



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**CÓMO REALIZAR UN PLAN ESTRATÉGICO PARA MIGRAR DATOS DE UNA VERSIÓN
ANTIGUA DE INFORMIX A UNA VERSIÓN RECIENTE, SIN ESCALAR POR LAS
VERSIONES ANTERIORES**

José Arnulfo Roldán Caracún

Asesorado por el Ingeniero Saúl Alberto Pereira Puac

Guatemala, mayo de 2009

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**CÓMO REALIZAR UN PLAN ESTRATÉGICO PARA MIGRAR DATOS DE
UNA VERSIÓN ANTIGUA DE INFORMIX A UNA VERSIÓN RECIENTE, SIN
ESCALAR POR LAS VERSIONES ANTERIORES**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR:

JOSÉ ARNULFO ROLDÁN CARACÚN

ASESORADO POR EL ING. SAÚL ALBERTO PEREIRA PUAC

AL CONFERÍRSELE EL TÍTULO DE
INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, MAYO DE 2009

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Inga. Glenda Patricia García Soria
VOCAL II	Inga. Alba Maritza Guerrero de López
VOCAL III	Ing. Miguel Ángel Dávila Calderón
VOCAL IV	Br. José Milton De León Bran
VOCAL V	Br. Isaac Sultán Mejía
SECRETARIA	Inga. Marcia Ivónne Véliz Vargas

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Murphy Olympo Paiz Recinos
EXAMINADOR	Ing. Edgar Estuardo Santos Sutuj
EXAMINADOR	Ing. Ludwing Federico Altán Sac
EXAMINADOR	Ing. César Augusto Fernández Caceres
SECRETARIA	Inga. Marcia Ivónne Véliz Vargas

HONORABLE TRIBUNAL EXAMINADOR

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

**CÓMO REALIZAR UN PLAN ESTRATÉGICO PARA MIGRAR DATOS
DE UNA VERSIÓN ANTIGUA DE INFORMIX A UNA VERSIÓN RECIENTE,
SIN ESCALAR POR LAS VERSIONES ANTERIORES,**

tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, en noviembre 2005.


~~José Arnulfo Roldán Caracún~~

Guatemala, 06 de Octubre de 2008

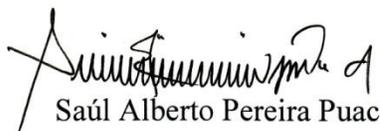
Escuela de Ciencias y Sistemas
Facultad de Ingeniería
Universidad San Carlos de Guatemala

Apreciables señores:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante JOSÉ ARNULFO ROLDÁN CARACÚN, titulado: “COMO REALIZAR UN PLAN ESTRATÉGICO PARA MIGRAR DATOS DE UNA VERSIÓN ANTIGUA DE INFORMIX A UNA VERSIÓN RECIENTE SIN ESCALAR POR LAS VERSIONES ANTERIORES”, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Sin otro particular, me suscribo de usted.

Atentamente,



Saúl Alberto Pereira Puac

Ingeniero en Ciencias y Sistemas

Colegiado No. 5186

Asesor y Revisor de Trabajo de Graduación



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 02 de Marzo de 2009

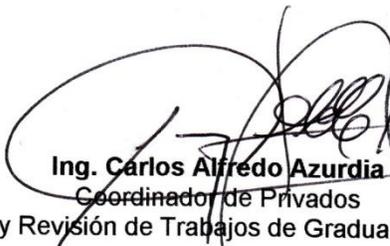
Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante **JOSÉ ARNULFO ROLDÁN CARACÚN**, titulado: **"COMO REALIZAR UN PLAN ESTRATÉGICO PARA MIGRAR DATOS DE UNA VERSIÓN ANTIGUA DE INFORMIX A UNA VERSIÓN RECIENTE SIN ESCALAR POR LAS VERSIONES ANTERIORES"**, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, de trabajo de graduación titulado **“COMO REALIZAR UN PLAN ESTRATEGICO PARA MIGRAR DATOS DE UNA VERSIÓN ANTIGUA DE INFORMIX A UNA VERSIÓN RECIENTE SIN ESCALAR POR LAS VERSIONES ANTERIORES”** presentado por el estudiante **JOSÉ ARNULFO ROLDAN CARACÚN**, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”



Ing. Marlon Antonio Perez Turk
Director, Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 14 de mayo 2009



El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **COMO REALIZAR UN PLAN ESTRATÉGICO PARA MIGRAR DATOS DE UNA VERSIÓN ANTIGUA DE INFORMIX A UNA VERSIÓN RECIENTE, SIN ESCALAR POR LAS VERSIONES ANTERIORES**, presentado por el estudiante universitario **José Arnulfo Roldán Caracún**, procede a la autorización para la impresión del mismo.

IMPRÍMASE.

Ing. Murphy Olimpo Paiz Recinos
DECANO



Guatemala, abril de 2009

/gdech

Este trabajo lo dedico a:

Dios, por permitirme la vida y por enviarme a las personas que me han acompañado en cada reto de mi vida. Por ser mi fuente de fortaleza y bendición en cada batalla emprendida.

Mis padres, José Arnulfo Roldán Monterroso y Melma Adela Caracún Gómez, por cada una de las puertas que me han facilitado abrir, por todos los buenos ejemplos que me han transmitido, por ser la motivación y el apoyo que Dios me ha regalado y del que me he sostenido toda la vida.

Mis hermanos, Johana Paola, Paulo César y Carlos Alberto, por los sacrificios a los que me han acompañado, por los innumerables favores que me han salvado y su compañía incondicional.

Mi novia, Rosa González, por el apoyo incondicional para terminar mi carrera. Por ser el hombro donde he desahogado frustraciones y éxitos, por su comprensión, paciencia, y sobre todo por el amor que cada día me demuestra.

Mis familiares, por sus oraciones, consejos, apoyo y muestras de afecto transmitidos. Elementos importantes para mi desarrollo y crecimiento.

Mis amigos de sistemas y sus familias, por todos los recuerdos que nos deja el paso en la preparación de la carrera, momentos alegres, de trabajo, de desvelo, de sacrificio, de frustración, y principalmente los de éxito. Gracias a sus familias por abrirnos las puertas de sus casas, por su confianza, hospitalidad y cariño. Gracias por su amistad y ser una pieza clave en mi formación personal y universitaria.

Mis amigos y compañeros de trabajo, que fueron un ejemplo para poder terminar mi carrera universitaria, especialmente, mi trabajo de graduación.

Mis amigos, gracias por su amistad incondicional, por darme la mano cuando lo he necesitado y por ser parte de muchos momentos de mi vida.

José Arnulfo Roldán Caracún

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	V
RESUMEN.....	XIII
PLAN ESTRATÉGICO	XXI
LÍMITES Y ALCANCES.....	XXIII
OBJETIVOS.....	XXV
INTRODUCCIÓN	XXVII
1. ANÁLISIS DE LA ESTRUCTURA ACTUAL DE LA BASE DE DATOS DE INFORMIX.....	1
1.1. Análisis de la integridad de tablas en la base de datos de Informix (versión antigua).....	1
1.1.1 Entidades:.....	2
1.2. Describir pruebas de migración de datos con un pequeño subconjunto de tablas de la versión antigua de Informix a la versión nueva de Informix	4
1.3. Analizar las pruebas con estos subconjuntos de tablas en aplicaciones con la nueva versión de Informix.....	10
1.3.1 Ventajas:	10
1.3.2 Desventajas:	11
2. ANÁLISIS DE MIGRACIÓN DE DATOS DE INFORMIX A SQL SERVER.....	13
2.1 Análisis de pruebas de migración de datos de Informix a SQL Server.....	13
2.2 Identificación de pros y contras con SQL Server.....	16
3. EXPORTACIÓN DE LA BASE DE DATOS DE INFORMIX HACIA ARCHIVOS UNL DE UN SERVIDOR A OTRO	19

3.1	¿Cómo realizar la exportación de data hacia archivos físicos Hexadecimales? (archivos UNL de Informix)	19
3.2	Utilización del protocolo FTP para el traslado de archivos UNL del servidor antiguo hacia el nuevo de Informix.....	24
4.	CONFIGURACIÓN DE LA BASE DE DATOS DE INFORMIX SOBRE ARCHIVOS BINARIOS EN EL NUEVO SERVIDOR DE INFORMIX	27
4.1	Levantar la base de datos de Informix sobre archivos binarios.....	27
4.1.1	Nombre del propietario de cada tabla	28
4.1.2	Propiedad Extent Size	28
4.1.3	Propiedad Revoke All	29
4.2	¿Cómo configurar manualmente el tamaño de paginación de tablas en la nueva versión de Informix?.....	29
4.2.1	Explicación de la gráfica “Proceso de Asignación del EXTENT”:	31
4.2.2	extent size 22,346.....	35
4.2.3	next extent 4,000;.....	35
4.3	Importación de data al nuevo servidor de Informix	38
4.3.1	Parámetros del comando “dbimport”	39
4.3.2	Parámetros Adiciones del comando “dbimport”	39
4.3.3	Parámetro –c.....	40
4.3.4	Parámetro –i /volu1/migracion.....	41
4.3.5	Parámetro –d data1dbs fha	42
4.4	Configuración de permisos a usuarios en la nueva versión de Informix.....	42
4.5	Análisis de pruebas para la verificación y comprobación del número de registros trasladados frente al número de registros en el servidor antiguo.....	43

5. “PERSONALIZACIÓN Y ÚLTIMAS CONFIGURACIONES EN LA NUEVA BASE DE DATOS DE INFORMIX”	47
5.1 Instanciamiento de la base de datos de Informix	47
5.2 ¿Cómo configurar el Driver Informix-Cli del lado del cliente y la forma de comunicación de datos a la nueva versión de Informix?.....	50
5.3 Configuración del driver para Informix “Informix-Cli” y los archivos del sistema hacia el nuevo servidor DBLNX	51
5.3.1 Archivos del Sistema:	51
5.4 Cambios en el driver Informix-Cli	52
5.4.1 Ilogin Demo	52
5.4.2 Setnet 32.....	53
5.5 Análisis del comportamiento de aplicaciones con la nueva versión de Informix.....	54
5.6 Descripción de resultados de pruebas	55
CONCLUSIONES	59
RECOMENDACIONES	61
REFERENCIAS ELECTRÓNICAS	63

ÍNDICE DE ILUSTRACIONES

FIGURAS

1. Bosquejo de Diagrama Entidad/Relación	3
2. Estructura de la tabla “caso”	14
3. Proceso de Asignación del EXTENT.....	31
4. Los Dbspaces enlazan las unidades de almacenamiento lógicos y físicos.....	32
5. Configuración de la propiedad Extet Size.....	34
6. Listado de tablas seleccionadas para configuración del Extent Size en kilobytes.	36
7. Reporte de Comparación de tablas entre el servidor anterior frente al servidor actual	44
8. Configuración de Instancias de la Base de Datos creadas en el sistema del FHA.....	49
9. Flujograma del proceso de migración de datos en Informix.....	57

GLOSARIO

Archivo binario	Un archivo binario es un archivo informático que contiene información de cualquier tipo, codificada en forma binaria para el propósito de almacenamiento y procesamiento en computadoras.
Db	El administrador de base de datos (DBA), es la persona responsable de los aspectos ambientales de una base de datos.
Dbexport	Comando utilizado por Informix para exportar información de la base de datos en archivos binarios hacia algún dispositivo de almacenamiento masivo.
Dbimport	Comando utilizado por Informix para importar información hacia la base de datos en archivos binarios desde algún dispositivo de almacenamiento masivo.
Db	Son las iniciales en inglés de Database Owner, que significan Propietario de la Base de Datos y es un usuario que tiene todos los permisos de realizar cualquier actividad sobre la base de datos.

Download	Término en inglés utilizado en informática para indicar que se bajará información de algún lugar hacia otro.
Driver	Programa que se utiliza para poder interactuar con algún dispositivo hardware.
Extent size	Instrucción de la Base de Datos de Informix que se utiliza para poder definir el espacio de almacenamiento por páginas de una tabla.
Ftp	Protocolo de Transferencia de Archivos, es un protocolo de transferencia de archivos entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor, de manera que desde un equipo-cliente nos podemos conectar a un servidor para descargar archivos desde él o para enviarle nuestros propios archivos independientemente del sistema operativo utilizado en cada equipo.
Grant all on	Esta instrucción permite dar permisos totales al objeto especificado de la base de datos de Informix.
Grant privileges	Privilegios de administrador dentro de la base de datos en Informix.
Host	Es un dispositivo de red que ofrece servicios a otros ordenadores conectados a dicha red.

Informix	Sistema de Base de Datos adquirido por IBM.
Informix-cli	Programa intermedio entre el cliente y la base de datos de Informix, que sirve para poder interactuar entre el cliente y la base de datos.
Instrucción bin	Instrucción del comando FTP para indicar que se está trabajando con archivos binarios.
Instrucción mput *	Instrucción en el comando FTP para indicar que se copiaran todos los archivos del directorio actual.
Instrucción quit	Instrucción del comando FTP que indica la salida de la sesión abierta del mismo.
Login	Término en computación que se utiliza para indicar que se iniciará sesión en el sistema donde se desea interactuar.
Odbc	Administrador de Origen de datos ODBC, se utiliza para configurar las aplicaciones de forma que puedan obtener datos desde una gran variedad de sistemas de administración de bases de datos.
Olsoctcp	Online soctcp, significa que el protocolo utilizado por la nueva versión de Informix será en línea.

Prompt	Comando interno del sistema operativo para indicar el símbolo del sistema.
Script	Es un archivo de texto con instrucciones de SQL. (en el contexto de base de datos)
Sesoctcp	Standar Engine soctcp, significa que el protocolo de comunicación de Informix es un moto estándar.
Sqlhosts	Archivo de configuración de Informix, en el cual cada línea de este archivo define un servidor Informix.
Stored procedures	Define una serie de instrucciones de SQL dentro de la base de datos para realizar alguna tarea programada, de allí su nombre “procedimientos almacenados”
Telnet	TELEcommunication NETwork, es el nombre de un protocolo de red (y del programa informático que implementa el cliente), que sirve para acceder mediante una red a otra máquina, para manejarla remotamente como si estuviéramos sentados delante de ella.
Triggers	Un trigger o un disparador en una base de datos es un evento que se ejecuta cuando se cumple una condición establecida al realizar una operación de inserción (insert), actualización (update) o borrado (delete).

Unl

Son archivo que contienen datos en formato hexadecimal y son generados al momento de bajar la información de la base de datos con la opción dbexport. Este crea uno por cada objeto de la base de datos.

