



“ARQUITECTURA DEL SERVIDOR DE INTELIGENCIA DE NEGOCIOS DE ORACLE ”

Ms C. Yunelsy Ortiz Chávez.

Universidad de Holguín “Oscar Lucero Moya”

Ing. Freddy Ramón Laffita Almaguer

Desarrollo de Programas Informáticos

Ing. Zaily Lizandra Ortiz Chávez

Cadena Hotelera Isla Azul

yortiz@facii.uho.edu.cu

A los lectores:

El contenido de este artículo está orientado a personas con experiencia en las tecnologías de Inteligencia de Negocios de Oracle. Los ejemplos utilizados se refieren a la versión 11.1.1.5.

Resumen

El presente trabajo ayuda de manera fácil e intuitiva, a comprender la estructura del Servidor de Inteligencia de Negocios de Oracle, y se ejemplifican las transformaciones que sufren las consultas desde los clientes en los servicios de presentación hasta los orígenes de datos físicos.

Introducción

Muchas veces, al realizar proyectos utilizando herramientas de Inteligencia de Negocios de la prestigiosa empresa Oracle, nos encontramos con problemas durante la construcción del repositorio y la creación de respuestas (answers), que en su mayoría, están asociados al desconocimiento parcial o total de la estructura y funcionamiento del servidor. En ocasiones percibimos que no coinciden los datos calculados de forma manual con los que proporcionan los reportes generados por la utilidad *Analytics* del paquete OBI (Oracle Business Intelligence) en edición empresarial y demás, o tal vez que devuelven errores al relacionar determinadas dimensiones y tablas de hechos. Situaciones como estas pudieran ser resueltas

sin incurrir en grandes gastos de tiempo si se conocen a fondo las potencialidades y características de los elementos empleados.

Arquitectura del Servidor de Inteligencia de Negocios de Oracle

Oracle BI Server (OBIS) es un componente de Inteligencia de Negocios que procesa las solicitudes de los usuarios y consulta las fuentes de datos accesibles. Además soporta el modelo de datos lógicos y permite al cliente acceder al mismo a través del puente ODBC¹.

La arquitectura del Servidor y el repositorio de OBI provee una capa de abstracción que posibilita a los usuarios enviar consultas simples de SQL Lógico² contra complejos conjuntos de fuentes de datos.

El OBIS utiliza los metadatos que se encuentran en el repositorio de Oracle BI (OBIR) para realizar las siguientes tareas:

1. Interpreta las consultas SQL Lógicas y escribe las correspondientes consultas físicas contra las fuentes de datos apropiadas.
2. Transforma y combina los result set³ físicos, y realiza los cálculos finales.

La siguiente figura muestra de manera gráfica la arquitectura del OBIS:

