Capítulo 5

Construcción de un prototipo

Se implementó un prototipo basado en la arquitectura para análisis de información Zombi. Se utilizaron principalmente dos herramientas de uso gratuito: el servidor de procesamiento en línea Mondrian y el minero de datos Weka. El resto de los componentes fue desarrollado usando Java.

El prototipo, denominado también Zombi, se propuso como una solución arquitectural y un desarrollo que incluyera la funcionalidad de procesamiento analítico en línea y minería de datos y los servicios de integración y construcción de un almacén de datos. Las fuentes de datos de este sistema fueron los datos de los sistemas operacionales que usa actualmente la biblioteca de la Universidad de las Américas -Puebla y que ha generado a través de los años, en particular los datos de circulación de material bibliográfico que se encuentran en el sistema de administración de bibliotecas Sydney.

En este capítulo se describen los componentes y en general las actividades relacionadas a la implementación de Zombi.

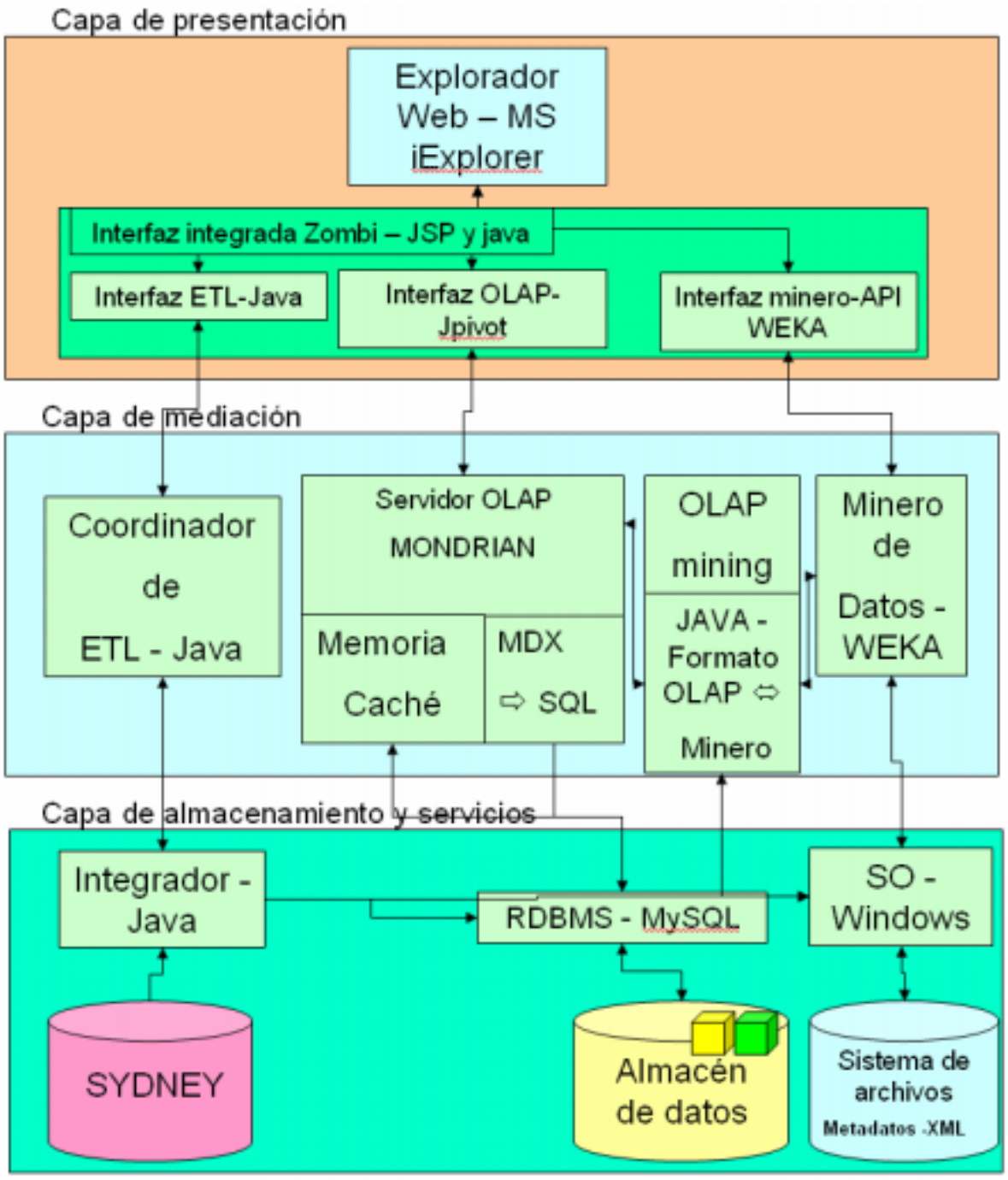

Figura 5.1. Construcción de Zombi basada en la arquitectura propuesta

La figura 5.1 muestra cada uno de los componentes de software que se construyeron o implementaron para construir el prototipo Zombi basado en la arquitectura propuesta en el capítulo anterior.