Views

Una vista de base de datos es un resultado de una consulta SQL, de una o varias tablas; también se le puede considerar una tabla virtual.

RESUMEN

En la actualidad, muchas de las empresas guatemaltecas trabajan con una gran variedad de sistemas de cómputo, operativos, de bases de datos y cliente/servidor. En fin, existen muchas empresas en el país, sobre todo las grandes que trabajan con sistemas un tanto más complejos y poco comunes, sin embargo, son sistemas que por su estructura fueron diseñados para mantener una forma de trabajo muy ortodoxa, además, muchas de las grandes empresas en Guatemala tienen por lo regular también una gran trayectoria y por ende tienen un mayor tiempo que las demás.

Con el transcurrir del tiempo, los sistemas han venido evolucionando sin embargo, hay muchos que por su gran desempeño y fiabilidad no evolucionaron en el tiempo dentro de la empresa, como es el caso del Instituto de Fomento de Hipotecas Aseguradas, “FHA”. Esta empresa semi-estatal tiene una gran trayectoria de hace muchos años en Guatemala y se dedica al fomento de hipotecas para la vivienda, exclusivamente para ciudadanos guatemaltecos. Con el transcurrir de los años el FHA ha trabajado con el sistema operativo Unix y una base de datos Informix. Hace unos pocos años se cambió el sistema operativo Linux que tiene como origen Unix. Para el FHA cambiar de un sistema operativo a otro no fue un cambio considerable o de una gran magnitud en cuanto al impacto que las aplicaciones tuvieran debido a que Linux tiene una gran similitud con Unix. Ahora bien, hace unos meses atrás Linux como sistema operativo funciona actualmente muy bien, no así para la versión de Informix como base de datos, pues el instituto ha venido mejorando y automatizando muchos procesos que son necesarios de una gran precisión y exactitud, requerimientos que para la versión de Informix ya no es posible cumplirlos.

Aunque es muy probable que la razón principal por la que Informix está limitado sea porque muchas de sus aplicaciones no corren en Linux sino en Windows, y además de esto, las herramientas de desarrollo son Visual Basic 6 y .Net recientemente. En todas las aplicaciones, las conexiones a Informix se hacen por medio de un ODBC. Este control de origen de datos tiende a limitar a las herramientas de desarrollo en muchas de las funciones de Informix, además que esta versión de Informix ha venido quedándose muy obsoleta y no es posible utilizar, por ejemplo, nuevas funciones propias de SQL Informix entre otras.

Debido a la gran limitación de la actual versión de Informix, en el FHA se realizó un estudio sobre la problemática de este punto, llegando a la conclusión que es necesario realizar un cambio de la base de datos y para esto se realizó un plan estratégico para migrar datos de una versión antigua de Informix a una versión reciente sin escalar por las versiones anteriores. Brevemente se comentará en qué consiste este plan estratégico de migración de datos, debido a que su contenido se explicará más detalladamente a lo largo de este trabajo de investigación de fin de carrera.

Para poder entender realmente qué envuelve una migración de datos podemos mencionar las siguientes definiciones: ¹ “una migración incluye la conversión (mejoramiento) hacia una última versión de un servidor de base de datos, reversión hacia una versión anterior de un servidor de base de datos, mover data dentro de las bases de datos, servidores de base de datos sobre el mismo sistema operativo, servidores de base de datos sobre diferentes sistemas operativos y diferentes clases de servidores de base de datos.

¹ **IBM CORP.** “IBM Informix Migration Guide versión 9.4”, Segunda Edición, Agosto de 2004, Capítulo Introduction, About This Manual, página ix-x

La conversión o reversión a menudo involucra cambios en la información de conectividad en los archivos `sqlhosts` o en las llaves del registro, en variables de ambiente de los *host*, parámetros de configuración y otras características de la base de datos”.

Como primer paso para llevar a cabo un cambio de la versión de Informix hacia una versión mucho más reciente, se planeó verificar la integridad de todas las tablas de la base de datos de Informix “fha”. Esto se llevó a cabo porque se consideró que para poder realizar una migración de datos es necesario que la nueva versión tenga una integridad de estructura tanto de tablas como de datos, debido a que en años anteriores el FHA ha sufrido muchos cambios tanto de políticas como de formas de trabajo en sus procesos básicos y con el tiempo se han quedado muchos datos sin depurar o sin tener una normalización correcta con los procesos actuales.

Luego de esta revisión de integridad de tablas se realizó como algo muy importante llevar a cabo pruebas con la nueva versión de Informix. Para esto se consideró un subgrupo de tablas del sistema, las más utilizadas dentro de los procesos de trabajo, se crearon las mismas estructuras de estas tablas en la nueva versión de Informix cargada en un nuevo servidor Linux que soportará dicha versión de la base de datos. Este cambio de servidor fue también tomado en consideración debido a que el actual servidor donde está instalada la versión de Informix antigua no cumple los requerimientos de la nueva versión. Al momento de terminar la creación de la estructura de prueba de la base de datos en la nueva versión de Informix, se hicieron pruebas sobre un conjunto de aplicaciones donde se utilizó el subgrupo de tablas que se crearon con anterioridad, y cuando se corrieron las aplicaciones no hubo que realizar ningún cambio, incluso la conexión aún se realizó por medio de ODBC y el funcionamiento fue correcto, por lo que las pruebas ante esta nueva versión de la base de datos fue totalmente transparente, y esto dio lugar a que se trazaran bien los pasos para llevar a cabo el proceso de migración de datos global.

Pero aun sabiendo que las pruebas fueron un éxito y que la migración de datos total podría llevarse a cabo sin ningún riesgo de incompatibilidad ante las aplicaciones, surgió una nueva alternativa, migrar hacia una nueva base de datos totalmente nueva y diferente a Informix. Aprovechando que se iba a realizar esta migración de datos se pensó en migrar hacia la versión 2005 de SQL Server.

Se llevó a cabo el mismo proceso de prueba que anteriormente se realizó con la nueva base de datos de Informix, se crearon tanto la base de datos como el subconjunto de tablas que se escogieron como parte principal en los procesos de trabajo, sin embargo, debido a que la estructura de las tablas es bastante compleja por el tamaño de estas, el archivo de secuencia de SQL fue generado por parte de Informix incrustando en tal *script* estructuras y sentencias que son propias por el mismo Informix y que SQL Server 2005 no pudo interpretar debido a que no conformaban parte de un SQL Estándar. Pero aun así aunque fue arreglado el *script* para que corriera en SQL Server 2005 surgieron inconvenientes más graves. Uno de los problemas que surgieron mayormente fue la conexión que se tiene en el código fuente de las aplicaciones del sistema del FHA que en su mayoría están diseñadas en Visual Basic 6.0.

Existe mucho código que debe cambiar para poder conectarse a SQL Server, además la forma de conexión que se tiene en estas aplicaciones requerían de reestructurar totalmente esta parte, junto a esto hay funciones propias de Informix que están inmersas en el código fuente y habría que cambiarlas y/o sustituirlas por las equivalentes en SQL Server, otro problema se originó en los *stored procedures*, vistas y triggers que se tienen programados en Informix, estas estructuras también deberían cambiar casi en su totalidad pues existen consultas que tienen funciones propias de Informix y que en SQL Server 2005 habría también que buscar su equivalente para tales funciones y probar su funcionamiento y comparar si devuelven el mismo resultado de información.

Por otra parte, la herramienta con la cual fueron hechos la inmensa cantidad de reportes existentes en el FHA es *Cristal Reports*, éste tiene incrustado en cada reporte y sub-reportes dentro de estos mismos, el nombre del servidor al cual se está conectando para obtener los datos. Esto hubiera implicado modificar la totalidad de todos los reportes cambiando el nombre del servidor de origen de datos y examinando cuidadosamente la existencia de sub-reportes en cada reporte, debido a que también en esta parte se hubiera tenido que modificar el nuevo nombre del servidor de origen de datos.

Como resultado, debido a la magnitud de estructuras como procedimientos, vistas, *triggers*, tablas y código fuente se llegó a la conclusión de que el impacto de migrar de la versión antigua de Informix hacia la nueva base de datos de SQL Server 2005 es sumamente grande a nivel del sistema y habría que reprogramar totalmente cada estructura en el sistema y probar su funcionamiento para poder confiar totalmente en la migración de datos. Dicho proceso lleva consigo un costo y tiempo que en el FHA no se había tomado en consideración, ya que el proceso de migración de datos se planteo de tal forma que se considerara y se pudiera llevar a cabo de una forma transparente y no de una forma que impactara radicalmente todo el proceso de trabajo que se ha venido realizando durante años atrás, por lo que llevar a cabo esta prueba de migración de datos con SQL Server sirvió para tener conocimiento del esfuerzo tan grande que se llevaría al cambiar de base de datos.

Después de haber realizado las pruebas correspondientes para la migración de datos y funcionamiento de las aplicaciones en la nueva versión de Informix se debe proceder a realizar toda la migración completa de la estructura y data globalmente.

Este procedimiento consiste primero en exportar la base de datos hacia el disco duro (*download*) de la estructura y data, por medio del comando de exportación *dbexport* de Informix.

Este comando exporta toda la data y la estructura hacia un directorio del disco duro indicado por el usuario, dejando en él físicamente un archivo de extensión .unl por cada tabla con su data dentro del sistema de la base de datos, junto con estos archivos unl también se crea un archivo de estructura de la base de datos el cual contiene secuencias de instrucciones SQL para poder generar la estructura completa de la base de datos, este archivo lleva el nombre de la base de datos más una extensión sql, de esta forma el archivo se llamaría fha.sql por el nombre de la base de datos.

Terminado el proceso de *download* de la base de datos hacia el disco duro del servidor, se procede a pasar estos archivos hacia un directorio en el servidor nuevo donde se instalará la nueva base de datos de Informix. Para realizar este traslado de archivos de un servidor a otro se utilizó el comando FTP. Una vez trasladados los archivos binarios con extensión unl se procede a asegurar el buen funcionamiento de las tablas principalmente de las más utilizadas en el sistema, para esto se configura manualmente el tamaño de paginación de cada tabla en el servidor nuevo de Informix para evitar fragmentación interna en estas tablas. Esto se realiza ajustando la propiedad *Extent Size* de cada tabla en el archivo de estructura de la base de datos que el comando dbexport creó para la base de datos cuando se bajó hacia el disco duro, este archivo *script* contiene las instrucciones SQL para crear toda la estructura de la base de datos, esto quiere decir que tiene instrucciones SQL como *create table*, *create view*, etc. La propiedad *Extent Size* se modificó únicamente en las tablas más grandes y más utilizadas en el sistema debido a que el almacenamiento de datos es muy grande que puede llegar a provocar una fragmentación interna en esta información haciendo más lento el acceso a estas tablas.

Hecho estos cambios en el archivo de estructura de la base de datos se levanta la base de datos, primero su estructura y luego las tablas en la nueva versión de Informix, para esto utilizamos el comando `dbimport` de Informix, este comando lo que hace es interpretar los archivos `unl` y crear la base de datos junto con sus tablas y data en la nueva base de datos indicada por el usuario.

Para poder configurar los permisos en la base de datos se hizo un programa sencillo en *Visual Basic 6*, dicho programa se utilizó para que de forma automática al ingresar el nombre del usuario, este configurara fácilmente los permisos necesarios en las tablas seleccionadas de la base de datos, agilizando así el proceso de asignación de permisos.

Para poder comprobar la exportación exitosa de los datos hacia la nueva base de datos se procedió primeramente a comparar el número de tablas exportadas al nuevo servidor con el número de tablas en el servidor origen, sin tomar en cuenta las tablas del sistema ya que la nueva versión de Informix tiene más tablas del sistema que la versión anterior, seguidamente, se procedió a comparar la cantidad de registros en cada tabla de la nueva versión de la base de datos con la antigua versión, únicamente para verificar el traslado exitoso de los registros en las tablas. Como una configuración del uso y manejo de la base de datos y como forma de seguridad para la data en la base de datos se llevó a cabo la configuración de instancias o copias idénticas de la nueva base de datos de producción de Informix “fha”. Estas copias se hicieron para diversos propósitos, por ejemplo: una instancia para producción, otra instancia para desarrollo de aplicaciones y otra instancia para pruebas en casos particulares.

Una vez configurada y lista la base de datos en el servidor nuevo se configuró en cada máquina cliente el *driver* que sirve como configurador de la base de datos en las aplicaciones, este *driver* es único para Informix y se llama Informix-Cli.

Aquí se configuran parámetros de la base de datos como el DBDATE que es configurado como DMY4/, esto porque la data y la forma de acceder a los registros que contienen información de fechas y horas están por día, mes y año, este último de cuatro caracteres, también el DBPATH cambió debido a que en la versión antigua de Informix aquí se especificaba la ruta de acceso a la base de datos, en cambio, en la versión nueva esta configuración es explícita y no lleva ninguna información en esta opción.

Otra propiedad es el INFORMIXSERVER que es ajustada con el nombre del servidor de Informix que en este caso es “fha_tcp”. El HOST es “dblhx”, el servicio es ajustado en la propiedad SERVICE con el valor “sqlturbo”, el protocolo utilizado por la nueva versión de la base de datos cambio de “SESOCTCP” que significa *standard engine sock tcp* por “OLSOCTCP” que significa *on line sock tcp*. Esto debido a que la base de datos antigua era una versión *standard engine* y ahora la nueva versión de la base de datos es *on line engine*. También otra de las propiedades que se configura en el Informix-Cli es el usuario y contraseña para iniciar una conexión en el servidor de Informix.

Debido a que la configuración en la conexión hacia la base de datos en las aplicaciones está configurada haciendo uso de un ODBC, es necesario entonces hacer un ODBC escogiendo entre el listado de controladores, el Informix *Driver*, configurando el ODBC con este *driver* y con el nombre “FHA” la conexión en los programas de aplicación es transparente al cambio de la nueva versión de la base de datos. Por lo que únicamente resta verificar el comportamiento, funcionamiento y rendimiento de las aplicaciones con la nueva versión de la BD de Informix.

PLAN ESTRATÉGICO

MISIÓN

Realizar el proceso de migración de datos de forma transparente para los usuarios, actualizando todos los programas y módulos que conforman el sistema del Instituto de Fomento de Hipotecas Aseguradas, FHA, hacia la nueva versión de la base de datos de Informix. Llevando también a cabo una nueva forma y metodología de trabajo en ambientes de desarrollo, pruebas y producción.