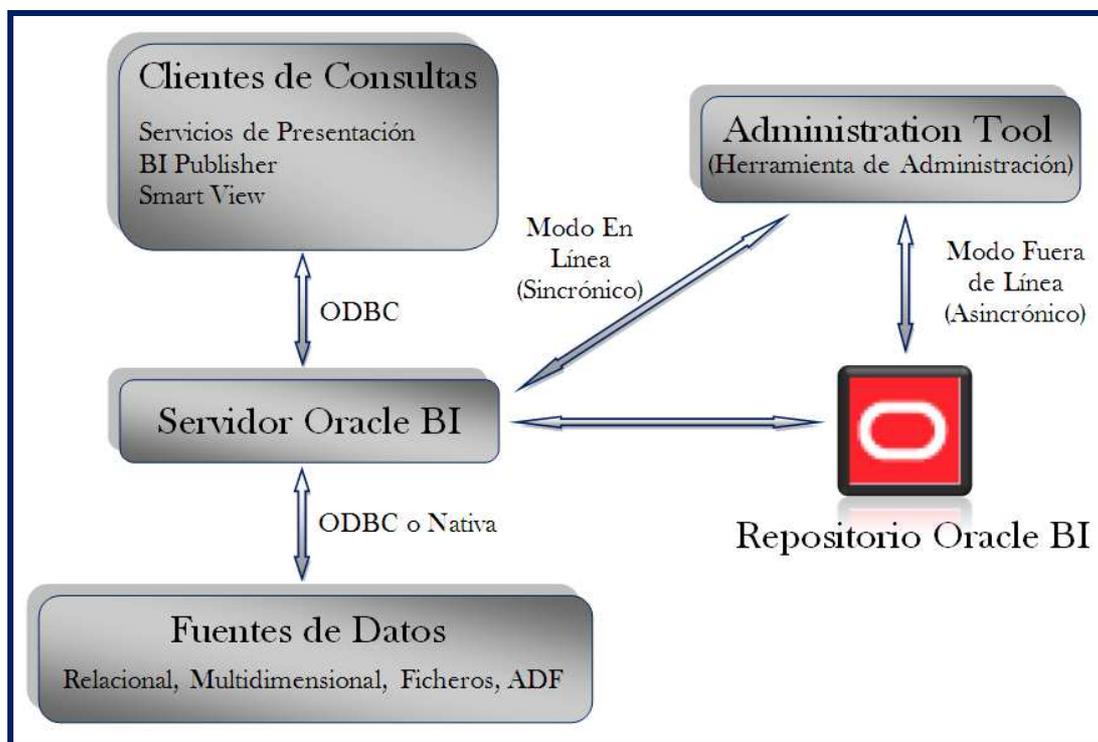


Figura 1. Arquitectura del Servidor de Inteligencia de Negocios de Oracle.

¹ Conectividad Abierta de la Base de Datos (en inglés, Open DataBase Connectivity) es un estándar de acceso a Bases de datos desarrollado por Microsoft Corporation, el objetivo de ODBC es hacer posible el acceder a cualquier dato desde cualquier aplicación, sin importar qué Sistema Gestor de Bases de Datos (DBMS por sus siglas en inglés) almacene los mismos.

² SQL Lógico se denomina a las sentencias SQL que son enviadas al OBIS desde los clientes de consultas.

³ Conjunto de datos resultante de una consulta SQL realizada a una Base de datos.

Los clientes de consultas se comunican con el OBIS y reciben los resultados mediante el puente ODBC, de manera similar funciona entre el OBIS y las distintas fuentes de datos a las que tenga acceso, debido a que, originalmente las herramientas de BI pertenecían a la compañía Siebel e implementaron la transmisión de esta forma. Luego al pasar a manos de Oracle se le incorpora una API⁴ nativa (modelo OCI⁵) para la conexión específica a su base de datos. Todos los metadatos que procesa el servidor están referenciados en el OBIR, el cual se gestiona a través de la herramienta Administration Tool perteneciente al paquete de OBI. Esta utilidad interactúa con el repositorio de dos formas (en línea y fuera de línea), la diferencia radica en que al usar la primera variante los cambios se pueden actualizar en el servidor, por lo que es imprescindible que el servicio (Oracle BI Server) esté en ejecución, en la segunda los cambios no son actualizables ni requieren que el servidor se encuentre funcionando.

La siguiente figura muestra las consultas SQL Lógicas cruzando las distintas capas del repositorio:

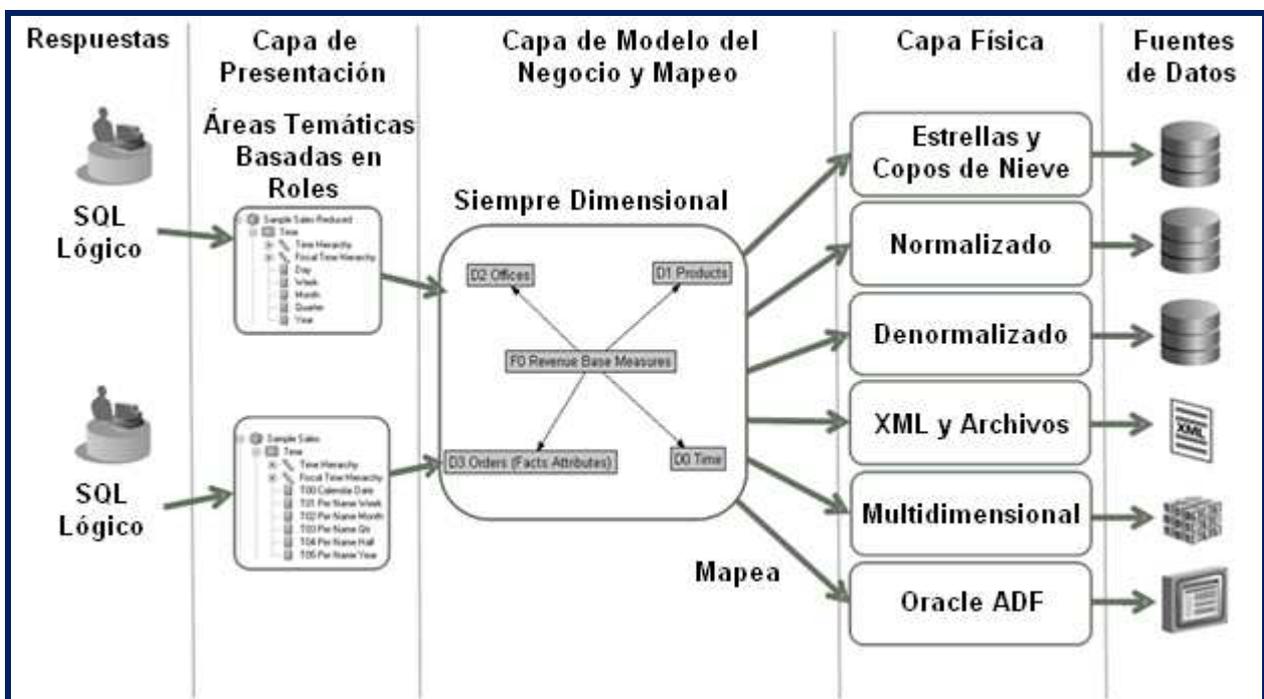


Figura 2. Consultas SQL Lógicas a través de las capas del repositorio de OBI.

Cuando los usuarios de consulta utilizan algunas de las herramientas de generación de reportes y crean respuestas, los campos de estas pertenecen a las áreas temáticas resultantes de la capa de presentación del repositorio, y ellos a su vez son columnas de las tablas lógicas en la capa intermedia (Modelo del Negocio y Mapeo). A continuación se muestra un ejemplo del SQL Lógico que recibe el servidor producto a una respuesta creada:

```
SELECT
  "D0 Time"."T02 Per Name Month" saw_0,
  "D4 Product"."P01 Product" saw_1,
  "F2 Units"."2-01 Billed Qty (Sum All)" saw_2
FROM "Sample Sales"
```

⁴ **Interfaz de Programación de Aplicaciones** (en inglés, Application Programming Interface) Es el conjunto de funciones y procedimientos o métodos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

⁵ **Interfaz de llamada a Oracle** (en inglés, Oracle Call Interface) Es una interfaz hacia la Base de Datos Oracle desarrollada en el lenguaje de programación C, que muestra el mayor rendimiento, integralidad y poder de la misma.

ORDER BY saw_0, saw_1

El OBIS convierte la consulta SQL Lógica en una consulta física sofisticada como sigue:

```
WITH SAWITH0 AS (  
  select T986.Per_Name_Month as c1, T879.Prod_Dsc as c2,  
         sum(T835.Units) as c3, T879.Prod_Key as c4  
  from  
    Product T879 /* A05 Product */ ,  
    Time_Mth T986 /* A08 Time Mth */ ,  
    FactsRev T835 /* A11 Revenue (Billed Time Join) */  
  where ( T835.Prod_Key = T879.Prod_Key and T835.Bill_Mth = T986.Row_Wid)  
  group by T879.Prod_Dsc, T879.Prod_Key, T986.Per_Name_Month  
  )  
select SAWITH0.c1 as c1, SAWITH0.c2 as c2, SAWITH0.c3 as c3  
from SAWITH0  
order by c1, c2
```

En el momento de ejecutarla el OBIS comprueba dos aspectos fundamentales:

Primero: Que exista una relación (camino) lógica (capa lógica) entre los campos ya sean provenientes de tablas de hechos o de dimensiones.

Segundo: De existir dicha relación entonces comprueba si hay un camino físico (capa física) entre los campos en los orígenes de datos hacia los cuales están mapeados.

Si todo está correcto realiza los cálculos finales (escogidos en la respuesta), si no envía los correspondientes mensajes de inconsistencia.

A continuación se muestra un ejemplo de estos dos aspectos:

(1) Comprobación lógica.