5.1 Construcción del almacén de datos

Debido a que la biblioteca de la UDLA-P opera con el sistema Sydney, se desarrolló un servicio de integración de datos para construir el almacén de datos. Los datos de Sydney se encuentran en la forma de reportes tipo texto. Se utilizó el reporte con datos históricos de los últimos cinco años de circulación de material bibliográfico.

5.1.1 Modelado de datos de circulación

Los reportes de circulación de Sydney contienen la información siguiente:

DATOS EN REPORTE DE CIRCULACIÓN DE SYDNEY (en este extracto aparecen operaciones de préstamos y devoluciones de material bibliográfico)
Estudiante
Fecha
Clasificación del libro
Transacción (préstamo, devolución, reserva, renovación)
Título
Autor
Tipo material
Tipo documento
Número de copia
Ubicación
Sub-Ubicación
División
Departamento

Los datos de circulación de material bibliográfico se llevaron al almacén de datos y con base en ellos se construyó un modelo dimensional como el que se muestra en la figura 5.2.

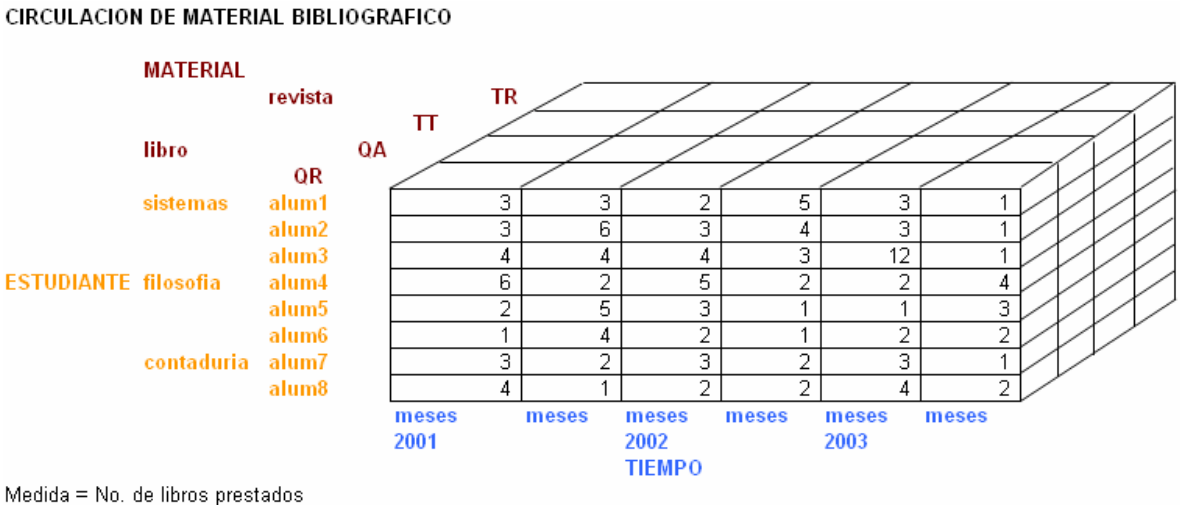

Figura 5.2 Modelo dimensional de circulación de material bibliográfico

Debido a que el modelo de datos se utilizará en Mondrian usando MySQL, entonces tiene que ser llevado a una representación relacional. El esquema multidimensional con su tabla de hechos y dimensiones se muestra en la figura 5.3.

