

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación



**“IMPLEMENTACIÓN DE LA HERRAMIENTA DE ANÁLISIS ORACLE
BUSINESS INTELLIGENCE, EN REEMPLAZO DE ORACLE
DISCOVERER VIEWER EN UNA ENTIDAD BANCARIA”**

EXAMEN DE GRADO (COMPLEXIVO)

PREVIO A LA OBTENCIÓN DEL TÍTULO DE

**MAGÍSTER EN SISTEMAS DE INFORMACIÓN
GERENCIAL**

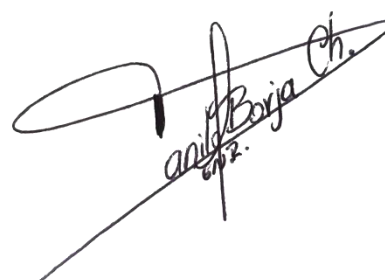
AUTOR

DANILO TOMÁS BORJA CHAVES

GUAYAQUIL, ABRIL 2021

AGRADECIMIENTO

A mis padres por darme mis primeros estudios, a mi esposa por darme el apoyo para terminar esta maestría y a Belén por cuidar de mis hijos cuando me dedique a este proyecto.



Carlos Bojia Ch.
MSc.

DEDICATORIA

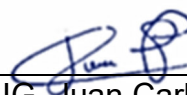
Quiero dedicar este trabajo a mis hijos, ser un ejemplo de que la paciencia y perseverancia rinde frutos, que nunca dejen de estudiar, investigar y aprender.

TRIBUNAL DE SUSTENTACIÓN



MSIG. Lenin Freire Cobo

COORDINADOR DE LA MSIG



MSIG. Juan Carlos García

PROFESOR DE LA MSIG

RESUMEN

Este proyecto tiene como objetivo principal mejorar y aumentar la eficiencia en la que la alta gerencia de una entidad bancaria recibe la información del estado actual de la institución, de esta manera la toma de decisiones es más acertada en el menor tiempo posible, brindando así los mejores servicios a los clientes.

Para ello, se definió migrar desde el aplicativo de DISCOVERER hacia Oracle Business Intelligence todos los reportes que actualmente usa la institución bancaria, además de crear otros nuevos reportes según crezca las necesidades de los usuarios.

Al migrar de aplicativo de reporte, se minimiza el impacto de merma de rendimiento que ahora mismo sufre el sistema de base de datos, se mejoran los tiempos de respuesta de la base de datos hacia el usuario, haciendo más rápida y efectiva la consulta de información.

Adicionalmente, Oracle Business Intelligence posee mejores características con las que no cuenta el aplicativo actual como gráficos para visualizar la data en lugar de mostrar únicamente tablas de valores y permite bloquear el acceso a campos confidenciales de los clientes.

ÍNDICE GENERAL

AGRADECIMIENTO	i
DEDICATORIA	ii
TRIBUNAL DE SUSTENTACIÓN	iii
RESUMEN.....	iv
ABREVIATURAS Y SIMBOLOGÍA	viii
INTRODUCCIÓN.....	ix
CAPÍTULO 1.....	1
GENERALIDADES	1
1.1 DESCRIPCIÓN DEL PROBLEMA	1
1.2 SOLUCIÓN PROPUESTA.....	3
CAPÍTULO 2.....	4
METODOLOGÍA PARA EL DESARROLLO DE LA SOLUCIÓN	4
2.1 LEVANTAMIENTO DE LA INFORMACIÓN	4
2.2 DIMENSIONAMIENTO DEL HARDWARE Y SOFTWARE REQUERIDOS	6
2.2.1 DIMENSIONAMIENTO DE SOFTWARE REQUERIDO.....	6
2.2.2 DIMENSIONAMIENTO DEL HARDWARE	8
2.2.3. PERSONAL REQUERIDO	8
2.2.4. Costo del proyecto	9

2.3 GENERACIÓN DEL PLAN DE ACCIÓN.....	9
2.3.1 Adquisición e Implementación de Hardware	12
2.3.2 Levantamiento de Información de libros de Trabajo	13
2.3.2.1 Conocer los libros de trabajo	13
2.3.2.2 Identificar las tablas de base de datos que se necesitan para crear los libros.....	14
2.3.3 Implementación de Base de datos para OBI (Desarrollo)	15
2.3.4 Implementar Aplicativo OBI en maquina cliente (Desarrollo)	22
2.3.5 Creación de Dashboards en OBI (Desarrollo)	24
2.3.6 Implementación de Base de Datos para OBI (Producción).....	25
2.3.7 Creación de Dashboards en OBI (Producción).....	26
2.3.8 Ajustes en nuevos Dashboards	27
2.3.9 Capacitación al personal del Banco.....	28
2.4 SEGURIDAD DE DATOS	28
CAPÍTULO 3.....	30
EVALUACIÓN DE RESULTADOS.....	30
3.1 ACOMPAÑAMIENTO A USUARIOS AL USAR LA HERRAMIENTA DE OBI.....	30
3.2 DIFERENCIAS ENTRE EL APLICATIVO DE ORACLE BUSINESS INTELLIGENCE Y DISCOVERER VIEWER.....	31

3.3 MONITOREO DE RENDIMIENTO EN EL SERVIDOR DE OBI	33
3.4 BENEFICIOS DE LA SOLUCIÓN	38
CONCLUSIONES Y RECOMENDACIONES	40
BIBLIOGRAFÍA.....	43
GLOSARIO.....	44

ABREVIATURAS Y SIMBOLOGÍA

DWH	DataWareHouse – Almacén de Datos
KPI	Key Performance Indicator – Indicador Clave de Rendimiento
OBI	Oracle Business Intelligence.
ODBC	Conectividad abierta de base de datos
REDO	Archivos que se usan para deshacer o rehacer un cambio en la manipulación de data en la base de datos.

INTRODUCCIÓN

El presente documento describe la problemática de una entidad bancaria para migrar de Aplicativo de Visualización de Datos orientado a usuarios, la solución que se implementó con el fin de resolverlo y los resultados obtenidos después de la misma. A continuación se resumen los apartados del documento. Capítulo 1.- Descripción del problema y la solución propuesta con el fin de resolverlo. Capítulo 2.- Metodología para el desarrollo de la información, trata del levantamiento de información, la construcción de los puntos a resolver, la dimensión de la solución requerida tanto en hardware como en software y el costo de los mismos, se determina el plan de acción y el tiempo total que se requirió, por último la implementación de la solución. Capítulo 3.- Corresponde a la evaluación de los resultados, el monitoreo del nuevo aplicativo y la diferencia entre el viejo y el nuevo aplicativo. Se finaliza este documento con las conclusiones y recomendaciones.

CAPÍTULO 1

GENERALIDADES

1.1 DESCRIPCIÓN DEL PROBLEMA

Una entidad Bancaria Ecuatoriana adquirió en el año 2010 la herramienta Oracle Discoverer Viewer, la cual se usaba para generar algunos reportes financieros hacia la alta gerencia de la institución, estos reportes fueron necesarios y claves para la toma de decisiones comerciales, como la de ofrecer productos y hacia que nicho de mercado, por esa razón, también se compraron más licencias del producto y ya no solo la alta

gerencia sino también las subgerencias y jefaturas recibían informes, de manera que se logre dar una solución más rápida a cualquier situación.

La generación de reportes desde la herramienta de Oracle Discoverer Viewer era muy buena pero no la mejor, ya que sus reportes eran columnas, y si bien daba una idea de la situación de un servicio o de la cantidad de clientes nuevos, se debían de tomar estos datos y colocarlos en Microsoft Excel para generar gráficos en los que se interpretaba mejor la información, es decir, que cada usuario tenía que procesar nuevamente los datos entregados en el informe.

Adicional a este problema se sumaron otros por el tema de seguridad y confidencialidad, en algunos reportes era necesario mostrar datos desde algunas tablas de base de datos en las que se contaba con datos confidenciales, sin embargo no se contaba con la forma para bloquear de algún modo algunos campos para que sean no visibles.

A medida que más usuarios usaban la herramienta, más consultas directas eran realizadas diariamente hacia la base de datos por lo que el rendimiento de la misma se fue mermando al paso de los años.

1.2 SOLUCIÓN PROPUESTA

La solución propuesta es implementar la herramienta Oracle Business Intelligence (OBI), la cual permitirá crear reportes (dashboards) para representar los principales indicadores (KPI) como ya lo venía haciendo Oracle Discoverer Viewer, pero con una mejora visual, adicional OBI cuenta con algunas herramientas que llenaran los vacíos de seguridad que tenía Oracle Discoverer Viewer, OBI también cuenta con apartados visuales analíticos para la creación de gráficos y modificación de los mismos en tiempo real.

Gracias a la configuración interna de OBI, también se puede crear capas de seguridad y ocultar los campos de tipo confidencial de las tablas de la base de datos de manera que la información de los clientes se encuentre protegida en todo momento.