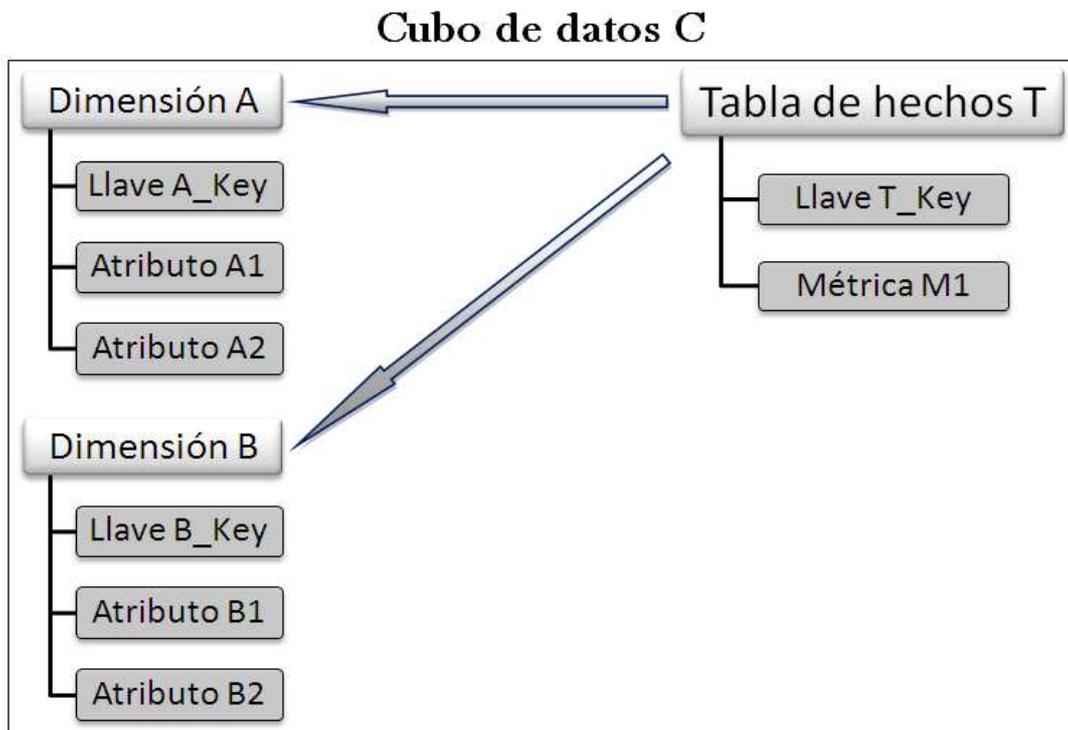


Figura 3. Ejemplo 1 de un Modelo del Negocio.

Dado este diagrama (Figura 3) si se procesa una solicitud que involucre los atributos A.A1, B.B1 y la métrica T.M1 se tiene que existe una relación lógica ya que pertenecen al mismo

cubo de datos, y los atributos (A.A1 y B.B1) se relacionan de manera directa a la métrica T.M1.

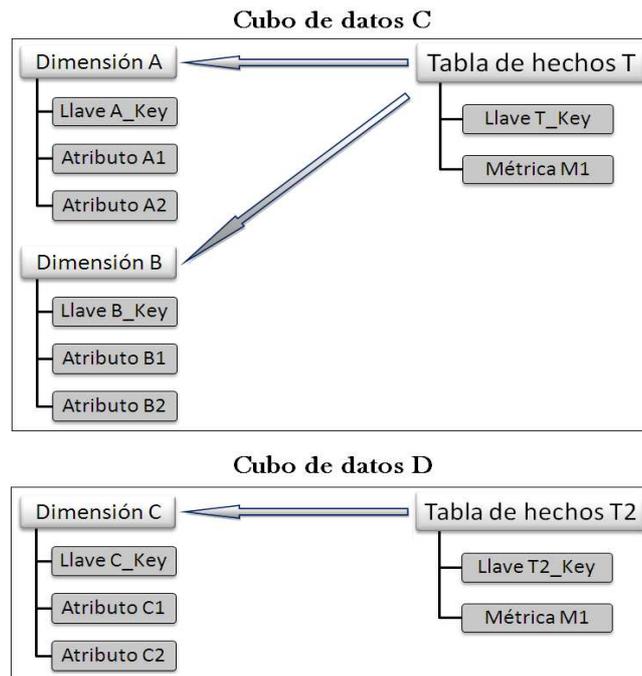


Figura 4. Ejemplo 2 de un Modelo del Negocio.

En el caso de la imagen anterior (Figura 4) el modelo del negocio está formado por 2 cubos de datos (C y D). Si se procesa una solicitud que involucre los atributos A.A1, B.B1, C.C1 (C se refiere a la dimensión) y la métrica T2.M1 se tiene que no existe una relación lógica entre C.C1 y ninguno de los otros componentes de la solicitud. Los problemas de inconsistencia lógica no están dados por pertenecer a cubos de datos diferentes, puesto que en un mismo diagrama (modelo del negocio) pueden coexistir más de uno, sino por la falta de relaciones entre las dimensiones y tablas de hechos a las cuales pertenecen los atributos de la solicitud realizada.