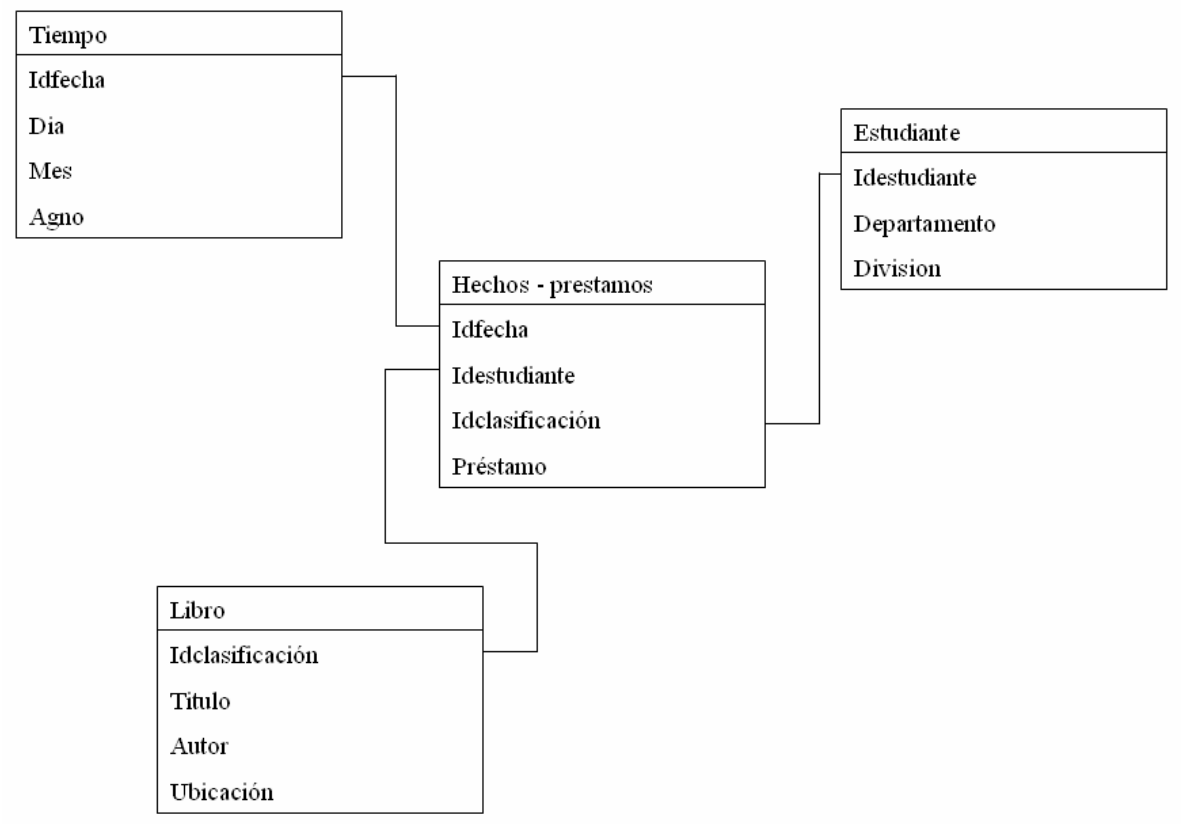

Figura 5.3 Esquema multidimensional de circulación de material bibliográfico

Se propone el siguiente esquema con las jerarquías en cada dimensión para el conjunto de datos de circulación:

Tiempo	**Estudiante**	**Libro**
Año	División	IdClasificación
Mes	Departamento	
Día	IdEstudiante	

5.1.2 Propuesta de modelado de datos de adquisiciones

Aunque no se implementó, se tuvo acceso a los datos de adquisiciones de Sydney, los cuales contienen la información siguiente:

DATOS EN REPORTES DE CUENTA DEL PROVEEDOR DE SYDNEY (aparecen todas las transacciones con proveedores)
Nombre del proveedor
Tipo de transacción (Ejemplo: COMPRA)
Fecha de la transacción
“P.O. Number”
Presupuesto / Depósito
Cantidades
Cantidad prepagada
Cantidad gastada
Total
Balance

Se propone que estos datos sean llevados al almacén de datos y con base en ellos se construya un modelo dimensional como el que se muestra en la figura 5.4.