Con esta propuesta también se busca minimizar el impacto que tienen las consultas hacia la base de datos, mejorando el rendimiento de la misma, crear nuevos grupos de reportes para la alta gerencia y jefaturas de distintas áreas.

CAPÍTULO 2

METODOLOGÍA PARA EL DESARROLLO DE LA SOLUCIÓN

2.1 LEVANTAMIENTO DE LA INFORMACIÓN

El área de DataWareHouse es custodio de los datos históricos del banco, todos los días en la madrugada bajo procesos automáticos, se copia la nueva data desde la base transaccional hacia la base del DataWarehouse, de ser necesario también crea tablas con información agrupada y relevante para algunas áreas de negocio según lo requieran.

Con ayuda del área de seguridad bancaria también se crea grupos de usuarios a los que se envía de forma automática archivos de texto plano, necesarios para cumplir con el reporte de información hacia la Súper Intendencia de Bancos.

El personal del DataWareHouse es el encargado de crear si se requiere libros de trabajo en el actual aplicativo de Discoverer Viewer para las consultas que los usuarios pudieran tener, al momento de ejecutar una consulta, esta realiza una petición de información hacia la base de datos, la misma que tiempo después se visualiza en el aplicativo, los datos cargados en pantalla pueden ser descargados en archivos de texto plano y formato Excel.

La creación de estos libros de trabajo deben ser solicitados bajo requerimiento y puestos en la cola de atención del DataWareHouse.

Si un usuario avanzado no desea un libro de trabajo sino en su lugar obtener acceso a las tablas del DataWareHouse, para el mismo crear sus libros en Discoverer Viewer, el DataWareHouse se comunica con el Subgerente del área del usuario que realiza la solicitud y con el permiso verificado del mismo se les otorga acceso a dichas tablas.

El DataWareHouse posee un manual del aplicativo Discoverer Viewer por si un usuario lo requiere, en este se detalla cómo realizar consultas en el

aplicativo, la exportación de data, la creación de nuevos libros de trabajo, compartir los libros creados con otros usuarios, entre otros.

De ser necesario, el área del DataWareHouse también capacita a un nuevo grupo de colaboradores a fin de que logren obtener el máximo provecho del aplicativo.

En conjunto con el área de base de datos, el DataWareHouse monitorea el rendimiento de la base de datos, de esta manera se logra evidenciar si un usuario ha creado un libro de trabajo con una consulta hacia la base de datos con bajo rendimiento.

2.2 DIMENSIONAMIENTO DEL HARDWARE Y SOFTWARE REQUERIDOS

2.2.1 DIMENSIONAMIENTO DE SOFTWARE REQUERIDO

El área de Identidades y Accesos del banco cuenta con las licencias del aplicativo de Discoverer Viewer, para la implementación de OBI se requerirá nuevas licencias ya que es un aplicativo diferente a pesar de ser de la misma empresa, las nuevas licencias tendrán un costo normal, no se obtendrá descuento alguno por poseer las licencias de Discoverer Viewer. Las licencias para la implementación se detallan en la Tabla 1 – Licencias de Oracle Business Intelligence.

Tabla1 - Licencias de Oracle Business Intelligence

Cantidad	Descripción
50	Oracle Fussion Middleware
5	Oracle Data Visualization

Fuente: Elaborado por el autor

En la licencia de Oracle Fussion Middleware se incluye el paquete para la instalación del aplicativo base en el servidor de OBI, la consola de administración, y una licencia para visualizar los datos en un determinado computador, por el momento se han contratado cincuenta licencias, aunque el número de estas puede alcanzar las doscientas. Con la licencia de Oracle Data Visualization se obtiene mejores funciones en cuanto al apartado visual para la lectura y manipulación de datos, contando con mayor relevancia la parte grafica de los mismos, estas licencias poseen un costo adicional de la licencia general, por el momento solo se han obtenido unas cuantas licencias para los usuarios avanzados ya que se está valorando si es necesario continuar con estas licencias o si es suficiente con la general.

2.2.2 DIMENSIONAMIENTO DEL HARDWARE

Para implementar el aplicativo servidor de OBI se requiere un computador servidor orientado a bases DATAMART con 3Tb para el Filesystem, el detalle de las características se encuentra en la siguiente tabla.

Tabla 2 - Características de Hardware para OBI

Tipo	Modelo	Procesador	Core	RAM	Disco Duro
Servidor de Base de Datos	HP Proliant DL380 Gen 9	Intel Xeon E52650 V3 – 2.30Ghz	10	64GB	4 TB

Fuente: Elaborado por el autor

2.2.3. PERSONAL REQUERIDO

Con el fin de finalizar en el menor tiempo requerido, en este proyecto se contó con un miembro del personal interno de la institución y adicional se contrató a dos desarrollares expertos en OBI.

2.2.4. COSTO DEL PROYECTO

El costo del proyecto se visualiza en la siguiente tabla.

Tabla 2.1 – Costo Segregado del Proyecto

	Cantidad	Valor	Total
Licencia General	50	\$1,640/año	\$82,000
Licencia Analytics	5	\$1,800/año	\$9,000
Desarrollador Externo	2	\$2,000/mes	\$28,000
Servidor	1	\$4,200	\$4,200

Fuente: Elaborado por el autor

El costo total del proyecto terminó costando 123,200 dólares.

2.3 GENERACIÓN DEL PLAN DE ACCIÓN

Para la migración de los libros de trabajo de Discoverer Viewer a OBI se requieren 159 días laborables, como se indica en la Figura 2.1 – Cronograma de Actividades, días en los que se realizara la gestión de compra e instalación del servidor con su sistema operativo, cumplir con las políticas de seguridad del banco, instalación del aplicativo OBI en el servidor, la copia de data resumida a la base de datos de OBI desde la

base del DataWareHouse, migración de libros de trabajo de Discoverer Viewer a OBI y la capacitación a usuarios finales.

Cabe señalar que se crearon dos ambientes en los que se realizó el desarrollo, la base de datos de OBI Desarrollo y la base de datos OBI Producción.

	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1	➤ Migración Discoverer a OBI	159 días	lun 20/01/20	jue 27/08/20	
2	➤ Adquisición e Implementación de Hardware	7 días	lun 20/01/20	mar 28/01/20	
3	Gestionar la compra y recepción de Servidor	3 días	lun 20/01/20	mié 22/01/20	
4	Instalación de Sistema Operativo y el aplicativo de OBI Server	2 días	jue 23/01/20	vie 24/01/20	3
5	Hardenizar el servidor de OBI	2 días	lun 27/01/20	mar 28/01/20	4
6	➤ Levantamiento de Información de libros de trabajo	20 días	lun 20/01/20	vie 14/02/20	
7	Conocer los libros de trabajo del area de Mercadeo	2 días	lun 20/01/20	mar 21/01/20	
8	Conocer los libros de trabajo del area de TyF	2 días	mié 22/01/20	jue 23/01/20	7
9	Conocer los libros de trabajo del area de Recaudaciones	1 día	vie 24/01/20	vie 24/01/20	8
10	Identificar las tablas de base de datos que se necesitan para crear los libros	15 días	lun 27/01/20	vie 14/02/20	9
11	➤ Implementación de Base de datos para OBI (Desarrollo)	49 días	lun 17/02/20	jue 23/04/20	
12	Crear nueva base de Datos para OBI	1 día	lun 17/02/20	lun 17/02/20	10
13	Codificar procesos para la copia de información	43 días	mar 18/02/20	jue 16/04/20	12
14	Pruebas Internas de los procesos realizados	5 días	vie 17/04/20	jue 23/04/20	13
15	➤ Implementar Aplicativo OBI en maquina cliente (Desarrollo)	5 días	vie 24/04/20	jue 30/04/20	
16	Comprobar Conexión ODBC	1 día	vie 24/04/20	vie 24/04/20	14
17	Consultas directas a las tablas de base de datos	3 días	lun 27/04/20	mié 29/04/20	16
18	Manipulación de datos con funcionalidades visuales	1 día	jue 30/04/20	jue 30/04/20	17
19	➤ Creación de Dashboards en OBI (Desarrollo)	35 días	vie 01/05/20	jue 18/06/20	
20	Migración de Libros de Trabajo de Mercadeo a Dashboards de OBI	10 días	vie 01/05/20	jue 14/05/20	18
21	Migración de Libros de Trabajo de TyF a Dashboards de OBI	10 días	vie 15/05/20	jue 28/05/20	20
22	Migración de Libros de Trabajo de Recaudaciones a Dashboards de OBI	10 días	vie 29/05/20	jue 11/06/20	21
23	Pruebas Internas con la Jefatura del Area	5 días	vie 12/06/20	jue 18/06/20	22
24	➤ Implementación de Base de Datos para OBI (Producción)	8 días	vie 19/06/20	mar 30/06/20	
25	Crear nueva base de Datos para OBI	1 día	vie 19/06/20	vie 19/06/20	23
26	Pase a producción de Creación de tablas de OBI	1 día	lun 22/06/20	lun 22/06/20	25
27	Pase a producción de procesos para copia de información	1 día	mar 23/06/20	mar 23/06/20	26
28	Pruebas de los procesos realizados	5 días	mié 24/06/20	mar 30/06/20	27
29	➤ Creación de Dashboards en OBI (Producción)	21 días	mié 01/07/20	mié 29/07/20	
30	Migración de Libros de Trabajo de Mercadeo a Dashboards de OBI	5 días	mié 01/07/20	mar 07/07/20	28
31	Revisión con los usuarios del Area de Mercadeo	2 días	mié 08/07/20	jue 09/07/20	30
32	Migración de Libros de Trabajo de TyF a Dashboards de OBI	5 días	vie 10/07/20	jue 16/07/20	31
33	Revisión con los usuarios del Area de Tesorería y Finanzas	2 días	vie 17/07/20	lun 20/07/20	32
34	Migración de Libros de Trabajo de Recaudaciones a Dashboards de OBI	5 días	mar 21/07/20	lun 27/07/20	33
35	Revisión con los usuarios del Area de Recaudaciones	2 días	mar 28/07/20	mié 29/07/20	34
36	➤ Ajustes en nuevos Dashboards	12 días	jue 30/07/20	vie 14/08/20	
37	Adicionar o eliminar algunos campos en los Dashboards de Mercadeo	2 días	jue 30/07/20	vie 31/07/20	35
38	Revisión con los usuarios del Area de Mercadeo	1 día	lun 03/08/20	lun 03/08/20	37
39	Adicionar o eliminar algunos campos en los Dashboards de TyF	4 días	mar 04/08/20	vie 07/08/20	38
40	Revisión con los usuarios del Area de Tesorería y Finanzas	2 días	lun 10/08/20	mar 11/08/20	39
41	Adicionar o eliminar algunos campos en los Dashboards de Recaudaciones	2 días	mié 12/08/20	jue 13/08/20	40
42	Revisión con los usuarios del Area de Recaudaciones	1 día	vie 14/08/20	vie 14/08/20	41
43	➤ Capacitación al personal del Banco	9 días	lun 17/08/20	jue 27/08/20	
44	Capacitación al personal de Mercadeo	3 días	lun 17/08/20	mié 19/08/20	42
45	Capacitación al personal de TyF	3 días	jue 20/08/20	lun 24/08/20	44
46	Capacitación al personal de Recaudaciones	3 días	mar 25/08/20	jue 27/08/20	45

