



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN

MONITOREO Y AFINACIÓN DE BASES DE  
DATOS INFORMIX

T E S I S

QUE PARA OBTENER EL TÍTULO DE  
INGENIERO EN COMPUTACIÓN

P R E S E N T A N:

JUAN VALERIANO VILLEGAS

JORGE OSORIO LÓPEZ

ASESOR: ING. RAÚL BARRÓN VERA



MÉXICO

MARZO, 2007



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**ÍNDICE**

---

**ÍNDICE**

<b>CARTA DE AUTORIZACIÓN DE IMPRESIÓN DE TESIS</b>	<b>4</b>
<b>AGRADECIMIENTOS</b>	<b>5</b>
<b>OBJETIVO</b>	<b>6</b>
<b>INTRODUCCIÓN</b>	
a.    Antecedentes	7
b.    Historia del proyecto SIIC	9
c.    Módulos del sistema SIIC en operación	11
d.    Recursos de hardware y software del sistema SIIC en operación	18
e.    Consideraciones para desarrollar un modulo de monitoreo	22
<b>1. - CONCEPTOS GENERALES</b>	
1.1    Bases de datos	24
1.2    Conceptos básicos	24
1.3    Diagrama entidad-relación	29
1.4    Diagrama de flujo de control	30
1.5    Especificación de procesos	30
<b>2. - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS</b>	
2.1    Replicación continua de datos	32
2.2    Definición del servidor central y los 77 servidores remotos	33
2.3    Monitoreo del CDR en servidor central y sus 77 servidores remotos	34
2.4    Estado del modulo de replicación	38
2.5    Monitoreo de las replicas de datos inconsistentes	41
2.6    Monitoreo de las replicas pendientes de envío y/o recepción	45
2.7    Monitoreo de las replicas de datos inactivas	48
<b>3. - MONITOREO DEL I/O POR DBSPACE</b>	
3.1    Accesos por segundo de I/O por dbspaces	50
3.2    Balanceo del I/O por dbspaces	60
3.3    Ocupación de cada dbspaces y por chunk	73
<b>4. - MONITOREO DE LAS TABLAS</b>	
4.1    La seguridad en el acceso a las tablas	83
4.2    Actualización de las estadísticas de las tablas	86

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX ÍNDICE

---

4.3	Verificación de la estructura interna de las tablas	93
4.4	Acceso secuencial a las tablas	95
4.5	Los extents de las tablas	98
4.6	La fragmentación de las tablas	101
<b>5. - MONITOREO DE LAS TRANSACCIONES</b>		
5.1	Transacciones en espera debido a registros y/o tablas bloqueadas	103
5.2	Transacción(es) que esta(n) bloqueando registros y/o tablas	106
5.3	Transacciones en proceso	110
5.4	Transacciones con código de error	114
5.5	Transacciones sin terminar del día anterior	117
5.6	Transacciones sin terminar en log's anteriores	121
5.7	Transacciones no autorizadas	123
<b>6. - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS</b>		
6.1	Para los dbspaces	127
6.2	Para los índices, Foreign Key y Primary Key.	132
6.3	Para las tablas	133
6.4	Para los parámetros de configuración de la Base de Datos	134
6.5	Para el equipo UNIX	138
6.6	Para los discos externos	140
6.7	Para el respaldo de la BD y SO por medio de una librería	143
<b>CONCLUSIONES</b>		
a.	Riesgos de no efectuar un monitoreo adecuado	144
b.	Ventajas y beneficios al efectuar un monitoreo adecuado	144
c.	Monitoreo, una actividad permanente y cambiante	145
<b>BIBLIOGRAFIA</b>		146
<b>ANEXO 1</b>		
	Parámetros configurables del servidor de base de datos	147
<b>ANEXO 2</b>		
	Definición de una replica de datos hacia los servidores remotos	148

**OBJETIVO:**

Presentar los parámetros más relevantes a monitorear en una base de datos INFORMIX y la afinación de los parámetros mencionados, para obtener el mejor desempeño de la misma.

## **INTRODUCCIÓN**

### **a) Antecedentes**

Pemex Refinación es la empresa paraestatal que transforma el petróleo crudo en los diferentes productos petrolíferos que requiere el desarrollo del país. Esto a través de la red de refinerías con las cuales cuenta la empresa, distribuidas por todo el territorio nacional. Con lo cual se satisfacen los requerimientos de combustibles que demanda la nación para todas las necesidades, ya sea de transporte (terrestre, aéreo, marítimo), industrial, etc.

La Subdirección Comercial es la encargada de comercializar los productos petrolíferos, por lo cual cuenta con almacenes de distribución en las 77 ciudades principales, con los cuales garantiza el abastecimiento de los productos a los clientes que lo requieran.

El traslado de los productos petrolíferos de las 5 refinerías, a los 77 centros de almacenamiento y distribución se realiza por ductos (oleoductos, poliductos), también por ferro tanque, auto tanque y buque tanques.

Pemex Refinación cuenta con una relación comercial con 7000 gasolineras y 200 grandes consumidores. Los clientes de Pemex Refinación cuentan con un contrato comercial mediante el cual se rigen la relación de negocios entre ambos.

El traslado de los productos petrolíferos al expendedor final se realiza por auto tanques propiedad de la empresa, y por auto tanques propiedad de terceros.

El administrar y operar lo expuesto anteriormente requiere de la participación coordinada de los trabajadores de la empresa, de sus proveedores de servicios, de los clientes, de diversas secretarías de estados, tales como la Secretaría de Hacienda, la Secretaría de Energía, etc.

Para mantener los niveles de servicio que los clientes de Pemex Refinación requieren, la Subdirección Comercial se ha apoyado en los Sistemas de Información.

En 1988 se implanto el Sistema de Automatización de Terminales (SAT), el cual contaba con un equipo Main-Frame IBM 9121-521, ubicado en las Oficinas Centrales de la empresa, y terminales IBM terminales 3178 (monocromática) y 3179 (color) con impresoras modelos 3287 y 4224, instaladas en las terminales de almacenamiento y distribución. En esta etapa de sistematización de la operación comercial, únicamente se capturaban los movimientos de facturación.

En 1992 se implanto el sistema INFOSAT, para robustecer y mejorar la funcionalidad del sistema comercial anterior (SAT), se conservo toda la infraestructura y se fortaleció instalando 77 servidores UNIX mediante los cuales el personal de los centros de almacenamiento

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

---

tuvieron mas funciones operativas automatizadas dentro del sistema, otro de los beneficios fue el de poder transmitir vía electrónica y automatizada la facturación del día, esto a través del protocolo de comunicación SNA-RJE de IBM.

Este sistema INFOSAT opero satisfactoriamente hasta el año de 1997, pero debido a los cambios operativos y de las políticas comerciales de la empresa a lo largo de los años anteriores, el sistema ya había dejado de ser funcional, y los cambios al mismo eran muy complicados, difícil de implantarlos. Aunado a que la plataforma tecnológica en el cual fue desarrollado ya estaba en obsolescencia y era cerrada.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

---

### b) Historia del proyecto SIIC

En 1997 se tomo la decisión de desarrollar el Sistema Institucional de Información Comercial (SIIC), con las premisas de contar con un sistema de información que cumpliera al 100% las necesidades actuales (comerciales, operativas) de la empresa y de los clientes de Pemex Refinación, el sistema debía de ser modular y flexible a los cambios de la política comercial de la empresa y además que este implantado sobre una tecnología abierta y que resolviera el Problema del año 2000.

El desarrollo del SIIC inicio en enero de 1998 y la implantación del mismo se realizo en septiembre de 1999, entre las nuevas características del sistema SIIC es de resaltar:

El SIIC es utilizado por las Oficinas Centrales, Gerencias de Zona y Terminales de Almacenamiento y Distribución TAD's y la información podrá ser actualizada y consultada en los siguientes niveles:

- Local - Sólo información que le pertenece al centro.
- Regional - Sólo información que le pertenece a la Gerencia Regional y todos los centros que le corresponden.
- Central - Toda la información de acuerdo al módulo que opera.

Ofrece la máxima integridad en la información, todas las transacciones realizadas en el sistema por los usuarios serán actualizadas EN LÍNEA en el equipo central HP9000 de la siguiente forma:

- Transacciones de usuarios de Oficinas Centrales y Gerencias de Zona mediante conexión en forma directa a la base de datos central.
- Transacciones de usuarios de Terminales de Almacenamiento y Distribución mediante la replicación de datos a través de la Tecnología CDR (Continuos Data Replication) de INFORMIX.

Garantiza que la operación en las TAD's no se detengan debido a caídas de enlace o fallas en las comunicaciones, ya que podrán seguir operando con el computador local mientras no exista comunicación porque este contendrá los últimos datos actualizados hasta el momento del corte, una vez restablecida la comunicación se actualizarán las transacciones que se realizaron a nivel central y regional hacia nivel local y viceversa.

Toda la información esta en línea, ya que a través del modulo de replicación, toda la información generada en los 77 centros de almacenamiento y distribución es actualizada en el servidor central en un período de tiempo menor a los 60 segundos.

Con lo anterior se evito realizar los cierres diarios y mensuales de toda la información contable y financiera resultado de toda la operación de las terminales de almacenamiento y distribución.



## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

---

Generación, cálculo y distribución en línea de los precios de producto elaborados por la Subdirección Comercial.

Eliminación de transferencia de documentos físicos y archivos batch.

Se implemento una plataforma única para los servidores remotos, con los beneficios que esto genera, tanto para la generación de nuevas versiones del sistema, el mantenimiento a los equipos, versiones de software de los diferentes componentes del sistema, etc.

Los servidores centrales y remotos son de tecnología abierta tanto en software y hardware.

Cuenta con la opción de generar consultas y reporte abiertos, de acuerdo al criterio de búsqueda del usuario.

Cuenta con un Portal Comercial para Internet e Intranet por medio del cual los clientes pueden consultar su información desde cualquier PC con acceso a Internet.

Es un sistema de misión crítica para la empresa (24x7), que debe de estar operando, las 24 horas del día, los 7 días de la semana, con el mínimo de tiempo fuera de operación, ya sea de actividades programadas de mantenimiento y/o caídas del sistema.

El numero promedio de usuarios concurrentes es de 1,100 ya sea por la red interna, por la Intranet, por la WWW o Internet.

El número de transacciones al nivel de bases de datos por día es de 50,000

El número de accesos a disco por día es de 80 millones.

El espacio reservado actualmente para la base de datos es de 200 GB, 80 GB de datos y 120 GB de índices.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

---

### c) Módulos del sistema SIIC en operación

**Administración:** Controla y administra los perfiles de usuario para acceso al sistema.

**Catálogos:** Controla y administra todos los catálogos que complementan los diferentes módulos que conforman este sistema.

**Clientes:** Controla y administra los datos generales del cliente, ubicación de cada uno de los lugares donde se le puede proveer de producto, los productos que tiene autorizado adquirir y los centros que pueden proveerle producto a un cliente desde el inicio de la relación comercial entre el cliente y Pemex Refinación.

**Crédito y Garantías:** Controla y autoriza el crédito que proporciona Pemex Refinación a sus clientes respaldado por una afianzadora y una o más garantías. Permitiendo tener un plazo por cada producto que se le haya autorizado al cliente.

**Suministro:** Controla las existencias y los diferentes tipos de movimientos de entrada y salida de los productos comercializados por Pemex Refinación por cada Terminal de Almacenamiento y Distribución (TAD)

**Programa de Entregas:** Controla y administra el plan diario de entrega de producto mediante la programación de los diferentes medios de transporte por los cuales es distribuido el producto a los clientes o entre TAD's.

**Estructura de Precios:** Controla y administra las condiciones de venta y la aplicación de las estructuras de precios que son reflejadas en la facturación a los clientes de Pemex Refinación.

**Facturación:** Controla y genera la emisión de documentos comerciales (facturas y/o remisiones de productos, servicios, notas de crédito y débito) por la venta de productos refinados o servicios de Pemex Refinación.

**Estadísticas de Ventas:** Generación de informes estadísticos de las ventas de producto, incluyendo los documentos de ajuste a los mismos, agrupados de acuerdo a las necesidades de información de las diferentes áreas de Pemex Refinación. Lo anterior, para facilitar el análisis y toma de decisiones basándose en las ventas del organismo.

**Cobranza:** Registra el pago de documentos a cargo de los clientes de Pemex - Refinación a través de la cobranza electrónica que se efectúan con diversas instituciones bancarias y en la ventanilla de la Subdirección Comercial de Pemex Refinación.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

Los siguientes diagramas representan el flujo básico de información en el sistema SIIC