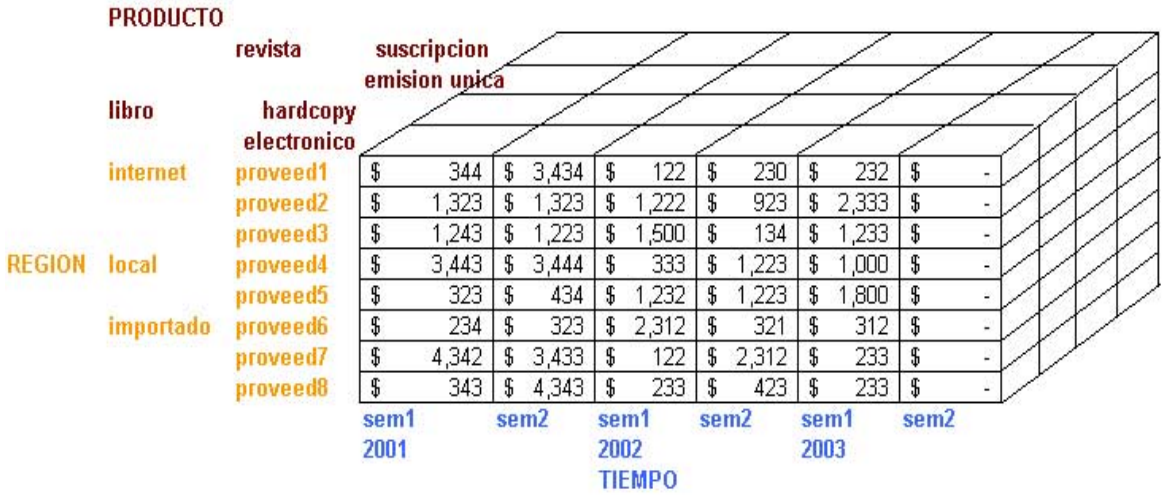

Figura 5.4 Modelo dimensional de adquisiciones de la biblioteca

Debido a que el modelo de datos se utilizará en Mondrian usando MySQL, entonces tiene que ser llevado a una representación relacional. El esquema multidimensional con su tabla de hechos y dimensiones se muestra en la figura 5.5.