VISIÓN

Mantener siempre óptimo el rendimiento del sistema de base de datos de Informix aprovechando todas las nuevas funciones, herramientas y seguridad para el acceso y manipulación de información del Instituto de Fomento de Hipotecas Aseguradas, FHA que el servidor Informix provee con sus nuevas versiones.

OBJETIVOS

1. Verificar integridad de la BD antigua de Informix.
2. Preparación de estructura y data de la BD antigua de Informix.
3. Traslado de BD antigua de Informix al nuevo servidor de Informix.
4. Configuración y pruebas de la nueva versión de la BD de Informix
5. Redactar un informe final.

LÍMITES Y ALCANCES

Límites

1. No profundizar en la configuración del nuevo servidor de Informix.
2. La migración de datos se llevará a cabo únicamente sobre la estructura de la base de datos con sus usuarios y privilegios respectivamente.
3. Del lado del cliente se limitará a explicar cómo configurar el *driver* Informix-Cli para la conexión a la base de datos.
4. No se analizarán problemas con software de aplicación que utilizaran el driver de Informix-Cli.
5. No se profundizará en la configuración y problemas de instancias de la base de datos de Informix.

Alcances

1. Explicar los pros y contras de realizar la migración de datos hacia una base de datos distinta como es SQL Server.
2. Poder describir la solución óptima a este tipo de migración de datos en Informix.
3. Analizar y desarrollar de forma práctica y sencilla los pasos para poder realizar una migración de datos tan especial como este tipo de base de datos.

OBJETIVOS

General

Realizar un documento de referencia.

Específicos

1. Explicar cuál es el fin de realizar esta migración de datos.
2. Identificar los pasos necesarios para realizar la migración de datos.
3. Documentar detalladamente cada uno de los pasos identificados.

INTRODUCCIÓN

El presente trabajo de graduación, tiene dos finalidades principales, una es realizar un documento de referencia para las personas que necesiten llevar a cabo una migración de datos con este tipo de base de datos y otra es recopilar de forma adecuada y ordenada el conocimiento de estos sistemas ortodoxos en un informe escrito bien elaborado. Este documento detalla cuáles son los pasos necesarios para realizar dicha migración, de una versión antigua de Informix a una versión mucho más reciente sin escalar por las versiones intermedias.

Uno de los temas principales que más se hace alusión es la verificación de la integridad de la base de datos antigua, debido a que no es posible ni recomendable realizar dicha migración de datos si la estructura de las tablas y la información no tienen correctamente una consistencia de datos, de esta forma se debe preparar la base de datos para poder migrar hacia la versión reciente. Un detalle muy importante es que la nueva versión de Informix puede no correr en el servidor donde está instalada la versión antigua de este mismo, en este documento se hace mención de un nuevo servidor de Informix, sin embargo, no se indica el detalle de su instalación, debido a que éste es un punto que no conlleva dentro del objetivo principal de este trabajo de investigación.

Por último, se describe una serie de pruebas después de la migración de datos sobre la nueva versión de Informix instalada en un nuevo servidor, dichas pruebas se realizan sobre el funcionamiento de las aplicaciones y se describe cuál ha sido su desempeño con la nueva versión de la base de datos.

1. ANÁLISIS DE LA ESTRUCTURA ACTUAL DE LA BASE DE DATOS DE INFORMIX

1.1. Análisis de la integridad de tablas en la base de datos de Informix (versión antigua)

Como un punto importante para poder iniciar un proceso de migración de datos hacia cualquier otra base de datos ya sea igual o distinta, es necesario comprobar la estructura de la base de datos mediante un modelo entidad/relación. En este caso se inició sacando un listado de tablas necesarias y de utilidad de la base de datos “FHA”, esto porque existían tablas que con el transcurrir del tiempo fueron creadas debido a ciertas eventualidades y necesidades, y que por esta razón no hubo tiempo para incluirlas dentro del modelo original.

El modelo entidad/relación de una base de datos lo podemos representar por medio de un ² “Diagramas E/R”, que es una técnica para representar la estructura lógica de una base de datos en forma de gráfica. Como tal, proporciona un medio sencillo y de fácil comprensión para comunicar las características sobresalientes del diseño de cualquier base de datos dada, (“una imagen vale más que mil palabras”).

² C.J. Date, Introducción a los Sistemas de Base de Datos, Séptima Edición, 2001, Capítulo 13
Modelo Semántico, páginas 427-428

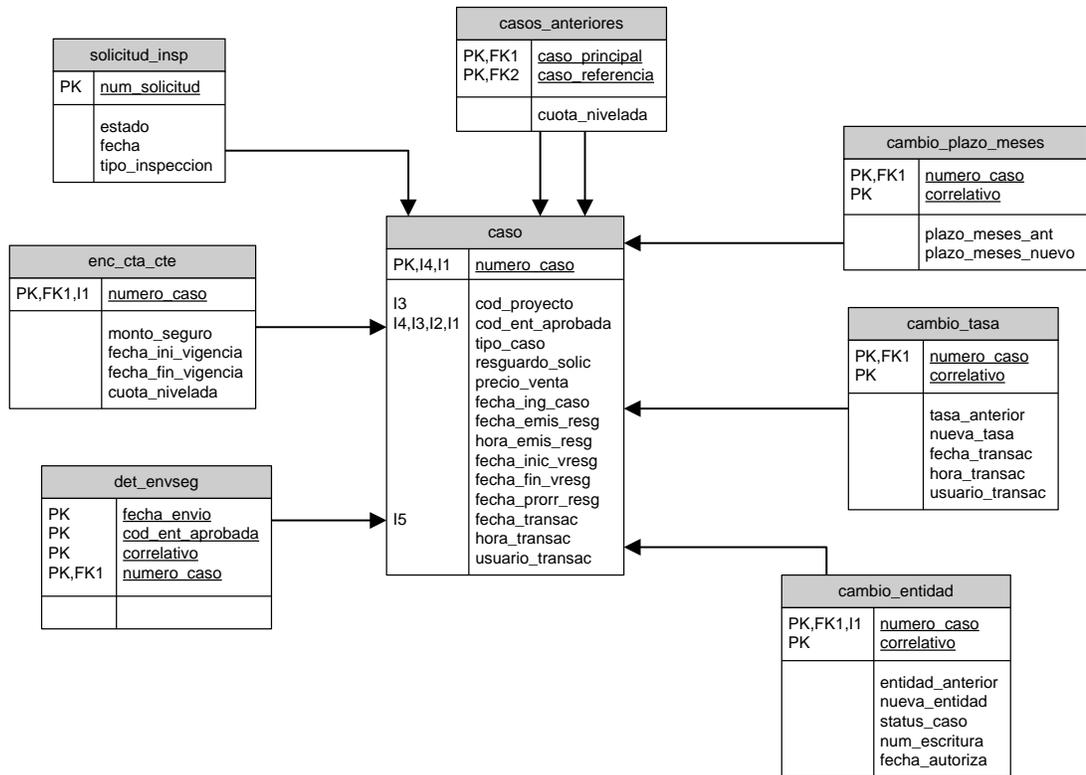
Para poder elaborar un diagrama entidad/relación primero identificamos las entidades y las propiedades de la base de datos. Realizaremos entonces un listado de las tablas que el sistema realmente utiliza. Por cuestiones de seguridad se describirán solamente algunas tablas principales y secundarias del sistema completo de la institución para luego hacer un ejemplo del diagrama entidad relación.

1.1.1 Entidades:

- **Caso:** Entidad que contiene la información del encabezado de los expedientes de solicitud de resguardo de asegurabilidad.
- **Enc_cta_cte:** Tabla de encabezado de la cuenta corriente de primas.
- **Solicitud_Insp:** Tabla donde se detallan las Solicitudes de Inspecciones.
- **Det_env_seg:** Tabla de detalle del documento con el cual se envían los seguros a la entidad aprobada.
- **Cambio_entidad:** Cambio de entidad aprobada cuando el caso ya es resguardo.
- **Cambio_tasa:** Cambio del % de la tasa cuando el caso ya es resguardo.
- **Cambio_plazo_meses:** Cambio del plazo en meses cuando el caso ya es resguardo.
- **Casos_anteriores:** Tabla que indica los casos que ha tenido un deudor con anterioridad.

En la figura 1 podemos observar el bosquejo de un diagrama entidad relación.

Figura 1. Bosquejo de Diagrama Entidad/Relación



Esta fue una de las formas para analizar la integridad de la base de datos “fha”, debido a que principalmente lo que se deseaba corregir fue la relación de las tablas, la integridad referencial y la consistencia de datos.

Otro de los pasos que se realizaron dentro del análisis de la base de datos, fue realizar un informe de nombres de tablas así como su número de registros, número de columnas y filas que cada tabla contiene. Toda esta información es necesaria para que después de haber realizado completamente la exportación de datos, podamos comparar la información dada por el sistema antiguo.

1.2. Describir pruebas de migración de datos con un pequeño subconjunto de tablas de la versión antigua de Informix a la versión nueva de Informix

El propósito principal de estas pruebas es poder llevar a cabo una migración completa de toda la estructura de la base de datos de forma confiable y segura en el nuevo servidor de Informix. Como primer paso lo que se hizo fue elegir las tablas más utilizadas y complementarias que el sistema utiliza frecuentemente en las aplicaciones. Básicamente el concepto es elegir un pequeño subconjunto de tablas de toda la estructura de la base de datos para poder realizar nuestras pruebas en la nueva versión de Informix. Esto nos permitirá poder montar una muestra de la estructura completa de la base de datos en el nuevo servidor y poder probar algunas aplicaciones. Para esto se llevo a cabo una selección de tablas así como también se rectificaron sus índices y data en algunos casos (subconjunto de tablas).

Para este subconjunto de tablas utilizado, se tomaron las tablas que utiliza el modulo conocido como Mayor Contable. Este es el encargado de centralizar todos los registros contables de la Institución, el cual está integrado por un subconjunto de tablas que aproximadamente son 20.

El conjunto de tablas utilizadas se describe a continuación:

- **bita_datos_caso:** Bitácora de la tabla caso.
- **det_cta_cte:** Contiene el detalle de la cuenta corriente de primas.
- **control_casos:** Lleva el control de expedientes.
- **new_ctrl_casos:** Se utiliza para el traslado de casos.
- **det_cedula_hipot:** Contiene el detalle de la información de cedulas de hipoteca.
- **det_mayor_hist:** Contiene el detalle del historial del mayor.

- **saldo_capital_seg:** Lleva el control de seguimiento al saldo de capital.
- **deudores_caso:** Contiene la información del detalle de los deudores del caso.
- **Caso:** Contiene información del encabezado de los expedientes de solicitud de resguardo de asegurabilidad.
- **tmp_cta_cte:** Es una tabla temporal de la tabla cuenta corriente.
- **Casas:** Representa la conformación de un proyecto, típicamente en manzanas y lotes.
- **new_inspecciones:** Almacena las inspecciones realizadas.
- **enc_recibo_caja:** Contiene la información del encabezado de los recibos de caja.
- **enc_cta_cte:** Contiene el encabezado de la cuenta corriente de primas.
- **solicitud_insp:** Se almacena la información de las solicitudes de inspecciones.
- **sus_det_notas:** Contiene la información del detalle de notas de suspensión.
- **det_cxc_proyecto:** Se almacenan los movimientos por pagos a cuenta, ya sean débitos o créditos.

Después de haber seleccionado el conjunto de tablas que nos servirá para hacer pruebas, los pasos que se van a definir a continuación se describen detalladamente en el capítulo tres donde se amplía cada proceso que se llevo a cabo, resumidamente estos son:

1. Se utilizó el comando `dbexport` para bajar la información de las tablas seleccionadas a archivos binarios `unl`. Se creó una carpeta en el servidor actual para poder almacenar los archivos binarios que el comando `dbexport` creó al bajar la información de la base de datos llamada “FHA”, este comando permite crear un archivo de secuencia de instrucciones de SQL que contiene la estructura completa de la base de datos, aparte de esto crea un archivo binario por cada tabla existente en el sistema.

2. Corregir el archivo de secuencia de instrucciones SQL que contiene la estructura de la base de datos. Este archivo de secuencia de instrucciones de SQL se corrigió en las siguientes instrucciones:
 - Nombre del propietario de cada tabla: estas instrucciones se corrigieron a un solo nombre de propietario debido a que en la mayoría de las tablas el nombre del propietario variaba según el usuario que creaba la tabla, esto se corrigió a un solo usuario como un `dbo` (propietario de base de datos)
 - Propiedad `Extent Size`: en algunas tablas con mayor peso de datos, por ejemplo la tabla “caso”, esta instrucción se agregó para poder ampliar el manejo de información en las tablas seleccionadas, como resultado esta instrucción define el tamaño de información que soporta cada tabla para evitar la fragmentación interna de estas.

3. Trasladar esta información a una carpeta nueva en el servidor destino utilizando el comando `FTP`. Por medio de este comando se paso toda la información creada por el comando `dbexport` hacia una carpeta del servidor destino. Para poder trasladar todos los archivos binarios hacia el nuevo servidor, se abrió una consola de comandos ubicados en el servidor origen (actual). En esta consola se abrió una sesión de `FTP` hacia el servidor nuevo y por medio de las instrucciones del comando `FTP` se hicieron los siguientes pasos:

- Se creó una nueva carpeta donde se almacenaran los archivos a trasladar.
 - Se ingreso a esta carpeta nueva y con la instrucción prompt se mostró la ubicación actual para poder trasladar los archivos.
 - Dentro de la sesión que se abrió con el comando FTP se utilizo la instrucción “bin” para indicar que se trasladaran archivos binarios.
 - Con la instrucción “mput *” se inicio el traslado de todos los archivos binarios hacia la carpeta donde nos ubicamos anteriormente.
 - Después de haber llevado a cabo todo el traslado de archivos completamente hacia la nueva carpeta se procede a salir de la sesión abierta de FTP con la instrucción “quit”.
4. Restaurar la información que está en archivos binarios utilizando la herramienta dbimport. Primero nos ubicamos en la carpeta donde esta almacenados los archivos binarios trasladados por medio del comando FTP en el servidor nuevo, luego utilizando la instrucción dbimport se llevo a cabo la carga total de los archivos binarios que contienen toda la información de la base de datos, por procedimiento la instrucción dbimport correrá primero el archivo de secuencia de instrucciones de SQL de las tablas seleccionadas en la nueva versión de Informix. Este archivo restaurara toda la estructura de la base de datos en el servidor nuevo de Informix. En resumen la instrucción dbimport subirá toda la información almacenada en los archivos binarios hacia la nueva base de datos creada por el archivo de secuencia de instrucciones de SQL en el servidor nuevo.
5. Asignación de permisos a las tablas para el usuario de prueba. Esta asignación se hizo de forma manual utilizando instrucciones de SQL en cada tabla creada en la nueva base de datos.