Figura 2.1 – Cronograma de Actividades

Fuente: Elaborado por el autor

2.3.1 ADQUISICIÓN E IMPLEMENTACIÓN DE HARDWARE

Para el proyecto se necesitó importar el servidor a usar, una vez llegado este a la granja de servidores, se procedió a instalar el sistema operativo Centos y luego de esto a instalar la base de datos Oracle en su versión 12c junto con el aplicativo de OBI, una vez realizado esta instalación se procedió a Hardenizarlo con el equipo detallado en la siguiente tabla.

Tabla 2.2 – Equipo para Implementación de Servidor OBI

Departamento	Cargo	Procedimiento
Sistemas Operativos	Operador	Configuración del Sistema Operativo
Base de Datos	Operador	Instalación de Base de Datos y OBI
Seguridad Bancaria	Jefe	Supervisión y Soporte
Seguridad Bancaria	Operador	Protocolos de Seguridad
Seguridad Bancaria	Operador	Permisos en Puertos

Fuente: Elaborado por el autor

Este proceso se llevó a cabo dos veces, ya que el aplicativo de OBI debía estar disponible en dos ambientes: Desarrollo y Producción.

2.3.2 LEVANTAMIENTO DE INFORMACIÓN DE LIBROS DE TRABAJO

2.3.2.1 Conocer los libros de trabajo

Mientras se realizaba el proceso de compra del servidor, el equipo de DWH se acercó a las áreas donde usaban los libros de trabajo de Discoverer Viewer y se tomaron todos los nombres de estos para revisión interna.

Un punto a tomar en cuenta es que la herramienta Discoverer Viewer permite conocer mediante un query los libros de trabajo que pertenecen a un usuario de la institución y a su vez los de un área, pero no se tomó esta opción debido a que muchos de los libros creados tenían años sin usarse, algo que se corroboró cuando los usuarios solicitaron la migración de un 60% de los libros creados, ya que estos son los que realmente usaban, el 40% restante se trataba de libros repetidos por mes, o libros que cumplieron su función en su momento pero luego dejaron de ser útiles.

Para el área de Mercadeo se recolectaron 32 libros de trabajo, en Tesorería y Finanzas 41 y para el área de Operaciones se obtuvieron 19 libros.

2.3.2.2 Identificar las tablas de base de datos que se necesitan para crear los libros.

Para identificar las tablas que son usadas al realizar la consulta en un libro de trabajo, se debe tomar el query que se realiza en el aplicativo de Discoverer Viewer. En la mayoría de casos se encontró que las tablas de base de datos que se usaban en los libros de trabajo eran tablas resumen de grupos de tablas, por lo que se recurrió a la identificación de todas las tablas origen como se explica en la Figura 2.2 – Identificación tablas origen, ya que es necesario para la migración. Por cada libro de trabajo se encontró entre una a cinco tablas finales y entre tres a diez tablas origen, cabe indicar que algunos libros de trabajo por área usaban las mismas tablas origen.

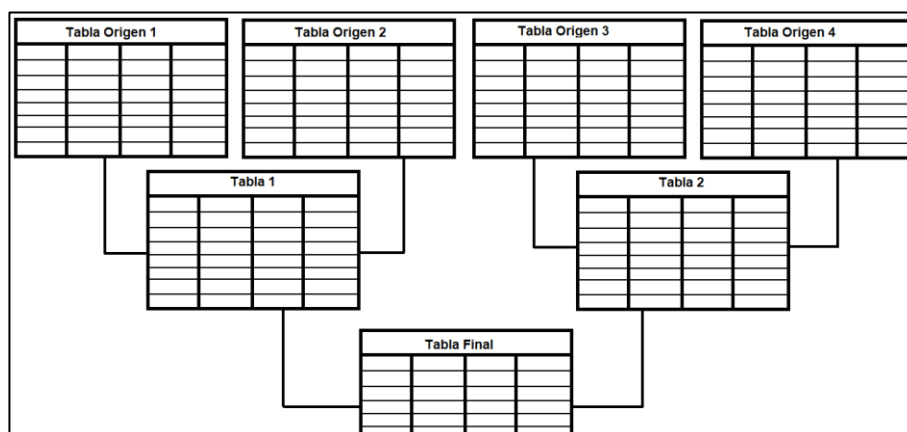


Figura 2.2 – Identificación tablas origen.
Fuente: Elaborado por el autor

De los libros de trabajo se identificaron 57 tablas en total, en las cuales se encontraron campos duplicados entre tablas y otros que no deberían ser visible a los usuarios finales, por lo que se optó por generar nuevas tablas para el servidor de OBI con un formato definido en el que consten campos relevantes para el usuario final y en el que no exista un campo igual a otro en todo el conjunto de tablas.

2.3.3 IMPLEMENTACIÓN DE BASE DE DATOS PARA OBI (DESARROLLO)

2.3.3.1 Crear nueva Base de Datos para OBI

Una vez que en el servidor de OBI fue instalado el sistema operativo con su base de datos Oracle para el guardado de data histórica y luego hardenizado, se procedió a crear la base de datos con tres usuarios, uno como dueño de las tablas y secuencias, otro como custodio de los objetos como paquetes, procedimientos y funciones, y un último como consultas de las tablas.

2.3.3.2. Codificar procesos para la copia de información

Para el traslado de información en la base de OBI, los desarrolladores externos crearon procedimientos y funciones

en la base de datos con los cuales se copia la data resumida desde la base de datos del DATAMART. En la base de datos de OBI se crearon dos tipos de tablas, las cuales se detallan a continuación.

Tablas de Dimensión: Estas tablas son copias resumidas de las tablas catálogo del DATAMART, son resumidas porque no se copian todos los campos desde las tablas del DATAMART a OBI, sino solo los necesarios para realizar los reportes como se puede apreciar en la *Figura 2.3 – Creación de Tablas de Dimensión en OBI*. Estas tablas deben estar indexadas bajo su código único, aquí podemos encontrar algunas tablas como Clientes, Provincias, Cantones, Tipos de Inversión, Catalogo de Cuentas Contables, entre otras.

