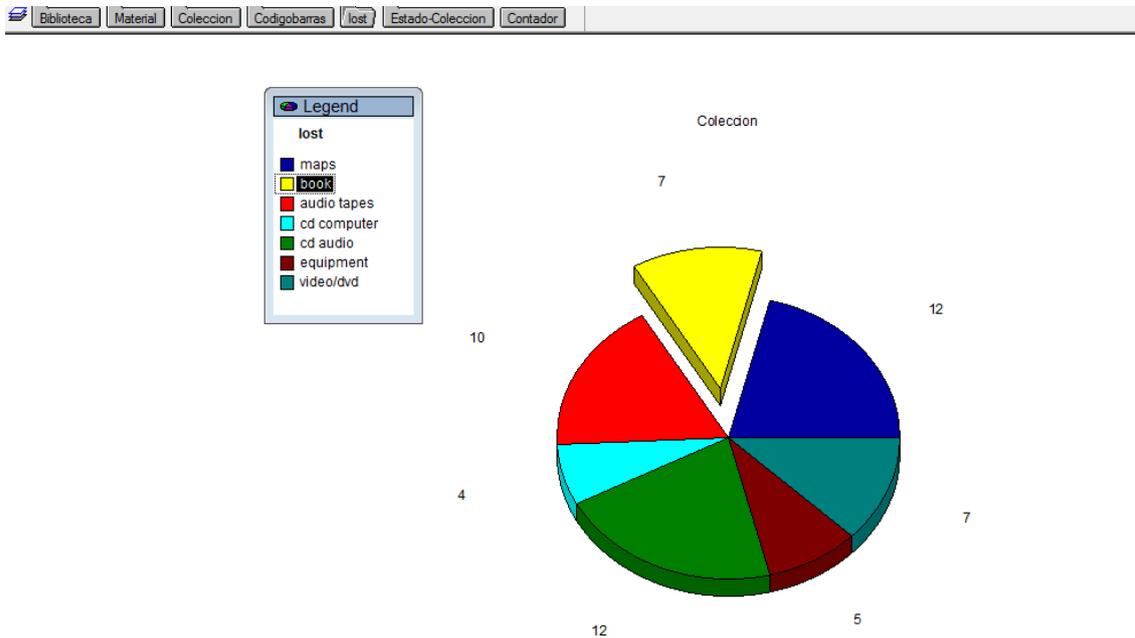


Ilustración 30: Gráfico 1 PowerPlay Estado Documentos

6. Valoración económica.

Estudio y cuantificación de los costes imputados al Personal Técnico así como suministros requeridos para la realización del proyecto.

El proyecto está pensado para una red de bibliotecas y podrán acceder continuamente entre 50 y 100 empleados para acceder a los servicios.

Recursos Técnicos

Con objetivo de evaluar el impacto que el proyecto significará se prevé que su implantación significará los siguientes esfuerzos.

Actividad	Recurso	Dedicación (Días/pers ona)	Duración (Días)	Esfuerzo (Días/persona)	Precio unitario (€/día/ pers)	Coste	
Selección Proveedor							1400
Requerimientos solución DWH	Coordinador	1	1	1 A	350	350	
Requerimientos solución HW	Coordinador	1	1	1 A	350	350	
Selección, contratación proveedor	Coordinador	1	2	2 A	350	700	
Adecuaciones a realizar							6000
Sistema Operacional	Equipo técnico	1	15	15 C	200	3000	
Adecuaciones Proveedor	Equipo técnico	1	15	15 C	200	3000	
Operación y puesta en marcha							1000
Pruebas	Equipo técnico	1	5	5 C	200	1000	
Formación y Administración							400
Formación en Herramienta	Equipo Técnico	1	2	2 C	200	400	
Coordinación General del Proyecto							1575
	Jordi Blanca	0,10	45	4,50 A	350	1575	
Cierre							175
	Jordi Blanca	0,5	1	0,5 A	350	175	
TOTAL Recursos humanos							10.550

Tarifas		
Tipo	€/dia-persona	Observaciones
A	350	Alto cargo. Sueldo bruto de 70.000 €/año
B	320	Tarifa media de personal externo especializado 50 €/hora
C	200	Sueldo bruto anual de equipo técnico 30000 € / año
D	125	Sueldo bruto anual de equipo administrativo 20000 € / año

Otros recursos

Adicionalmente al equipo de técnicos que se implicarán en el proyecto debe considerarse:

Hardware.

Un servidor Dell PowerEdge T710 con un procesador Xeon X5550, 2,66GHz de 4 núcleos.

Software.