6. Configuración de los archivos de control Hosts y Services, en la maquina del cliente. Una vez terminada la configuración de la nueva base de datos de prueba procedemos a configurar una maquina cliente de prueba para direccionar los nuevos servicios de la nueva versión de la base de datos para que el sistema pueda tener acceso a esta, para esto el primer paso es configurar los siguientes archivos:

- Archivo Hosts: en la dirección C:\WINDOWS\SYSTEM32\DRIVERS\ETC\hosts se agrego la siguiente línea: 192.9.200.3 dblnx esta instrucción especifica la dirección IP de el nuevo servidor de Informix llamado dblnx.
- Archivo Services: en la misma ubicación del archivo hosts también se encuentra el archivo Services, en este archivo se introdujo esta nueva instrucción Sqlturbo 1540/tcp especifica el nombre del servicio sqlturbo así como también el numero que lo identifica.

7. Configurar el driver Informix-Cli direccionándolo al nuevo servidor de Informix. Este driver Informix-Cli lo utiliza Informix como un configurador de la base de datos que el cliente utilizara para conectarse al sistema, actualmente los usuarios ya tenían ese driver instalado en su máquina, por lo que la configuración de este driver únicamente constituyo modificar el nombre del nuevo servidor así como también el nombre de la nueva base de datos, otra instrucción que se modifiko fue el protocolo de acceso a esta nueva versión debido a que la versión antigua de Informix era una versión Standar Engine y su protocolo es “sesoctp”, que significa standar engine socket tcp y esta nueva versión de Informix es una versión Online Engine por lo que su protocolo de comunicación es olsoctp que significa on line socket tcp.

8. Correr el modulo de mayor contable para revisión del funcionamiento en nueva versión de Informix. Las pruebas que se hicieron al correr este modulo fueron probar las funciones de mayor uso e importancia en el sistema de la institución por lo que se probaron, las siguientes opciones del sistema:

- Funcionalidades básicas: prueba de mantenimientos, esto implico inserciones, actualizaciones y eliminaciones de la información de prueba corroborando que no existieran errores o cambios que implicaran un mayor esfuerzo.
- Reportes: la gran mayoría de reportes están hechos con la herramienta Cristal Reports esto implica que el acceso a los datos debería de ser transparente en estas aplicaciones, aquí si se tuvo que volver a configurar la forma de reconexión a la base de datos igual que el driver Informix-Cli, estos reportes tienen su acceso a la base de datos donde se especifica el nombre del servidor y el nombre de la base de datos.

1.3. Analizar las pruebas con estos subconjuntos de tablas en aplicaciones con la nueva versión de Informix

Al correr el modulo Mayor Contable se llevo a cabo varias pruebas como las descritas anteriormente. En el último paso de las pruebas se analizaron accesos a la base de datos tanto en mantenimientos hechos en Visual Basic 6 como en accesos a datos por reportes hechos en Cristal Reports.

1.3.1 Ventajas:

- Acceso rápido a la información de la base de datos. Se observo el acceso a la información de la base de datos el cual fue lo suficientemente más rápido que en la versión antigua de Informix aunque se trataba de una prueba, no existían todas las tablas en el sistema pero el volumen de datos en las tablas que se utilizaron en las pruebas si coincidía con el numero de datos de las tablas de la versión origen dando como resultado una observación más real del rendimiento observado.
- Herramienta para acceso a la base de datos (DBAccess). La versión nueva Informix, tiene una mejor presentación mucho más cómoda para el usuario permitiéndole tener acceso a los datos de una forma ordenada y sencilla, desempeñándose de una forma más óptima que su versión anterior.
- La nueva versión tiene más funciones de SQL que la versión anterior, se tiene la disponibilidad de muchas y nuevas funciones y procedimientos para poder manipular mejor los datos.
- La forma de trabajo de esta base de datos, nos permite tener acceso real y actual a la información de una forma rápida y consistente no importando la cantidad de usuarios que estén manipulando la información desde el sistema.

- Esta nueva versión de Informix nos permitirá poder tener acceso a actualizaciones, del mismo sistema de base de datos y compatibilidades con nuevos sistemas tales como, Visual Basic .Net, otros sistemas operativos como Linux, nuevas versiones de Windows, programas visuales para manipular la base de datos que la versión antigua no lo permitía o no los soportaba.

1.3.2 Desventajas:

- Tiempo que se llevara el modificar los reportes y cuantos subreportes tengan estos con respecto a la nueva forma de conexión a la base de datos.

2. ANÁLISIS DE MIGRACIÓN DE DATOS DE INFORMIX A SQL SERVER

2.1 Análisis de pruebas de migración de datos de Informix a SQL Server

El objetivo de realizar esta prueba con SQL Server 2005, se hizo para contemplar la posibilidad de cambiar de base de datos, y aprovechar muchas ventajas que SQL Server tiene sobre la versión antigua de Informix. Esta prueba incluye la migración completa de la base de datos “FHA” y también del servidor en funcionamiento, así como el reforzamiento del sistema de seguridad utilizado por dicha entidad que se especializa sobre el fomento de hipotecas en bienes inmuebles.

Para realizar esta prueba se utilizaron las tablas que se emplean para el manejo del Modulo conocido como Mayor Contable, el cual como se menciono anteriormente tiene aproximadamente 20 tablas, la principal tabla quizás podría ser por su utilización e importancia la tabla “caso”. Esta tabla contiene toda la información de los casos del FHA, el concepto de caso está relacionado con un cliente y una propiedad. La estructura de esta tabla se muestra de forma resumidamente, por razones de seguridad, ver figura 2.

Figura 2. Estructura de la tabla “caso”

```
CREATE TABLE informix.caso(  
  numero_caso          CHAR(7),  
  cod_proyecto         SMALLINT,  
  corr_tipo_viv        SMALLINT,  
  cod_ent_aprobada     SMALLINT,  
  cod_promotor         SMALLINT,  
  cod_constructor      SMALLINT,  
  cod_acreedor_hipot  SMALLINT,  
  tipo_caso            SMALLINT,  
  resguardo_solic     DECIMAL(7, 0),  
  resguardo_recom     DECIMAL(7, 0),  
  resguardo_aprob     DECIMAL(7, 0),  
  valor_fha           DECIMAL(7, 0),  
  resguardo_recom_an  DECIMAL(7, 0),  
  gastos_cierre       DECIMAL(7, 0),  
  cod_tipo_ope        SMALLINT,  
  tipo_cuota          SMALLINT,  
  plazo_meses         SMALLINT,  
  tipo_tasa           SMALLINT,  
  porc_tasa           DECIMAL(5, 2),  
  porc_prima_fha      DECIMAL(5, 2),  
  porc_admon          DECIMAL(5, 2),  
  precio_venta        DECIMAL(9, 2),  
  inversion_inic      DECIMAL(9, 2),  
  prima_fha_ano1     DECIMAL(9, 2),  
  cuota_nivelada     DECIMAL(9, 2),  
  costo_produc       DECIMAL(9, 2),  
  direc_inmueble     CHAR(80),  
  finca              CHAR(8),  
  folio              CHAR(8),  
  libro              CHAR(6),  
  area_terreno       DECIMAL(11, 5),  
  fecha_ing_caso     DATE,  
  fecha_emis_resg    DATE,  
  hora_emis_resg     CHAR(8),  
  fecha_inic_vresg   DATE,  
  fecha_fin_vresg    DATE,  
  fecha_prorr_resg   DATE,  
  aprobado_promo    SMALLINT,  
  usu_aprobo_promo  CHAR(10),
```

fecha_aprobo_promo	DATE,
hora_aprobo_promo	CHAR(8),
...	
...	
...	
)	

Otra de las tablas con mayor uso que también fue elegida para esta prueba es la tabla “bita_datos_caso”, su estructura es similar a la tabla caso, debido a que su propósito es tener la función de bitácora de transacciones sobre la tabla caso. Otra de las tablas es la “tipo_vivienda”, que tiene toda la información requerida para contemplar una nueva propiedad en la entidad. Y así se fueron escogiendo una serie de tablas que se relacionan con estas tablas principales, entre ellas están la new_inspecciones, solicitud_insp, proyectos, enc_cta_cte, caso_x_casa.

La prueba consiste en realizar manualmente la estructura de estas tablas e importar mediante un script un porcentaje pequeño de data hacia SQL Server.

Uno de los objetivos que se esperaba con esta prueba era que se demostrara que el cambio de versión de base de datos fuera totalmente transparente para los usuarios, ya que otro objetivo es que sigan usando la misma interfaz de siempre, pero con mejores resultados en cuanto a rendimiento y rapidez en acceso a datos. Lo que si cambio fue la última capa de proceso, vale decir, que las aplicaciones deberían de ser modificadas para que ya no recojan la información de una base de datos Informix, sino que ahora de una base de datos SQL Server 2005.

Cuando se hicieron los cambios pertinentes en el código de prueba de una de las aplicaciones, se tuvo que modificar en su totalidad la forma de conexión a la base de datos SQL Server 2005, además de eso el programa tiene vínculos con procedimientos almacenados que tuvieron que ser también migrados y adaptados a SQL Server 2005.

Como resultado de un trabajo difícil y largo únicamente con este pequeño conjunto de tablas, si se noto el aumento en la velocidad de las transacciones alcanzado con la nueva plataforma. Sin embargo el costo y tiempo que se llevaría la migración completa al 100% de todo el sistema es demasiado grande debido a la magnitud de datos y el presupuesto limitado de la institución.

2.2 Identificación de pros y contras con SQL Server

Aprovechando que se iba a realizar esta migración de datos se pensó en migrar hacia la versión 2005 de SQL Server. Se llevo a cabo el mismo proceso de prueba que anteriormente se realizo con la nueva base de datos de Informix, se crearon tanto la base de datos como el subconjunto de tablas que se escogieron como parte principal en los procesos de trabajo, sin embargo debido a que la estructura de las tablas es bastante compleja por el tamaño de estas, el archivo de secuencia de SQL fue generado por parte de Informix incrustando en tal script estructuras y sentencias que son propias por el mismo Informix y que SQL Server 2005 no pudo interpretar debido a que no conformaban parte de un SQL Estándar. Pero aun así aunque fue arreglado el script para que corriera en SQL Server 2005 surgieron inconvenientes más graves.

Uno de los problemas que surgieron mayormente fue la conexión que se tiene en el código fuente de las aplicaciones del sistema del FHA que en su mayoría están diseñadas en Visual Basic 6.0. Existe mucho código que debe de cambiar para poder conectarse a SQL Server, además la forma de conexión que se tiene en estas aplicaciones requerían de reestructurar totalmente esta parte, junto a esto hay funciones propias de Informix que están inmersas en el código fuente y habría que cambiarlas y sustituirlas por las equivalentes en SQL Server, otro problema se origino en los stored procedures (procedimientos almacenados), vistas y triggers que se tienen programados en Informix, estas estructuras también deberían de cambiar casi en su totalidad pues existen consultas que tienen funciones propias de Informix y que en SQL Server 2005 habría también que buscar su equivalente para tales funciones y probar su funcionamiento y comparar si devuelven el mismo resultado de información.

Por otra parte la herramienta con la cual fueron hechos la inmensa cantidad de reportes existentes en el FHA es Cristal Reports, este tiene incrustado en cada reporte y subreporte, el nombre del servidor al cual se está conectando para obtener los datos. Esto hubiera implicado modificar la totalidad de todos los reportes cambiando el nombre del servidor de origen de datos y examinando cuidadosamente la existencia de subreportes en cada reporte, debido a que también en esta parte se hubiera tenido que modificar el nuevo nombre del servidor de origen de datos.

Como resultado debido a la magnitud de estructuras como procedimientos, vistas, triggers, tablas y código fuente se llevo a la conclusión de que el impacto de migrar de la versión antigua de Informix hacia la nueva base de datos de SQL Server 2005 es sumamente grande a nivel del sistema y habría que reprogramar totalmente cada estructura en el sistema y probar su funcionamiento para poder confiar totalmente de la migración de datos.

Dicho proceso lleva consigo un costo y tiempo que en el FHA no se había tomado en consideración, ya que el proceso de migración de datos se planteo de tal forma que se considerara y se pudiera llevar a cabo de una forma transparente y no de una forma que impactara radicalmente todo el proceso de trabajo que se ha venido realizando durante años atrás, por lo que llevar a cabo esta prueba de migración de datos con SQL Server sirvió para tener conocimiento del esfuerzo tan grande que se llevaría al cambiar de base de datos.

Todos estos cambios son complejos y costosos, y no se tienen contemplados dentro del plan estratégico para la migración de datos hacia SQL Server 2005.

3. EXPORTACIÓN DE LA BASE DE DATOS DE INFORMIX HACIA ARCHIVOS UNL DE UN SERVIDOR A OTRO

3.1 ¿Cómo realizar la exportación de data hacia archivos físicos Hexadecimales? (archivos UNL de Informix)

Luego de haber verificado la integridad de las tablas y data en la base de datos Informix se procede a la exportación de la base de datos hacia un directorio físico en el servidor actual llamado “new_fha”. Para llevar a cabo este procedimiento lo primero que se tiene que hacer es lo siguiente:

- Hacer un login al servidor de Informix con permisos de DBA.
- Crear un directorio donde se almacenaran los archivos UNL que contienen la estructura y data de la base de datos.
- Luego se utilizara el comando dbexport para iniciar la exportación de la data hacia el directorio que acabamos de crear.

Durante la exportación de la base de datos ésta es configurada a modo exclusivo para evitar cualquier actualización de datos que pueda llevar a corromper la integridad de la información en la base de datos. ¿Cómo se realizan todos estos procesos? Para empezar se eligió un lugar en el servidor donde se tuviera fácil acceso y disponibilidad en espacio para almacenar los archivos UNL que contienen la base de datos, nos posicionamos hacia ese directorio y luego de esto se creó un subdirectorio con la instrucción mkdir seguido del nombre, de esta forma:

Mkdir //new_fha/volu1/migracion

Ahora que ya se ha creado el directorio procedemos a la exportación de la base de datos por medio del comando dbexport. Existen varios métodos para realizar la exportación e importación de data, el método que uno escoja depende de la cantidad de data que se planea mover. Todos estos métodos entregan un desempeño similar y tienen la disponibilidad para modificar el esquema de la base de datos. El método que se utilizara será por medio de las utilidades dbexport y dbimport, porque el propósito es mover completamente toda la base de datos.