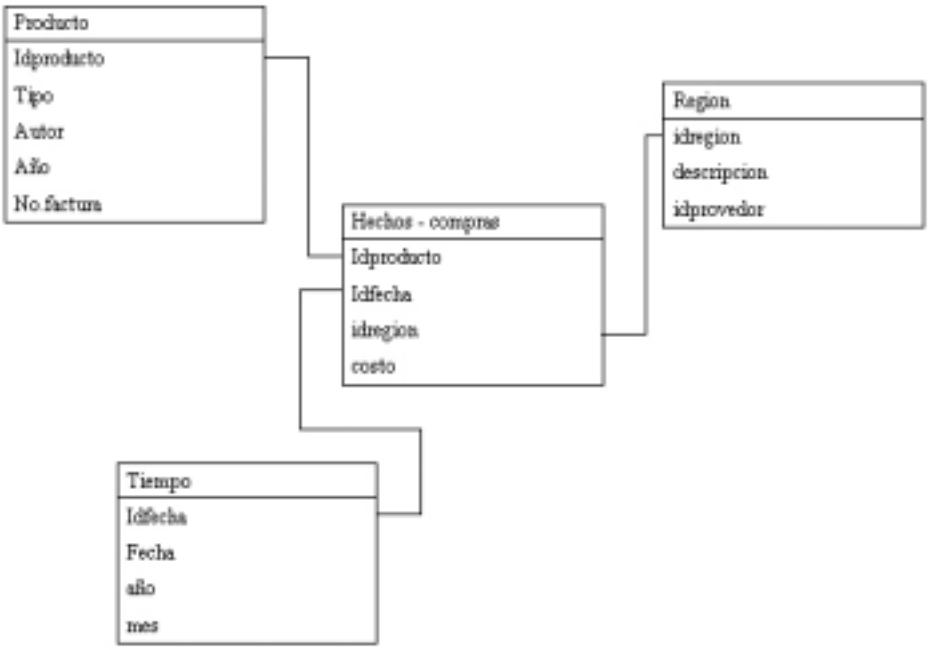

Figura 5.5 Representación relacional del modelo multidimensional para adquisiciones

5.2 Componentes de integración de datos

La arquitectura para análisis de información sugiere construir un generador de código para el proceso de ETL basado en las definiciones del usuario, en este prototipo se construyó un programa “Integrador” basado en el diseño de cubos, este programa construye el modelo multidimensional para circulación en MySQL y se codificó en Java y SQL, se muestra parcialmente en la figura 5.6.

```
            c14=(thisLine.substring(293,301)).replace('"', '\'');
                                                c15=(thisLine.substring(257,291)).replace('"', '\'');
                                                c16=(thisLine.substring(303,322)).replace('"', '\'');
                                                c17=(thisLine.substring(323,338)).replace('"', '\'');

// Almacena una transaccion en la tabla de hechos
                                            if (c4.compareTo("Checked-out       ")==0)
                                            {

                                                query = "INSERT INTO hechos_prestamos (
id_estudiante, id_fecha, id_clasificacion, circ_actividad ) values (\"" + c1 + "\",\"" + c2 + "\",\""
+ c3 + "\",\"" + nc4 + "\")";

                                                fm = stmt.executeUpdate (query);
                                            }

// Para no almacenar duplicados en libros
                                                checa="SELECT * FROM dim_clasificacion WHERE
id_clasificacion=\""+c5+"\"";
                                                rs = stmt.executeQuery (checa);



                                                if (!rs.next())
                                                {
// Inserta en la dimension clasificacion (los libros)
                                                query2 = "INSERT INTO dim_clasificacion
(id_clasificacion, TITULO, CLASIFICACION,    AUTOR_INDICE, T_MATERIAL,
T_DOCUMENTO, Copy_number, Location, Sublocation) values (\"" + c5 + "\",\"" + c6 +
"\",\"" + c7 + "\",\"" + c8 + "\",\"" + c9 + "\",\"" + c10 + "\",\"" + c11 + "\",\"" + c12 + "\",\""
+ c13 + "\")";
                                                query=query2;
```

Figura 5.6 Segmento del programa para construir el modelo multidimensional de circulación

El programa que define el esquema multidimensional para circulación en MySQL se codificó en SQL y se muestra en la figura 5.7. Adicionalmente se construyeron un conjunto de índices en cada tabla para ofrecer un mejor desempeño.

```
CREATE TABLE hechos_prestamos (
        id_estudiante     VARCHAR(255),
        id_fecha          date,
        id_clasificacion VARCHAR(255),
        circ_actividad    VARCHAR(255)
);

CREATE TABLE dim_clasificacion (
        id_clasificacion  VARCHAR(255),
        titulo            VARCHAR(255),
        clasificacion     VARCHAR(255),
        autor_indice      VARCHAR(255),
        t_material        VARCHAR(255),
        t_documento       VARCHAR(255),
        copy_number       VARCHAR(255),
        location VARCHAR(255),
        sublocation       VARCHAR(255)
);

CREATE TABLE dim_estudiante (
        Id_estudiante     VARCHAR(255),
        borrower          VARCHAR(255),
        division          VARCHAR(255),
        department        VARCHAR(255)
);

CREATE TABLE dim_tiempo (
        id_fecha          date,
        activity_date     date,
        Hora              VARCHAR(255),
        dia               VARCHAR(255),
        mes               VARCHAR(255),
        agno              VARCHAR(255)
);
```