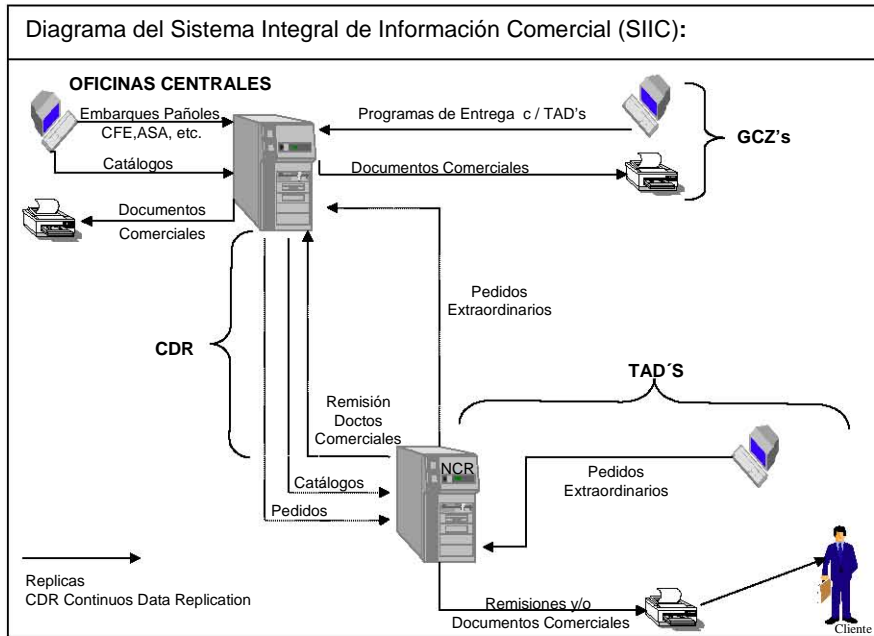


Diagrama c1

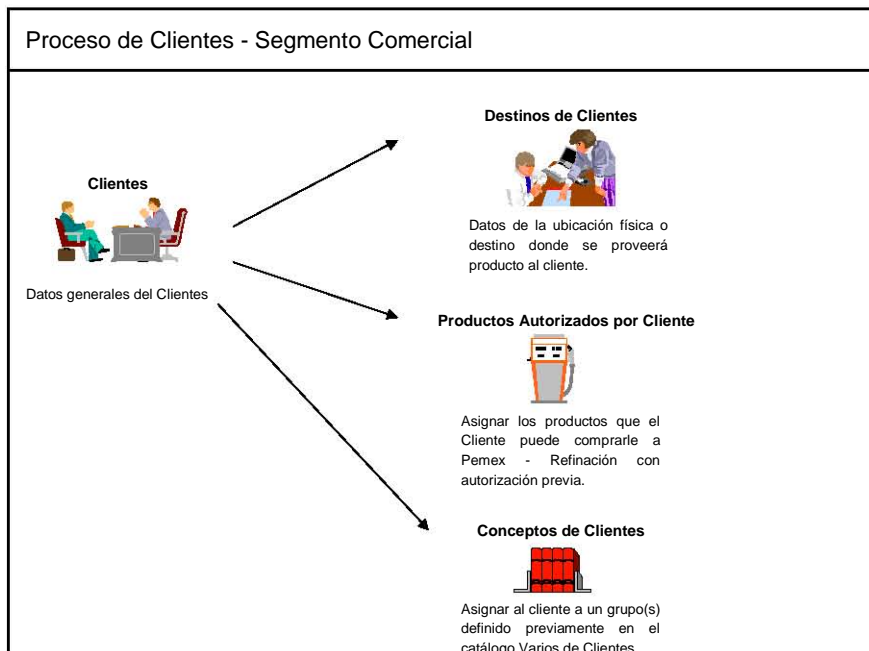


Diagrama c2

# MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

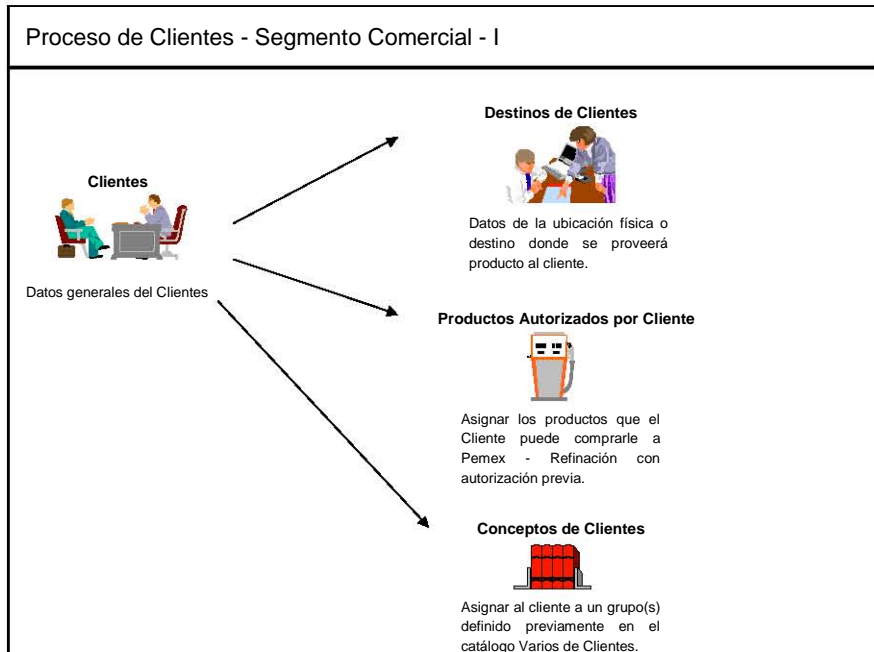


Diagrama c3

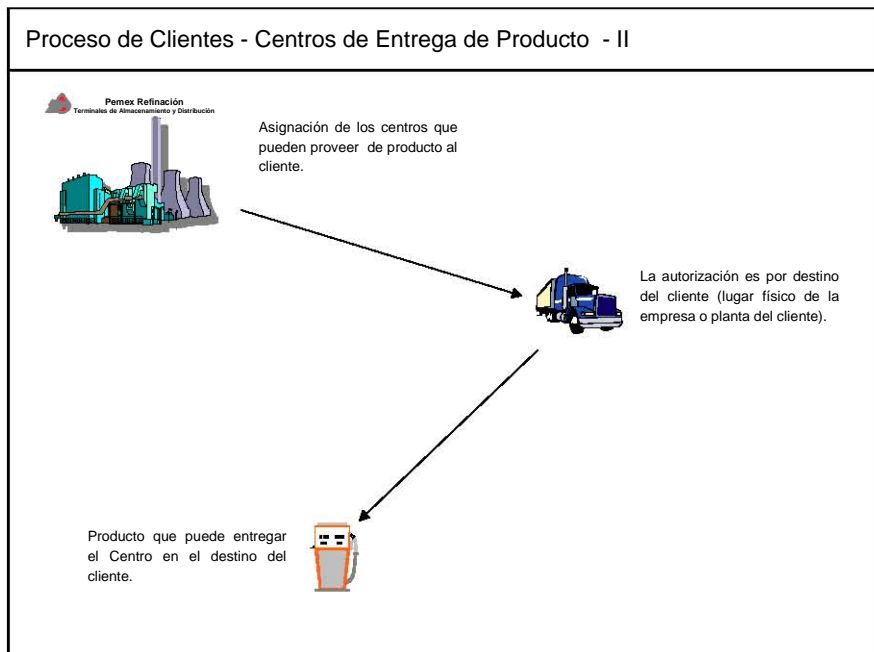


Diagrama c4

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

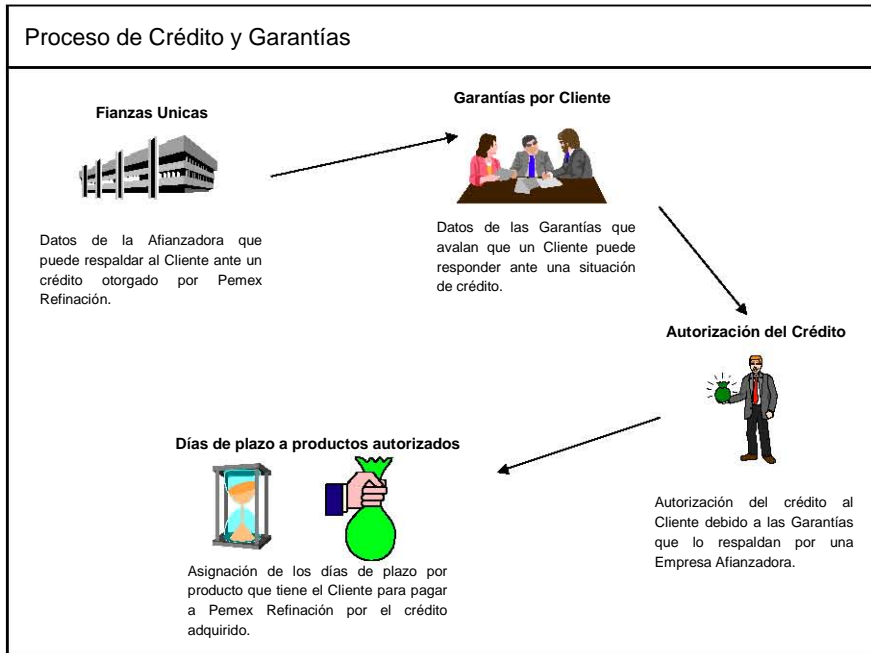


Diagrama c5

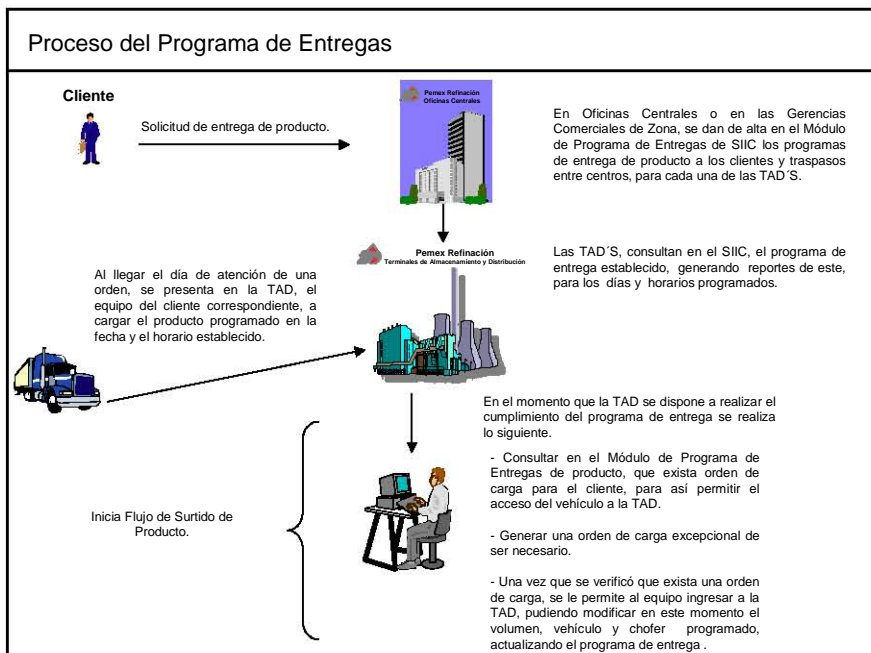


Diagrama c6

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

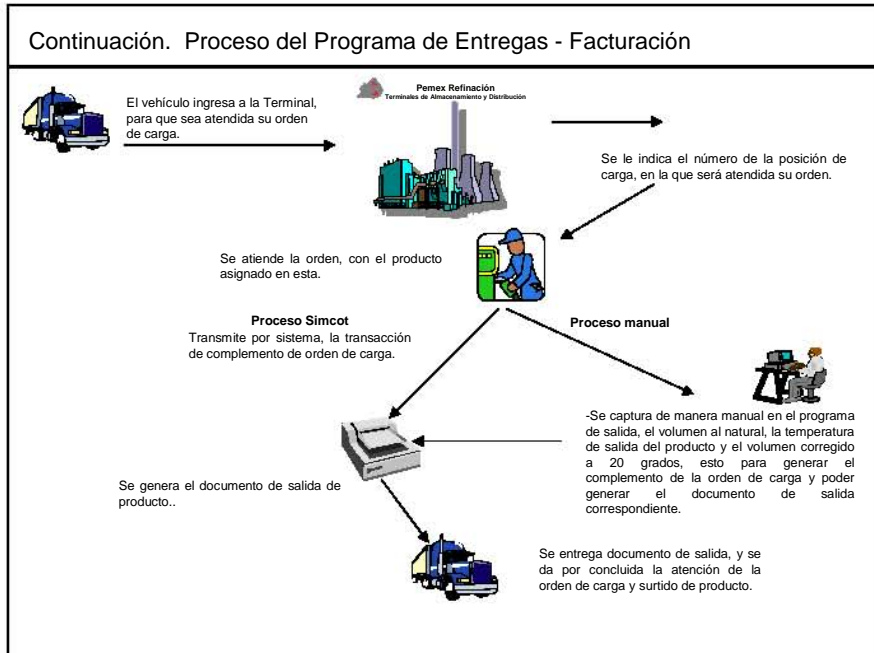


Diagrama c7

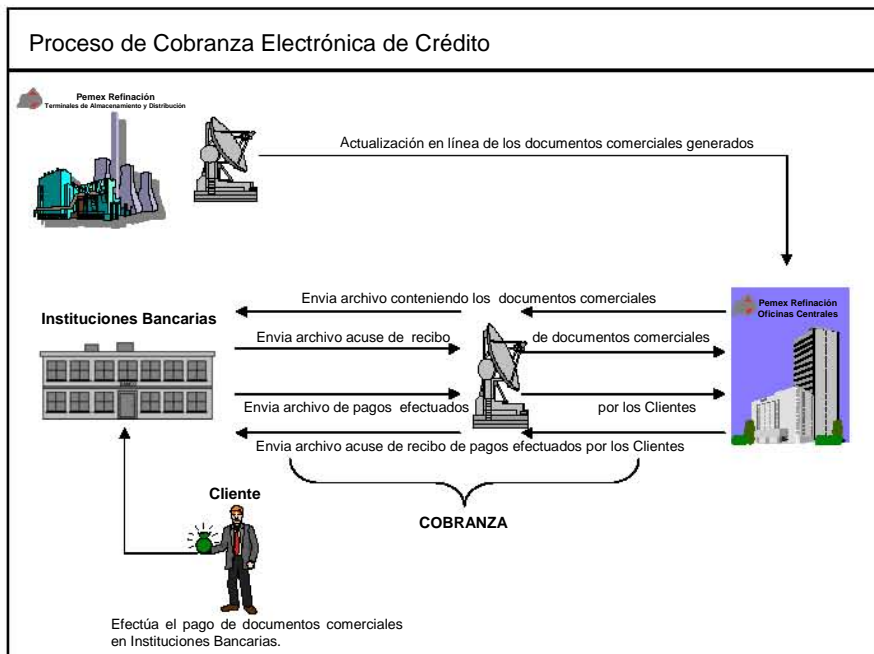


Diagrama c8

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

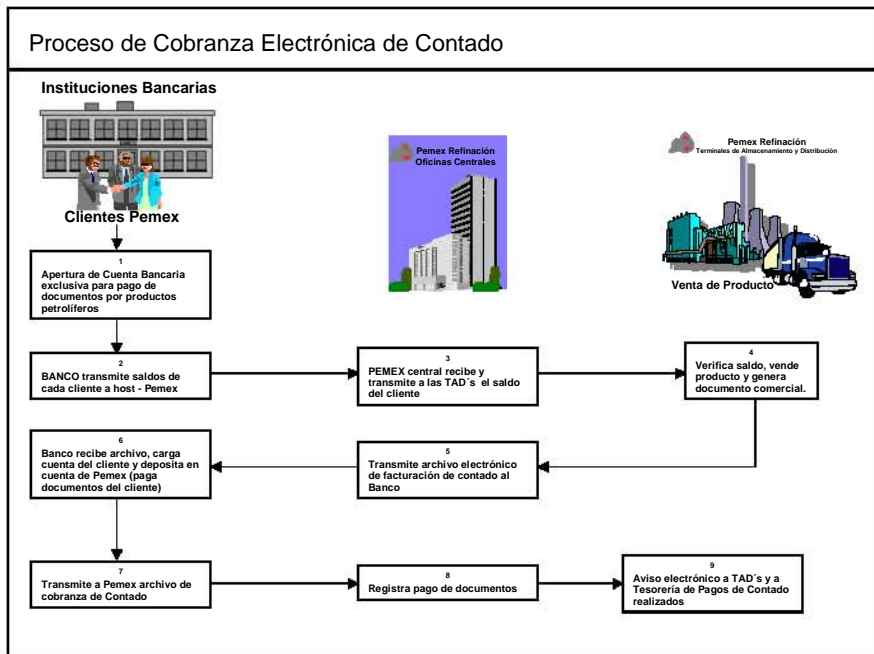


Diagrama c9

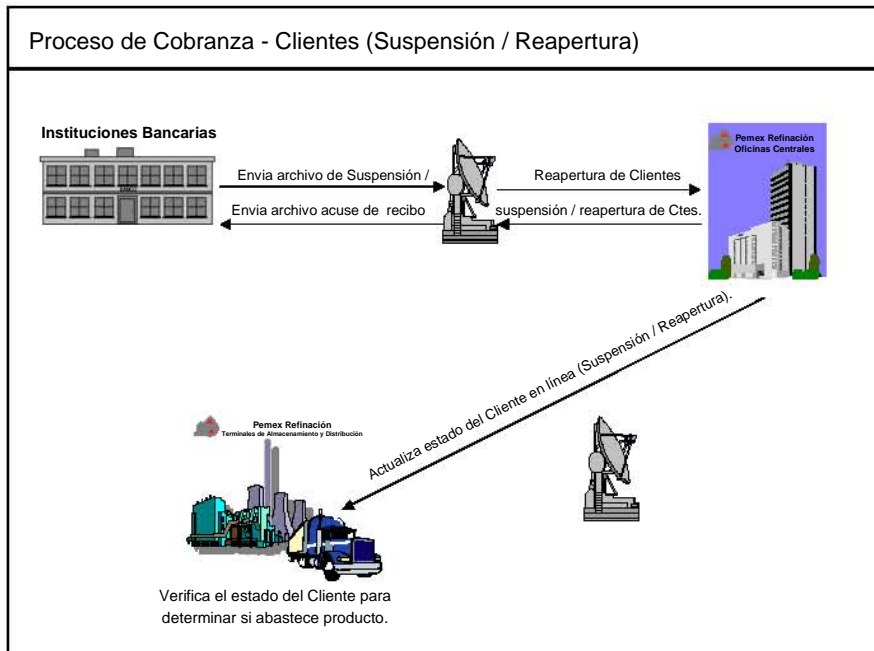


Diagrama c10

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### INTRODUCCIÓN

---

Existen también interfaces que sirven para intercambiar información con las instituciones bancarias con las que opera Pemex Refinación para actualizar la información de cobranza. Dichas interfaces están programadas para correr en horario predeterminados a lo largo del día. Es una comunicación en 2 vías en la cual se transfieren y reciben archivos planos.



## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

### d) Recursos de hardware y software del sistema SIIC en operación

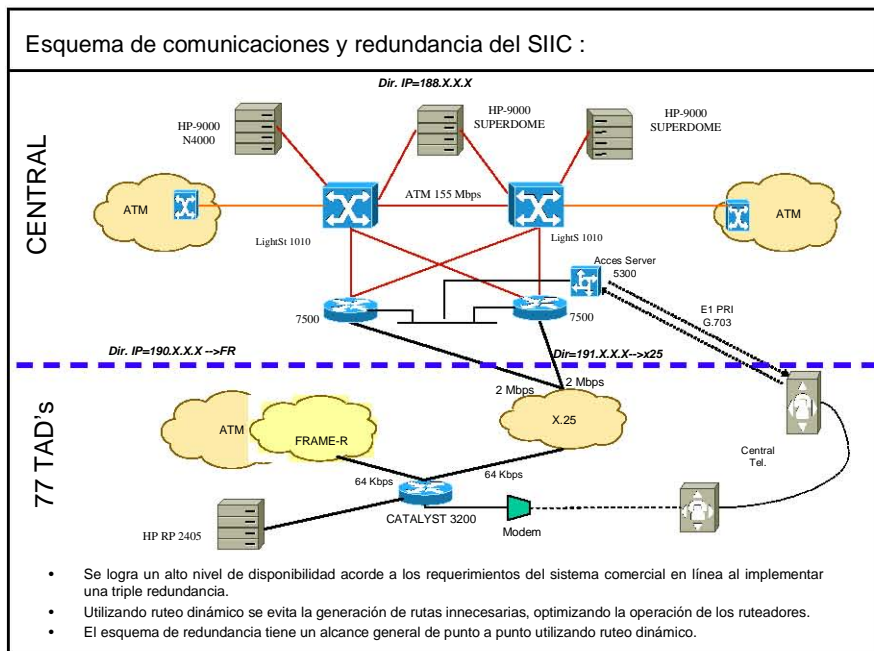


Diagrama d1

La red incluye un sitio central y 77 sitios remotos y/o locales. El uso de la misma es exclusivamente para la aplicación del SIIC.

El sitio central está conformado por dos segmentos Gigabit independientes para asegurar el servicio continuo de la red. Estos segmentos son utilizados por los servidores del sitio central.