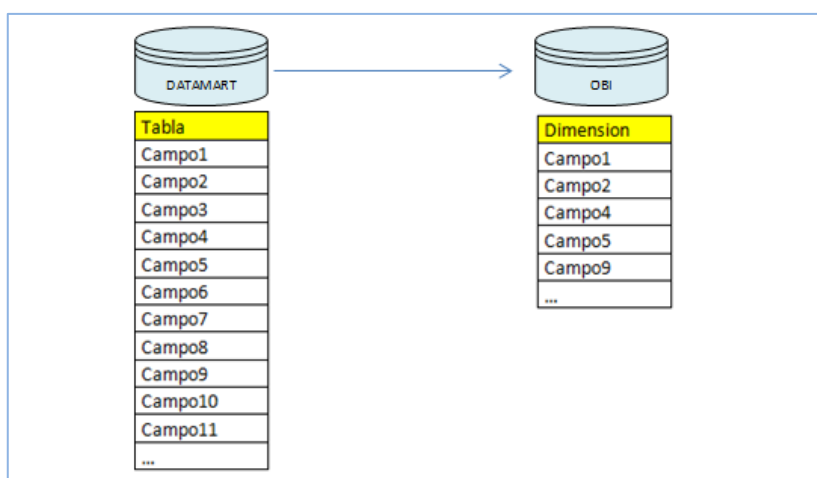


Figura 2.3 – Creación de Tablas de Dimensión en OBI.

Fuente: Elaborado por el autor

Tablas de hechos: Son copias resumen de tablas transaccionales del DATAMART, a diferencia de las tablas catálogo, una sola tabla de hecho puede estar conformada por una o varias tablas transaccionales del DATAMART como se ilustra en la *Figura 2.4 – Creación de Tablas de hechos en OBI*. Estas tablas necesitan mucho espacio de almacenamiento, deben estar correctamente particionadas por mes o día según el volumen de datos, además deben estar indexadas por los campos de búsqueda más frecuentes. En este tipo de tablas están las transacciones que se realizan por internet, cajeros, ventanillas y segregados en depósitos, retiros, inversiones, entre otras.

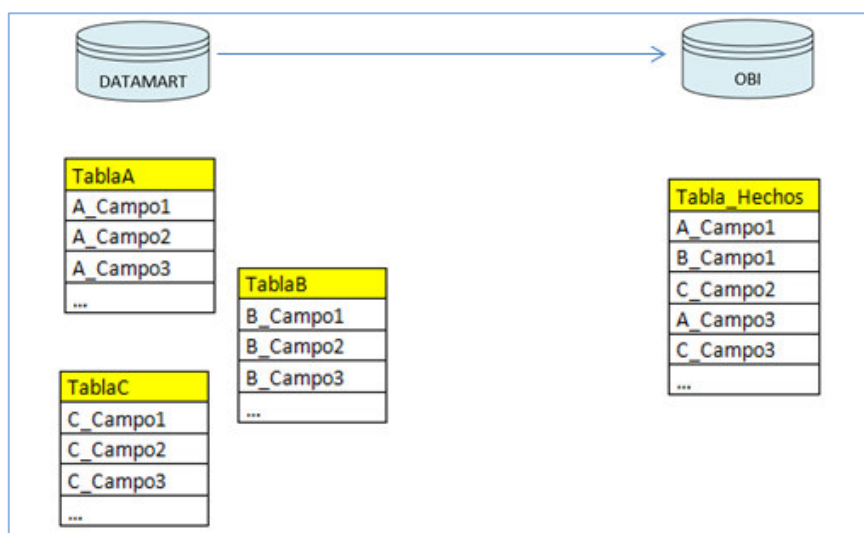


Figura 2.4 – Creación de Tablas de hechos en OBI

Fuente: Elaborado por el autor

Una vez que se realizó el estudio de copia de data, se obtuvo que de las 57 tablas identificadas en el uso de los libros de trabajo de Discoverer Viewer fueran reemplazados en la base de datos de OBI por 11 tablas de Dimensión y 26 tablas de hechos.

Para la migración de data de las tablas catalogo a las tablas de dimensión se codificaron paquetes y procedimientos detallados en la siguiente tabla en un su formato general.

Tabla 3.- Objetos para migración de tablas de Dimensión

Paquete	Procedimiento	Objetivo
Pck_Area	Pro_Tbl_Insert_Data	Inserta la carga inicial de la data
Pck_Area	Pro_Tbl_Update_Data	Actualiza la tabla con la data diaria
Pck_Area	Pro_Tbl_Truncate_Data	Elimina toda la data de la tabla

Fuente: Elaborado por el autor

Las áreas que corresponden a la migración y que fueron detalladas en el cronograma son Mercadeo, Tesorería y Finanzas, Recaudaciones. Para cada una de estas áreas se creó un paquete con los procedimientos de Inserción,

Actualización y Borrado completo de data para cada tabla, además se creó un paquete adicional con estos mismos procedimientos para el contenido de las tablas de dimensión.

Los procedimientos de inserción serían usados para la copia inicial de la data, los procedimientos de actualización están hechos para la copia de data diaria de las tablas y para el borrado de data se usó el comando de Truncate, el cual aplica un borrado completo de toda la data de una tabla, esto por si hubiese un error en la data y se deba cargar toda la información de nuevo.

La carga de información diaria será realizada automáticamente por el aplicativo de Control-M, el cual se usa en la institución financiera, en el cual se creó la hoja SIG0100 con todos los procesos que implica la actualización diaria de la data de OBI, para usar esta herramienta basta con hacer referencia al procedimiento que se va a ejecutar, el día y la hora para correr dicho proceso y en caso de estar encolados con más procedimientos, detallar todos estos, como se muestra en la *Figura 2.5 Control M – Procesos DataMart*.

Cabe indicar que los procesos automáticos de Control-M no fueron ejecutados en el proceso inicial, ya que como se indicó, para la carga inicial se ejecuta un proceso diferente al del diario automático.

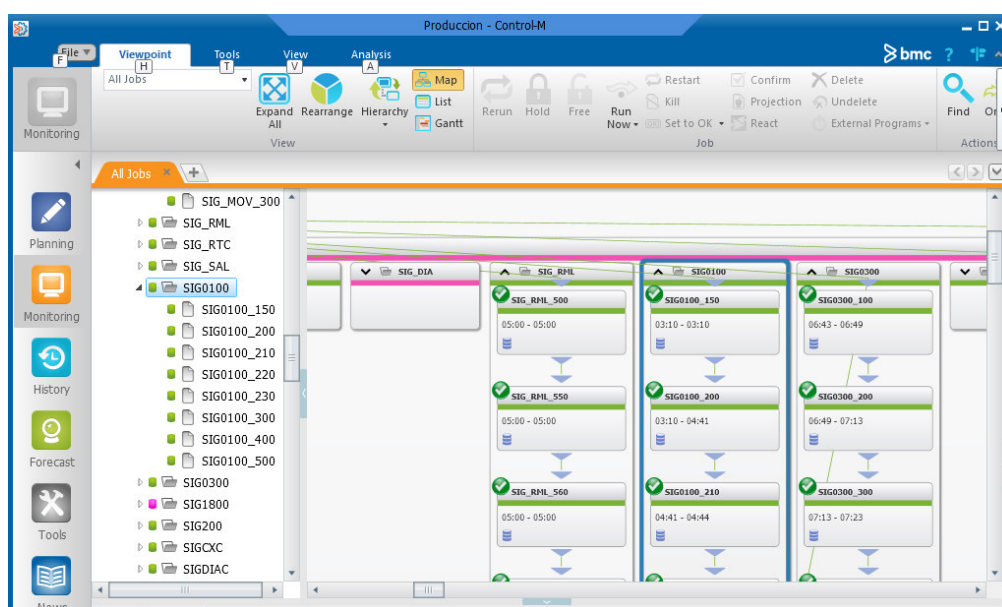


Figura 2.5 Control M – Procesos DataMart.