Los componentes a instalar son Oracle Database 11g Standard Edition One y IBM Cognos Express versión 8. De los ambos productos se deben adquirir 2 licencias puestos que ambas compañías valoran cada núcleo Intel como 0,5 licencias. Por otra parte, con una buena negociación se puede llegar a obtener un 30% de descuento sobre los precios de lista. Consideramos una cotización de 1\$ = 0,74€,

			Unidades	Precio unitario	Coste	
Suministros externos	Precio base	Variación				33.272
Dell PowerEdge T710	2.710	0%	1	2.710	2.710	
Oracle Database 11g Std Ed	12.950	30%	2	9.065	18.130	
IBM Cognos Express version 8	8.880	30%	2	6.216	12.432	

Presupuesto Total

Concepto	Coste
Recursos Personal	10.550
Suministros externos	33.272
Contingencias 10%	4.500
TOTAL	48.322

7. Conclusiones

Desde el la perspectiva del desarrollo del proyecto, éste se ha desarrollado desde el principio hasta el final dentro de la planificación establecida en el plan de proyecto.

Tan sólo tuve una incidencia dentro de la elaboración del proyecto cuando presenté el plan de proyecto y el consultor me recomendó reconducirlo a un planteamiento más conservador de definición de requerimientos, análisis, desarrollo y pruebas. La idea inicial era realizar un proceso de ingeniería inversa de un aplicativo 'open source' llamado OpenBiblio que a mi entender encajaba bastante con el enunciado del proyecto. Este cambio me retrasó una semana la planificación prevista, pero fue al inicio del curso y fácilmente pude posicionarme al día.

Ahora que ya estoy finalizando el proyecto, debo reconocer que el resultado ha sido más académico con el nuevo enfoque y que la idea inicial me podía haber llevado a un consumo de horas excesivo hasta el punto de poner en peligro la finalización del proyecto.

Por lo demás, la planificación se ha ido cumpliendo escrupulosamente semana a semana, los resultados obtenidos creo que son los esperados y he podido presentar las PAC's en las fechas previstas con buenos resultados.

Desde el punto de vista tecnológico, me ha interesado especialmente la creación del sistema del almacenamiento multidimensional. Subjetivamente, me ha parecido más interesante y gratificante que la parte del operacional. Probablemente porque esta parte ha sido mucho más recurrente en varias asignaturas de la carrera, mientras que la definición y explotación del datawarehouse me ha permitido introducirme y conocer, aunque sea de manera muy incipiente, una nueva dimensión en la gestión de la información contenida en una base de datos.

Para diseñar una buena arquitectura de almacén de datos es necesario conocer muy bien los requerimientos del negocio, conocer bien las fuentes externas y hacer un buen diseño del área de transformación de datos definiendo cuáles son las transformaciones que se van a realizar y cómo se va a implementar el modelo multidimensional con sus tablas de hechos, dimensiones y medidas.

Con toda seguridad continuaré trabajando en esta área del análisis de los datos. He estado viendo otros sistemas Business Intelligence y me ha parecido muy interesante otro producto también de IBM llamado AlphaBlox que fácilmente se integra en entornos J2EE e incluso pueden realizarse aplicaciones de portlets con esta herramienta. Esta combinación permitiría un portal tan flexible como la nueva web de la UOC con gráficos personalizables por el propio usuario.

Por otra parte, la utilización de procedimientos almacenados como único método de actualización de las tablas integrado con un sistema de registro diario de movimientos de la base de datos permite mantener de forma automática un sistema de auditoría de movimientos. Con este sistema se puede detectar y reproducir cualquier error puesto que tenemos un histórico cronológico de todas las llamadas realizadas.

Bibliografía

- **Oracle9i SQL Reference 10g , Release 1 (10.1). Part No.B10759-01**
Copyright © 1996, 2003, Oracle Corporation.
- **PL/SQL Programming. O'Reilly. Steven Feuerstein**
Oracle PL/SQL, Third Edition.
- **Cognos Impromptu. Impromptu User Reference.**
Copyright (C) 2003 Cognos Incorporated.
- **Cognos Impromptu. Discovering Impromptu**
Copyright (C) 2003 Cognos Incorporated.
- **Cognos Transformer. Step-by-step Transformer**
Copyright (C) 2003 Cognos Incorporated.
- **Cognos Transformer. Discovering Transformer**
Copyright (C) 2003 Cognos Incorporated.
- **Cognos PowerPlay for Windows. Discovering PowerPlay**
Copyright (C) 2003 Cognos Incorporated.

http://es.wikipedia.org/wiki/Base_de_datos_multidimensional

http://es.wikipedia.org/wiki/Cubo_OLAP

<http://www-01.ibm.com/software/data/cognos/customercenter/>