El dbexport nos permite bajar información hacia diferentes medios, ya sea por ejemplo hacia el disco duro o hacia una cinta magnética, o cualquier otro dispositivo de almacenamiento masivo.

La forma de bajar esta información se puede hacer en su totalidad hacia el dispositivo de nuestra elección, que en este caso lo haremos hacia el disco duro.

De la misma manera se utilizara el comando dbimport para importar el esquema y la data de la base de datos hacia la nueva versión de Informix, dicho proceso se explicara en capítulos siguientes.

Los parámetros que utilizaremos con el comando dbexport son los siguientes:³

- -c, esta opción nos indica si el dbexport tiene algún error en exportar la data hacia el dispositivo que configuramos. Dichos errores pueden ser la no disponibilidad del disco duro o también algún error físico al escribir sobre el disco duro. También puede darnos algún error si configuramos un parámetro inválido y por ultimo uno de los errores más comunes es que el dbexport no puede abrir la base de datos o no existen permisos en el sistema.
- -q, suprime el despliegue de los mensajes de error, advertencias y genera una secuencia de definición de datos de SQL.
- -X, esta opción se utilizara para reconocer los archivos binarios hexadecimales que contienen la data.
- Database, aquí especificamos el nombre de la base de datos que deseamos exportar.
- -o, este parámetro especifica el directorio sobre el disco en el cual dbexport creara los archivos de datos y el archivo de esquema de la base de datos, así como también el directorio que se crea por default “database.exp”.

Como en este caso que se escribirá al disco duro, el dbexport creara un subdirectorio, database.exp en el directorio que se especifico con el parámetro -o. En la línea de comando escribimos la siguiente instrucción:

dbexport -c -q fha -o /volu1/migracion

³ **IBM CORP.** “IBM Informix Migration Guide versión 9.4”, Segunda Edición, Agosto de 2004, Capítulo The DBEXPORT and DBIMPORT Utilities, página 8-3

La utilidad dbexport creara un archivo con extensión .unl por cada tabla en la base de datos FHA. El archivo que contiene el esquema de la base de datos tendrá el nombre de esta con extensión .sql es decir “fha.sql”, entonces los archivos .unl y .sql quedaran creados en el directorio database.exp. Además se utilizo el parámetro -q para poder evitar la creación del archivo dbexport.out, dicho archivo contiene todos los mensajes desplegados durante el proceso de exportación de la base de datos con el comando dbexport, mensajes como: de error, advertencias y despliegue de los comandos de SQL para la definición de la data que dbexport está generando, esto hace más lento el proceso y por la cantidad de tablas y mensajes se opto por no generarlo ya que quedaría únicamente como una bitácora de mensajes, de cualquier forma los mensajes se están desplegando en pantalla en tiempo de ejecución del comando dbexport, por lo que si existiera algún error o si deseáramos detener el proceso en cualquier momento podemos cancelar el proceso presionando la tecla esc.

Luego de haber ejecutado la instrucción anterior, podemos ir a verificar la existencia de los archivos .unl y .sql. El contenido de este último es el esquema de la base de datos que contiene todas las instrucciones SQL que son necesarias para reconstruir la base de datos exportada.

Este archivo puede ser editado para modificar el esquema de la base de datos. Este proceso se explicará más adelante.

⁴ El archivo de esquema contiene sentencias de SQL que son necesarias para recrear la base de datos exportada, este archivo se puede modificar para cambiar alguna sentencia del esquema de la base de datos. Por precaución al editar el archivo que contiene el esquema de la base de datos se debe de considerar no eliminar ni añadir ningún comentario nuevo debido a que cuando estemos importando o subiendo de nuevo la base de datos hacia el nuevo servidor con el comando dbimport no se produzca un error que aborte la operación o peor aun que se produzcan resultados impredecibles.

Dentro de las sentencias de SQL que contiene el archivo de esquema de la base de datos se encuentran: creación de tablas (create tables), vistas (views), índices (indexes), particionamiento de las tablas fragmentadas y sus índices, roles y privilegios (grant privileges) además de los nombres de los usuarios que dieron origen a la base de datos y a cada objeto dentro de ella.

También el archivo de esquema que crea el dbexport contiene comentarios, llaves que encierran información acerca de archivos que fueron exportados. La información que es catalogada como comentario en el archivo de esquema es también importante para la reconstrucción de la base de datos.

De esta forma los propietarios originales retienen sus privilegios de DBA en la base de datos así como los propietarios de todas las tablas, índices y vistas. Por lo que la persona que ejecuta el comando dbimport también tiene privilegios de DBA en la base de datos.

⁴ **IBM CORP.** “IBM Informix Migration Guide versión 9.4”, Segunda Edición, Agosto de 2004, Capítulo The DBEXPORT and DBIMPORT Utilities, página 8-6

3.2 Utilización del protocolo FTP para el traslado de archivos UNL del servidor antiguo hacia el nuevo de Informix

En este momento tenemos toda la información en el directorio volu1/migracion del servidor new_fha, el siguiente paso sería trasladar todos estos archivos de este servidor hacia el servidor nuevo cuyo nombre es dblnx. Para trasladar la información de un servidor a otro utilizaremos el comando FTP de la siguiente forma:

1. Nos ubicamos en el servidor origen y abrimos una consola de comandos, desde allí vamos a conectarnos vía FTP hacia el servidor destino que es el servidor nuevo. Tecleamos entonces la siguiente instrucción:

ftp dblnx

2. Donde ftp es el nombre de la instrucción y “dblnx” es el nombre del servidor al que nos deseamos conectar, este nombre viene de “db” que significa database o base de datos y lnx de Linux que es el sistema operativo, es decir hace referencia a la base de datos que se encuentra en el servidor Linux. El comando anterior nos conecta con el nuevo servidor, aquí se requiere que nos firmemos para poder tener acceso. Nos pide que ingresamos el usuario y la contraseña con privilegios de administrador de la nueva Base de Datos.
3. Una vez conectados en el servidor nuevo, creamos un directo temporal para copiar allí todos los archivos de la base de datos. Creamos el directorio de esta forma:

mkdir fha.exp

4. Ingresamos al directorio creado de la siguiente forma:

cd fha.exp

prompt

5. con el comando interno cd ingresamos al directorio fha.exp seguidamente la instrucción “prompt” la usamos para que nos muestra en pantalla que estamos dentro del directorio correcto.
6. Para hacer el traslado de archivos se debe de indicar al protocolo FTP qué tipo de archivos se trasladaran, en nuestro caso los archivos creados por la instrucción dbexport son de tipo binario, con la siguiente sentencia indicamos esto:

bin

7. Después de haber configurado el tipo de archivo que vamos a trasladar, procedemos a copiar todos los archivos, esto de la siguiente forma:

mput *

8. Utilizando el comando “mput *” procedemos al traslado de todos los archivos de un servidor a otro.

9. Nos desconectamos del comando ftp utilizando la instrucción siguiente:

quit

10. Utilizando la herramienta telnet en el nuevo servidor, abrimos una sesión y verificamos que hayan sido trasladados todos los archivos.

4. CONFIGURACIÓN DE LA BASE DE DATOS DE INFORMIX SOBRE ARCHIVOS BINARIOS EN EL NUEVO SERVIDOR DE INFORMIX

4.1 Levantar la base de datos de Informix sobre archivos binarios

En este paso se describirá como se llevo a cabo el procedimiento de subir o restaurar la base de datos FHA en el nuevo servidor de Informix “dblnx”. Para poder realizar este paso primero debemos de recordar que hasta este punto nuestra información o backup de la base de datos esta en archivos binarios, en la carpeta fha.exp.

Lo que se hizo primero fue corregir el archivo de secuencia de instrucciones de SQL que creó el comando dbexport, esta corrección se llevo a cabo porque en la base de datos original existen usuarios que ya no están activos y se encuentran configurados como propietarios de algunas tablas, vistas y procedimientos.

Nota: para poder corregir todas las instrucciones de SQL en el archivo de secuencias se abrió el archivo en un editor de texto como Wordpad de Windows para facilitar las búsquedas y reemplazar fácilmente secuencias de instrucciones de SQL, es de mencionar que puede usar cualquier otro tipo de editor de texto que pueda ayudar en este tipo de búsquedas y reemplazo de caracteres.

Una vez abierto el archivo de secuencias de instrucciones de SQL se buscarán y reemplazarán las siguientes opciones:

4.1.1 Nombre del propietario de cada tabla

En el archivo de secuencia de instrucciones está especificado en cada tabla la instrucción GRANT ALL ON seguido del nombre del propietario de la tabla junto con el nombre de la misma de esta forma: 'nombre_propietario'.nombre_tabla. Lo que se pretende aquí es cambiar ese nombre de propietario a un nuevo nombre, por lo que se utilizó el editor para que busque y reemplace los nombres de usuarios que tienen como propiedad algunas tablas, estos usuarios fueron identificados por el listado de usuarios que está en el archivo de secuencia.

Por ejemplo, se buscó el usuario "juan" y se reemplazó por el usuario "Informix". Prácticamente solo se dejó este usuario como único propietario de todos los objetos que pertenecen a la nueva base de datos, por cuestión de seguridad y mejor control, además de poder evitar cualquier conflicto por cambios en actualizaciones futuras de datos en la aplicación del sistema agilizando de esta forma este proceso.

4.1.2 Propiedad Extent Size

Esta propiedad no está dentro del esquema de la base de datos por lo que fue añadida de algunas tablas con mayor peso de datos. En el siguiente subcapítulo se detallará más detenidamente esta instrucción así como también se definirá el listado de tablas que fueron seleccionadas por parte del Administrador del Sistema para su utilización.

4.1.3 Propiedad Revoke All

En el archivo de secuencia de instrucciones está definida esta instrucción. La se elimino debido a que su propósito es eliminar todo privilegio sobre cada tabla. Debido a que todas las tablas tienen esta instrucción. Para poder realizar esto, se abrió una sesión de telnet hacia el servidor nuevo y utilizando el editor de Informix se usó la siguiente instrucción:

```
cat fha.sql | grep -e "revoke all" -v | more
```

Con esta instrucción se hizo que el editor de Informix eliminará toda instrucción "revoke all" del archivo de secuencia de instrucciones de SQL, para evitar que las tablas no quedaran sin ningún privilegio de acceso por parte de sus propietarios. (La edición de este archivo se pudo haber llevado a cabo también por cualquier otro editor de texto)

4.2 ¿Cómo configurar manualmente el tamaño de paginación de tablas en la nueva versión de Informix?

El tamaño de paginación de las tablas es un concepto que utiliza Informix para asignar un espacio en disco a las tablas, esto para evitar de alguna forma la fragmentación de la misma, es necesario definir primero algunos conceptos.

Definiremos el concepto de un EXTENT (rango) en Informix y para qué se utiliza.⁵ Cundo se crea una tabla en Informix, el servidor de la base datos asigna una cantidad fija de espacio que ocupara la información a ser almacenada en la tabla. Cuando este espacio es ocupado completamente el servidor de la base de datos debe de asignar espacio adicional para poder seguir almacenando la información. La unidad física de almacenamiento que el servidor de la base datos usa para asignar ambos espacios “rango de inicio” y “rango siguiente” es llamada un “EXTENT”. En resumen un extent size (rango de tamaño) consiste en una colección contigua de páginas que almacenan información para una tabla dada.

Cada tabla permanente en la base de datos tiene dos Extent Sizes asociados dentro de ella. El tamaño initial-extent es el numero de kilobytes (bytes es la unidad de medida en kilos para almacenar información utilizado por el sistema de archivos) asignados a la tabla cuando esta se crea por primera vez. El tamaño next-extent es el numero de kilobytes asignados a la tabla cuando el initial-extent (y cualquier subsiguiente extent) se llena.

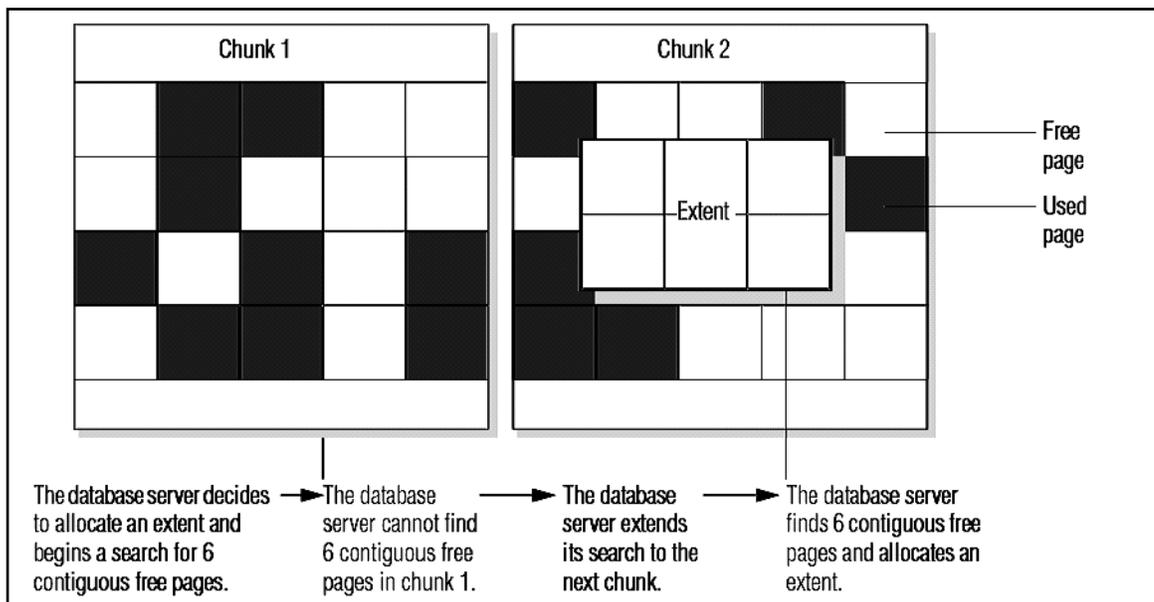
Algunas recomendaciones para la utilización del extent size son las siguientes:

- Un extent esta siempre contenido dentro de un pedazo de espacio determinado (en inglés se define como “*chunk*”), por lo que un extent no puede cruzar los límites de este pedazo de espacio.

⁵ **IBM CORP.** “IBM Informix Dinamic Server Administrators Guide”, Versión 9.4, Capítulo 9-14 Extents, páginas 328 y 329.