En los sitios locales y/o remotos cuentan con 3 salidas: Frame Relay, X.25 y acceso telefónico vía MODEM. En operación normal, los segmentos de Frame Relay y X.25 están en servicio continuo y solo se recurre al MODEM en caso de caída o falla de las dos primeras opciones.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

---

El equipamiento en el centro de cómputo central del SIIC, cuenta con lo siguiente:

### **Servidor central (Servidor primario)**

Marca: HP  
Modelo: SUPER DOME  
Sistema Operativo: HPUX 11.11  
24 Procesadores a 700 Mhz  
24 GB de RAM  
300 GB de disco interno  
1.2 TB de disco externo (EMC DMX1000)  
6 tarjetas FC y 4 tarjetas Gigabit-Ethernet  
Software de Base de Datos: INFORMIX RDBMS 7.31.UD7W4



### **Servidor de respaldo (Servidor secundario)**

Marca: HP  
Modelo: SUPER DOME  
Sistema Operativo: HPUX 11.11  
24 Procesadores a 700 Mhz  
24 GB de RAM  
300 GB de disco interno  
1.2 TB de disco externo (EMC DMX1000)  
6 tarjetas FC y 4 tarjetas Gigabit-Ethernet  
Software de Base de Datos: INFORMIX RDBMS 7.31.UD7W4



Los 2 servidores anteriores forman un cluster de alta disponibilidad los cuales están en comunicación constante uno con el otro, de tal forma que si el servidor principal sufre una caída, el servidor de respaldo toma automáticamente el rol de servidor principal garantizando un mínimo de tiempo fuera de servicio del sistema SIIC, máximo 15 minutos.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

---

### Disco externo (1)

Marca: EMC  
Modelo: DMX 1000  
8 GB RAM  
1.2 TB de disco, Max 5 TB  
8 tarjetas FC



### Disco externo (2)

Marca: EMC  
Modelo: DMX 1000  
8 GB RAM  
1.2 TB de disco, Max 5 TB  
8 tarjetas FC



### Librería de respaldos

Marca: Storageteck  
Modelo: SureStore 10  
588 slots, para cintas  
10 drives DLT7000  
Capacidad por cinta: 70 GB



## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

---

El equipamiento en los 77 centros foráneos de cómputo del SIIC, tiene lo siguiente:

### **Servidor foráneo**

Marca: HP  
Modelo: RP 2405  
Sistema Operativo: HP UX 11.11  
1 Procesador RISC a 700 Mhz  
512 Mb de RAM  
36 GB de disco interno en mirror  
1 tarjeta ETHERNET  
Software de Base de Datos: INFORMIX RDBMS 7.31.UC6



## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

---

### e) Consideraciones para desarrollar un modulo de monitoreo

Por todo lo expuesto anteriormente estamos ante un sistema de misión crítica para la empresa, por el cual se realiza toda la facturación de los productos que comercializa Pemex Refinación, por lo cual se debe garantizar que siempre estará en operación.

Para lograr lo expuesto anteriormente, se debe de contar con un sistema de monitoreo de bases de datos, eficiente y oportuno, mediante el cual se auxilie a los administradores del sistema a operar, mantener y optimizar el uso del sistema SIIC.

Se realizaron la evaluación de diversos software's de monitoreo tales como: Tivoli de IBM, Patrol de BMC, Unicenter de CA.

Los requerimientos fueron:

Monitoreo de la base de datos INFORMIX 7.31.XX

Monitoreo del sistema operativo UNIX HP UX 11.11 en servidores marca HP

Los resultados de la evaluación del software de monitoreo fueron los siguientes:

Software de monitoreo	Equipos HP RP 2405	Equipos HP	Base de datos INFORMIX	Monitoreo regular	Monitoreo bueno	Monitoreo excelente
Tivoli		X	X	X		
Patrol	X	X	X	X		
Unicenter		X			X	

El único software que cumplió con los requerimientos fue el software de Patrol de la compañía BMC, se instaló en los 20 principales servidores remotos y en el servidor central, es de mencionar que el nivel de monitoreo es "regular", esto es, no notifica con el detalle requerido todos los procesos y eventos que se requieren.

Con el sistema SIIC en producción, la demanda de recursos al hardware y software que integran el sistema se incremento día a día, al proporcionar la información más confiable de los ingresos de la empresa, pues cada día más usuarios se integraban al mismo.

En sistemas con pocos usuarios jamás se presentan problemas de desempeño y/o cuellos de botella.

Pero en un sistema como el SIIC, la demanda de la explotación de la información contenida en el mismo, que es requerida tanto por usuarios externos (clientes de Pemex Refinación) como por usuarios internos. Puede llegar al promedio actual de 1,100 usuarios simultáneos en el servidor de la base de datos, con unos 80 millones de accesos a disco al día, y un consumo de 1 GB de log's por día. Con potenciales problemas de desempeño y/o cuellos de botella en: accesos a discos, manejo de memoria, uso de la red WAN, etc, etc.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX INTRODUCCIÓN

---

Por todo lo anterior, es evidente que es un sistema de alta demanda, por lo cual se decidió construir un módulo de monitoreo propio, el cual debe de monitorear y notificar oportunamente y de manera muy precisa, los aspectos más importantes: Del sistema operativo y del servidor de la base de datos INFORMIX, todo lo anterior componentes fundamentales del sistema SIIC.

La notificación de los eventos será por correo electrónico y hacia teléfonos móviles.

De manera general los eventos a monitorear por este módulo de monitoreo serán:

- Monitoreo de la replicación continua de datos
- Monitoreo de las transacciones
- Monitoreo del i/o por dbspace
- Monitoreo de las tablas
- Monitoreo de la red WAN/LAN que soporta al SIIC

Cada uno de los monitoreos realizados, proporciona al DBA, información importante para realizar la afinación de la Base de Datos, esta afinación se debe de llevar a cabo, paso a paso, es decir realizar un cambio a la vez, y volver a monitorear, debe de ser una actividad constante

## CAPITULO 1                      CONCEPTOS GENERALES

### 1.1    Bases de datos

Una Base de datos, es una colección de datos interrelacionados, almacenados sin redundancia innecesaria, para servir a múltiples aplicaciones; los datos son almacenados de tal manera que son independientes de los programas que los usan, y además estructurados de tal manera que sirven para aplicaciones futuras.

### 1.2    Conceptos básicos

- **Modelos de Bases de datos:** Un modelo de Bases de Datos se constituye con las herramientas que permiten describir la estructura de una base de datos. Estos modelos deben cumplir con la tarea de describir a los datos, sus relaciones y sus limitantes.

Existen 2 modelos de bases de datos:

- a) **Modelos jerárquicos y de red:** han sido utilizados como estructuras básicas de los sistemas de manejo de base de datos hasta el principio de la década de los setenta, donde las características principales son que los datos se organizan por medio de un conjunto de árboles.
- b) **Modelo relacional:** surgió a principios de la década de los setenta, con un nuevo planteamiento relacional en su simplicidad, es decir, la facilidad de comprensión por parte del usuario final. Los usuarios finales no se tienen que preocupar por la estructura de almacenamiento físico; se les puede orientar hacia el contenido de la información de sus datos. En el modelo relacional las entidades y sus relaciones se representan con tablas bidimensionales, compuestas por diferentes atributos.

El concepto matemático sobre el que está basado el modelo relacional es el de "relación" de la teoría de conjuntos. Una relación es un subconjunto del producto cartesiano de una lista de dominios. Un dominio es simplemente un conjunto de valores.

#### – Ventajas del modelo relacional

**Simplicidad:** Sus solicitudes se formulan en términos del contenido de la información y no reflejan la complejidad de los aspectos relacionados con el sistema.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 1 - CONCEPTOS GENERALES

---

**Consultas no planeadas:** Debido a que no hay una dependencia de posición entre las relaciones, las consultas no tienen que reflejar ninguna estructura preferida y por lo tanto pueden ser de tipo procedimientos no estándar de consulta.

**Independencia de los datos:** Esto debe constituir uno de los principales objetivos de cualquier sistema de manejo de bases de datos. El modelo relacional elimina los detalles relativos a la estructura del almacenamiento y la estrategia del acceso de la interfase con el usuario. Este modelo proporciona un grado de independencia de los datos relativamente más alta que los otros dos modelos antes mencionados.

**Fundamentos teóricos:** El modelo teórico está basado en la bien desarrollada teoría matemática de las relaciones. El riguroso método del diseño de una base de datos usando la normalización, da a este modelo un fundamento sólido. Esta clase de fundamento no existe para los otros modelos.

El campo del diseño estructurado ofrece guías para ayudar al diseñador a determinar los módulos, y sus interconexiones, que mejor realizarán los requerimientos especificados por el analista. Las dos reglas más importantes son las referentes al acoplamiento y la cohesión.

**Cohesión:** Grado en el cual los componentes de un módulo (típicamente las instrucciones individuales que conforman un módulo) son necesarias y suficientes para llevar a cabo una sola función bien definida. En la práctica, esto significa que el diseñador debe asegurarse de no fragmentar los procesos esenciales en módulos, y también de no juntar procesos no relacionados (que se representan por burbujas en DFD) en módulos sin sentido. Los mejores módulos son aquellos que son funcionalmente cohesivos (es decir, módulos en los cuales cada instrucción es necesaria para poder llevar a cabo una sola tarea bien definida) Los peores módulos son los que son coincidentemente cohesivos (es decir, aquellos cuyas instrucciones no tienen una relación significativa entre uno y otro)

**Acoplamiento:** Grado en el cual los módulos se interconectan o se relacionan entre ellos. Entre más fuerte sea el acoplamiento entre módulos en un sistema, más difícil es implantarlo y mantenerlo, pues entonces se necesitará un estudio cuidadoso para la modificación de algún módulo o módulos. En la práctica, esto significa que cada módulo debe tener interfaces sencillas y limpias con otros, y que se debe compartir un número mínimo de datos entre módulos. También significa que un módulo dado no debe modificar la lógica interna o los datos de algún otro módulo; lo que se conoce como una conexión patológica.

**Tamaño del módulo:** De ser posible, cada módulo debe ser lo suficientemente pequeño como para caber en una sola página (o para que pueda desplegarse en una sola pantalla). Desde luego, a veces no es posible determinar que tan grande va a ser un módulo hasta haberlo escrito, pero las actividades iniciales de diseño a menudo darán al diseñador una buena pista de que el módulo va a ser grande y complejo. Si es así, debe partirse en uno o más submódulos.



**Alcance del control:** El número de subordinados inmediatos que un módulo administrador puede llamar se conoce como el alcance del control. Un módulo no debe poder llamar a más de una media docena de módulos de nivel inferior. La razón es evitar la complejidad: si el módulo tiene digamos, 25 módulos de nivel inferior, entonces probablemente contendrá tanta lógica compleja de programa (HACERMIENTRAS anidadas, etc.) que nadie lo podrá entender. La solución es introducir un nivel intermedio de módulos administradores, como haría un administrador de una organización humana, así se ve en la necesidad de tratar de supervisar directamente a 25 subordinados.

**Ámbito del efecto/alcance del control:** Esta regla sugiere que cualquier módulo afectado por el resultado de alguna decisión debe ser subordinado (aunque no necesariamente un subordinado inmediato) del módulo que toma la decisión. Es un tanto a la regla de administración que dice que cualquier empleado afectado por los resultados de la decisión de algún administrador (es decir, dentro del alcance del efecto de la decisión) debe estar dentro del alcance de control de el administrador (es decir trabajando entre la jerarquía de personas que se reportan con el administrador). Violar esta regla en un ambiente de diseño estructurado usualmente lleva paso innecesario de banderas y condiciones (lo cual incrementa el acoplamiento entre módulos), la toma redundante de decisiones o (en el peor de los casos) conexiones patológicas entre módulos.

- **Bases de datos:** Es una colección de datos interrelacionados, almacenados juntos sin redundancia innecesaria, para servir a múltiples aplicaciones; los datos son almacenados de tal manera que son independientes de los programas que los usan, y además estructurados de tal manera que sirven para aplicaciones futuras.
- **Entidad:** *Es* una persona, un lugar, una cosa, un evento o un concepto, acerca del cual se registra información.
- **Registro:** Es una colección de datos o valores tomados por atributos relacionados entre sí.
- **Atributo:** Es la unidad mínima o elemento de datos que describen a las entidades.
- **Llaves:** Existen dos tipos de llaves:
  - Llave primaria. Es un atributo o combinación de atributos que identifican como única una instancia de una entidad.
  - Llave secundaria o foránea. Es un atributo que debe hacer referencia a una llave primaria en otra entidad, para establecer relación entre entidades. Esta puede contener nulos y duplicados.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 1 - CONCEPTOS GENERALES

---

- **Diccionario de datos:** Es el diseño referente al uso, relaciones y significado de los datos. Y a la herramienta que da la posibilidad de controlar y manejar la información sobre los datos en las fases de diseño, implantación, operación y expansión de una base de datos, se llama diccionario de datos.
- **Sistema relacional de Manejo de la Base de Datos (RDBMS):** *Un RDBMS hace posible acceder datos integrados que cruzan límites operacionales, funcionales u organizacionales dentro de una empresa. Para reconocer el potencial de un RDBMS debe satisfacer los siguientes requisitos:*
  - Que el RDBMS atienda de manera efectiva las diferentes funciones de la empresa.
  - Minimizar la cantidad de redundancia en los datos almacenados.
  - Suministrar información consistente al proceso que toma decisiones.
  - Aplicar controles de seguridad.
  - Desarrollar, mantener y mejorar los programas de aplicación de manera más rápida y económica, con menor personal calificado.
  - Facilitar la reorganización física de los datos almacenados.
  - Posibilitar el control centralizado de la base de datos.
- **Administrador de Base de Datos (DBA):** Es la persona responsable de administrar la base de datos. Se encarga de la definición de la estructura de almacenamiento, la creación de métodos de acceso apropiados, la concesión de autorización a los usuarios para el proceso a los datos y la especificación de personas autorizadas para efectuar ciertas operaciones, del desempeño de la misma, del respaldo de la información.
- **Transacción:** Es una secuencia de operaciones que deben ser completadas totalmente o de lo contrario no se hace ninguna, la transacción no es meramente una protección contra errores, es una rutina que regresa todo a su estado anterior de los elementos involucrados al detectar un error lógico.
- **Integridad:** Las restricciones de integridad aseguran que la información no pueda ser borrada y que los datos insertados cumplan con las especificaciones definidas, ya que algunas tablas o entidades dependen de otras haciendo referencia a su información. Entre las restricciones de integridad disponibles se pueden mencionar las siguientes:
  - **Integridad de entidades:** Una entidad es el objeto fundamental que se registrará en la base de datos, la cual está representada por una tabla. Para identificar cada registro en la tabla, ésta debe tener una llave primaria (valor único de identificación del registro) a este requerimiento se le llama restricción de integridad de entidad.
  - **Integridad semántica:** Asegura que los datos de los registros reflejen un valor permitido dentro de un dominio, es decir, dentro de un conjunto de valores para esa columna. Para forzar la integridad semántica, se usan las siguientes restricciones:

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 1 - CONCEPTOS GENERALES

---

- ◆ **Tipo de datos:** Define los tipos de valores que se almacenarán en una columna.
  - ◆ **Valores por omisión:** Son los valores insertados en una columna cuando no se especifica el valor explícito.
  - ◆ **Chequeo de constraint:** Especifica condiciones en la inserción de datos a una columna, cada registro insertado en la tabla debe cumplir esas condiciones.
- **Integridad referencial:** La integridad referencial es una dependencia lógica de una llave foránea sobre una llave primaria. Para preservar la integridad referencial, INFORMIX requiere que, para poder afectar el registro con la llave primaria, primero se deben afectar los registros que dependen del primero a través de una llave foránea.

La integridad referencial permite a los usuarios definir y forzar relaciones entre columnas. Por ejemplo, puede garantizarse mediante una definición de integridad referencial que un registro de una tabla maestra no pueda ser borrado si existe información dependiente de éste registro en la tabla detalle.

- **Normalización:** Es un proceso de depuración de las relaciones y especificaciones de los archivos que forman el esquema, de tal manera que se garantice que las relaciones eviten en lo más posible la redundancia entre las diferentes entidades; Y de esta manera lograr eficacia y eficiencia.

La teoría de la normalización está basada en el concepto de formas normales, se dice que una relación está en una forma normal particular si satisface cierto conjunto específico de restricciones como las que a continuación se detallan.

**Primera forma normal:** Incluye la eliminación de grupos repetidos y la identificación de la llave que define al criterio primario.

**Segunda forma normal:** Asegura que todos los atributos no llaves sean completamente dependientes de la llave del criterio primario.

**Tercera forma normal:** Elimina cualquier dependencia transitoria, una dependencia transitoria es aquella en la cual sus atributos no llaves son dependientes de otros atributos no-llave.

**Cuarta forma normal:** No se cumple con esta forma normal cuando una relación contiene dos o más datos multivaluados de una entidad.

**Quinta forma normal:** Establece que un registro no puede estar estructurado de tal forma que sus contenidos se puedan reconstruir a partir de registros más pequeños. Por supuesto siempre es posible reestructurar un registro en otros más

pequeños con el mismo campo clave. Este caso es una excepción de la restricción anterior.

Las formas normales se aplican secuencialmente, sobre un diseño de base de datos, primero debe comprobarse la primera, luego la segunda, seguida de la tercera, y así sucesivamente.

Las relaciones normalizadas tienden a tener un pequeño número de atributos. De hecho, si se siguen cuidadosamente todas las reglas de normalización, probablemente el diseñador acabará con muchas relaciones binarias. Los teóricos hacen un gran énfasis en procesos de normalización como herramienta para el diseño de una buena base de datos. El hecho es que la *sobre normalización* del diseño puede degradar el rendimiento. La normalización debería utilizarse sólo como una de las herramientas para el diseño de bases de datos. El diseñador debe familiarizarse con el sistema y las necesidades de los usuarios finales. Las decisiones de agrupar campos deben tener en cuenta las necesidades de los usuarios. Las bases de datos estáticas, que no están sujetas a muchas actualizaciones, empeoran su rendimiento si están *sobre normalizadas*. Sólo se puede conocer el hecho de que una base de datos sea estática hablando con los usuarios finales.

La normalización, si se aplica ciegamente, consigue la eliminación de los datos redundantes. Normalmente la redundancia de los datos no es deseable.