(2) Comprobación física.

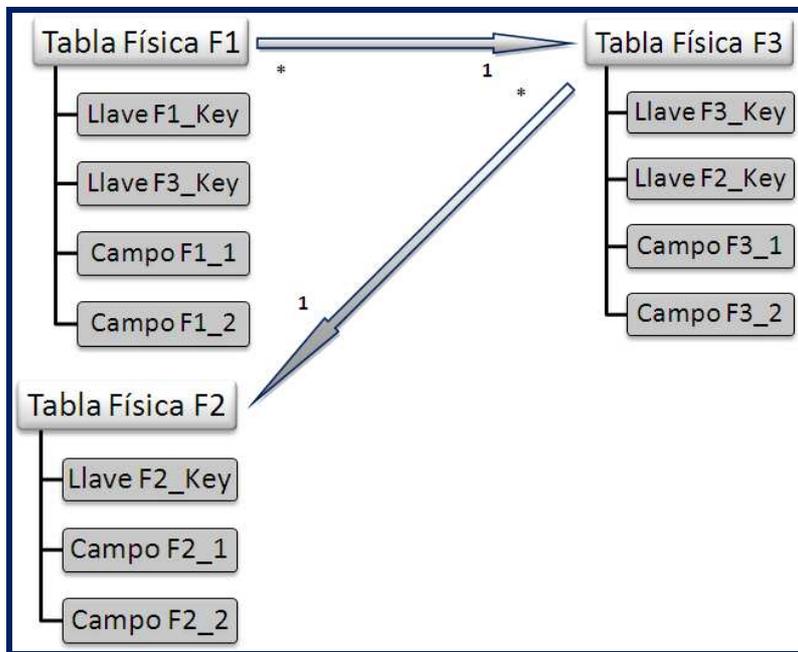


Figura 5. Ejemplo 1 de un Diagrama Físico.

Tomando la solicitud de **A.A1**, **B.B1** y la métrica **T.M1** correspondiente al diagrama lógico de la figura 3, y como mapeo de los datos el siguiente:

- **T.M1** procede del campo **F2_Key**.
- **A.A1** procede del campo **F2_1**.
- **B.B1** procede del campo **F3_1**.

Para este caso se conoce que no hay problemas en la comprobación lógica, y tampoco habrá en la (2) puesto que existe un camino físico que permite llegar desde los campos que representan los atributos de las dimensiones **A.A1 (F2_1)** y **B.B1 (F3_1)** hasta los que representan las métricas en la tabla de hechos **T.M1 (F2_Key)**.

Ahora analicemos el mismo ejemplo pero utilizando como mapeo de los datos el siguiente:

- **T.M1** procede del campo **F2_Key**.
- **A.A1** procede del campo **F1_1**.
- **B.B1** procede del campo **F3_1**.

Y basado en el diagrama físico que se muestra a continuación:

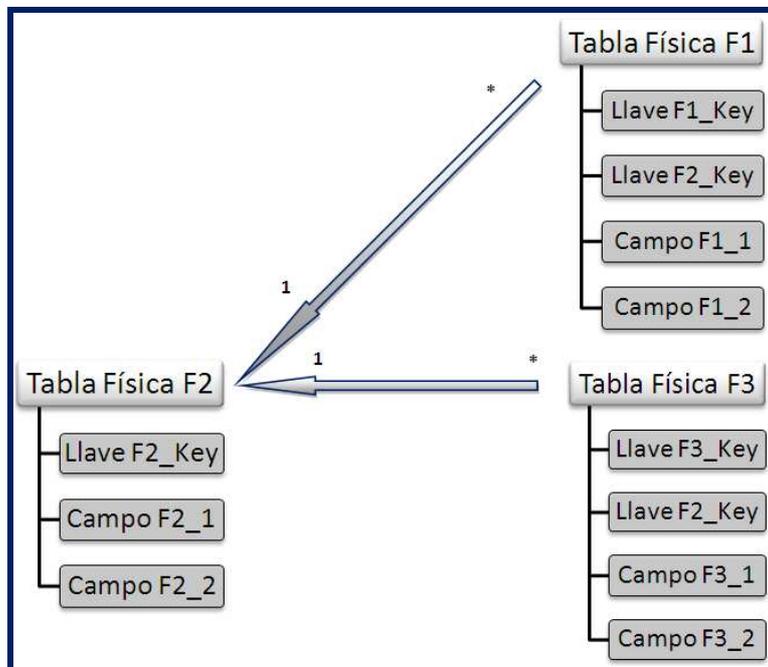


Figura 6. Ejemplo 2 de un Diagrama Físico.