Fuente: Elaborado por el autor

2.3.3.3. Pruebas internas de los procesos realizados

En esta fase, los desarrolladores externos presentaron una prueba total de la migración con el jefe del área del DataWarehouse, desde los procedimientos de copia de data inicial hasta la copia diaria de la misma. En este punto se realizaron pruebas con data de producción por lo que

salieron a la luz algunos casos que no fueron contemplados en el desarrollo, los cuales fueron corregidos en el lapso de tiempo destinado a estas pruebas. Entre algunos casos que se evidencio en OBI fue que al contrario de Discoverer Viewer, todas las tablas deberán estar relacionadas, es decir no podrá presentarse una tabla solitaria, este tema se resolvió duplicando en la Interfaz de Administración de Herramientas OBI la tabla a visualizar, creando una relación entre la original y la copia para que esta pueda visualizarse en el browser como se ve en la *Figura 2.6 Administrador de Herramientas OBI - Duplicar Tabla*

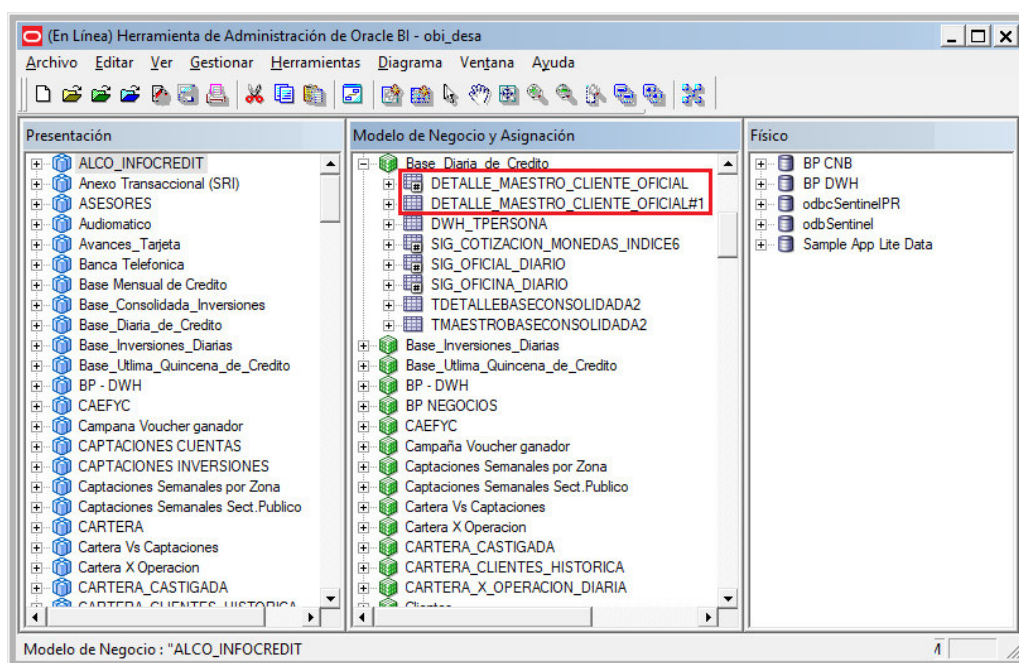


Figura 2.6 Administrador de Herramientas OBI - Duplicar Tabla

Fuente: Elaborado por el autor

2.3.4 IMPLEMENTAR APLICATIVO OBI EN MAQUINA CLIENTE (DESARROLLO)

Con la base de datos creada, se debía ver la data en el browser de una terminal, por lo que se usó una de las licencias de OBI para la instalación del aplicativo en el ambiente de Desarrollo, junto al área de Seguridad Bancaria se permitió la conexión por ODBC a la base de datos y conexión por SSH al servidor de OBI.

Antes de visualizar los datos en el browser, primero se debe tomar las tablas de la base de datos e ir creando las relaciones entre estas tablas para que el motor de Oracle sea eficiente, además de crear las carpetas que contendrán estas consultas, para esto se usó el aplicativo del Administrador de OBI. En la *Figura 2.7 – Administrador de Herramientas de OBI* podemos ver las tres capas: Física, en donde tenemos las tablas, Modelo, donde definimos la relación entre las tablas y la de Presentación donde podemos poner los nombres que estarán en el browser, así como los filtros que deseemos.

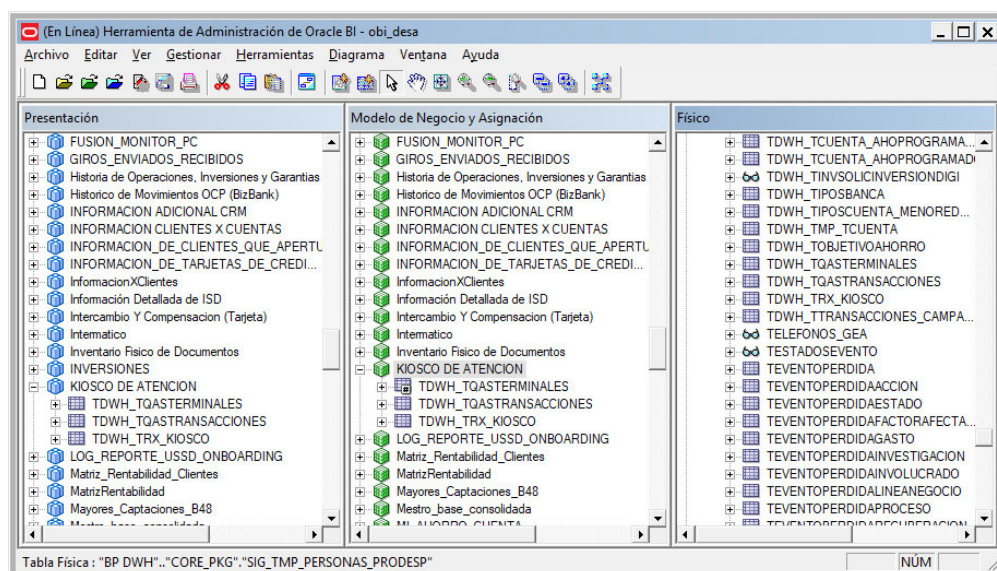


Figura 2.7 – Administrador de Herramientas de OBI

Fuente: Creado por el autor

Mediante el aplicativo del Administrador de Herramientas de OBI se realizaron consultas directas a las tablas migradas y estas fueron visualizadas después en el browser, en el que se realizaron consultas y se manipularon los datos para crear diagramas de barras, de quesos e histogramas; un ejemplo de lo que se puede hacer en OBI se muestra en la *Figura 2.8 – Aplicativo Oracle Business Intelligence*.

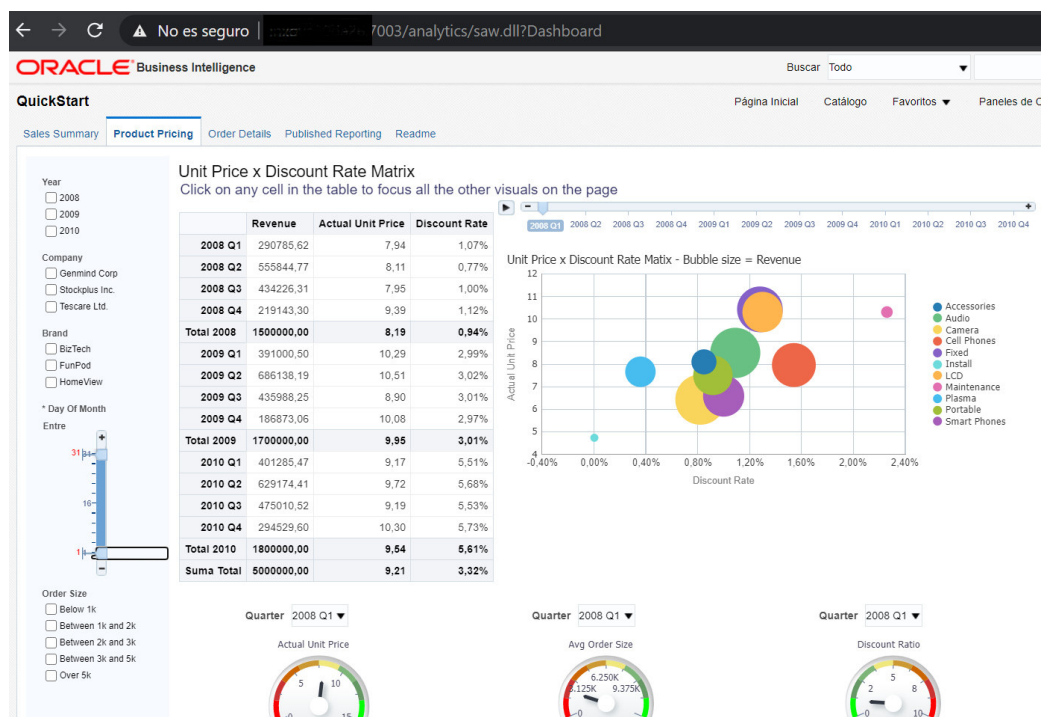


Figura 2.8 – Aplicativo Oracle Business Intelligence.

Fuente: Creado por el autor

2.3.5 CREACIÓN DE DASHBOARDS EN OBI (DESARROLLO)

Una vez que se creó la conexión a la base de datos y se logró visualizar las tablas en el browser, se procedió a realizar todos los dashboards que los usuarios requerían migrar desde el Discoverer Viewer, empezando con los libros de trabajo del área de Mercadeo, luego los de Tesorería y Finanzas para terminar con los de Recaudaciones.

Una vez concluida la creación de los dashboards, se realizaron pruebas internas en el área del DataWareHouse, en el que participaron los dos desarrolladores externos, un desarrollador interno y la jefa del área. Todos los dashboards fueron revisados y comparados con su homólogo del Discoverer Viewer, en esta revisión se constató que no hubo inconvenientes en la migración y que se podría realizar el paso a producción de todo el proceso.