### 1.3 Diagrama entidad-relación

Entre las estrategias de diseño se utiliza el diagrama de entidad-relación u otras formas de diagramas de estructura de datos como punto de partida para obtener el diagrama de estructura apropiado.

Todos los sistemas almacenan y usan información acerca del ambiente en el cual interactúan; a veces, la información es mínima, pero en la mayoría de los sistemas actuales es bastante compleja. No sólo deseamos conocer en detalle que información hay en cada agregado de datos, sino que también queremos conocer la relación que existe entre agregados. Este aspecto del sistema no es resaltado por el diagrama de flujo de datos, pero si lo hace el diagrama de entidad-relación.

El diagrama de entidad-relación consta de cuatro elementos principales:

- **Entidades:** Se representan por medio de un rectángulo en el diagrama. Esto representa una colección o conjunto de objetos (cosas) del mundo real cuyos miembros juegan algún papel en el desarrollo del sistema.
- **Atributos:** Representan las características de los objetos que participan en el modelo de datos, generalmente columnas de tablas.

- **Relaciones:** Son líneas que representan la asociación que existe entre dos o más entidades de un modelo de datos. Una relación puede a su vez llegar a ser una entidad misma, por lo que incluso puede tener atributos.
- **Ligas:** Son líneas que conectan a las relaciones con sus entidades y estas con sus atributos y sirven para representar la cardinalidad de las relaciones.

#### **1.4 Diagrama de flujo de control**

En un diagrama de flujo de datos se describen simplemente los conductores a lo largo donde viajan los paquetes de datos entre procesos y almacenes. Similarmente, las burbujas en los DFD pudieran considerarse como procesadores de datos. Para una amplia clase de sistemas, sobre todo de negocios, existen sólo dos tipos de flujos necesarios en el modelo del sistema. Pero para otra clase de sistemas, los de tiempo real, necesitamos alguna manera de modelar flujos de control. Y se requiere una manera de mostrar procesos de control (esto es, burbujas cuya única labor es de coordinar y sincronizar las actividades de otras burbujas del DFD)

Un flujo de control puede imaginarse como un conducto que porta una señal binaria (esto es, está encendido o está apagado), el flujo de control se manda de un proceso a otro (o de un terminador externo a un proceso)

Un proceso de control puede considerarse como una burbuja supervisora o ejecutiva, cuya labor es coordinar las actividades de otras burbujas en el diagrama; sus entradas y salidas consisten sólo de flujos de control. Los flujos de control salientes del proceso de control se utilizan para despertar a otras burbujas; los flujos de control entrantes generalmente indican que una de las burbujas ha terminado su labor o que se ha presentado alguna situación extraordinaria, de la cual necesita informarse a la burbuja de control. Por lo tanto sólo hay un proceso de control de estos en un DFD dado.

Sin embargo, el comportamiento interno de un proceso de control es diferente; aquí es donde el comportamiento dependiente del tiempo del sistema se modela con detalle. El interior del proceso de control se modela con un diagrama de transición de estados, que muestra los varios estados en los que se puede encontrar todo el sistema y las circunstancias que lo llevan a cambiar de estado.

#### **1.5 Especificación de procesos**

La especificación de procesos, es la descripción de qué es lo que sucede en cada burbuja primitiva de nivel más bajo en un DFD. La especificación de procesos define lo que debe hacerse para transformar entradas en salidas. Es una descripción detallada de la política de negocios del usuario que cada burbuja lleva a cabo. Existe una variedad de herramientas que se pueden utilizar para la especificación de procesos: tablas de decisiones, lenguaje estructurado, pre/pos condiciones, diagramas de flujo, diagramas de Nassi/Shneiderman, etc. Es importante tener presente que en cualquiera de estas herramientas se deben considerar dos requerimientos cruciales:

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 1 - CONCEPTOS GENERALES

---

- La especificación del proceso debe expresarse de una manera que puedan verificar tanto el usuario como el analista. Precisamente por esta razón se evita utilizar el lenguaje narrativo como herramienta de especificación.
- El proceso debe especificarse en una forma que pueda ser comunicada efectivamente al público amplio que este involucrado. A pesar de que el analista es típicamente quien describe la especificación del proceso, habitualmente será un público bastante diverso de usuarios, administradores, auditores, personal de control de calidad y otros, el que leerá la especificación del proceso. Una especificación pudiera expresarse tal vez con un cálculo de predicados, o en Pascal, o en enfoque de diagramación formal.

La mayoría de los analistas se inclinan por utilizar el lenguaje estructurado como método favorito para escribir especificación de procesos, pero también se puede utilizar una combinación de herramientas de especificación.

Como conclusión de una especificación de procesos se dice que representan la mayor parte del trabajo detallado que se tiene en la construcción de un modelo de sistemas; también la actividad de escribir especificaciones de proceso sirve como "prueba de cordura" para los DFD que ya se hallan desarrollado.

Así, podría descubrirse que la especificación del proceso requiere flujos de datos de entrada o salida adicionales (es decir flujos que no aparecieron en el DFD)

Por lo tanto se pueden esperar cambios, revisiones y correcciones del modelo de DFD, basadas en el trabajo detallado de la escritura de las especificaciones del proceso.

## CAPITULO 2 MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS

### 2.1 Replicación continua de datos

El sistema SIIC, cuenta con una base de datos central y 77 bases de datos remotas, las cuales se actualizan entre sí, a través del modulo de replicación "CDR" (Continuos Data Replication), funcionalidad de la base de datos INFORMIX, funcionalidad disponible a partir de la versión 7.24, la cual consiste en que la información se mantiene sincronizada entre el servidor central y los 77 servidores remotos.

La sincronía en el sistema SIIC se definió al nivel de tabla, por medio de la definición de replicas; por cada tabla se define una replica, la replica puede definirse al nivel de toda la tabla o en un campo específico de la tabla.

En el sistema SIIC se definió la replica al nivel de tabla, esto implica que si se inserta, modifica, borra, un registro de la tabla, la transacción se replicara en la tabla "gemela" en la base de datos remota.

Una de las condiciones es que la tabla a replicar, tenga la misma estructura en ambas bases de datos, (central y remota). No se puede modificar la estructura de la tabla mientras tenga definida la replicación entre ambas.

La gran funcionalidad de la replicación es que si el enlace de red (TCP/IP) entre ambas bases de datos remotas, se pierde por un periodo de tiempo, minutos u horas, las transacciones independientemente realizadas en la tabla central y/o remota, efectuadas en ese lapso de tiempo, no se pierden, pues se acumulan en áreas temporales o colas de envío/recepción.

Posteriormente cuando se recupere el enlace de red, las transacciones pendientes en esa tabla, se aplicaran secuencialmente, ya que cada transacción lleva asociada un timestamp, lo cual permite al servidor de base de datos aplicarlas en la secuencia correcta.

Lo anterior es la funcionalidad más importante del CDR, ya que no existen perdidas de transacciones entre ambas bases de datos (central y remota).

Una de las condiciones para la replicación por CDR, es que la fecha y hora de ambos equipos UNIX (central y remoto) estén sincronizados, esto se logra por medio del comando UNIX `rdate`, el cual se ejecuta cada minuto desde cada uno de los 77 servidores remotos, para que el servidor central le indique cual es la hora central o de referencia.

Para los catálogos del sistema SIIC que se actualizan en el servidor central, la replica definida en cada catalogo y/o tabla, es unidireccional, esto es cualquier modificación a la información de este catalogo, se replicara del servidor central a cada uno de los servidores remotos en forma unidireccional.

Las transacciones generadas para replicación tomaran algunos segundos en aplicarse en los servidores remotos, en promedio el tiempo es menor a 4 segundos, con lo cual la información contenida en el sistema SIIC, se considera que esta "en línea" en todo momento, esto asegura que todas las consultas y reportes generados por los usuarios, sea con información reciente, fresca, actualizada.

En el ámbito de los sistemas de información, lo anterior es muy importante, pues toda la información, esta actualizada. Con lo anterior se evita tener que realizar procesos "Batch" entre las bases de datos central y remotas para sincronizar la información entre

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS**

---

ambas. Los famosos cierres mensuales en el sistema SIIC se evitan, pues al final de cada día se generan reportes que permiten verificar que los estados financieros de egresos vs. ingresos sean correctos.

Todo el monitoreo de los diferentes parámetros del CDR, que se comentan en este capítulo, no los realizan los diferentes productos-programa comerciales de monitoreo, tales como Patrol de BMC, TIVOLI de IBM, esta fue una de las principales razones por las cuales se decidió elaborar un módulo de monitoreo propio. Las razones por las cuales los productos-programa mencionados, no la tienen implementada, son principalmente que es una funcionalidad "nueva" de las bases de datos INFORMIX, la replicación CDR es una funcionalidad que tiene aun pocos usuarios y/o instalaciones a escala mundial.

Lo anterior no motiva aun a las compañías desarrolladoras de Software de monitoreo de bases de datos a incluir el monitoreo del CDR en sus nuevas versiones.

De acuerdo a información proporcionada por la compañía IBM, en el sistema SIIC se tiene la instalación más grande de replicación CDR en el mundo, con 1 servidor central y 77 servidores remotos, con 4928 replicas de datos entre ellas.

## 2.2 Definición del servidor central y los 77 servidores remotos

Se tienen definidos un servidor central llamado prod\_300 y 77 servidores remotos llamados prod\_6XX, los cuales están instalados en las 77 Terminales de Almacenamiento y Distribución, ubicadas en las 77 principales ciudades del país.

La instrucción para definir el servidor central es:

```
cdr def serv -c prod_300 -i 0 -s rlcdr -r rlcdr -I -A /siic_onbar2/ats  
-R /siic_onbar2/ris prod_300
```

La instrucción para definir cada uno de los 77 servidores remotos es:

```
cdr def serv -c prod_$1 -i 0 -s cdrproddb -r cdrproddb  
-A /usr/informix/ats -R /usr/informix/ris -I -S prod_300  
-L prod_$1
```

Donde \$1 varía de 601 al 699, el cual es el rango asignado a las 77 bases de datos remotas, mediante el cual están organizadas administrativamente, por zonas regionales.



### 2.3 Monitoreo del CDR en servidor central y sus 77 servidores remotos

El monitoreo para comprobar el estado de la conexión entre el servidor central y sus 77 servidores remotos, nos permite verificar que bases de datos remotas están conectadas, las bases de datos remotas deben de estar permanente conectadas a la base de datos central, por lo que si alguna de las bases de datos remotas esta "desconectada", los administradores del sistema necesitan estar informados de este evento, para revisar las causas, las causas pueden ser varias:

- Problemas de red MAN
  - Lan-Switch apagado
  - Red congestionada o "muy lenta"
- Equipo UNIX remoto "Off-line" o "apagado"
- Base de datos remota "Off-line" o "apagada"
- Base de datos remota con problemas

El comando manual para comprobar el estado de conexión de los servidores remotos hacia el servidor central es:

**cdr list server**

SERVER	ID	STATE	STATUS	CONNECTION	CHANGED
prod_300	300	Active	Local		
prod_602	602	Active	Connected	Apr 24 11:07:14	2006
prod_604	604	Active	Connected	Apr 24 11:07:14	2006
prod_605	605	Active	Connected	Apr 24 11:07:14	2006
prod_606	606	Active	Connected	Apr 24 11:07:14	2006
prod_607	607	Active	Connected	Apr 24 11:07:15	2006
prod_608	608	Active	Connected	Apr 24 11:07:14	2006
prod_609	609	Active	Connected	Apr 24 11:07:14	2006
prod_611	611	Active	<b>Connecting</b>	Apr 24 11:07:15	2006
prod_612	612	Active	Connected	Apr 24 11:07:16	2006
prod_613	613	Active	Connected	Apr 24 11:07:14	2006
prod_614	614	Active	Connected	Apr 24 11:07:15	2006
prod_615	615	Active	Connected	Apr 24 11:07:16	2006
prod_617	617	Active	<b>Connecting</b>	Apr 24 11:07:15	2006
prod_619	619	Active	Connected	Apr 24 11:07:14	2006
prod_620	620	Active	Connected	Apr 24 11:07:14	2006
prod_621	621	Active	Connected	Apr 24 11:07:15	2006
prod_622	622	Active	Connected	Apr 24 11:07:15	2006
prod_623	623	Active	Connected	Apr 24 11:07:15	2006
prod_624	624	Active	Connected	Apr 24 11:07:15	2006
prod_625	625	Active	Connected	Apr 24 11:07:14	2006
prod_627	627	Active	Connected	Apr 24 11:07:15	2006
prod_628	628	Active	Connected	Apr 24 11:07:15	2006
prod_629	629	Active	Connected	Apr 24 11:07:15	2006
prod_630	630	Active	Connected	Apr 24 11:07:14	2006
prod_631	631	Active	Connected	Apr 24 11:07:15	2006

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS**

---

prod_632	632	Active	Connected	Apr 24 11:07:15	2006
prod_633	633	Active	Connected	Apr 24 11:07:15	2006
prod_636	636	Active	Connected	Apr 24 11:07:15	2006
prod_637	637	Active	Connected	Apr 24 11:07:15	2006
prod_638	638	Active	Connected	Apr 24 11:07:15	2006
prod_639	639	Active	Connected	Apr 24 11:07:15	2006
prod_640	640	Active	Connected	Apr 24 11:07:15	2006
prod_641	641	Active	Connected	Apr 24 11:07:15	2006
prod_644	644	Active	Connected	Apr 24 11:07:15	2006
prod_645	645	Active	Connected	Apr 24 11:07:15	2006
prod_646	646	Active	Connected	Apr 24 11:07:15	2006
prod_647	647	Active	Connected	Apr 24 11:07:15	2006
prod_648	648	Active	Connected	Apr 24 11:07:15	2006
prod_649	649	Active	Connected	Apr 24 11:07:15	2006
prod_650	650	Active	Connected	Apr 24 11:07:15	2006
prod_652	652	Active	Connected	Apr 24 11:07:14	2006
prod_654	654	Active	Connected	Apr 24 11:07:14	2006
prod_655	655	Active	Connected	Apr 24 11:07:15	2006
prod_656	656	Active	Connected	Apr 24 11:07:14	2006
prod_657	657	Active	Connected	Apr 24 11:07:14	2006
prod_658	658	Active	Connected	Apr 24 11:07:14	2006
prod_659	659	Active	Connected	Apr 24 11:07:15	2006
prod_660	660	Active	Connected	Apr 24 11:07:15	2006
prod_661	661	Active	Connected	Apr 24 11:07:14	2006
prod_663	663	Active	Connected	Apr 24 11:07:14	2006
prod_664	664	Active	Connected	Apr 24 11:07:14	2006
prod_665	665	Active	Connected	Apr 24 11:07:15	2006
prod_666	666	Active	Connected	Apr 24 19:27:54	2006
prod_667	667	Active	Connected	Apr 24 11:07:14	2006
prod_668	668	Active	Connected	Apr 24 11:07:16	2006
prod_669	669	Active	Connected	Apr 24 11:07:16	2006
prod_672	672	Active	Connected	Apr 24 11:07:15	2006
prod_673	673	Active	Connected	Apr 24 11:07:14	2006
prod_674	674	Active	Connected	Apr 24 11:07:14	2006
prod_675	675	Active	Connected	Apr 24 11:07:49	2006
prod_676	676	Active	Connected	Apr 24 11:07:14	2006
prod_677	677	Active	Connected	Apr 24 16:01:53	2006
prod_678	678	Active	Connected	Apr 24 11:07:14	2006
prod_681	681	Active	Connected	Apr 24 11:07:15	2006
prod_682	682	Active	Connected	Apr 24 11:07:15	2006
prod_683	683	Active	Connected	Apr 24 11:07:15	2006
prod_684	684	Active	Connected	Apr 24 11:07:15	2006
prod_685	685	Active	Connected	Apr 24 11:07:14	2006
prod_688	688	Active	Connected	Apr 24 11:07:15	2006
prod_689	689	Active	Connected	Apr 24 11:07:15	2006
prod_690	690	Active	Connected	Apr 24 11:07:15	2006
prod_693	693	Active	Connected	Apr 24 11:07:15	2006
prod_695	695	Active	Connected	Apr 24 11:07:14	2006
prod_696	696	Active	Connected	Apr 24 11:10:06	2006
prod_697	697	Active	Connected	Apr 24 11:07:14	2006
prod_698	698	Active	Connected	Apr 24 11:10:09	2006
prod_699	699	Active	Connected	Apr 24 11:07:15	2006

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS

---

Del listado anterior, podemos observar que el estado normal para cualquier servidor remoto debe de ser "Active" y "Connected", sin embargo puede suceder lo siguiente:

```
prod_611          611 Active   Connecting  Apr 24 11:07:15 2006
```

Podemos observar que el estado de conexión para el servidor remoto prod\_611, esta en un estado de "connecting", si este estado persiste por mas de 60 minutos, los administradores del sistema deben de revisar las causas, de acuerdo a los puntos enunciados al inicio de este sub capitulo

Si después de revisar las causas y tratar de corregirlas, no se tiene éxito, lo más probable es que el servidor central ya no intente reconectarse nuevamente al servidor remoto, lo anterior significa problemas en la operación del sistema SIIC en ese centro de trabajo remoto, pues la información que debe de replicarse, ya no se actualizara entre ambos servidores, por lo cual los usuarios ya no contarán con su información actualizada, mientras no se corrija el problema.