En esta situación existirán dos variantes suponiendo que se haga una unión lógica⁶ entre las tablas involucradas (T y A, T y B):

Mejor caso: El valor en la columna (**F2_Key**) de la tabla **F2** es el mismo en las tablas **F1** y **F3**, el servidor no arroja errores de consistencia física pero los resultados de los cálculos no serán los esperados (solo calculará la métrica donde coincida en las tres tablas el valor del campo **F2_Key**).

Peor caso: El valor en la columna (**F2_Key**) de la tabla **F2** se encuentra repartido entre las tablas **F1** y **F3** pero nunca en ambas, el servidor manda un error de consistencia física en el que especifica que es imposible calcular la métrica **T.M1** según las dimensiones seleccionadas puesto que no se pueden mapear los datos.

Veamos un ejemplo, para el juego de datos:

F2_Key	F2_1	F2_2
1	Descripción 1	A
2	Descripción 2	B
3	Descripción 3	C
4	Descripción 4	D
5	Descripción 5	E

F3_Key	F2_Key	F3_1	F3_2
1	1	Auto	4 Ruedas
2	2	Motor	2 Ruedas

F1_Key	F2_Key	F1_1	F1_2
1	1	Francisco	Hombre
2	2	Pedro	Hombre

El resultado haciendo $T.M1 = \text{Count}(F2_Key)$, sería el siguiente:

Count(F2_Key)	A.A1	B.B1
1	Francisco	Auto
1	Pedro	Motor

Pero al adicionar nuevos registros en las tablas **F1** y **F3**, no contenidos en ambas se generan inconsistencias al no existir coincidencias de **F2_Key** de la tabla **F3** en **F1** o viceversa.

⁶ **Unión Lógica:** Se refiere a la unión entre los descriptores de tablas físicas que se encuentran en cada dimensión.

En general es de destacar que el OBIS está diseñado para realizar sus operaciones sin atacar las bases de datos, por tanto su diagrama físico (en el repositorio) no verificará ni se interpondrá con el esquema físico de ninguno de los orígenes de datos.

Esperamos que el artículo les sirva de ayuda en el desarrollo de soluciones de Inteligencia de Negocios, en especial la construcción del repositorio como el paso fundamental en este sentido.

Bibliografía

Azriel Marla, Bob Ertl. *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition, 11g Release 1 (11.1.1)*, April 2011.

Benchmarking Report - Strategic Foresight in Multinational Companies, Report of the European Corporate Foresight Group / R. Rohrbeck...[et.al.]. Berlin, Germany, 2009.

Cano, Josep Lluís. *Business Intelligence: Competir con Información*, Barcelona, ESADE, 2007.

Concepción Díaz Mayans. *Referencias Bibliográficas Estilo Vancouver*. Tomado de: <http://www.cpintz.sld.cu/normvanc.htm>, 6 de octubre del 2011.

CREME, Phyllis y Mary R. LEA. *Escribir en la universidad*, Barcelona, Gedisa, 2001.

Ganczarski, Joe. *Data Warehouse Implementations: Critical Implementation Factors Study*. VDM Verlag, 2009.

INMON, W.H., *Building the Data Warehouse*, 4 Edition, Wiley, 2005.

Laudon, Jane y Kenneth. *Sistemas de información gerencial. Administración de la empresa digital*. Pearson Educación, Prentice Hall, 2006.

MARCO, D., *Building and Managing the Meta Data Repository*, New York: John Wiley & Sons, 2000.

MOSS, L. & ATRE, S., *Business Intelligence roadmap: the complete project lifecycle for decision-support applications*, Addison Wesley, 2003.

Normas Cubanas para la descripción bibliográfica. Disponible en: http://www.sld.cu/galerias/pdf/sitios/pdvedado/normas_cubanas_para_la_descripcin_bibliogrifica.pdf, consultado el 9 de octubre del 2011.

Oracle Call Interface. Tomado de: <http://www.oracle.com/technetwork/database/features/oci/index.html>, 19 de octubre del 2011

Referencias bibliográficas Harvard. Disponible en: <http://www.sld.cu/galerias/pdf/sitios/ecimed/harvard.pdf>, consultado el 11 de octubre del 2011.