2.3.6 IMPLEMENTACIÓN DE BASE DE DATOS PARA OBI (PRODUCCIÓN)

En conjunto con el área de base de datos, seguridad bancaria y sistemas operativos, se realizó el pase del proceso de OBI del ambiente de Desarrollo a Producción, al ser bastante extenso, se lo hizo por partes, primero creando la base de datos, luego la creación de tablas, la compilación de los paquetes y procedimientos.

Se realizaron pruebas de los procesos realizados, en los que se copió la data de la base de datos del DataMart a la de OBI, se tomó más días de los esperados debido a ciertos problemas de seguridades en los que no era posible hacer un Debug en producción por cuenta propia, por lo que se hizo esta revisión de fallos en

conjunto con el área de Base de Datos hasta que los procesos realizaron la copia de data y se verifico el éxito de la misma.

2.3.7 CREACIÓN DE DASHBOARDS EN OBI (PRODUCCIÓN)

Con la data de producción actualizándose día a día, se procedió a realizar los dashboards en los que se visualizaría la data solicitado por los usuarios, en este caso primero se migro los dashboards de un área y luego se revisó con los mismos, en lugar de migrarlo todo. Esto es porque los usuarios muchas veces aprovechan una migración para solicitar pequeños cambios en determinadas consultas.

Para cada área se colocaron los mismos dashboards que estuvieron en Desarrollo, la diferencia estaba en que ahora se visualizaba la data de producción, cada cierto tiempo los usuarios llegaban al puesto de los desarrolladores para solicitar un nuevo grafico de barras o pay y la forma en la que debía estar la data. Una vez que todos los cambios fueron realizados se convocó una reunión con el área a afectarse y se revisaban estos dashboards.

2.3.8 AJUSTES EN NUEVOS DASHBOARDS

Adicional a los dashboards de la migración, se realizaron unos pocos dashboards nuevos solicitados por los usuarios, en los que se realizó un proceso en la base de datos para enmascarar datos sensibles que se visualizarían en estos. Los dashboards fueron compartidos entre las tres áreas: Mercadeo, Tesorería y Finanzas, y Recaudaciones.

En la Figura 2.9 Dashboard OBI, podemos apreciar uno de los gráficos detallados que se pueden hacer con OBI.

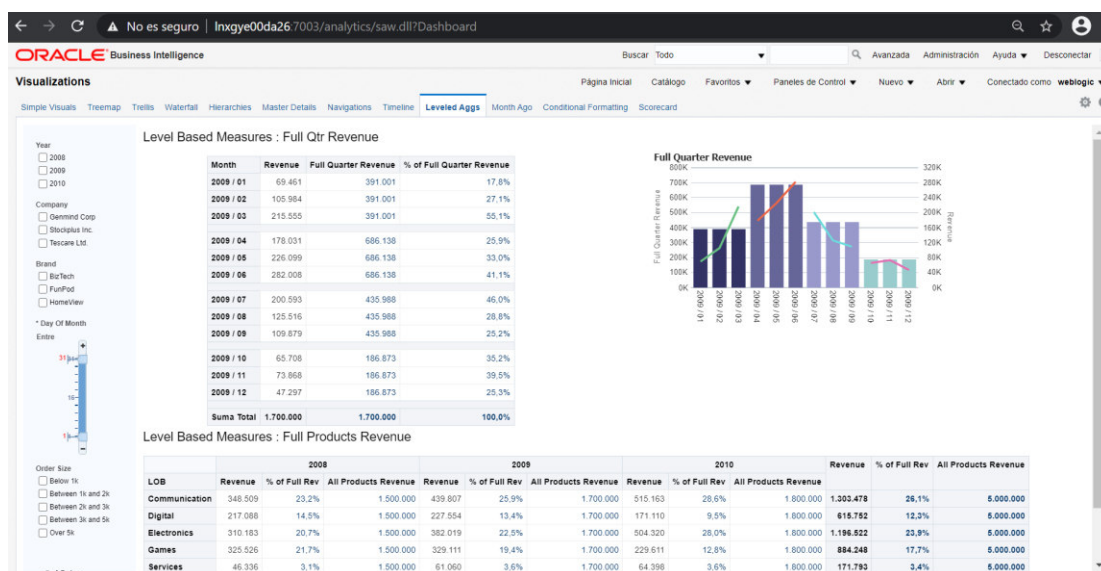


Figura 2.9 – Dashboard OBI

Fuente: Elaborado por el autor

2.3.9 CAPACITACIÓN AL PERSONAL DEL BANCO.

Una vez que se realizó la migración de los libros de trabajo de Discoverer Viewer a OBI y se verificó con los usuarios que obtuvieron lo requerido, era necesario que para los futuros Dashboards, el usuario no solo sea capaz de interactuar con el Dashboard creado sino también crear uno, para esto los desarrolladores externos se encargaron de realizar capacitaciones con clases prácticas. Los usuarios después de esta capacitación estuvieron preparados para crear y moldear un Dashboard según sus necesidades, el área del DataWareHouse solo estaría presente en el punto de inicio, para crear la conexión de una nueva tabla de la base de datos de OBI hasta el browser.

2.4 SEGURIDAD DE DATOS

Seguridad de datos de Clientes.

Para garantizar la seguridad confidencial en los datos de los clientes de la institución, el departamento de Seguridad Bancaria desarrolló un procedimiento en el servidor de OBI, en el cual los datos como Numero de Identificación, Números de Cuentas y Números de Tarjetas sean encriptados, protegiendo así los códigos originales de los mismos, a la vez

que permite realizar las consultas entre las tablas usando los códigos encriptados.

Respaldo de la data de OBI.

En caso de que el servidor de OBI sufra algún desperfecto, o un siniestro natural termine con la conexión del servidor, la entidad bancaria decidió colocar un respaldo completo en un servidor adicional con las mismas características del servidor original, el área de base de datos en conjunto con Seguridad Bancaria tomo dicho servidor y lo reutilizaron ya que estuvo en mantenimiento después de usarse en un proyecto ajeno a este y se encuentra en la ciudad de Quito, la forma en la que se transmite la información es bajo el proceso de Data Guard, el cual genera una entrada y salida de los datos en una base activa de Oracle, el proceso de escritura de la base de datos continuamente actualiza la información de OBI al servidor de Data Guard, mientras los archivos ReDo generan entradas y salidas adicionales, es la forma más rápida y eficiente en la que se puede replicar la información hasta el momento.

CAPÍTULO 3

EVALUACIÓN DE RESULTADOS

3.1 ACOMPAÑAMIENTO A USUARIOS AL USAR LA HERRAMIENTA DE OBI.

Aún con la capacitación realizada en días anteriores se tuvo que dar acompañamiento a los usuarios, ya que la herramienta posee demasiadas funciones y cuando se trata de crear un Dashboard muy elaborado y personal puede ser fácil perderse en el camino, esto se tuvo contemplado y a pesar de que ya no se contaba con el personal externo, el personal interno de la institución ayudo en el acompañamiento.

Al inicio era un poco difícil descifrar los gráficos que los usuarios requerían, por lo que se usó la herramienta de Microsoft Excel para que ahí desarrollen la idea, luego con la idea clara se les enseñó como hacerlo en la herramienta de OBI.

Los usuarios más avanzados que requerían de una gama de opciones mucho más completa para crear dashboards, se les otorgó una licencia de Oracle Data Visualization, solo dos personas requirieron esta licencia.

Actualmente todo el personal del DataWarehouse está capacitado para dar acompañamiento a un requerimiento del usuario, y si bien al inicio demandaba demasiado tiempo en el mismo, poco a poco fue disminuyendo, hasta hacerse poco frecuente, después de los dos primeros meses de liberar OBI para el uso de los usuarios, solo se han recibido 3 requerimientos en los últimos meses para dar acompañamiento en la creación de dashboards muy complejos.

3.2 DIFERENCIAS ENTRE EL APLICATIVO DE ORACLE BUSINESS INTELLIGENCE Y DISCOVERER VIEWER

Cuando se levantó la información acerca del aplicativo de OBI, se contaba con la enorme mejora visual en los reportes que manejaban los usuarios, sin embargo no se contaban con pequeñas diferencias que vale la pena mencionar.

Una diferencia notable es la forma en la que OBI muestra los resultados nulos o en blanco, ya que Discoverer Viewer mostraba la palabra *NULL* en algunos reportes cuando el registro no contaba con un valor en cierto campo, sin embargo en OBI el campo se presenta vacío, esto ocasionaba molestias en los usuarios ya que usaban este *NULL* para filtrar la información, llegando a solicitar que cuando un registro tenga un campo vacío, este sea rellenado con la palabra *NULL*; cabe indicar que no todos los libros de trabajo mostraban la palabra *NULL* cuando un campo estaba vacío, por ese motivo no se tomó el requerimiento y se instruyó a los usuarios para que tomen el campo vacío como nuevo filtro.