- Si el servidor de base de datos no puede encontrar un pedazo contiguo de espacio en el disco, el cual fue especificado por el next-extent size, este buscara en el siguiente pedazo de espacio dentro del dbspace para tener más espacios contiguos.

Figura 3. Proceso de Asignación del EXTENT



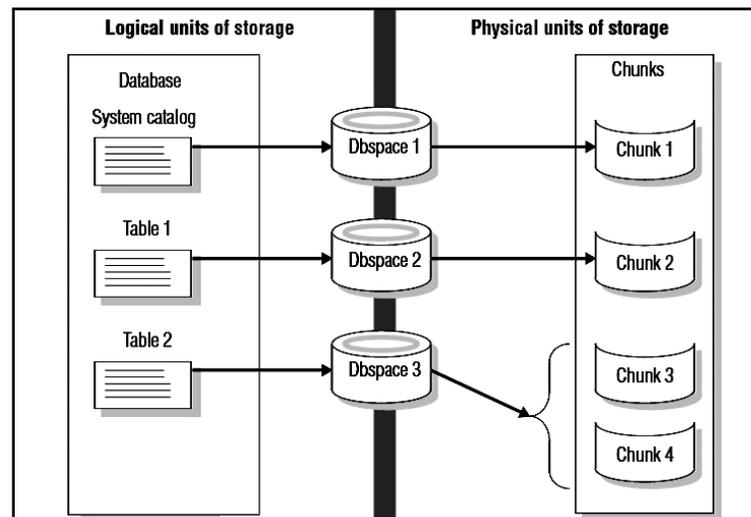
4.2.1 Explicación de la gráfica “Proceso de Asignación del EXTENT”:

Aquí vemos dos pedazos de espacio, como podemos observar cada uno tiene un número determinado de páginas similares, el proceso de asignación de un extent es el siguiente:

1. El servidor de la base de datos decide asignar un extent e inicia en la búsqueda de 6 espacios contiguos libres,
2. El servidor de la base de datos no encuentra 6 páginas contiguas libres en el pedazo de espacio 1,
3. El servidor de la base de datos, extiende la búsqueda en el siguiente pedazo de espacio,
4. El servidor de la base de datos encuentra 6 páginas contiguas libres y asigna un extent.

Otro concepto muy importante que debemos de tomar en cuenta es ⁶ el dbspace, este es una unidad lógica que puede contener entre 1 y 32,767 pedazos de espacio. En este lugar se almacenan las bases de datos, tablas, archivos log “lógicos” y físicos. Como lo muestra siguiente grafica.

Figura 4. Los Dbspaces enlazan las unidades de almacenamiento lógicos y físicos



⁶ **IBM CORP.** “IBM Informix Dinamic Server Administrators Guide”, Versión 9.4, Capítulo 9-16 y 9-17 “DBSPACE”, página 330 y 331.

El servidor de base de datos utiliza el dbspace para almacenar las bases de datos y tablas.

Ya que tenemos una idea más clara de los conceptos extent size, pedazos de espacios y el dbspace, procedemos a definir el tamaño de paginación en las tablas seleccionadas por el administrador de la base de datos “fha”. Para entender cómo es que vamos a realizar este paso procedemos a definir un ejemplo:

⁷ Si nosotros tenemos en el sistema una página de 2 kilobytes, su longitud mínima es de 8 bytes. La máxima longitud es igual al siguiente pedazo de espacio (*chunk*) En el siguiente ejemplo se especifica un extent size de 32 kilobytes.

alter table customer modify next size 32

Cuando se usa esta instrucción, el tamaño existente del extent no cambia. No se puede cambiar el tamaño existente del extent sin haber bajado toda la información del sistema.

Después de haber explicado este ejemplo, procedemos entonces a definir la sentencia de instrucciones que utilizamos para poder definir las propiedades del extent size en las tablas seleccionadas.

⁷ **IBM CORP.** “IBM Informix Guide to SQL: Sintaxis”, versión 9.4, Marzo de 2004, Capítulo “Modify next size clause 2-76”, página 120.

La siguiente instrucción fue ingresada en el archivo de secuencia de instrucciones SQL que el comando dbexport creo en el momento de bajar la información en archivos binarios, dicho archivo se llama fha.sql.

Al editar este archivo utilizando un editor de texto, se busco la definición de las tablas seleccionadas a las que se les configurará la propiedad de extent size, por ejemplo, se busco donde estaba definida la tabla caso y luego de esta definición se inserto la propiedad extent size junto con su información predefinida y calculada por el Administrador de la Base de Datos.

Para dar un ejemplo de que fue lo que se busco e inserto en el archivo de secuencia de instrucción de SQL vea la figura 5:

Figura 5. Configuración de la propiedad Extet Size

```
CREATE TABLE informix.caso(  
  numero_caso      CHAR(7),  
  cod_proyecto     SMALLINT,  
  corr_tipo_viv    SMALLINT,  
  cod_ent_aprobada SMALLINT,  
  cod_promotor     SMALLINT,  
  cod_constructor  SMALLINT,  
  ...,  
  ...)  
extent size 22,346 next extent 4,000;
```

Como vemos en este ejemplo la sentencia create table define la creación de la tabla informix.caso, podemos observar que después del paréntesis que finaliza esta instrucción se introdujo la propiedad extent size, y para su entendimiento la explicaremos por partes, de esta forma:

4.2.2 extent size 22,346

Aquí se define el initial-extent, definido anteriormente como el extent size inicial de la tabla, el cual tiene un valor estimado de 22,346 kbytes, luego la segunda parte se define así:

4.2.3 next extent 4,000;

Como podemos observar la instrucción es clara y lo que se está definiendo aquí es el siguiente extent size con un valor estimado de 4,000 kbytes. De esta forma se debe de alterar todas las tablas seleccionadas previamente a las cuales se les configurara esta propiedad. Al alterar el archivo de secuencia de instrucciones SQL se tuvo la precisión de cuidar que todos los cambios hechos estuvieran libre de errores debido a que si alguna coma o un punto y coma que no se introdujera bien el archivo de secuencia quedaría prácticamente inutilizable puesto que nos daría errores en el momento de ejecutarlo.

A continuación se muestra el listado de tablas que fueron seleccionadas por el tamaño y crecimiento que se ha estimado según la experiencia del DBA del “FHA”.

Figura 6. Listado de tablas seleccionadas para configuración del Extent Size en kilobytes.

1	2	3	4	5	6	7	8	9	10	11
tabname	rowsize	ncols	nindexes	Nrows	created	Crecimiento Esperado (Regs)	Extent Paginas	Next Extent Paginas	Extent KB	Next Extent KB
bita_datos_caso	338	51	1	551783	07/07/1999	115,000	110,357	23,000	220,714	46,000
det_cta_cte	83	18	4	438474	02/02/2004	90,000	18,270	3,750	36,540	7,500
control_casos	56	10	2	225428	01/03/2004	0	6,262	8	12,524	16
md_hist_cred	77	13	2	204751	03/07/2002	90,000	7,876	3,462	15,752	6,924
mdr_hist_cred	65	10	2	193151	19/01/2004	0	6,231	8	12,462	16
conc_mayor_hist	111	8	3	183912	01/03/2004	75,000	10,218	4,167	20,436	8,334
pago_cap_e_interes	68	14	3	175017	04/09/2005	150,000	6,036	5,173	12,072	10,346
new_ctrl_casos	56	10	1	123987	28/11/2001	50,000	3,445	1,389	6,890	2,778
det_cedula_hipot	81	17	1	99534	06/09/1999	20,000	4,148	834	8,296	1,668
det_mayor_hist	70	11	2	84813	01/03/2004	33,000	3,030	1,179	6,060	2,358
conc_ht_ing_p	115	4	2	79392	01/03/2004	4,500	4,671	265	9,342	530
saldo_capital_seg	37	8	1	78025	31/12/2004	78,025	1,445	1,445	2,890	2,890
deudores_caso	347	25	5	77574	22/08/2001	14,000	15,515	2,800	31,030	5,600
conc_notas_suspen	1013	4	2	52681	25/02/2002	2,000	52,681	2,000	105,362	4,000
systabauth	28	4	2	45613	01/03/2004				0	0
caso	428	68	9	44691	01/10/2003	8,000	11,173	2,000	22,346	4,000
obs_cta_cte_g	361	3	1	41208	01/03/2004	9,000	8,242	2,000	16,484	4,000
tmp_cta_cte	99	13	0	37730	21/07/2004	887	1,887	50	3,774	100
casas	356	20	4	37555	23/11/2004	9,000	7,511	2,000	15,022	4,000
new_inspecciones	385	32	2	37523	23/04/2003	6,000	7,505	1,200	15,010	2,400
enc_recibo_caja	287	17	3	34289	22/08/2001	6,000	4,899	900	9,798	1,800
enc_cta_cte	88	17	2	33984	01/03/2004	8,000	1,545	500	3,090	1,000
solicitud_insp	429	23	4	33643	13/08/2003	18,000	8,411	4,500	16,822	9,000
sus_det_notas	765	6	2	22078	13/02/2003	8,000	11,039	4,000	22,078	8,000
det_cxc_proyecto	390	21	3	21112	08/08/2003	11,000	4,223	2,200	8,446	4,400

En la tabla anterior se definieron las siguientes características, que se describen a continuación:

1. tabname: es el nombre de la tabla seleccionada para configuración del extent size.
2. rowsize: muestra el tamaño total de las columnas de la tabla.
3. ncols: especifica el número de columnas que tiene la tabla.
4. nindexes: especifica la cantidad de índices asociados a la tabla.
5. nrows: especifica el número de filas que tiene la tabla.
6. created: especifica la fecha en que fue creada la tabla.
7. crecimiento esperado: muestra una estimación del crecimiento calculado en número de registros de la tabla.
8. extent (páginas): muestra el tamaño actual del inicial-extent size de la tabla antes de haberla bajado a archivos binarios, esta propiedad estaba definida en páginas.
9. next extent (páginas): muestra el tamaño actual del next-extent size de la tabla antes de haberla bajado a archivos binarios, esta propiedad estaba definida en páginas.
10. extent (Kb): muestra el tamaño estimado del initial-extent size de la tabla, esta propiedad se definirá en kilobytes.
11. next extent (Kb): muestra el tamaño estimado del next-extent size de la tabla, esta propiedad se definirá en kilobytes.

4.3 Importación de data al nuevo servidor de Informix

Luego de haber terminado con la modificación del archivo de secuencia de instrucciones SQL procedemos a importar la base de datos hacia el servidor nuevo de Informix llamado “dblnx”. Para completar este procedimiento debemos empezar por lo siguiente:

- Hacer un login al servidor de Informix con permisos de DBA.
- Ingresar al directorio donde están almacenados los archivos binarios que exportamos por medio de ftp.
- Luego se utilizara el comando dbimport para iniciar la importación de la base de datos hacia el nuevo servidor.

Primero para empezar abrimos una consola de comandos en el servidor nuevo de Informix y ejecutamos la siguiente instrucción para poder ingresar al directorio donde se encuentran los archivos binarios:

```
cd //dblnx/volu1/migracion
```

Ahora que ya estamos en el directorio procedemos a la importación de la base de datos por medio del comando dbimport. El dbimport nos permite subir información hacia diferentes medios, ya sea por ejemplo hacia el disco duro o hacia una cinta magnética, o cualquier otro dispositivo de almacenamiento masivo. La forma de subir esta información se puede hacer en su totalidad hacia el dispositivo de nuestra elección, que en este caso lo haremos hacia el disco duro. De la misma forma como se utilizó el dbexport así se utilizará el comando dbimport con sus respectivos parámetros.

4.3.1 Parámetros del comando “dbimport”

Los parámetros que utilizaremos con el comando dbimport son los siguientes: ⁸

- -c, le indica al dbimport que complete el proceso de importación aunque este encuentre errores insignificantes pero certeros.
- -q, suprime el despliegue de los mensajes de error, advertencias y genera una secuencia de definición de datos de SQL.
- -V, despega la información de la versión del producto.
- -X, esta opción se utilizara para reconocer los archivos binarios hexadecimales que contienen la data.
- Database, especifica el nombre de la base de datos a crear.

4.3.2 Parámetros Adiciones del comando “dbimport”

- -i directory, especifica la ubicación completa en el disco del directorio database.exp, que contiene los archivos binarios y el archivo de esquema que el dbimport utilizara para crear y cargar la nueva base de datos.
- -d dbspace, especifica el dbspace donde la base de datos será creada. La ubicación por defecto del dbspace es el rootdbs.

⁸ **IBM CORP.** “IBM Informix Migration Guide versión 9.4”, Segunda Edición, Agosto de 2004, Capítulo The DBEXPORT and DBIMPORT Utilities, página 8-8

El usuario que corra el comando dbimport debe de tener el privilegio de DBA en la nueva base de datos creada. Además este proceso cuando inicia bloquea el acceso en cada tabla mientras se carga y la desbloquea cuando la tabla está cargada completamente.

Para cancelar la operación iniciada por el comando dbimport se puede presionar la tecla de cancelar o *Esc* del teclado, en cualquier momento.

Escribimos entonces en la línea de comando la siguiente instrucción:

```
dbimport -c -i / volu1/migracion -d data1dbs fha
```

Esta instrucción realizará el proceso de cargar completamente la nueva base de datos, principalmente su estructura y la información de cada tabla.

A continuación se explicará detenidamente por partes lo que significa toda esta instrucción del dbimport:

4.3.3 Parámetro -c

El parámetro -c lo utilizamos para obviar todos los mensajes de precauciones y errores que durante el proceso el comando dbimport despliega sin mayor importancia pero por la cantidad de información e instrucciones de las tablas importadas estos mensajes se desplegarán incontablemente.

Además al incluir esta opción el dbimport ignorará los siguientes errores:

- Una fila de datos que contenga demasiadas columnas.
- La dificultad de colocarle bloqueo a una tabla.
- La dificultad de quitarle bloqueo a una tabla.

Aun si se está usando la opción `-c`, el dbimport interrumpe el proceso si uno de los errores fatales sucede:

- Incapacidad de abrir el dispositivo especificado.
- Mala escritura en la cinta o el disco.
- Comando de parámetros invalido.
- No puede abrir la base de datos o no se tiene permisos en el sistema.
- No puede convertir la información.

⁹El dbimport crea un archivo de mensajes llamado dbimport.out en el directorio actual. Este archivo contiene cualquier mensaje de error y precaución relacionados con el proceso del dbimport. La misma información también es escrita a una salida estándar a menos que usted especifique la opción `-q`.