Figura 5.7 Programa para construir el modelo multidimensional de circulación

5.3 Componentes de análisis de información

El servidor OLAP que se utilizó para el prototipo fue Mondrian. Este sistema contiene los componentes necesarios para realizar la funcionalidad de procesamiento analítico en línea, proporcionar un lenguaje de manipulación de modelos multidimensionales llamado MDX y componentes que traducen instrucciones generadas en ese lenguajes a instrucciones de SQL

que se pueden ejecutar en la capa de almacenamiento en un RDBMS, en este caso se utilizó MySQL. La otra funcionalidad de Mondrian es la optimización de la ejecución de consultas a través del uso de memoria caché. La implementación de Mondrian requirió de la revisión y modificación de algunos de sus componentes:

a) Al nivel de la interfaz de usuario se utilizo Jpivot, otro prototipo que ofrece la visualización y manipulación de los cubos administrador por Mondrian. En este se tuvo que hacer modificaciones a los componentes que presentan botones en la pantalla y al componente que genera gráficos.
b) Al nivel del servidor OLAP se tuvo que compilar todo el código fuente de Mondrian y reconfigurarlo para que pudiera ser ejecutado bajo Microsoft Windows con el servidor de aplicaciones Tomcat en esa plataforma, esto porque Mondrian fue desarrollado en Linux.
c) Al nivel de datos se tuvo que investigar como representar modelos multidimensionales en Mondrian y hacer todo el trabajo manual puesto que Mondrian no está muy bien documentado todavía.

El minero de datos que se selecciono fue Weka, se hicieron algunos experimentos para entender el software y se realizaron pruebas sencillas de minería. Durante la construcción debido a los problemas de memoria que se tenían con Mondrian se decidió que antes de construir componentes para la traducción entre los formatos de Weka y Mondrian sería necesario en integrar un componente que permita optimizar el uso de cubos esparcidos puesto que con grandes volúmenes de información los cubos que se generan pueden ser enormes sin embargo con pocas celdas ocupadas con información.

5.4 Operación del prototipo Zombi.

La operación del prototipo Zombi es sencilla, el administrador debe extraer los datos de reportes de de Sydney y posteriormente ejecutar el programa “Integrador 1”, previamente es su responsabilidad borrar los datos que existen en MySQL puesto que el programa carga todos los datos del extracto. Este primer programa genera un archivo de datos cuyos registros tienen el formato de campos delimitados por un separador como se ve en el ejemplo siguiente:

The blue rider inthe lenbachhaus, Munich / Armin (dfou)|ND568/Z9.413/1989/ 202063 |Zweite, Armin. |PAPEL |MONOGRAFIA |1 |Biblioteca |COL. GENERAL |113917NOMBRE DEL ESTUDIANTE |113917 |ESC DE HUMAN |DISEÑO GRAFICO |Renewed |2002/11/25|07:47

Debido a que el extracto proviene de un reporte de Sydney, lo que hace “Integrador 1” es eliminar todo lo relacionado a encabezados, pies de página y formato del reporte, dejando los registros de transacciones de circulación limpios de esa información que no es de interés para el análisis.

Este archivo es posteriormente cargado en el almacén de datos sobre el modelo multidimensional previamente definido usando el programa “Integrador 2”.

La interfaz del prototipo se construyó como una lista de reportes, se muestra en la figura 5.8, cada reporte de la lista fue previamente construido.