También se presentaron diferencias entre el número de registros arrojados por el Dashboard y el presentado en el archivo exportado del mismo, de este tipo hubo dos casos similares, sin embargo, en la revisión interna ejecutada por el equipo del DataWareHouse no se mostraban estas diferencias, cuando se solicitó a los usuarios que ejecutaran de nuevo los reportes estas diferencias desaparecían, esto se daba porque antes de exportar los datos, los usuarios añadían o removían un filtro en la consulta, por lo que se recordó a los usuarios la forma en la que debían descargar los datos y la afectación que ocurría en la visualización de la data por cada acción ejecutada por los mismos.

Debido a que las diferencias reportadas no eran más que errores o confusión a la hora de manejar la nueva herramienta por parte de los usuarios, estas novedades no fueron tomadas en cuenta para los requerimientos a atenderse, se recordó a los usuarios el compromiso de los departamentos para lograr que el proyecto funcione, a la vez se les motivo para que experimenten e investiguen acerca de nuevas formas de mostrar la data, de manera que logren tomar más información con los mismos datos.

3.3 MONITOREO DE RENDIMIENTO EN EL SERVIDOR DE OBI

Uno de los puntos fuertes por el que se tomó la iniciativa de migrar de aplicativo y a su vez colocar la data en otro servidor fue por el bajo rendimiento que se estaba evidenciando en el servidor del DATAMART, ya que en este servidor se colocaba toda la data histórica y no estaba resumida, por lo que cuando un usuario ejecutaba una consulta para conocer el estado actual de un servicio, el servidor debía realizar cálculos de sumatorias o conteos de registros campo a campo, con más de treinta consultas a la vez era demasiado para el servidor, además en este servidor se ejecutan procesos batch para el guardado de toda la data histórica de la base transaccional lo que la vuelve una base pesada y también es usada para generar los reportes que van hacia los diferentes organismos de control como la Súper Intendencia de Bancos o el Servicio

de Rentas Internas lo que la hace una base un tanto ocupada. Con los usuarios usando Discoverer Viewer el rendimiento de la base era muy bajo y llegaba a usarse hasta el 90% de la RAM del servidor en algunas ocasiones, esto mejoro considerablemente una vez que OBI fue implementado y los usuarios abandonaron el servidor de DATAMART para trasladarse al servidor de OBI, esto se puede visualizar en la *Figura 3.1 Rendimiento del servidor DATAMART*, en la que se muestra el rendimiento del servidor en un día de Julio y otro en Octubre.

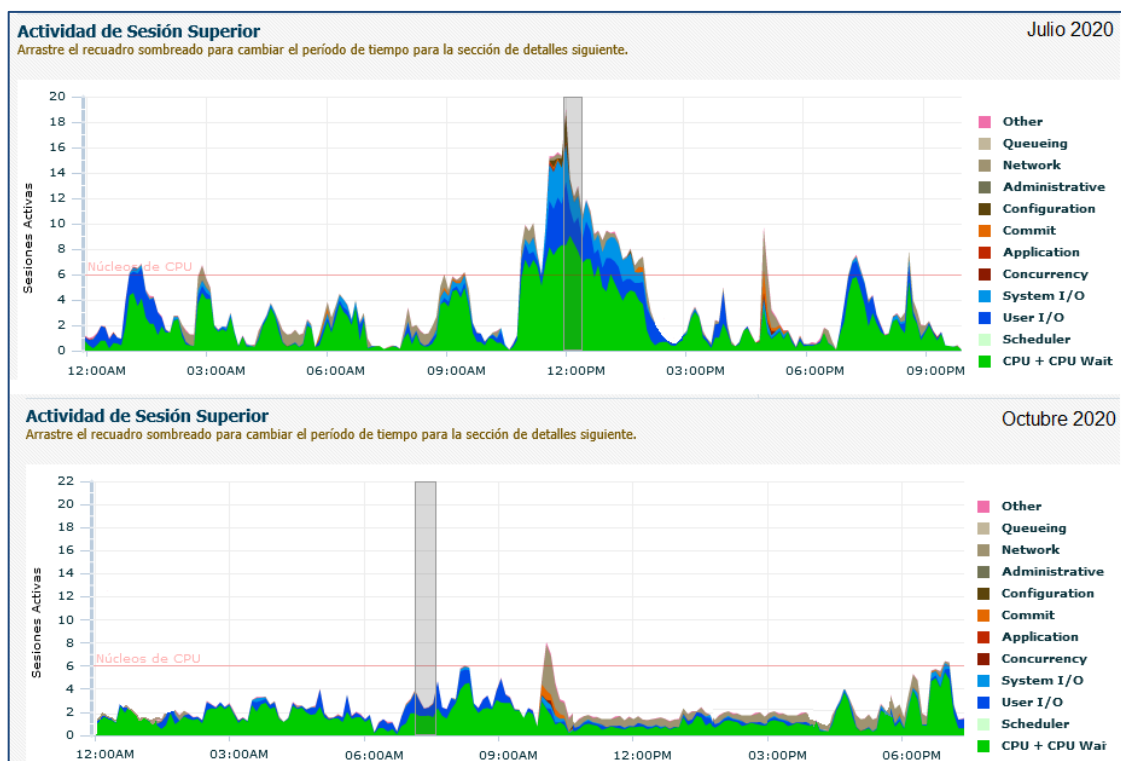


Figura 3.1 – Rendimiento del servidor DATAMART.

Fuente: Elaborado por el autor

Adicional a esto los desarrolladores del equipo del DataWareHouse también han notado un cambio notable en el rendimiento de la base, esto es porque la base se encuentra más libre de conexiones y de requerimientos, esto también ha afectado positivamente a los procesos automáticos de la base DATAMART, en la cual al medio día tenía un comportamiento muy lento y pesado, esto se visualiza en la *Figura 3.2 Rendimiento del servidor DATAMART, procesos automáticos* en el que podemos ver que al momento de realizar un Insert Masivo a la tabla de la base Datamart, se demoraba hasta 38 segundos por cada Insert masivo en el mes de Julio, sin embargo en Octubre se tomó la foto del mismo proceso y tuvimos tiempos máximos de 2.4 segundos por Insert Masivo.

Julio 2020										
Status	Duration	Type	SQL Plan Hash	User	Parallel	Database Time	IO Requests	Start	Ended	SQL Text
✓	21.0s	SQL		CORE_PKG		20.9s	15K	12:19:25 PM	12:19:46 PM	INSERT INTO SIG_TARJETA...
✓	24.0s	SQL		CORE_PKG		24.7s	15K	12:19:01 PM	12:19:25 PM	INSERT INTO SIG_TARJETA...
✓	27.0s	SQL		CORE_PKG		27.7s	15K	12:18:33 PM	12:19:00 PM	INSERT INTO SIG_TARJETA...
✓	30.0s	SQL		CORE_PKG		30.6s	15K	12:18:03 PM	12:18:33 PM	INSERT INTO SIG_TARJETA...
✓	32.0s	SQL		CORE_PKG		32.7s	15K	12:17:30 PM	12:18:02 PM	INSERT INTO SIG_TARJETA...
✓	35.0s	SQL		CORE_PKG		35.4s	15K	12:16:55 PM	12:17:30 PM	INSERT INTO SIG_TARJETA...
✓	38.0s	SQL		CORE_PKG		38.7s	15K	12:16:17 PM	12:16:55 PM	INSERT INTO SIG_TARJETA...
Octubre 2020										
Status	Duration	Type	SQL Plan Hash	User	Parallel	Database Time	IO Requests	Start	Ended	SQL Text
✓	3.0s	SQL		CORE_PKG		2.1s	648	12:41:21 PM	12:41:24 PM	INSERT INTO SIG_TARJETA...
✓	3.0s	SQL		CORE_PKG		2.2s	623	12:39:23 PM	12:39:26 PM	INSERT INTO SIG_TARJETA...
✓	3.0s	SQL		CORE_PKG		2.4s	629	12:39:09 PM	12:39:12 PM	INSERT INTO SIG_TARJETA...
✓	3.0s	SQL		CORE_PKG		2.1s	637	12:38:25 PM	12:38:28 PM	INSERT INTO SIG_TARJETA...
✓	3.0s	SQL		CORE_PKG		2.1s	636	12:34:42 PM	12:34:45 PM	INSERT INTO SIG_TARJETA...
✓	3.0s	SQL		CORE_PKG		2.0s	908	12:31:36 PM	12:31:39 PM	INSERT INTO SIG_TARJETA...
✓	3.0s	SQL		CORE_PKG		2.7s	2,481	12:30:46 PM	12:30:49 PM	INSERT INTO SIG_TARJETA...