La solución para este tipo de evento, es "tirar" y "levantar" el servidor de base de datos central del sistema SIIC. Si esto ocurre en cualquier día de la semana, se realizara la actividad a las 21:30 horas, mientras las replicas se encuentran pendientes de envío, en las colas locales del modulo CDR, en la cual se tiene una ventana de tiempo, ya que se realiza el cambio de turno en la operación de los 77 centros de trabajo foráneos o remotos.

Antes de actualizar a la versión **7.31.UD1X5** de la base de datos INFORMIX, este tipo de evento era muy común y muy problemático para los administradores y para los usuarios del sistema SIIC, pues se presentaba 2 veces en promedio por semana. Al instalar la versión **7.31.UD1X5**, este tipo de evento se redujo sustancialmente pues ya no se presenta en semanas, aun así es un problema latente, por lo cual los administradores necesitamos estar informados cuando se presente.

Para automatizar el monitoreo de este tipo de eventos y ser notificados oportunamente los administradores, ya sea por correo electrónico ó teléfono móvil, sé decidió realizar programas en lenguaje "Shell", los cuales se muestran a continuación:

Se programo la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
30 7,12,17,21,0 * * * /siic_onbar2/monitoreo/connecting/conn2.sh 1>/tmp/en_connec.bit 2>&1
```

El programa Shell conn.sh, monitorea si algún centro remoto se queda su conexión en estado "**connecting**", si ese es el caso enviara un aviso mediante correo electrónico a los administradores del sistema SIIC, el mismo se ejecuta en el minuto 30 de las 7,12,17,21 y 24 horas, su contenido es el siguiente:

```
==> cat /siic_onbar2/monitoreo/connecting/conn.sh
export DBDATE=dmy4/
export INFORMIXDIR=/informix
export INFORMIXSERVER=prod_300shm
export INFORMIXSQLHOSTS=/informix/etc/sqlhosts
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS

---

```
export INFORMIXTERMCAP=TERMCAP
export ONCONFIG=prod_300shm
export PATH=$INFORMIXDIR/bin:/usr/bin

cd /siic_onbar2/monitoreo/connecting
#date                               1>>      connecting.bit 2>&1
cdr list server                      1> /tmp/connecting.bit 2>&1
hay=`grep -ic connecting /tmp/connecting.bit`
#echo "-----" 1>>      connecting.bit 2>&1
if [ $hay -gt 0 ]
then
    grep -i connecting /tmp/connecting.bit 1>f1
    sleep 420
    cdr list server                  1> /tmp/connecting.bit 2>&1
    hay1=`grep -ic connecting /tmp/connecting.bit`
    if [ $hay1 -gt 0 ]
    then
        grep -i connecting /tmp/connecting.bit >f2
        diff f1 f2>/dev/null
        if [ $? -eq 0 ]
        then
            date                       1>>      connecting.bit 2>&1
            grep -i connecting /tmp/connecting.bit 1 >>
                connecting.bit
            echo "Hay centros en Connecting, `date +%d/%b-%H:%M` "
                >/tmp/pager_conn_asunto.txt
            echo "Verificar: `cat f2`" >/tmp/pager_conn_mensaje.txt

            rcp /tmp/pager_conn_asunto.txt sun_correo:/tmp
            rcp /tmp/pager_conn_mensaje.txt sun_correo:/tmp
        fi
    fi
fi
else
    echo "No hay centros en Connecting en este momento, `date` "
fi
```

El programa Shell `conn.sh`, a través del programa `rcp` del sistema operativo UNIX envía a el equipo UNIX "sun\_correo", el archivo con el texto, con la alerta respectiva. El servicio de correo en el servidor central del SIIC, se encuentra deshabilitado por cuestiones de seguridad.

Por lo que el equipo UNIX "sun\_correo" proporciona el servicio de correo hacia la Intranet (correo electrónico) e Internet (localizadores, teléfonos móviles, etc.).

## 2.4 Estado del modulo de replicación de datos

La facilidad de replicación continua de datos (CDR) de la base de datos INFORMIX esta muy ligada al área de LOG de la misma base, por lo cual cuando los apuntadores a esos LOG's se corrompen, todo el modulo CDR queda deshabilitado, esto es una condición muy grave, pues afecta la replicación y/o actualización de la información entre todos los servidores del sistema SIIC, lo cual afecta la facturación por la venta de los productos petrolíferos en todo el país, es decir se detiene la operación del mas prioritario de los objetivos del sistema SIIC.

Motivo por el cual es uno de los eventos a monitorear, manualmente la instrucción para monitorearlo es la siguiente:

```
> onstat -g ddr
```

```
Informix Dynamic Server Version 7.31.UD7W4 -- On-Line -- Up 2 days 12:43:33 --  
1265688 Kbytes
```

```
DDR -- Running --
```

# Event	<b>Snoopy</b>	Snoopy	<b>Replay</b>	Replay	<b>Current</b>	Current
Buffers	ID	Position	ID	Position	ID	Position
1024	21743	410b078	21743	3ee1018	21743	410c000

```
DDR events queue
```

```
Type TX id Partnum Row id
```

De lo anterior lo mas importante es que el modulo DDR este en estado "**Running**", el numero de LOG, tanto del proceso Snoopy y Replay deben de estar el numero de LOG "Current", esto es deben de estar en el ultimo LOG usado. No se deben de atrasar, pues de lo contrario se podría generar problemas en el modulo de CDR.

Al restaurar por alguna razón la base de datos, es cuando se puede tener problemas de este tipo en particular. También cuando existen varios servidores remotos "desconectados", lo cual se debe de evitar, todos los servidores remotos deben de estar siempre "conectados".

Para automatizar el monitoreo de este tipo de eventos y ser notificados oportunamente los administradores, ya sea por correo electrónico ó teléfono móvil, sé decidió realizar programas en lenguaje "Shell", los cuales se muestran a continuación:

Se programo la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
0,15,30,45 * * * * /informix/envia_pagers/pager_ddr_down.sh  
1>/tmp/pager_ddr_down.bit 2>&1
```

El programa Shell **pager\_ddr\_down.sh**, monitorea si el modulo de replicación esta en estado "**down**", si ese es el caso enviara un aviso mediante correo electrónico a los

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS

---

administradores del sistema SIIC, el mismo se ejecuta en el minuto 15, de todas las horas, su contenido es el siguiente:

```
==> cat /informix/envia_pagers/pager_dds_down.sh
#####INFORMIX#####
export PATH=/usr/bin:/informix/bin
export INFORMIXDIR=/informix
export INFORMIXTERM=termcap
export TERM=vt100
export INFORMIXSERVER=prod_300shm
export ONCONFIG=prod_300shm
export ONCONFIG
export DBDATE=dmy4/
#####INFORMIX#####
#
> /tmp/pager_dds_down_asunto.txt
> /tmp/pager_dds_down_mensaje.txt

echo "SIIC DDR OK `date +%d/%b-%H:%M`" > /tmp/pager_dds_down_asunto.txt
echo "SIIC DDR OK " > /tmp/pager_dds_down_mensaje.txt

IS_RUNNING=`onstat -g ddr|grep -ci running`
if [ $IS_RUNNING -eq 0 ]
then
echo "ERROR: DDR Down">/tmp/pager_dds_down_asunto.txt
echo "Revisar ERROR: DDR Down" > /tmp/pager_dds_down_mensaje.txt
onstat -g ddr >> /tmp/pager_dds_down_mensaje.txt
fi
#
# DETECTA CUANDO HAY REPLICAS DETENIDAS EN DDR
DDR_STOP=`onstat -g ddr|wc -l`
if [ $DDR_STOP -gt 16 ]
then
echo "SIIC DDR HAY $DDR_STOP REPLICAS DETENIDAS `date +%d/%b-%H:%M`" \
>> /tmp/pager_dds_down_asunto.txt
echo "Revisar DDR hay $DDR_STOP replicas detenidas" \
>> /tmp/pager_dds_down_mensaje.txt
onstat -g ddr >> /tmp/pager_dds_down_mensaje.txt
fi
#
# DETECTA CUANDO UN CENTRO ESTA TRATANDO DE CONECTARSE POR SHM.
#
log="/informix/prod_300shm.log"
IN_SHM=`tail -50 $log|grep -c "conerr=-25555"`
if [ $IN_SHM -gt 0 ]
then
echo "Hay $IN_SHM intentos de conectarse a SHM">
/tmp/pager_dds_down_asunto
.txt
cto=`tail -75 $log|grep 'conerr=-25555'|head -1|awk -F" " '{ print $8 }'`
echo "Centro `echo $cto|cut -c2-12` trata de conectarse a SHM">
/tmp/pager_
dds_down_mensaje.txt
tail -75 $log|grep "conerr=-25555"|head -1>>
/tmp/pager_dds_down_mensaje.tx
t
fi
#
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
***CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS***

---

```
DIA=`date +%u`
export DIA
HORA=`date +%H`
export HORA
MINUTOS=`date +%M`
export MINUTOS
#
if [ $IS_RUNNING -eq 0 ] || [ $DDR_STOP -gt 16 ] || [ $IN_SHM -gt 0 ]
then
    rcp /tmp/pager_ddr_down_asunto.txt sun_correo:/tmp
    rcp /tmp/pager_ddr_down_mensaje.txt sun_correo:/tmp
fi
```

## 2.5 Monitoreo de las replicas de datos inconsistentes

Mediante la definición de las replicas se establece que tablas son las que estarán en sincronía en ambos servidores, central y remotos, la replicación es al nivel de tabla, cuando se inserte, modifique o se de baja un registro, este se replicara y/o actualizará en su tabla "gemela" en los servidores remotos.

Los catálogos básicamente serán modificados desde el servidor central y se replicaran en las 77 bases de datos remotas en la tabla correspondiente. Las replicas se definen de acuerdo a los procesos que se automaticen en el sistema SIIC.

La definición de la replica del catálogo de centros de trabajo, (es una replica **unidireccional**, únicamente se actualiza la información desde el servidor central a los 77 servidores remotos), es de acuerdo a lo siguiente:

```
cdr define repl -C ignore -A -R -S trans r_centro$1 \  
"P sis_com@prod_300:informix.sccat_cto" "select * from sccat_cto" \  
"R sis_com@$2:informix.sccat_cto" "select * from sccat_cto"
```

Analizando la instrucción anterior:

El parámetro **\$1** es el identificador de la terminal de almacenamiento

El parámetro **\$2** es el nombre del servidor remoto de la terminal de almacenamiento.

El servidor primario es el servidor central prod\_300

El servidor secundario o remoto es el prod\_XXX, donde XXX varía del 600 al 699

El nombre de la base de datos es sis\_com en ambos servidores

El nombre de la replica es **r\_centroXXX**, donde **XXX** varia del 600 al 699

Se tiene que definir una replica por cada centro remoto a sincronizar, para cada tabla. Para el catalogo de centro **sccat\_cto**, se repite la definición de la replica 77 veces, para las 77 terminales de almacenamiento y distribución.

La tabla origen y destino tienen el mismo nombre, pudieran tener diferente nombre, pero la estructura de la misma, tiene que ser igual.

La opción de **-C ignore** nos indica que las transacciones se aplicaran en orden riguroso de acuerdo a como lleguen al servidor destino.

La opción **-A**, en que en caso de error se genera un archivo **ATS** en el directorio /informix/ats, el cual ayuda a le personal del área de "control de la producción" a determinar cual fue el error, un error común es que el registro a modificar este bloqueado por otra transacción.

La opción **-R**, en que en caso de error se genera un archivo **RIS** en el directorio /informix/ris, el cual ayuda a le personal del área de "control de la producción" a determinar cual fue el error, este archivo contiene todo el detalle de la transacción rechazada, la cual puede afectar a 1 registro o muchos.

La opción **-S trans** indica al servidor que la replicación esta basándose en transacciones, esto es una instrucción SQL, la cual puede afectar a 1 registro o muchos.



## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS

---

Lo anterior implica que los servidores, central y remotos tengan activada el **LOG**, el cual es un área especial del servidor de base de datos en la cual se van registrando todas las transacciones realizadas. En el sistema SIIC se tiene activado el LOG (10 GB) de tipo **Unbufferedd**, en el cual cada transacción realizada se escribe inmediatamente a disco, con lo cual si hay una caída del servidor, la pérdida de transacciones en memoria es mínima, aunque el I/O a disco es mayor. El uso de LOG en un sistema es un buen indicador del número de transacciones generadas.

La definición de una replica **bidireccional**, (esto es que si la información es alterada en una tabla, se actualiza en su contraparte, no importa si quien genera la transacción es el servidor central o remoto), es de acuerdo a lo siguiente:

```
cdr define repl -C timestamp -A -R -S trans r_prog_entr$1 \  
"sis_com@prod_300:informix.scmae_prog_entr" "select * from \  
scmae_prog_entr where icto=$3" \  
"sis_com@$2:informix.scmae_prog_entr" "select * from scmae_prog_entr \  
where icto=$3"
```

Analizando únicamente las diferencias de la instrucción anterior:

No hay servidor primario y secundario para este tipo de replica, los dos pueden originar la transacción y su posterior replicación y/o actualización.

La opción de **-C timestamp** nos indica que las transacciones se aplicaran en orden riguroso de acuerdo a la fecha y hora en que se genero la transacción.

La instrucción contiene un **where**, con lo cual nos permite ser aun más selectivos, y afectar únicamente los datos que correspondan, esto es una gran facilidad técnica del CDR y una de las razones técnicas para seleccionar este producto.

Para monitorear inconsistencias en el estado de las replicas, es mediante una sentencia SQL, el resultado debe ser 0 registros encontrados, de lo contrario las replicas desplegadas seguramente tienen problemas en la replicación de la información, en este caso en específico se deben de borrar las definiciones de las replicas con problema, y volverlas a crearlas.

La instrucción es la siguiente:

```
echo "  
database syscdr ;  
select repname  
  from repdef  
  where not exists (select repid from pendingstates where id=repid)  
                    order by repname ;  
                    " > /tmp/repl_inc.sql  
dbaccess -e sis_com /tmp/repl_inc.sql
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS

---

Para automatizar el monitoreo de este tipo de eventos y ser notificados oportunamente los administradores, ya sea por correo electrónico ó teléfono móvil, sé decidió realizar programas en lenguaje "Shell", los cuales se muestran a continuación:

Se programo la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
47 * * * * /informix/repl_no_ok/repl_no_ok.sh 1>/tmp/repl_no_ok.bit 2>&1
```

El programa Shell **repl\_no\_ok.sh**, monitorea si alguna replica no se encuentra registrada en la tabla "pendigstates", si ese es el caso enviara un aviso mediante correo electrónico a los administradores del sistema SIIC, el mismo se ejecuta en el minuto 47 de cada hora, su contenido es el siguiente:

```
==> cat /informix/repl_no_ok/repl_no_ok.sh
#####
#
#####
#
EQUIPO=`uname -a | awk -F" " ' { printf "%s", $2 } ' `
export EQUIPO
#
HORA=`date +%H`
export HORA
#
SERVIDORES="prod_300shm" # UN SOLO SERVIDOR
export SERVIDORES
#
cd /informix/repl_no_ok
#
rm /tmp/pager*_chk-rpt-$EQUIPO.lst > repl_no_ok.lst
#
for i in `echo $SERVIDORES`
do
#
#
INFORMIXSERVER=$i
export INFORMIXSERVER
#
INFORMIXDIR=/informix
export INFORMIXDIR
INFORMIXSQLHOSTS=$INFORMIXDIR/etc/sqlhosts
export INFORMIXSQLHOSTS
PATH=$INFORMIXDIR/bin:/usr/bin
export PATH
#
$INFORMIXDIR/bin/saceprep repl_no_ok
$INFORMIXDIR/bin/sacego repl_no_ok $i $EQUIPO
echo "$EQUIPO REPLICAS OK" > /tmp/pager_asunto_chk-rpt-$EQUIPO.lst
cat repl_no_ok.lst >> /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
#
done
CUANTOS=`grep TOTAL repl_no_ok.lst | awk -F":" ' { printf "%d", $2 } ' `
export CUANTOS
#
HORA=`date +%H`
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS**

---

```
export HORA
#
MINUTOS=`date +%M`
export MINUTOS
#
# EL CRON DEBE DE ESTAR EN EL MINUTO 47
#
if [ $CUANTOS -gt 0 ] || [ "${HORA}" = "12" ] && [ "${MINUTOS}" = "47" ]
then
rcp /tmp/pager_asunto_chk-rpt-$EQUIPO.lst sun_correo:/tmp &
rcp /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst sun_correo:/tmp &
fi
#
```

El contenido del correo es el siguiente, en este caso no se tienen replicas inconsistentes:

PAGINA : 1 Thu 25/07/2006 12:47:01

REPLICAS QUE POSIBLEMENTE NO ESTAN REPLICANDO : prod\_300shm EN EL EQUIPO : siicl

NOMBRE

TOTAL DE REPLICAS CON PROBLEMAS : 0

## 2.6 Monitoreo de las replicas pendientes de envío y/o recepción

El modulo de replicación CDR de la base de datos, cuenta con colas y/o queue's de salida y entrada, en las cuales se van acumulando las transacciones que tienen desde el servidor central hacia un servidor remoto, y las transacciones que vienen de los 77 servidores remotos hacia el servidor central, el estado de estas estructuras de datos se tienen que monitorear continuamente.

No es conveniente que las transacciones se acumulen, una de las razones causantes del problema es que algunos de los centros remotos estén "desconectados" por lapsos de tiempo prolongados, otra de las razones es que se tenga un error interno en el modulo de replicación CDR.

Este tipo de evento es crítico y los administradores del sistema, necesitan estar informados de la anomalía inmediatamente, para tomar las acciones necesarias.

El monitoreo manual de este tipo de evento es con la siguiente instrucción:

```
> onstat -g rqm brief
```

```
Queue trg_send (all totals approximate)
# Txns In Memory: 1
Server prod_605 (605) # txns in queue: 1
Server prod_607 (607) # txns in queue: 1
Server prod_608 (608) # txns in queue: 1
Server prod_614 (614) # txns in queue: 1
Server prod_624 (624) # txns in queue: 1
Server prod_625 (625) # txns in queue: 1
Server prod_627 (627) # txns in queue: 1
Server prod_628 (628) # txns in queue: 1
Server prod_629 (629) # txns in queue: 1
Server prod_630 (630) # txns in queue: 1
Server prod_631 (631) # txns in queue: 1
```

```
Queue trg_receive (all totals approximate)
# Txns In Memory: 3
```

Del resultado anterior se observa que se tienen 11 transacciones en la cola de envío, 3 transacciones en la cola de recepción, el numero total de transacciones pendientes en memoria debe mantenerse al mínimo.

Para automatizar el monitoreo de este tipo de eventos y ser notificados oportunamente los administradores, ya sea por correo electrónico ó teléfono móvil, sé decidió realizar programas en lenguaje "Shell", los cuales se muestran a continuación:

Se programo la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
0,15,30,45 * * * * /informix/envia_pagers/pager_dds_down.sh
1>/tmp/pager_dds_down.bit 2>&1
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS**

---

El programa Shell **pager\_cdr.sh**, monitorea si el modulo de replicación tiene transacciones pendientes en las "colas y/o "queue's", si ese es el caso enviara un aviso mediante correo electrónico a los administradores del sistema SIIC, el mismo se ejecuta cada 15 minutos, de cada hora, su contenido es el siguiente:

```
==> cat /informix/envia_pagers/pager_cdr.sh
##### INFORMIXR #####
export PATH=$PATH:/informix:/informix/bin
export INFORMIXDIR=/informix
#
export INFORMIXTERM=termcap
#
export TERM=vt100
#
export INFORMIXSERVER=prod_300shm
#
export ONCONFIG=prod_300shm
export ONCONFIG
#
export DBDATE=dmy4/
#
#####INFORMIX#####
cont=0
tot=0
until [ $cont -eq 25 ]
do
    onstat -g rqm brief>> OGR.TXT
    cont=`expr $cont + 1`
    # sleep 3
done
    for i in `cat OGR.TXT | awk -F " " ' { print $4 $5 } ' `
    do
#sleep 2
        #echo "datos completos: $i"
        uno=`echo $i | cut -b 1-7`
        dos=`echo $i | cut -b 8-11`
        if [ $uno = "Memory:" ]
        then
            #echo "dos=$dos"
            if [ $dos -ge 2100 ]
            then
                tot=`expr $tot + 1`
                if [ $tot -eq 20 ]
                then
                    clear
                    echo "HAY $dos REPLICAS
                    PENDIENTES">/tmp/pager_cdr_asunto.txt
                    echo "Existen mas de $dos Transacciones en espera:
                    `date +%d/%b-%H:%M` ">/tmp/pager_cdr_mensaje.txt
                    rcp /tmp/pager_cdr_asunto.txt sun_correo:/tmp
                    rcp /tmp/pager_cdr_mensaje.txt sun_correo:/tmp
                    tot=0
                fi
            fi
        fi
    fi
fi
```

MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX  
*CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS*

---

```
        fi
      else
        tot=0
      fi
    fi
  done
rm OGR.TXT
```

## 2.7 Monitoreo de las replicas de datos inactivas

El estado de las replicas, debe de ser normalmente "ACTIVE", los otros 2 estados posibles son "FAILED" e "INACTIVE", cuando se están definiendo las replicas es cuando se pueden presentar replicas con el estado "FAILED", el estado "INACTIVE" se puede presentar cuando se "tira" y "levanta" el servidor de base de datos.

El monitoreo manual del estado de las replicas, es el siguiente

**cdr list repl**

(solo se despliegan las primeras de un total de 4930 replicas definidas)

REPLICATE	STATE	CONFLICT	FREQUENCY	OPTIONS
r_acum_inv602	ACTIVE	timestamp	immediate	transaction,r,s,ats
r_acum_inv604	ACTIVE	timestamp	immediate	transaction,r,s,ats
r_acum_inv605	ACTIVE	timestamp	immediate	transaction,r,s,ats
r_acum_inv606	ACTIVE	timestamp	immediate	transaction,r,s,ats
r_acum_inv607	ACTIVE	timestamp	immediate	transaction,r,s,ats
r_acum_inv608	ACTIVE	timestamp	immediate	transaction,r,s,ats
r_centro612	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro613	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro614	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro615	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro617	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro619	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro620	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro621	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro622	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro623	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro624	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro625	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro627	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro628	ACTIVE	ignore	immediate	transaction,r,s,ats
r_centro629	ACTIVE	ignore	immediate	transaction,r,s,ats

El estado de las replicas debe de ser "ACTIVE", de lo contrario se tiene problemas con el flujo de la información para el servidor remoto afectado.

Para automatizar el monitoreo de este tipo de eventos y ser notificados oportunamente los administradores, ya sea por correo electrónico ó teléfono móvil, se decidió realizar programas en lenguaje "Shell", los cuales se muestran a continuación:

Se programo la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
45 * * * * /informix/envia_pagers/pager_cdr_rep_inac.sh
1>/tmp/pager_cdr_rep_inac.bit 2>&1
```

El programa Shell **pager\_cdr\_rep\_inac.sh**, monitorea si hay replicas en estado diferente a "ACTIVE", si ese es el caso enviara un aviso mediante correo electrónico a los

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 2 - MONITOREO DE LA REPLICACIÓN CONTINUA DE DATOS**

---

administradores del sistema SIIC, el mismo se ejecuta en el minuto 45 de cada hora, su contenido es el siguiente:

```
==> cat pager_cdr_rep_inac.sh
#####INFORMIX#####
#
#####INFORMIX#####
export PATH=/usr/bin:/informix/bin
export INFORMIXDIR=/informix
export INFORMIXTERM=termcap
export TERM=vt100
export INFORMIXSERVER=prod_300shm
export ONCONFIG=prod_300shm
export ONCONFIG
export DBDATE=dmy4/
#####INFORMIX#####
#
> /tmp/pager_cdr_rep_inac_asunto.txt
> /tmp/pager_cdr_rep_inac_mensaje.txt

REP_INA=` cdr list repl |egrep -cvi " active|replicate|\-\-\-\-" `
if [ $REP_INA -gt 0 ]
then
    echo "ERROR: Hay replicas inactivas " >
    /tmp/pager_cdr_rep_inac_asunto.txt
    echo "ERROR: Hay replicas inactivas [$REP_INA] `date +%d/%b-%H:%M`" >
    /tmp/pager_cdr_rep_inac_mensaje.txt
    rcp /tmp/pager_cdr_rep_inac_asunto.txt sun_correo:/tmp
    rcp /tmp/pager_cdr_rep_inac_mensaje.txt sun_correo:/tmp
fi
```



## CAPITULO 3 MONITOREO DEL I/O POR DBSPACE

### 3.1 Accesos por segundo de I/O por dbspace

En los dbspaces se encuentra almacenada la información de la base de datos del sistema SIIC, y un dbspace puede tener más de un chunk, el tamaño máximo de cada uno es de 2 GB, actualmente cuenta con 56 dbspaces y 149 chunks, lo que nos indica que hay varios dbspaces con más de un chunk.

Con el comando onstat -d, podemos obtener el detalle de la distribución de los dbspaces:

```
Informix Dynamic Server Version 7.31.UD7W4 -- On-Line -- Up 16 days 19:20:58 --
1665688 Kbytes
```

Dbspaces							
address	number	flags	fchunk	nchunks	flags	owner	name
c34cd158	1	1	1	2	N	informix	rootdbs
c3f266c8	2	1	2	1	N	informix	rlgran1
c3f26788	3	1	3	1	N	informix	rlgran2
c3f26848	4	1	4	1	N	informix	rlgran3
c3f26908	5	1	5	1	N	informix	rlgran4
c3f269c8	6	1	6	1	N	informix	rlgran5
c3f26a88	7	1	7	1	N	informix	rlgran6
c3f26b48	8	1	8	1	N	informix	rlgran7
c3f26c08	9	1	9	1	N	informix	rlgran8
c3f26cc8	10	1	10	2	N	informix	rlgran9
c3f26d88	11	1	11	1	N	informix	rlgran10
c3f26e48	12	1	12	1	N	informix	rlgran11
c3f26f08	13	1	13	1	N	informix	rlgran12
c3f2b018	14	1	14	1	N	informix	rlgran13
c3f2b0d8	15	1	15	2	N	informix	rlgran14
c3f2b198	16	1	16	2	N	informix	rlgran15
c3f2b258	17	1	17	2	N	informix	rlgran16
c3f2b318	18	1	18	1	N	informix	rlgran17
c3f2b3d8	19	1	19	1	N	informix	rlgran18
c3f2b498	20	1	20	1	N	informix	rlgran19
c3f2b558	21	1	21	1	N	informix	rlgran20
c3f2b618	22	1001	22	9	N	informix	rltab1
c3f2b6d8	23	1	24	1	N	informix	rltab2
c3f2b798	<b>24</b>	1	25	1	N	informix	<b>rltab3</b>
c3f2b858	25	1	26	1	N	informix	rltab4
c3f2b918	26	1	27	2	N	informix	rltab5
c3f2b9d8	27	1	28	1	N	informix	rltab6
c3f2ba98	28	1	29	1	N	informix	rlcdr
c3f2bb58	29	1	30	1	N	informix	rllog1
c3f2bc18	30	1	31	1	N	informix	rllog2
c3f2bcd8	31	2001	32	1	N T	informix	rltemp1
c3f2bd98	32	2001	33	1	N T	informix	rltemp2
c3f2be58	33	1	35	1	N	informix	rllog3
c3f2bf18	34	1	36	1	N	informix	rllog4
c3f2bfd8	35	1	37	1	N	informix	rllog5
c3f2c098	36	1	43	1	N	informix	rlidx1
c3f2c158	37	1	44	1	N	informix	rlidx2
c3f2c218	38	1	45	1	N	informix	rlidx3
c3f2c2d8	39	1	46	1	N	informix	rlidx4
c3f2c398	40	1	47	1	N	informix	rlidx5
c3f2c458	41	1	50	2	N	informix	rlgran23
c3f2c518	42	1	52	1	N	informix	rlgran25

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

c3f2c5d8	43	1	53	1	N	informix rlix6
c3f2c698	44	1	54	1	N	informix rlix7
c3f2c758	45	1	55	1	N	informix rlix8
c3f2c818	46	1	56	1	N	informix rlix9
c3f2c8d8	47	1	58	1	N	informix rlgran26
c3f2c998	48	1	59	1	N	informix rlgran27
c3f2ca58	49	2001	64	1	N T	informix rltemp3
c3f2cb18	50	2001	65	1	N T	informix rltemp4

50 active, 2047 maximum

Chunks

address	chk/dbs	offset	size	free	bpages	flags	pathname
c34cd218	1	1	0	1000000	889679	P0-	/dev/vgrootdbs/rlrootdbs
c34ce9a0	2	2	0	1000000	315993	P0-	/dev/vggrandbs/rlgran1
c34ceaa8	3	3	0	1000000	472656	P0-	/dev/vggrandbs/rlgran2
c34cebb0	4	4	0	1000000	73478	P0-	/dev/vggrandbs/rlgran3
c34cecb8	5	5	0	1000000	390356	P0-	/dev/vggrandbs/rlgran4
c34cedc0	6	6	0	1000000	132404	P0-	/dev/vggrandbs/rlgran5
c34ceec8	7	7	0	1000000	196904	P0-	/dev/vggrandbs/rlgran6
c34f7430	8	8	0	1000000	68972	P0-	/dev/vggrandbs/rlgran7
c34f7538	9	9	0	1000000	484276	P0-	/dev/vggrandbs/rlgran8
c34f7640	10	10	0	1000000	29915	P0-	/dev/vggrandbs/rlgran9
c34f7748	11	11	0	1000000	105148	P0-	/dev/vggrandbs/rlgran10
c34f7850	12	12	0	1000000	310827	P0-	/dev/vggrandbs/rlgran11
c34f7958	13	13	0	1000000	399552	P0-	/dev/vggrandbs/rlgran12
c34f7a60	14	14	0	1000000	390409	P0-	/dev/vggrandbs/rlgran13
c34f7b68	15	15	0	1000000	1126	P0-	/dev/vggrandbs/rlgran14
c34f7c70	16	16	0	1000000	55678	P0-	/dev/vggrandbs/rlgran15
c34f7d78	17	17	0	1000000	3608	P0-	/dev/vggrandbs/rlgran16
c34f7e80	18	18	0	1000000	341379	P0-	/dev/vggrandbs/rlgran17
c34f7f88	19	19	0	1000000	422664	P0-	/dev/vggrandbs/rlgran18
c34f8090	20	20	0	1000000	309328	P0-	/dev/vggrandbs/rlgran19
c34f8198	21	21	0	1000000	333392	P0-	/dev/vggrandbs/rlgran20
c34f82a0	22	22	0	1000000	1294	P0-	/dev/vgtabdds/rltab1
c34f83a8	23	23	0	1000000	9688	P0-	/dev/vgrootdbs/rllee1
c34f84b0	24	23	0	1000000	713391	P0-	/dev/vgtabdds/rltab2
c34f85b8	<b>25</b>	<b>24</b>	0	1000000	121696	P0-	/dev/vgtabdds/ <b>rltab3</b>
c34f86c0	26	25	0	1000000	257181	P0-	/dev/vgtabdds/rltab4
c34f87c8	27	26	0	1000000	52278	P0-	/dev/vgtabdds/rltab5
c34f88d0	28	27	0	1000000	355680	P0-	/dev/vgtabdds/rltab6
c34f89d8	29	28	0	1000000	753477	P0-	/dev/vgrootdbs/rlcdr
c34f8ae0	30	29	0	1000000	7947	P0-	/dev/vgrootdbs/rllog1
c34f8be8	31	30	0	1000000	7947	P0-	/dev/vgrootdbs/rllog2
c34f8cf0	32	31	0	1000000	967037	P0-	/dev/vgrootdbs/rltemp1
c34f8df8	33	32	0	1000000	966301	P0-	/dev/vgrootdbs/rltemp2
c34f8f00	34	22	0	1000000	17198	P0-	/dev/vgtabdds/rltab7
c3504630	35	33	0	1000000	49947	P0-	/dev/vgrootdbs/rllog3
c3504738	36	34	0	1000000	49947	P0-	/dev/vgrootdbs/rllog4
c3504840	37	35	0	1000000	7947	P0-	/dev/vgrootdbs/rllog5
c3504948	38	22	0	1000000	36259	P0-	/dev/vgtabdds/rltab8
c3504a50	39	22	0	1000000	15299	P0-	/dev/vgtabdds/rltab9
c3504b58	40	22	0	1000000	2102	P0-	/dev/vgtabdds/rltab10
c3504c60	41	22	0	1000000	2170	P0-	/dev/vgtabdds/rltab11
c3504d68	42	15	0	1000000	782349	P0-	/dev/vggrandbs/rlgran21
c3504e70	43	36	0	1000000	195844	P0-	/dev/vgidxdbs/rlidx1
c3f25018	44	37	0	1000000	296678	P0-	/dev/vgidxdbs/rlidx2
c3f25120	45	38	0	1000000	219329	P0-	/dev/vgidxdbs/rlidx3
c3f25228	46	39	0	1000000	350602	P0-	/dev/vgidxdbs/rlidx4
c3f25330	47	40	0	1000000	225912	P0-	/dev/vgidxdbs/rlidx5
c3f25438	48	22	0	1000000	826	P0-	/dev/vgtabdds/rltab12
c3f25540	49	17	0	1000000	835261	P0-	/dev/vggrandbs/rlgran22
c3f25648	50	41	0	1000000	93947	P0-	/dev/vggrandbs/rlgran23
c3f25750	51	41	0	1000000	699997	P0-	/dev/vggrandbs/rlgran24
c3f25858	52	42	0	1000000	687893	P0-	/dev/vggrandbs/rlgran25
c3f25960	53	43	0	1000000	242298	P0-	/dev/vgidxdbs/rlidx6
c3f25a68	54	44	0	1000000	300948	P0-	/dev/vgidxdbs/rlidx7

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

c3f25b70	55	45	0	1000000	223517	P0-	/dev/vgidxdbs/rlidx8
c3f25c78	56	46	0	1000000	217083	P0-	/dev/vgidxdbs/rlidx9
c3f25d80	57	22	0	1000000	689204	P0-	/dev/vgtabdbbs/rltab13
c3f25e88	58	47	0	1000000	656830	P0-	/dev/vggrandbs/rlgran26
c3f25f90	59	48	0	1000000	611344	P0-	/dev/vggrandbs/rlgran27
c3f26098	60	26	0	1000000	941545	P0-	/dev/vgtabdbbs/rltab14
c3f261a0	61	10	0	500000	499997	P0-	/dev/vggrandbs/rlgran28
c3f262a8	62	16	500000	500000	499997	P0-	/dev/vggrandbs/rlgran28
c3f263b0	63	1	0	1000000	999725	P0-	/dev/vgrootdbs/rlrootdbs2
c3f264b8	64	49	0	1000000	966813	P0-	/dev/vgrootdbs/rltemp3
c3f265c0	65	50	0	1000000	968348	P0-	/dev/vgrootdbs/rltemp4
65 active, 2047 maximum							

Se tienen dbspaces dedicados, para el área interna de la base de datos, para los datos, para los índices, para los logs, para las áreas temporales de trabajo. En una base de datos de alta demanda, es importante separar los dbspaces según sea la función y/o la información que se almacene en los mismos.