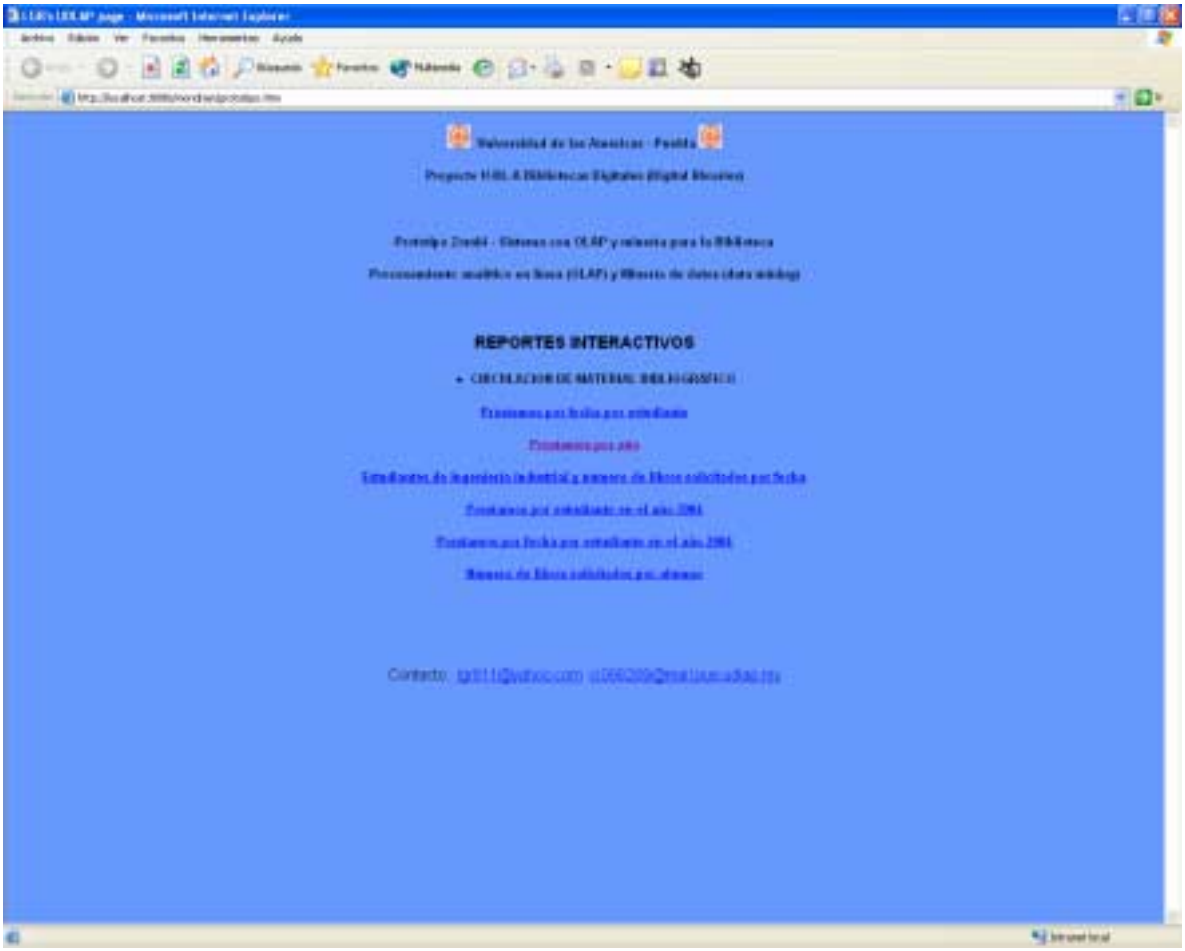

Figura 5.8 Pantalla principal del prototipo

Al seleccionar algún reporte, se genera el reporte en la forma de tabla dinámica como se ve en la figura 5.9. Estos reportes son dinámicos y el símbolo “+” permite la navegación sobre las dimensiones.