Figura 3.2 Rendimiento del servidor DATAMART

Fuente: Elaborado por el autor

Con estos cambios en el rendimiento del servidor de Datamart, se logró que los procesos automáticos terminen máximo hasta las 11 de la mañana a diferencia de las 6 de la tarde en un día laboral normal.

Por otra parte, en el servidor de OBI no se contaba con problemas iniciales de consultar toda la información de las tablas ya que la data colocada en su base de datos esta resumida, a su vez no cuenta con campos que los usuarios no desean ver y están relacionados entre ellos incluso antes de que el usuario realice una consulta, por lo que el rendimiento al realizar una consulta directa a la base mejoro en un 42% en el peor de los casos y hasta un 75% en el mejor de ellos, esto fue medido realizando consultas directas hacia un grupo de tablas desde las bases de OBI y Datamart, cada una con un peso de más de 4Gb, la diferencia entre ellas estaba en el número de campos que poseía cada tabla. Esta forma de medir el rendimiento se lo hizo así porque no era posible medir exactamente la misma consulta de un reporte de OBI en el aplicativo de Discoverer porque la estructura de las tablas es diferente por lo que sería en vano compararlas de esta forma.

Al finalizar el desarrollo de los dashboards e iniciar las consultas por los usuarios en el servidor de OBI, el rendimiento del servidor era alto, sin embargo poco a poco fue mermando con el tiempo, se obtuvo lecturas iniciales de que el porcentaje de RAM usado del servidor de OBI era del

5%, poco a poco fue aumentando hasta llegar al 42% lanzando alarmas al grupo de Base de Datos, esto ocurrió a los dos meses de que los usuarios usaban el aplicativo de OBI y dejaron de usar completamente Discoverer Viewer, después de una extensa revisión por parte del área del DataWareHouse se evidencio que los usuarios tenían la costumbre de realizar las mismas consultas hacia la base en lugar de guardarlas para trabajar con las mismas, es decir que si necesitaban el reporte del número de tarjeta habientes entre Enero y Junio, consultaban esta información en Julio, Agosto, Septiembre a manera de compararlos con los meses venideros, esto se resolvió con las opciones con las que cuenta OBI, el aplicativo permite guardar los resultados para realizar comparaciones con consultas venideras guardando las mismas funciones que posee al momento de realizarse como es filtrar información por campo o búsqueda de un registro específico, etc.

Una vez que los usuarios hicieron uso de la función del guardado de resultados de las consultas, el rendimiento del servidor mejoro llegando a usarse un máximo de 15% de RAM, esto se logra evidenciar con la *Figura 3.3 Rendimiento del servidor OBI* en el que podemos ver la foto cuando se lanzó la alarma del rendimiento bajo y otra dos meses después de la instrucción de reusar las consultas guardadas.

Octubre 2020												
CPU	TTY	PID	USERNAME	PRI	NI	SIZE	RES	STATE	TIME	%WCPU	%CPU	COMMAND
7	?	7160	oracle	178	20	3470M	15904K	run	42:27	74.51	66.65	oracle
3	?	4	root	152	20	144K	128K	run	62:42	1.85	1.85	unhashdaemon
0	?	21338	oracle	178	20	3470M	16652K	sleep	1:10	1.65	1.65	ora_dbw0_GYE
0	?	19639	root	168	20	15692K	1176K	sleep	20:43	0.41	0.40	utild
Diciembre 2020												
CPU	TTY	PID	USERNAME	PRI	NI	SIZE	RES	STATE	TIME	%WCPU	%CPU	COMMAND
2	?	6518	root	152	20	144K	128K	run	62:35	1.83	1.83	oracle
1	?	6551	oracle	178	20	9470M	17860K	sleep	0:00	3.14	1.73	ora_j000_GYE
2	?	771	root	122	20	15412K	178K	sleep	0:41	0.40	0.40	util
2	pts/0	5579	copnod	178	20	16612K	1356K	run	0:01	0.36	0.36	top

Figura 3.3 Rendimiento del servidor OBI

Fuente: Elaborado por el autor

3.4 BENEFICIOS DE LA SOLUCIÓN

- Con la implementación de OBI, la mejora visual fue notable al momento de presentar un informe, gracias al abanico de opciones que ofrece la herramienta se logra obtener más información de la misma data, lo mejor de todo es que todo se logra hacer en el mismo aplicativo en lugar de usar programas externos, esto hizo que los usuarios mejoren sus tiempos al momento de presentar sus informes y a la vez mejorar la presentación de los mismos.
- Al contar con la opción de guardar reportes con sus resultados, estos ahora son usados por algunos gerentes de la institución, ellos son capaces ahora de obtener información de los reportes sin pasar por sus subordinados en algunos casos.

- La mejora del rendimiento de la base de Datamart fue considerable, con la base libre de usuarios se incrementó su rendimiento hasta en un 60%, lo que hizo que el proceso automático total propio de la base logre bajar su tiempo de ejecución hasta en 7 horas.
- Los datos de los clientes se encuentran más seguros, ya que los campos confidenciales se encuentran encriptados y otros campos solo están habilitados para algunos usuarios y ocultos para otros.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

1. La migración de Discoverer Viewer a Oracle Business Intelligence definitivamente represento una mejora visual al usuario y abrió nuevas posibilidades para mejorar las presentaciones de los KPI hacia la alta gerencia.
2. El colocar el aplicativo de OBI en un nuevo servidor libero el número de conexiones hacia la base de datos del DataMart, lo que causo una mejora notable de tiempos en el proceso automático de copia de información.

3. Los tiempos de respuesta para las consultas realizadas por los usuarios mejoraron al menos en un 42%, lo que hace que los usuarios se sientan más cómodos al usar el aplicativo de OBI.
4. La confidencialidad de los datos de los clientes se mejoró con la implementación de OBI ya que el mismo ofrece la opción de mostrar u ocultar campos, además que internamente también se realizó un enmascaramiento de los datos más sensibles.
5. Acompañar a los usuarios a usar una nueva herramienta y estar pendiente de lo que necesita, hizo que los mismos tengan más confianza de adentrarse en una nueva tecnología, minimizando la resistencia al cambio.

RECOMENDACIONES

1. Contar con un equipo apto para la asistencia a los requerimientos de los nuevos dashboards que pueda solicitar el usuario.
2. Realizar monitoreo mensual al servidor con la finalidad de predecir o corregir cualquier proceso inadecuado que se esté realizando por parte de los usuarios o por los procesos internos de la base.
3. Tener muy presente la fecha de caducidad de las licencias, con alarma de un mes antes de que finalicen las mismas.

4. Alentar a los usuarios a probar todas las funcionalidades que la herramienta pueda ofrecer para obtener el máximo provecho del aplicativo.

BIBLIOGRAFÍA

[1] Conceptual Data Warehouse Design - Bodo Husemann, Jens Lechtenböcker, Gottfried Vossen

https://www.researchgate.net/profile/Gottfried-Vossen/publication/2455394_Conceptual_Data_Warehouse_Design/links/0912f5092b22bef1bc000000/Conceptual-Data-Warehouse-Design.pdf

[2] From Enterprise Models to Dimensional Models: A Methodology for Data Warehouse and Data Mart Design - Daniel L. Moody, 20/03/2021

http://neumann.hec.ca/sites/cours/6-060-00/MK_entreprise.pdf

[3] Capacidades Técnicas, Legales Y De Gestión Para Equipos Blueteam Y Redteam - Jorge Fabián Bracho 20/03/2021

<https://repository.unad.edu.co/bitstream/handle/10596/36907/77185066.pdf?sequence=1&isAllowed=y>

[4] Arimetrics – Glosario Digital 20/03/2021

<https://www.arimetrics.com/glosario-digital/query>

[5] HostGator – 20/03/2021

<https://www.hostgator.mx/blog/que-es-debug-en-programacion/#:~:text=Tambi%C3%A9n%20llamado%20como%20debugging%20%E2%80%93%20o,pueden%20cometer%20softwares%20y%20hardware>
s.

[6] Role of Oracle Active Data Guard in High Availability Database Operations - Rab Nawaz, Tariq Rahim Soomro

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.401.8890&rep=rep1&type=pdf>

GLOSARIO

DataWarehouse: es una colección integrada y variable de datos derivados de datos operativos y utilizados principalmente en decisiones estratégicas. [1]

Datamart: generalmente se adaptan a las necesidades de un grupo específico de usuarioso tarea de toma de decisiones. [2]

Hardenizar: buscan endurecer las medidas de seguridad y cerrar más las brechas para que las vulnerabilidades en los equipos informáticos tanto a nivel personal como los que suelen estar presentes en las redes de las organizaciones. [3]

Query: Petición precisa para obtener información en una base de datos o sistema de información. [4]

Debug: Nombre que se le da al proceso de encontrar y eliminar los errores que pueden cometer softwares y hardwares.

Dataguard: Data Guard es uno de los equipos de recuperación ante desastres de alta disponibilidad y arquitectura de protección de datos de Oracle.