La separación de los dbspaces por función, evitara la contención en el acceso al disco, esto es el I/O por dspace, al definir los grupos de volumen a nivel del sistema operativo, se debe de balancear el uso de cada disco, es decir se deben de separar los grupos que contendrán a los dbspaces, la regla primordial es evitar tener 2 dbspaces de alta demanda en un mismo disco.

Un parámetro importante es el numero de accesos al dspace por segundo, es decir el I/O por dspace por segundo, este es un parámetro que los administradores de la base de datos deben de monitorear continuamente, pues este parámetro afecta en gran medida el desempeño de la base de datos y por lo tanto del sistema SIIC, pues puede haber en un instante dado, decenas de usuarios tratando de acceder a un mismo dspace.

Para monitorear manualmente el I/O de los dbspaces, es con el siguiente comando:

**onstat -g iof**

==> onstat -g iof

```
Informix Dynamic Server Version 7.31.UD7W4 -- On-Line -- Up 16 days 20:24:01 --
1665688 Kbytes
```

AIO global files:

gfd	pathname	totalops	dskread	dskwrite	io/s
3	rlrootdbs	4263	3283	980	3.7
4	rlgran1	11313	11258	55	9.8
5	rlgran2	16913	16864	49	14.6
6	rlgran3	6664	6593	71	5.8
7	rlgran4	8820	8766	54	7.6
8	rlgran5	1995	1942	53	1.7
9	rlgran6	3587	3493	94	3.1
10	rlgran7	5407	5322	85	4.7
11	rlgran8	13699	13580	119	11.9
12	rlgran9	35239	35119	120	30.5
13	rlgran10	32540	32254	286	28.2
14	rlgran11	25691	25579	112	22.2
15	rlgran12	13056	13005	51	11.3
16	rlgran13	51813	51712	101	44.9

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

---

17	rlgran14	53051	53001	50	45.9
18	rlgran15	49363	49303	60	42.7
19	rlgran16	51405	51347	58	44.5
20	rlgran17	628	578	50	0.5
21	rlgran18	5895	5852	43	5.1
22	rlgran19	66789	66688	101	57.8
23	rlgran20	7216	7147	69	6.2
24	rltab1	77891	77471	420	67.4
25	rllee1	27543	26943	600	23.8
26	rltab2	84276	84176	100	73.0
27	<b>rltab3</b>	<b>339148</b>	<b>339007</b>	<b>141</b>	<b>293.6 ←</b>
28	rltab4	38311	38243	68	33.2
29	rltab5	48739	48654	85	42.2
30	rltab6	74449	74338	111	64.5
31	rlcdr	1070	1056	14	0.9
32	rllog1	0	0	0	0.0
33	rllog2	0	0	0	0.0
34	rltemp1	15291	10719	4572	13.2
35	rltemp2	10193	5918	4275	8.8
36	rltab7	22140	20929	1211	19.2
37	rllog3	0	0	0	0.0
38	rllog4	0	0	0	0.0
39	rllog5	12638	1116	11522	10.9
40	rltab8	22954	22200	754	19.9
41	rltab9	5019	4710	309	4.3
42	rltab10	458	325	133	0.4
43	rltab11	8140	7931	209	7.0
44	rlgran21	3772	3747	25	3.3
45	rlidx1	800	644	156	0.7
46	rlidx2	1306	1092	214	1.1
47	rlidx3	8502	8298	204	7.4
48	rlidx4	3340	3087	253	2.9
49	rlidx5	941	753	188	0.8
50	rltab12	9782	9546	236	8.5
51	rlgran22	410	393	17	0.4
52	rlgran23	0	0	0	0.0
53	rlgran24	0	0	0	0.0
54	rlgran25	89561	89537	24	77.5
55	rlidx6	926	894	32	0.8
56	rlidx7	200	135	65	0.2
57	rlidx8	41	33	8	0.0
58	rlidx9	210	162	48	0.2
59	rltab13	19021	18396	625	16.5
60	rlgran26	5556	5503	53	4.8
61	rlgran27	22267	22222	45	19.3
62	rltab14	134	99	35	0.1
63	rlgran28	0	0	0	0.0
64	rlgran28	0	0	0	0.0
65	rlrootdbs2	304	238	66	0.3
66	rltemp3	10868	6184	4684	9.4
67	rltemp4	10194	5711	4483	8.8

En el dbspace rltab3, tenemos un valor de 293.6 accesos por segundo, si este dbspace permanece varias horas con esa demanda de acceso, esto es un potencial "cuello de botella" de acceso a disco.

Para automatizar el monitoreo de este tipo de eventos y ser notificados oportunamente los administradores, ya sea por correo electrónico ó teléfono móvil, sé decidió realizar programas en lenguaje "Shell", los cuales se muestran a continuación:

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

Se programo la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
15 14 * * * /informix/chkiosum/chkiosum.sh 1> /tmp/chkiosum.bit 2>&1
```

El programa Shell **chkiosum.sh**, monitorea el numero de lecturas y escrituras a cada dbspaces 24 horas y genera un reporte sumarizado de la actividad de I/O el cual es enviado mediante correo electrónico a los administradores del sistema SIIC, el mismo se ejecuta a las 14 horas, cada día.

Una notificación típica de este evento es la siguiente:

```
Apr 24 2006 14:15:04 DBSPACES I/O -2- ORDER by writes desc
```

DBS	CHK	P/S	READS	%	WRITES	%	PAG_READS	%	PAG_WRITES	%	FECHA	HORA
rltemp2	33	P	672,453	0.5	452,553	19.4	2,820,299	0.4	2,701,308	22.0	24/04/2006	14:15
rltemp4	65	P	474,915	0.3	288,739	12.4	2,516,874	0.3	2,400,662	19.6	24/04/2006	14:15
rltemp1	32	P	447,385	0.3	276,881	11.9	2,456,289	0.3	2,360,802	19.2	24/04/2006	14:15
rltemp3	64	P	424,960	0.3	267,799	11.5	2,547,120	0.3	2,465,320	20.1	24/04/2006	14:15
rllog2	31	P	43,196	0.0	153,789	6.6	795,426	0.1	314,549	2.6	24/04/2006	14:15
rllog5	37	P	36,596	0.0	147,990	6.3	596,791	0.1	347,454	2.8	24/04/2006	14:15
rltab1	34	P	2,777,150	1.9	83,389	3.6	13,751,266	1.8	95,689	0.8	24/04/2006	14:15
rootdbs	1	P	3,184,995	2.2	57,995	2.5	3,244,114	0.4	796,256	6.5	24/04/2006	14:15
rltab1	23	P	2,816,592	1.9	55,244	2.4	9,193,345	1.2	74,937	0.6	24/04/2006	14:15
rltab1	38	P	3,253,763	2.2	51,985	2.2	21,377,636	2.8	60,259	0.5	24/04/2006	14:15
rltab1	57	P	1,943,650	1.3	39,424	1.7	21,476,888	2.8	51,242	0.4	24/04/2006	14:15
rltab1	22	P	7,047,108	4.8	32,221	1.4	11,286,197	1.5	34,478	0.3	24/04/2006	14:15
rltab1	39	P	1,472,032	1.0	31,061	1.3	3,838,366	0.5	37,276	0.3	24/04/2006	14:15
rltab1	41	P	2,172,071	1.5	27,157	1.2	9,095,032	1.2	34,814	0.3	24/04/2006	14:15
rlidx4	46	P	414,576	0.3	24,756	1.1	963,409	0.1	32,563	0.3	24/04/2006	14:15
rltab1	48	P	1,944,881	1.3	23,680	1.0	9,263,250	1.2	32,616	0.3	24/04/2006	14:15
rlidx5	47	P	519,651	0.4	23,595	1.0	1,192,345	0.2	29,657	0.2	24/04/2006	14:15
rlidx2	44	P	789,444	0.5	22,770	1.0	1,450,460	0.2	28,822	0.2	24/04/2006	14:15
rlgran10	11	P	6,073,596	4.1	21,018	0.9	32,003,142	4.2	26,885	0.2	24/04/2006	14:15
rlidx3	45	P	1,135,019	0.8	20,855	0.9	1,887,822	0.3	23,154	0.2	24/04/2006	14:15
rlidx1	43	P	716,540	0.5	18,839	0.8	2,146,720	0.3	21,449	0.2	24/04/2006	14:15
rltab1	40	P	894,983	0.6	14,661	0.6	2,086,632	0.3	17,309	0.1	24/04/2006	14:15
rootdbs	63	P	48,835	0.0	13,335	0.6	108,650	0.0	51,621	0.4	24/04/2006	14:15
rltab3	25	P	8,096,817	5.5	11,048	0.5	48,129,341	6.3	13,955	0.1	24/04/2006	14:15
rlgran19	20	P	6,228,016	4.2	10,768	0.5	27,417,751	3.6	13,532	0.1	24/04/2006	14:15
rlgran26	58	P	482,139	0.3	10,716	0.5	3,701,307	0.5	13,740	0.1	24/04/2006	14:15
rlgran9	10	P	5,084,570	3.4	9,542	0.4	26,276,998	3.4	13,131	0.1	24/04/2006	14:15
rltab6	28	P	7,963,146	5.4	9,394	0.4	46,874,976	6.1	11,272	0.1	24/04/2006	14:15
rlgran8	9	P	1,309,504	0.9	7,870	0.3	5,899,527	0.8	10,083	0.1	24/04/2006	14:15
rlgran11	12	P	5,171,821	3.5	7,435	0.3	29,750,315	3.9	9,096	0.1	24/04/2006	14:15
rltab2	24	P	6,002,184	4.1	6,879	0.3	38,317,249	5.0	7,901	0.1	24/04/2006	14:15
rltab4	26	P	4,852,073	3.3	6,635	0.3	24,979,362	3.3	8,140	0.1	24/04/2006	14:15
rlgran7	8	P	1,225,058	0.8	6,599	0.3	5,912,504	0.8	8,224	0.1	24/04/2006	14:15
rlgran6	7	P	1,454,900	1.0	6,565	0.3	6,462,915	0.8	8,752	0.1	24/04/2006	14:15
rltab5	27	P	7,539,428	5.1	6,227	0.3	40,661,715	5.3	7,954	0.1	24/04/2006	14:15
rlgran3	4	P	2,212,988	1.5	5,907	0.3	11,185,701	1.5	7,608	0.1	24/04/2006	14:15
rlgran13	14	P	8,257,770	5.6	5,823	0.3	54,643,876	7.1	7,811	0.1	24/04/2006	14:15
rlgran4	5	P	5,884,310	4.0	5,735	0.3	39,106,861	5.1	7,189	0.1	24/04/2006	14:15
rlgran15	16	P	4,240,691	2.9	5,159	0.2	20,774,974	2.7	8,091	0.1	24/04/2006	14:15
rlgran2	3	P	3,342,441	2.3	5,123	0.2	17,249,338	2.3	8,424	0.1	24/04/2006	14:15
rlgran5	6	P	1,025,187	0.7	4,917	0.2	5,263,260	0.7	6,539	0.1	24/04/2006	14:15
rlgran16	17	P	4,694,930	3.2	4,840	0.2	21,876,426	2.9	7,400	0.1	24/04/2006	14:15
rlgran17	18	P	794,858	0.5	4,365	0.2	3,619,868	0.5	6,220	0.1	24/04/2006	14:15
rlgran14	15	P	7,104,403	4.8	4,324	0.2	32,033,098	4.2	6,859	0.1	24/04/2006	14:15
rlgran20	21	P	2,815,082	1.9	4,299	0.2	16,125,618	2.1	5,983	0.1	24/04/2006	14:15
rlgran1	2	P	2,498,835	1.7	4,169	0.2	11,340,024	1.5	5,328	0.0	24/04/2006	14:15
rlidx7	54	P	196,902	0.1	4,115	0.2	762,353	0.1	4,156	0.0	24/04/2006	14:15
rlgran18	19	P	1,510,985	1.0	3,958	0.2	5,160,844	0.7	5,552	0.1	24/04/2006	14:15
rlgran27	59	P	1,303,133	0.9	3,861	0.2	7,583,641	1.0	4,947	0.0	24/04/2006	14:15
rlidx9	56	P	210,570	0.1	2,898	0.1	832,193	0.1	2,992	0.0	24/04/2006	14:15
rlgran14	42	P	853,814	0.6	2,680	0.1	7,248,044	1.0	3,312	0.0	24/04/2006	14:15
rlgran12	13	P	2,543,774	1.7	2,592	0.1	14,656,358	1.9	3,810	0.0	24/04/2006	14:15
rltab5	60	P	45,262	0.0	2,227	0.1	99,620	0.0	2,813	0.0	24/04/2006	14:15

Pagina : 1 ne\_aug\_13\_14:15\_04.lst

```
Apr 24 2006 14:15:04 DBSPACES I/O -2- ORDER by writes desc
```

DBS	CHK	P/S	READS	%	WRITES	%	PAG_READS	%	PAG_WRITES	%	FECHA	HORA
rlidx6	53	P	237,466	0.2	1,743	0.1	849,375	0.1	1,927	0.0	24/04/2006	14:15

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

rlgran25	52 P	2,962,242	2.0	1,605	0.1	32,559,454	4.3	2,238	0.0	24/04/2006	14:15
rlcdr	29 P	139,465	0.1	1,571	0.1	241,310	0.0	4,620	0.0	24/04/2006	14:15
rlidx8	55 P	201,086	0.1	1,207	0.1	834,895	0.1	1,220	0.0	24/04/2006	14:15
rlgran16	49 P	158,538	0.1	1,093	0.1	358,004	0.1	1,391	0.0	24/04/2006	14:15
rlgran23	50 P	112,027	0.1	29	0.0	983,465	0.1	29	0.0	24/04/2006	14:15
rlgran23	51 P	117,068	0.1	5	0.0	344,664	0.0	37	0.0	24/04/2006	14:15
rllog4	36 P	8	0.0		0.0		10	0.0	0.0	24/04/2006	14:15
rllog3	35 P	8	0.0		0.0		10	0.0	0.0	24/04/2006	14:15
rlgran15	62 P	2	0.0		0.0		2	0.0	0.0	24/04/2006	14:15
rlgran9	61 P	2	0.0		0.0		2	0.0	0.0	24/04/2006	14:15
rllog1	30 P	8	0.0		0.0		10	0.0	0.0	24/04/2006	14:15

Apr 24 2006 ne\_aug\_13\_14:15\_04.lst Page 3

Al analizar el reporte anterior se observa que los dbspaces asignados para las áreas temporales de trabajo y los de las áreas de LOG, son los mas demandados por el sistema, por lo cual cada uno de ellos, deben de estar en discos diferentes. El dbspace de datos rltab1 es un dbspace muy accesado.

El código del programa shell chkiosum.sh es el siguiente:

```
==> cat /informix/chkiosum/chkiosum.sh
#####
INFORMIXSERVER=prod_300shm
export INFORMIXSERVER
ONCONFIG=prod_300shm
export ONCONFIG
INFORMIXDIR=/informix
export INFORMIXDIR
INFORMIXSQLHOSTS=$INFORMIXDIR/etc/sqlhosts
export INFORMIXSQLHOSTS
DBDATE=dmy4/
export DBDATE
PATH=$INFORMIXDIR/bin:/usr/bin
export PATH
#####
EQUIPO=`uname -a | awk -F" " '{ printf "%s",$2 }'`
export EQUIPO
#
HORA=`date +%H`
export HORA
#
cd /informix/chkiosum
rm *.lst
#
FECHA_HORA=`date +%d/%m/%Y_%H:%M`
export FECHA_HORA

#####
echo '
database sysmaster ;
-- recolectar chunk i/o estadísticas into temp table "qaz"
SELECT
    name dspace,
    chknum,
    "P" chktype,
    reads,
    writes,
    pagesread,
    pageswritten
from
    syschktab c, sysdbstab d
where
    c.dbsnum = d.dbsnum