4.3.4 Parámetro `-i` /volu1/migracion

⁹ IBM CORP. “IBM Informix Migration Guide versión 9.4”, Segunda Edición, Agosto de 2004, Capítulo The DBEXPORT and DBIMPORT Utilities, páginas 8-9

Esta instrucción especifica la entrada de archivos. El parámetro `-i` le indica al `dbimport` donde se encuentra la información de la base de datos que se desea importar. Algo muy importante es que el directorio debe de ser el mismo directorio que se especifico en la opción `-o` del `dbexport`. El directorio que se especifica es el `“/volu1/migracion”` tanto en el comando `dbexport` como ahora se hace en la instrucción del comando `dbimport`.

4.3.5 Parámetro `-d data1dbs fha`

Como se menciono al inicio de este sub capitulo, el parametro `-d` especifica el database donde la base de datos será creada, que en este caso se llama `data1dbs`. Inmediatamente de esta instrucción sigue el siguiente parámetro que define el nuevo nombre la base datos, que en este caso se llama `“fha”`.

Luego de haber corrida esta instrucción del `dbimport` únicamente nos queda esperar a que la información de la nueva base de datos suba completamente para poder seguir con el proceso de migración de datos.

4.4 Configuración de permisos a usuarios en la nueva versión de Informix

Esta asignación de permisos a los usuarios se hizo debido a que ningún permiso fue exportado en el esquema de secuencias de intrusiones SQL. Ya que el archivo de esquema producía muchos errores y tardanzas al generarse por la magnitud de información. Esto indujo a que también se editara el archivo de esquema en la parte de permisos revocando todos estos en cada tabla.

Luego de haber llevado a cabo con éxito todo el proceso del comando dbimport, se procedió a revisar la información y por supuesto a revisar usuarios en la nueva base de datos. En ese momento no existía ningún usuario más que el de Informix para poder tener acceso a la información, debido a que en el momento que reemplazamos todos los usuarios por el único usuario “Informix” fue por seguridad y mejoras a la aplicación.

Se pensó en asignar a los usuarios por medio de instrucciones de SQL a través del dbaccess, en cada tabla restaurada. El problema es que son demasiadas tablas y realizarlo de esta forma no era factible hacerlo. Por lo que rápidamente se hizo un programa en Visual Studio 6 para que realizara esta tarea. Su función fue obtener el listado de usuarios de la tabla de “usuarios”, el listado de tablas que contiene la base de datos para poder hacer de forma automática la asignación de estas con los usuarios escogidos.

Así pues el programa se encargo de hacer búsquedas e inserciones para la información de los usuarios con los privilegios que nosotros fuimos otorgando a estos usuarios.

4.5 Análisis de pruebas para la verificación y comprobación del número de registros trasladados frente al número de registros en el servidor antiguo

Para poder estar seguros de la consistencia de datos en el momento después de haber subido o restaurado toda la base de datos junto con toda su información se procedió a efectuar un reporte de la siguiente manera, ver la figura 7.

Figura 7. Reporte de comparación de tablas entre el servidor anterior frente al servidor actual

Servidor new_fha					Servidor dblnx				Diferencia
tabname	owner	tabid	nrows		tabname	owner	tabid	nrows	
abono_capital_ced	root	471	11		abono_capital_ced	Informix	304	11	0
abono_normal_parti	root	555	165		abono_normal_parti	Informix	323	165	0
acabados	root	118	1		acabados	Informix	115	1	0
acreedor_hipot	root	1244	1363		acreedor_hipot	Informix	506	1363	0
acreedor_x_proy	root	110	987		acreedor_x_proy	Informix	108	987	0
actividad	root	402	14		actividad	Informix	287	14	0
activo_ant	saulp	562	1617		activo_ant	Informix	327	1617	0
activo_fijo	root	1510	1918		activo_fijo	Informix	529	1918	0
agua	root	156	8		agua	Informix	140	8	0
ambientes	root	1233	26		ambientes	Informix	500	26	0
amortiza_capital	root	906	1512		amortiza_capital	Informix	438	1512	0
amortiza_capital_n	root	1964	3871		amortiza_capital_n	Informix	665	3871	0
anticipo	root	205	0		anticipo	Informix	176	0	0
anticipos_diferido	root	479	987		anticipos_diferido	Informix	308	987	0
anula_resguardos	ricardo	166	1427		anula_resguardos	Informix	148	1427	0
aprobac_desactivad	root	207	431		aprobac_desactivad	Informix	177	431	0
asig_trabajo_tec	root	131	0		asig_trabajo_tec	Informix	123	0	0
asuetos	root	1853	12		asuetos	Informix	618	12	0
avaluos_en_mora	root	668	6		avaluos_en_mora	Informix	349	6	0
backup_recibo	root	1156	17816		backup_recibo	Informix	474	17816	0
bajas_employado	root	378	5		bajas_employado	Informix	275	5	0
banco	root	1864	5		banco	Informix	623	5	0
banos	root	1188	31		banos	Informix	487	31	0
beneficiario	root	249	159		beneficiario	Informix	203	159	0
bit_casas	root	1862	2287		bit_casas	Informix	621	2287	0
bita_asig_employado	root	1994	713		bita_asig_employado	Informix	678	713	0
bita_casos_cancel	root	766	364		bita_casos_cancel	Informix	384	364	0
bita_datos_caso	root	341	552559		bita_datos_caso	Informix	255	552559	0
bita_new_inspec	root	447	0		bita_new_inspec	Informix	296	0	0
bita_proyecto	root	196	432		bita_proyecto	Informix	169	432	0
bita_tipos_viv	root	343	672		bita_tipos_viv	Informix	256	672	0
bita_tiposviv	root	753	555		bita_tiposviv	Informix	376	555	0

bita_vac_nota_carg	root	561	893		bita_vac_nota_carg	Informix	326	893	0
bitacora_caso	root	186	0		bitacora_caso	Informix	161	0	0
bitacora_cta_cte	mynor	825	0		bitacora_cta_cte	Informix	407	0	0
bitacora_cta_cte_n	root	903	26		bitacora_cta_cte_n	Informix	437	26	0
bitacora_pgo_cta	root	1816	1		bitacora_pgo_cta	Informix	597	1	0
bn_nv_emision	root	772	0		bn_nv_emision	Informix	389	0	0
bono_x_caso	root	722	1120		bono_x_caso	Informix	369	1120	0
c_actuarial	root	201	0		c_actuarial	Informix	173	0	0
caja_chica	root	2020	6		caja_chica	Informix	694	6	0
califica	root	191	0		califica	Informix	165	0	0
cambio_deudor	root	133	0		cambio_deudor	Informix	125	0	0
cambio_entida_bono	root	735	230		cambio_entida_bono	Informix	371	230	0
cambio_entidad	root	762	29710		cambio_entidad	Informix	381	29710	0
cambio_monto	root	211	1896		cambio_monto	Informix	179	1896	0
cambio_plazo_meses	root	213	766		cambio_plazo_meses	Informix	181	766	0
cambio_por_admon	root	214	577		cambio_por_admon	Informix	182	577	0
cambio_tasa	root	212	3183		cambio_tasa	Informix	180	3183	0
cargo_miembros_jd	root	385	22		cargo_miembros_jd	Informix	277	22	0
casas	root	1996	38544		casas	Informix	679	38544	0
caso	root	1841	44739		caso	Informix	613	44739	0
caso_tempo	root	253	5260		caso_tempo	Informix	206	5260	0
caso_x_casa	root	2073	7728		caso_x_casa	Informix	722	7728	0
casos_actualizados	root	768	1		casos_actualizados	Informix	386	1	0
casos_anteriores	saulp	1517	463		casos_anteriores	Informix	531	463	0
casos_edwin	root	202	1343		casos_edwin	Informix	174	1343	0
casos_enjuiciados	root	469	913		casos_enjuiciados	Informix	302	913	0
casos_mora	root	188	540		casos_mora	Informix	162	540	0
casos_muestra	root	522	143		casos_muestra	Informix	319	143	0
cat_monedas	root	851	2		cat_monedas	Informix	420	2	0
cat_rutas	root	786	4		cat_rutas	Informix	395	4	0
cat_rutas_det	root	787	149		cat_rutas_det	Informix	396	149	0
catalogo_contable	root	1782	624		catalogo_contable	Informix	588	624	0
catalogo_egresosp	root	322	160		catalogo_egresosp	Informix	248	160	0
catalogo_ingresop	root	764	145		catalogo_ingresop	Informix	382	145	0
cci_admin	root	1884	0		cci_admin	Informix	630	0	0
cci_divisiones	root	1887	2		cci_divisiones	Informix	631	2	0
cci_modulos	root	1891	2		cci_modulos	Informix	632	2	0

Como podemos observar en esta tabla, la información está dividida en dos partes, el listado de tablas del servidor antiguo llamado “new_fha” y el servidor nuevo llamado “dblnx”, esto para poder comparar el traslado exitoso de la información en este último.

En la tabla se definen los campos siguientes:

- tabname: nombre la tabla.
- owner: indica el propietario de la tabla.
- tabid: es el número de identificador de la tabla en sistema.
- nrows: indica el número de filas que contiene cada tabla.
- Diferencia: este campo muestra si existe diferencia en el número de filas en cada tabla del sistema anterior y el sistema nuevo.

Prácticamente utilizamos esta información para visualizar de manera rápida y sencilla las posibles diferencias entre cada tabla del sistema.

5. “PERSONALIZACIÓN Y ÚLTIMAS CONFIGURACIONES EN LA NUEVA BASE DE DATOS DE INFORMIX”

5.1 Instanciamiento de la base de datos de Informix

Para empezar este concepto de instanciamiento, únicamente se pudo haber utilizado en esta nueva versión de Informix. No se llevara a cabo un análisis profundo ni la configuración de estas instancias en este contexto debido a que es parte de la configuración del sistema de administración de bases de datos y está fuera de los límites y alcances especificados en esta investigación.

Para tener una mejor idea del concepto de instanciar la base de datos, tenemos que imaginar a nuestro sistema de bases de datos en varios ambientes de trabajo distintos, por ejemplo: un ambiente de producción que es prácticamente donde el sistema de base de datos interactúa con la información propia del negocio es una instancia. Aquí es donde se almacena la información de clientes, casos, proyectos, etc.

Otra instancia podría ser un ambiente de desarrollo el cual está únicamente y exclusivamente para uso del personal de desarrollo de software, en esta instancia se hace pruebas de nuevas funcionalidad en información y datos de prueba pero que son un espejo de la instancia real del sistema de producción. Y así podemos imaginar un sin fin de ambiente de trabajo con distintos propósitos.

Pero para poder entrar un poco más de lleno en este tema de instancias de la base de datos, tenemos que recordar el concepto de dbSPACE enunciado en el capítulo 4, este concepto trasciendo a otro concepto más evolucionado denominado dbSPACETEMP que es al que nos vamos a enfocar en estos momentos.

¹⁰ El dbSPACETEMP especifica una lista de dbSPACES que el administrador de base de datos utiliza para administrar globalmente el almacenamiento de tablas temporales. La idea es que el servidor de base de datos utilice estos dbSPACES temporales durante los backups para almacenar imágenes de la data del sistema, los cuales puede ser utilizables mientras que el backup está corriendo.

Si la aplicación del cliente necesita especificar una lista alternativa de dbSPACES para uso de asignación de tablas temporales, el cliente puede usar el dbSPACETEMP utilizando las variables de ambiente para su uso.

Una variable de ambiente es aquella que contiene la información necesaria para especificarle al servidor de base de datos que dbSPACES usara para almacenar las tablas temporales para una sesión en particular, si el dbSPACETEMP no es ajustado la ubicación por defecto es el dbSPACE root.

Para que el servidor de base de datos pueda habilitar el uso de un nuevo dbSPACE temporal debe de seguir los siguientes pasos:

¹⁰ **IBM CORP.** “IBM Informix Dynamic Server Administrator’s Reference”, Versión 9.4, Capítulo 1-31 “Configuration Parameters”, página 63.

- Añadir el nombre del nuevo dbSPACE temporal a la lista de dbSPACES temporales en los parámetros del dbSPACEtemp, o en la variable de ambiente del dbSPACEtemp, o en ambos.
- Reiniciar el servidor de base de datos con el comando oninit de Unix.

En el caso que se utilice una variable de ambiente con el dbSPACEtemp para crear temporalmente un dbSPACE en una sesión de usuario, el cambio toma efecto inmediatamente y sobrecarga los valores del dbSPACEtemp en el archivo ONCONFIG.

A continuación se muestra una tabla de la configuración de las distintas instancias creadas en el sistema de la institución del fha, en el momento que se llevo a cabo la migración de datos hacia la nueva versión de Informix. Cabe mencionar que la primera instancia llamada fha_tcp es la instancia que se utiliza para producción, es decir el ambiente de trabajo real del sistema. La segunda instancia es la desa_tcp y como su nombre lo puede indicar es una instancia creada para el ambiente de desarrollo, la última instancia es la tmp_tcp esta instancia se creó como un segundo ambiente de desarrollo pero enfocado más a un ambiente de pruebas específicas que sean necesarias en algún momento determinado. Su configuración y ubicación son mostradas en la figura 8.

Figura 8. Configuración de Instancias de la Base de Datos creadas en el sistema del FHA

Instancia: fha_tcp	DBSPACES	Tamaño		Ubicación física	Bases de datos dentro de la instancia
1	rootdbs	0	15000	/dev/raw/raw1	
					sysmaster
					sysutils
					sysusers
2	physdbs	15000	25000	/dev/raw/raw1	
3	logsbds	40000	50000	/dev/raw/raw1	
4	tmp1dbs	90000	50000	/dev/raw/raw1	
5	tmp2dbs	140000	50000	/dev/raw/raw1	
6	data1dbs	0	3500000	/dev/raw/raw2	fha

Instancia: desa_tcp					
6	rootdbs	83000	1670000	/dev/raw/raw1	
6.625		91875	1933750		sysmaster
7.25		100750	2197500		sysutils
7.875		109625	2461250		sysusers
8.5	physdbs	118500	2725000	/dev/raw/raw1	
9.125	logsbds	127375	2988750	/dev/raw/raw1	
9.75	tmp1dbs	136250	3252500	/dev/raw/raw1	
10.375	tmp2dbs	145125	3516250	/dev/raw/raw1	
11	data1dbs	154000	3780000	/dev/raw/raw2	fha
Instancia: tmp_tcp					
11.625	rootdbs	162875	4043750	/dev/raw/raw1	
12.25		171750	4307500		sysmaster
12.875		180625	4571250		sysutils
13.5		189500	4835000		sysusers
14.125	physdbs	198375	5098750	/dev/raw/raw1	
14.75	logsbds	207250	5362500	/dev/raw/raw1	
15.375	tmp1dbs	216125	5626250	/dev/raw/raw1	
16	tmp2dbs	225000	5890000	/dev/raw/raw1	
16.625	data1dbs	233875	6153750	/dev/raw/raw2	fha

5.2 ¿Cómo configurar el Driver Informix-Cli del lado del cliente y la forma de comunicación de datos a la nueva versión de Informix?