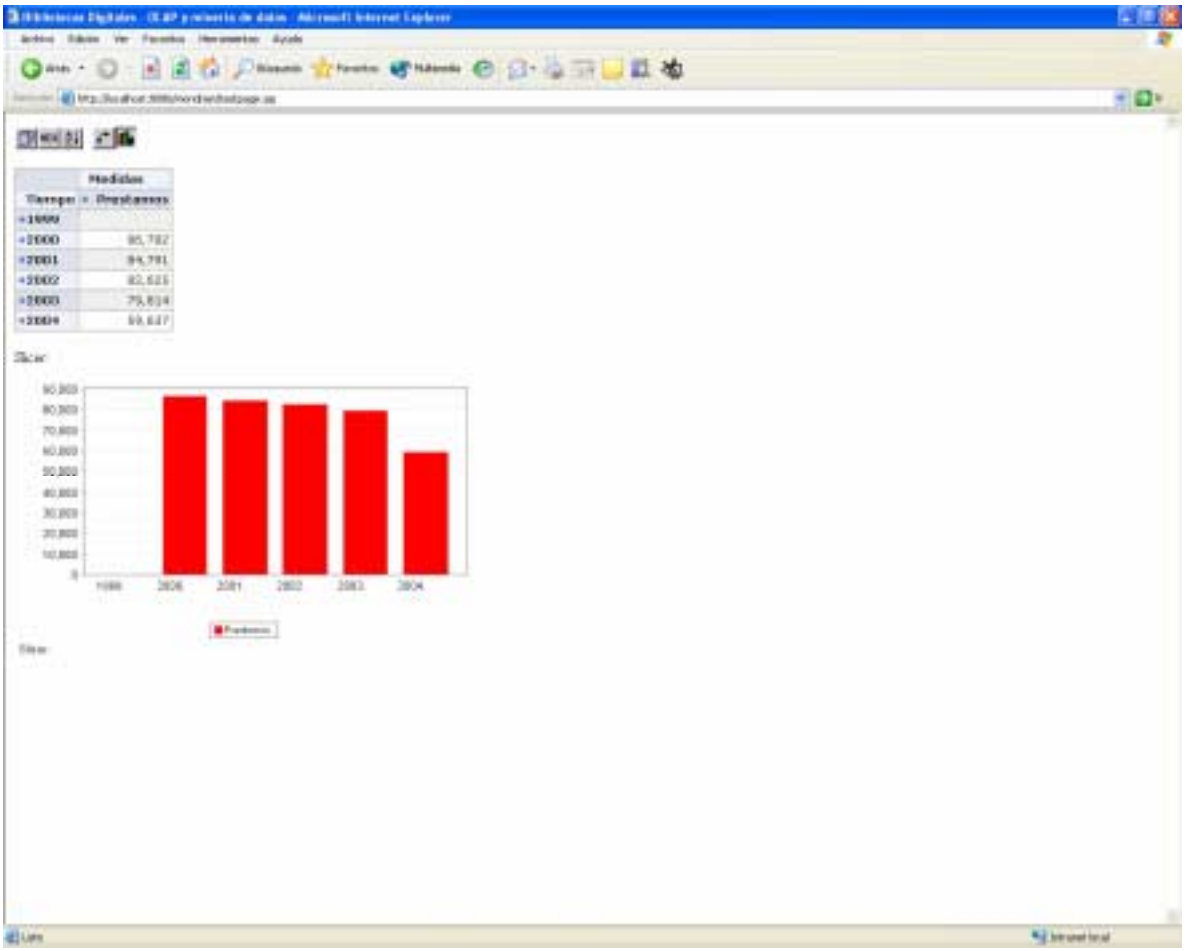

Figura 5.9 Ejemplo de un reporte dinámico del prototipo

Cada reporte de Mondrian tiene que ser construido por un administrador que conozca el modelo de datos y que sepa generar los reportes o gráficos con base a los requerimientos del usuario.

5.5 Resolución de problemas

Para poder implementar Zombi se tuvieron que solucionar algunos problemas, en el caso de Mondrian, tuvo que ser instalado y configurado para ser ejecutado en una PC con sistema

operativo Windows XP, lo cual implicó la compilación y revisión de cada componente de Mondrian así como la configuración del servidor de aplicaciones Tomcat.

La interfaz de Mondrian, llamada Jpivot, se instala por separado, tenía problemas en la configuración de botones y gráficos y tuvieron que ser corregidos.

La utilización de índices tuvo que ser desarrollada en la práctica basándose en pruebas de desempeño del software, esto puesto que no existe una documentación adecuada de mejores practicas en la creación de cubos para Mondrian.

Respecto a los datos, se tuvieron que definir de manera manual los modelos multidimensionales con base en los extractos obtenidos de reportes de Sydney. Y hacer una gran cantidad de pruebas hasta lograr visualizar los datos en la interfaz de Mondrian.