Antes de poder seguir con este paso debemos de tener claro que en este momento la base de datos esta ya cargada y configurada totalmente en el nuevo servidor de Informix llamado “dblrx”, la base de datos esta también instanciada y lista para poder usar e interactuar con el sistema. Únicamente nos queda llegar a la maquina del cliente y configurar el driver que utiliza Informix para poder comunicarse con el servidor de la base de datos. Dicho driver es llamado Informix-Cli.

Este driver es proporcionado por IBM para poder tener un protocolo de comunicación y configuración hacia Informix desde cualquier host hacia el servidor. A continuación mencionamos los parámetros que se reconfiguraron para poder llevar a cabo la comunicación del cliente hacia el nuevo servidor “dblnx” y la nueva versión de la base de datos “fha”.

5.3 Configuración del driver para Informix “Informix-Cli” y los archivos del sistema hacia el nuevo servidor DBLNX ¹¹

5.3.1 Archivos del Sistema:

1. Como primer paso se agrego en el archivo HOSTS del usuario el nuevo servidor, este archivo se encuentra en la siguiente ubicación:

c:\windows\system32\drivers\etc\hosts

Este es un archivo de configuración que maneja todos los dispositivos a donde el cliente puede comunicarse. Es un archivo simple de texto y se utilizo un editor de texto para agregar la siguiente línea, que registra la dirección IP del nuevo servidor.

192.9.200.3 dblnx

¹¹ **ING. SAÚL PEREIRA, DBA en Informix**, Entrevistas y documentos de información creados por él.

2. También en este directorio se encuentra el archivo SERVICES que a semejanza del anterior este también es un archivo de configuración (es un archivo de texto sencillo), y aquí se ingresan los servicios que el cliente tiene sobre el servidor de la base de datos de Informix, se añadió el siguiente servicio:

sqlturbo 1540/tcp

5.4 Cambios en el driver Informix-Cli

Este es un programa que está instalado en cada máquina cliente, sin este es imposible la comunicación al servidor de la base de datos de Informix. A continuación presentamos las opciones que se cambiaron en el menú de configuración del Informix-Cli.

5.4.1 Ilogin Demo

1. Server: fha_tcp.

Es el nombre de la instancia de trabajo que el cliente utilizará. Como se pueden dar cuenta esta instancia es la de producción, explicada anteriormente.

2. Hostname: dblnx

Es el nombre del servidor de la base de datos nuevo.

3. Servicename: sqlturbo

Se hace referencia al nuevo servicio agregado en los archivos de configuración.

4. Protocol: onsoctcp

Este protocolo es el que utiliza la nueva versión de Informix on-line. (ver inciso 7 página 8 de este documento)

5. Username: Informix

Nombre del usuario que se utilizará para crear la conexión hacia la base de datos.

5.4.2 Setnet 32

1. Host Information:

- Current Host: dblnx
- User Name: Informix
- Password Option: Password

2. Server Information:

- Informix Server: fha_tcp
- Hostname: dblnx
- Protocol: onsoctcp
- Service name: sqlturbo

En las siguientes opciones se eliminó la información que estaba configurada, debido a que ya no es necesaria su configuración.

- Eliminar dbpath
- Eliminar Environment.
- InformixServer: fha_tcp

Una vez terminada esta configuración en el driver Informix-Cli, aquí mismo existe una opción para probar la conexión hacia el servidor, prueba que fue útil para poder corroborar estos cambios. Y así de esta forma se llevo a cabo la configuración de todas las máquinas clientes en la institución del “FHA”.

5.5 Análisis del comportamiento de aplicaciones con la nueva versión de Informix

En esta parte se llevo a cabo un sin fin de pruebas que tuvieron como objetivo el probar el funcionamiento total del sistema en la nueva versión de Informix junto con su nuevo servidor.

Dentro de los módulos del sistema que se utilizó para corroborar el buen funcionamiento del sistema fue el módulo Mayor Contable, debido a que este módulo lleva todo el control contable de la institución y es el que mayor peso tiene dentro del sistema. Aquí se analizó el acceso a la base de datos tanto en mantenimientos hechos en la herramienta de desarrollo Visual Basic 6 como en accesos a datos por reportes hechos en Cristal Reports.

5.6 Descripción de resultados de pruebas

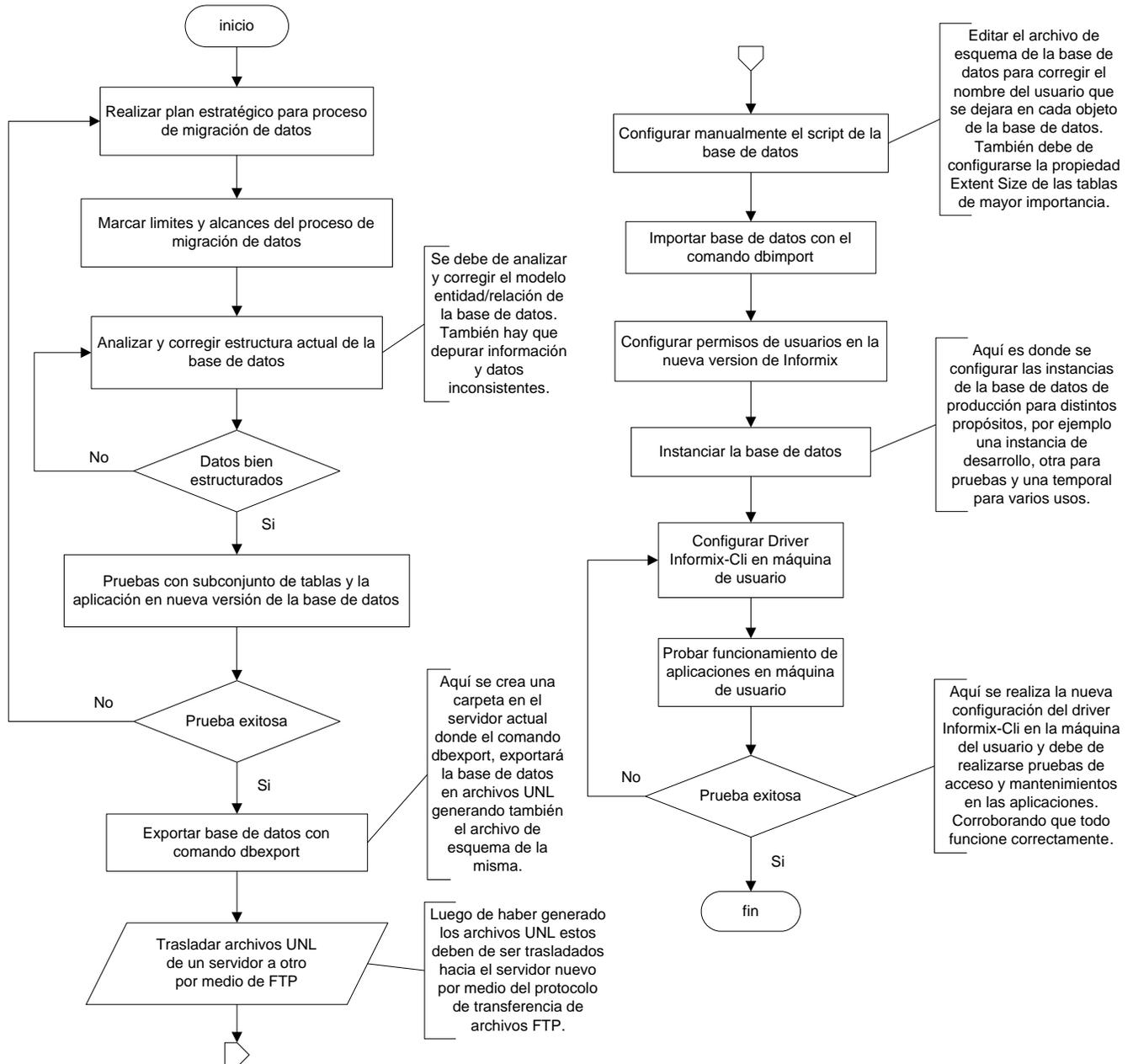
Uno de los parámetros que se pudo observar fue el acceso rápido a la información de la base de datos. Desde las maquinas clientes se probó el acceso a la información el cual por comentarios del Administrador de la Base de datos se observó que este acceso es lo suficientemente más rápido que en la versión antigua de Informix, aquí ya no se trataba de un prueba pequeña sino ya en un ambiente real y por consiguiente el sistema siguió funcionando y rindiendo mucho mejor tal y como se comportó en pruebas anteriores que se hicieron antes de realizar el proceso de migración de datos.

Otro análisis que se observó y mejoras que se obtuvieron con la nueva versión de Informix fue la herramienta de acceso a la base de datos, esta herramienta llamada DBAccess, nos presentó como desarrolladores del sistema un mejor acceso y manipulación de datos a través del lenguaje estructurado, además de tener un mejor rendimiento y control en la base de datos. También se obtuvieron mejoras en cuanto al conjunto de funciones soportadas ya en esta nueva versión de Informix, por ejemplo, funciones de SQL que la versión anterior no tenía ni soportaba, esto nos brindó un aumento en la disponibilidad de nuevas funciones que Informix ofrece hoy en día para poder manipular mejor los datos en nuestro sistema.

Otro análisis de pruebas que se pudieron realizar fue dentro del departamento de desarrollo, aquí se obtuvieron mejoras en cuanto a desarrollo de software se trata, debido a que se obtiene facilidad en cambios de ambientes de trabajo, es decir que las aplicaciones pueden muy bien trabajar en ambientes de producción, desarrollo y pruebas en cualquier momento que se configure las variables de ambiente que controlan las instancias de la base de datos “fha”, permitiendo así que se realicen pruebas y cambios con mayor seguridad.

Otro punto muy importante que se analizó en las aplicaciones es que la comunicación con nuevas herramientas de desarrollo para Informix, ya fue posible con esta nueva versión. Ya que con la versión anterior se obtuvieron muchas incompatibilidades incluso con el mismo driver de Informix (Informix-Cli) aquí el sistema nuevo de Reclamos y Liquidaciones desarrollado en Visual Basic .Net, ya interactuó de una forma más eficaz y sin problemas de compatibilidad, ya que se aprovechó a actualizar también el driver Informix-Cli a la última versión. Esta última versión del driver de Informix ya no era compatible con la versión antigua de Informix, así que esta es una ventaja más del cambio que se obtuvo en la migración de la base de datos Informix.

Figura 9. Flujograma del proceso de migración de datos en Informix



CONCLUSIONES

1. Considero que dentro de todo el proceso de migración de la base de datos es muy importante la parte de depuración y corrección de errores en la integridad de la información de la misma. En el caso del FHA, fue elemental este proceso debido a la trascendencia que el sistema ha tenido durante el transcurrir del tiempo. Esto nos demuestra que el sistema ha venido evolucionando y con ello ha venido expandiéndose a tal punto que fue necesario reestructurar algunas partes del modelo de la base de datos, por ejemplo la relación de tablas, la creación de usuarios, depuración de data, etc.
2. Que dentro del proceso de administración del sistema de base de datos, una de las mayores responsabilidades del DBA es velar porque su sistema tenga toda la disponibilidad de nuevas y mejores herramientas que existen en la actualidad.
3. De todos los beneficios que obtuvimos de este proceso de migración en Informix el más importante fue instanciar la base de datos para diversos propósitos. Esto permitió al Departamento de Desarrollo de *Software* tener un ambiente de desarrollo y un ambiente de pruebas, ambos idénticos al ambiente de producción. Esto es de mucha utilidad pues el desarrollo de *software* necesita de estos ambientes para poder realizar un buen producto de calidad.

4. Considero que el proceso de migración de datos fue un proceso exitoso porque fue un proceso transparente para los usuarios y se realizó de la mejor forma posible, porque hubo un plan estratégico a seguir durante todo el proceso, plan que está detallado completamente en este trabajo de graduación.

RECOMENDACIONES

Las siguientes recomendaciones que se mencionaran son derivadas de lo observado durante todo el proceso de migración de datos.

1. Antes de proceder a una migración de base de datos, es sumamente recomendable realizar un plan estratégico a seguir. Para poder realizar este plan, se puede empezar por documentar adecuadamente las diversas configuraciones que estén involucradas en nuestro sistema actual de trabajo, y visualizar cómo quedará configurado nuestro sistema futuro de trabajo al cual vamos a migrar.
2. Antes de llevar a cabo un proceso completo de migración de datos es necesario también realizar pruebas a pequeñas escalas sobre el ambiente nuevo, configurando, como en este caso, pequeñas estructuras donde se identifique que el sistema puede de alguna forma fallar en el nuevo ambiente, de cualquier forma mientras más real sea el ambiente de prueba mejores resultados obtendremos en este tipo de procesos.
3. Considerar las mejoras que se obtendrán así como también las compatibilidades de la herramienta de trabajo, de desarrollo y de *hardware*.
4. El proceso de migración de datos hacia una nueva versión de base de datos es necesario, debido a que cada día salen mejoras a nuestro sistema de trabajo y los cambios en cuanto a seguridad y rendimiento siempre van a mantener mejor nuestro sistema dentro de la empresa. Así también se aprovecha la compatibilidad de nuevas tecnologías, que en este caso, una mejora que realmente se obtuvo fue la integración de la tecnología .Net en nuestro sistema.

REFERENCIAS ELECTRÓNICAS

1. MIGRACIÓN DE DATOS EN INFORMIX
<http://publib.boulder.ibm.com/infocenter/idshep/v10/index.jsp?topic=/com.ibm.mig.doc/mig.htm>,
Fecha consultada 08/05/2006
2. MIGRACIÓN DE DATOS EN INFORMIX
<http://jgharris.com/informix-version-10/mig/25122930tfrm.htm>,
Fecha consultada 07/07/2006
3. MIGRACIÓN DE DATOS EN INFORMIX
<http://www-01.ibm.com/support/docview.wss?uid=swg21259394>
Fecha consultada 05/03/2007