UNION ALL
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

---

```

SELECT
    name[1,10] dspace,
    chknum,
    "S" chktype,
    reads,
    writes,
    pagesread,
    pageswritten
from
    sysmchktab c, sysdbstab d
where
    c.dbsnum = d.dbsnum
into
    temp qaz ;

-- Recolectar el Total de i/o estadísticas into temp table "wsx"

SELECT
    sum(reads)                total_reads,
    sum(writes)               total_writes,
    sum(pagesread)            total_pg_reads,
    sum(pageswritten)         total_pg_writes
FROM
    qaz
INTO
    TEMP wsx ;

-- Reporte que muestra cada chunk i/o en % individual
unload to "/tmp/chkiosum.txt"
delimiter "_"
SELECT
    dspace,
    chknum,
    chktype,
    reads,
    writes,
    pagesread,
    pageswritten,
    ROUND( (reads/total_reads) * 100 ,2 ) percent_reads,
    ROUND( (writes/total_writes) * 100 ,2 ) percent_writes,
    ROUND( (pagesread/total_pg_reads) * 100 ,2 ) perc_pg_reads,
    ROUND( (pageswritten/total_pg_writes) * 100 ,2 ) perc_pg_writes
FROM
    qaz, wsx
ORDER BY perc_pg_writes ;
' > /tmp/chkiosum.sql

#
dbaccess -e sysmaster - < /tmp/chkiosum.sql > /tmp/chkiosum.bit
#
cat /tmp/chkiosum.txt | awk -F"_" ' { printf "%s%s\n",ENVIRON["FECHA_HORA"],$0 }
' > /tmp/chkiosum.asc
#
dbaccess -e jvv <<EOF
delete from chkiosum_tmp;
EOF
#
dbload -d jvv -c chkiosum.dbl -l chkiosum.log
#
fglpc chkiosum.4gl
sleep 2
fglgo chkiosum 2
sleep 2
fglgo chkiosum 3
sleep 2
fglgo chkiosum 1
#
echo "$EQUIPO I/O STATISTICS" > /tmp/pager_asunto_chk-rpt-$EQUIPO.lst
pr *.lst > /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
#
rcp /tmp/pager_asunto_chk-rpt-$EQUIPO.lst prueba:/tmp &
rcp /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst prueba:/tmp &

```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

---

El código del programa en Informix\_4gl, chkiosum.4gl es el siguiente:

```
==> cat /informix/chkiosum/chkiosum.4gl
#####
##
#####
database jvv

GLOBALS
DEFINE

chkiosum_tmp_arreg      RECORD LIKE chkiosum_tmp.* ,
query_prep              char(200),
nomb_repo               char(30),
tipo_sort               char(50),
opcion_sort             char(1),
fecha_report            char(25)

END GLOBALS

MAIN

  LET opcion_sort      = UPSHIFT(ARG_VAL(1))

  LET query_prep = "select * from chkiosum_tmp  "

CASE
WHEN opcion_sort = "1"
  let query_prep = query_prep clipped, " ORDER by reads desc "
  let tipo_sort = " ORDER by pagesread desc "

WHEN opcion_sort = "2"
  let query_prep = query_prep clipped, " ORDER by writes desc "
  let tipo_sort = " ORDER by writes desc "

WHEN opcion_sort = "3"
  let query_prep = query_prep clipped, " ORDER by pagesread desc "
  let tipo_sort = " ORDER by pagesread desc "

WHEN opcion_sort = "4"
  let query_prep = query_prep clipped, " ORDER by pageswritten desc "
  let tipo_sort = " ORDER by percent_writes desc "

WHEN opcion_sort = "5"
  let query_prep = query_prep clipped, " ORDER by perc_pg_reads desc "
  let tipo_sort = " ORDER by perc_pg_reads desc "

WHEN opcion_sort = "6"
  let query_prep = query_prep clipped, " ORDER by perc_pg_writes desc "
  let tipo_sort = " ORDER by perc_pg_writes desc "

OTHERWISE
  let query_prep = query_prep clipped, " ORDER by perc_pg_writes desc "
  let tipo_sort = " ORDER by perc_pg_writes desc "

END CASE

PREPARE preparar_consulta1 FROM query_prep
DECLARE c_consulta CURSOR FOR preparar_consulta1
OPEN c_consulta

LET fecha_report = DATE , " ",time
let nomb_repo = "ne_",fecha_report[5,7], "_",fecha_report[9,10]
```



**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

---

```

using "&&","_",fecha_report[17,21],"_",fecha_report[23,24],".lst"
let nomb_repo = downshift(nomb_repo clipped)

START report chkio_lst to nomb_repo

FOREACH c_consulta INTO chkiosum_tmp_arreg.*

    OUTPUT TO REPORT chkio_lst (chkiosum_tmp_arreg.*)

END FOREACH

FINISH REPORT chkio_lst

END MAIN
#####
#####
REPORT chkio_lst (chkiosum_tmp_arreg)
define
chkiosum_tmp_arreg          RECORD LIKE chkiosum_tmp.*

output
left margin 1
format

page trailer
skip 1 lines
print "Pagina : ", pageno using "####", " ",nomb_repo

page header
skip 1 lines
print date," ",time," DBSPACES I/O  -",opcion_sort,"-",tipo_sort

skip 1 lines
print
    " DBS      ",
    " CHK",
    " P/S      ",
    " READS    ",
    " %",
    " WRITES   ",
    " % ",
    " PAG_READS ",
    " % ",
    " PAG_WRITES ",
    " % ",
    " FECHA    ",
    " HORA    "
skip 1 lines

on every row

print
    chkiosum_tmp_arreg.dbspace[1,8]," ",
    chkiosum_tmp_arreg.num_chk using "##", " ",
    chkiosum_tmp_arreg.pri_sec,
    chkiosum_tmp_arreg.reads using "###,###,###", " ",
    chkiosum_tmp_arreg.percent_reads using "##&.", " ",
    chkiosum_tmp_arreg.writes using "###,###,###", " ",
    chkiosum_tmp_arreg.percent_writes using "##&.", " ",
    chkiosum_tmp_arreg.pagesread using "###,###,###", " ",
    chkiosum_tmp_arreg.perc_pg_reads using "##&.", " ",
    chkiosum_tmp_arreg.pageswritten using "###,###,###", " ",
    chkiosum_tmp_arreg.perc_pg_writes using "##&.", " ",
    chkiosum_tmp_arreg.fecha," ",
    chkiosum_tmp_arreg.hora," "

```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
***CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE***

---

end report

==>

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

---

**3.2 Balanceo del I/O por dbspace**

Basándose en la información obtenida en el punto 3.1, los administradores deben de empezar a rastrear cuales son las tablas de datos, contenidas en los dbspace más accesados, que están siendo muy demandados por los procesos de los usuarios, para redistribuirlas a otros dbspaces y eliminar la contención en ese dbspace, en ese disco. Es decir balancear el I/O de una manera mas uniforme en todos los dbspaces que integran la base de datos.

Para conocer que tablas están almacenadas en un dbspace, se tiene el comando `oncheck -pe`, el cual reporta a detalle que objetos se encuentran almacenados en determinado dbspace.

De los listados generados en el punto 3.1 sabemos que el dbspace `rltab3` es de gran demanda, tiene el numero de dbpace 24, y el numero de chunk 25, el comando `oncheck -pe` nos genera el contenido a detalle de todos los dbspaces, únicamente mostraremos el contenido del dbspace `rltab3`:

**oncheck -pe**

DBspace Usage Report: `rltab3`

Owner: `informix` Created: 08/29/1999

Chunk: 25 /dev/vgtabdb/rltab3

Size	Used	Free
1000000	868867	131133

Disk usage for Chunk 25	Start	Length
-----	-----	-----
OTHER RESERVED Pages	0	2
CHUNK FREE LIST PAGE	2	1
TBLSPACE TBLSPACE	3	50
sis_com:informix.sccat_depto	53	32
sis_com:informix.sccat_param_ctrl	85	32
sis_com:informix.sccat_var_sum	117	32
sis_com:informix.sccat_vehiculo	149	32
sis_com:informix.sccat_prod	181	32
sis_com:informix.sccat_unidad	213	32
sis_com:informix.sccat_serv	245	32
sis_com:informix.sccat_imp_gpo	277	32
sis_com:informix.sccat_vehiculo	309	1392
sis_com:informix.scmae_docto_reimp	1701	8
sis_com:informix.sccat_vehiculo	1709	24
sis_com:informix.scdet_acum_sc4	1733	400
sis_com:informix.scmae_docto_reimp	2133	8
sis_com:informix.system-rowid	2141	200
sis_com:informix.sccat_vehiculo	2341	8
sis_com:informix.scmae_docto_reimp	2349	8
sis_com:informix.sccat_vehiculo	2357	16
sis_com:informix.system-rowid	2373	120
sis_com:informix.sccat_vehiculo	2493	32
sis_com:informix.scmae_docto_reimp	2525	8
sis_com:informix.system-rowid	2533	120
sis_com:informix.system-rowid	2653	1766
sis_com:informix.sccat_vehiculo	4419	8
sis_com:informix.system-rowid	4427	3209
sis_com:informix.system-rowid	7636	908
sis_com:informix.system-rowid	8544	400
sis_com:informix.sccat_vehiculo	8944	16
sis_com:informix.scmae_docto_reimp	8960	24
sis_com:informix.sccat_vehiculo	8984	16
sis_com:informix.system-rowid	9000	120

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

---

FREE	9120	3
sis_com:informix.scdet_mov_fact	9123	13500
sis_com:informix.system-rowid	22623	5375
sis_com:informix.scdet_mov_saldo	27998	23500
sis_com:informix.system-rowid	51498	3967
sis_com:informix.scmae_notas_emb_din	55465	4150
sis_com:informix.system-rowid	59615	235
sis_com:informix.scmae_docto_reimp	59850	8
sis_com:informix.sccat_vehiculo	59858	8
sis_com:informix.scdet_mov_notas_cr	59866	8
sis_com:informix.system-rowid	59874	124
FREE	59998	7
sis_com:informix.sccat_vehiculo	60005	8
sis_com:informix.system-rowid	60013	184
sis_com:informix.scmae_docto_reimp	60197	16
sis_com:informix.sccat_vehiculo	60213	16
sis_com:informix.scmae_docto_reimp	60229	64
sis_com:informix.sccat_vehiculo	60293	48
FREE	60341	72
sis_com:informix.system-rowid	60413	227
sis_com:informix.sccat_vehiculo	60640	8
sis_com:informix.scmae_docto_reimp	60648	96

Disk usage for Chunk 25	Start	Length
-----	-----	-----
sis_com:informix.sccat_vehiculo	60744	8
sis_com:informix.scmae_docto_reimp	60752	8
FREE	60760	3
sis_com:informix.scdet_mov_notas_emb	60763	450
sis_com:informix.system-rowid	61213	227
sis_com:informix.system-rowid	61440	400
sis_com:informix.system-rowid	61840	200
sis_com:informix.scmae_docto_reimp	62040	56
FREE	62096	5
sis_com:informix.system-rowid	62101	1766
sis_com:informix.system-rowid	63867	200
FREE	64067	185
sis_com:informix.system-rowid	64252	227
sis_com:informix.scmae_docto_reimp	64479	24
FREE	64503	4
sis_com:informix.scmae_fact_din	64507	3750
sis_com:informix.system-rowid	68257	908
FREE	69165	30
sis_com:informix.system-rowid	69195	227
sis_com:informix.sccat_vehiculo	69422	8
sis_com:informix.scmae_docto_reimp	69430	8
sis_com:informix.sccat_vehiculo	69438	8
sis_com:informix.scmae_docto_reimp	69446	8
sis_com:informix.sccat_vehiculo	69454	8
sis_com:informix.scmae_docto_reimp	69462	16
FREE	69478	4
sis_com:informix.system-rowid	69482	200
FREE	69682	400
sis_com:informix.system-rowid	70082	200
FREE	70282	40
sis_com:informix.scdet_acum_sc4	70322	400
sis_com:informix.scdet_mov_notas_cr	70722	750
FREE	71472	156
sis_com:informix.system-rowid	71628	227
sis_com:informix.system-rowid	71855	454
FREE	72309	206
sis_com:informix.system-rowid	72515	227
FREE	72742	13
sis_com:informix.scmae_docto_reimp	72755	72
sis_com:informix.system-rowid	72827	120
FREE	72947	400
sis_com:informix.system-rowid	73347	200
FREE	73547	40
sis_com:informix.system-rowid	73587	227
sis_com:informix.system-rowid	73814	454
FREE	74268	131

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

---

sis_com:informix.system-rowid	74399	227
sis_com:informix.system-rowid	74626	908
FREE	75534	441
sis_com:informix.system-rowid	75975	1766
sis_com:informix.sccat_vehiculo	77741	8
sis_com:informix.system-rowid	77749	227
FREE	77976	29
sis_com:informix.scmae_docto_reimp	78005	8
sis_com:informix.sccat_vehiculo	78013	8
FREE	78021	4
sis_com:informix.scdet_mov_bco_doct	78025	8
sis_com:informix.jun03_00	78033	4
sis_com:informix.system-rowid	78037	4
sis_com:informix.system-rowid	78041	200
sis_com:informix.system-rowid	78241	454

Disk usage for Chunk 25	Start	Length
-----	-----	-----
FREE	78695	235
sis_com:informix.system-rowid	78930	454
sis_com:informix.system-rowid	79384	454
sis_com:informix.system-rowid	79838	200
FREE	80038	12
sis_com:informix.system-rowid	80050	120
FREE	80170	496
sis_com:informix.sccat_vehiculo	80666	8
FREE	80674	385
sis_com:informix.system-rowid	81059	227
FREE	81286	228
sis_com:informix.scdet_mov_nota_cr	81514	750
sis_com:informix.scdet_mov_nota_emb	82264	2500
FREE	84764	95
sis_com:informix.system-rowid	84859	227
sis_com:informix.system-rowid	85086	120
FREE	85206	400
sis_com:informix.system-rowid	85606	200
FREE	85806	527
sis_com:informix.system-rowid	86333	200
FREE	86533	56
sis_com:informix.system-rowid	86589	200
sis_com:informix.scdet_mov_nota_cr	86789	750
FREE	87539	35
sis_com:informix.system-rowid	87574	200
sis_com:informix.scdet_mov_nota_emb	87774	2500
FREE	90274	221
sis_com:informix.system-rowid	90495	200
FREE	90695	640
sis_com:informix.scdet_mov_nota_emb	91335	2500
FREE	93835	759
sis_com:informix.system-rowid	94594	200
FREE	94794	425
sis_com:informix.system-rowid	95219	120
sis_com:informix.system-rowid	95339	454
FREE	95793	1427
sis_com:informix.scmae_docto_reimp	97220	8
FREE	97228	993
sis_com:informix.scdet_acum_sc4	98221	6850
sis_com:informix.system-rowid	105071	2661
sis_com:informix.scmae_fact_din	107732	57000
sis_com:informix.system-rowid	164732	4818
sis_com:informix.system-rowid	169550	4239
sis_com:informix.system-rowid	173789	2532
FREE	176321	1202
sis_com:informix.scdet_fact	177523	83500
sis_com:informix.system-rowid	261023	30659
FREE	291682	640
sis_com:informix.system-rowid	292322	227
sis_com:informix.system-rowid	292549	120
FREE	292669	6
sis_com:informix.sccat_vehiculo	292675	8
FREE	292683	840

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

---

sis_com:informix.system-rowid	293523	227
FREE	293750	353
sis_com:informix.system-rowid	294103	120
FREE	294223	200
sis_com:informix.scmae_saldo_docto	294423	15000
sis_com:informix.system-rowid	309423	1900
sis_com:informix.system-rowid	311323	4239
sis_com:informix.system-rowid	315562	4239

Disk usage for Chunk 25	Start	Length
-----	-----	-----
FREE	319801	436
sis_com:informix.system-rowid	320237	4239
sis_com:informix.system-rowid	324476	4534
FREE	329010	2338
sis_com:informix.scdet_fact	331348	6250
sis_com:informix.scmae_fact_din	337598	3750
sis_com:informix.scdet_acum_sc4	341348	400
sis_com:informix.system-rowid	341748	1766
FREE	343514	282
sis_com:informix.scdet_fact	343796	6250
FREE	350046	1536
sis_com:informix.system-rowid	351582	1766
FREE	353348	1020
sis_com:informix.scdet_acum_sc4	354368	400
sis_com:informix.scmae_saldo_docto	354768	2000
FREE	356768	1024
sis_com:informix.scmae_fact_din	357792	3750
FREE	361542	385
sis_com:informix.system-rowid	361927	1766
FREE	363693	1559
sis_com:informix.scdet_fact	365252	6250
FREE	371502	353
sis_com:informix.scdet_acum_sc4	371855	400
sis_com:informix.system-rowid	372255	1766
FREE	374021	456
sis_com:informix.system-rowid	374477	454
FREE	374931	1570
sis_com:informix.system-rowid	376501	454
FREE	376955	379
sis_com:informix.system-rowid	377334	227
sis_com:informix.scmae_fact_din	377561	3750
FREE	381311	400
sis_com:informix.scmae_saldo_docto	381711	2000
FREE	383711	1438
sis_com:informix.system-rowid	385149	200
sis_com:informix.scdet_acum_sc4	385349	400
FREE	385749	400
sis_com:informix.system-rowid	386149	1766
FREE	387915	2141
sis_com:informix.scdet_fact	390056	6250
sis_com:informix.system-rowid	396306	120
sis_com:informix.system-rowid	396426	227
FREE	396653	800
sis_com:informix.system-rowid	397453	454
FREE	397907	426
sis_com:informix.scmae_fact_din	398333	3750
FREE	402083	1633
sis_com:informix.scmae_saldo_docto	403716	2000
sis_com:informix.scdet_fact	405716	6250
sis_com:informix.system-rowid	411966	4534
sis_com:informix.system-rowid	416500	1766
sis_com:informix.system-rowid	418266	454
FREE	418720	199
sis_com:informix.scdet_mov_bco_doct	418919	130000
sis_com:informix.jun03_00	548919	13203
sis_com:informix.system-rowid	562122	26406
sis_com:informix.scmae_fact_din	588528	3750
sis_com:informix.scdet_acum_sc4	592278	15000
sis_com:informix.scdet_fact	607278	6250
sis_com:informix.scmae_fact_din	613528	3750

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

sis_com:informix.scdet_mov_fact	617278	20000
Disk usage for Chunk 25		
	Start	Length
-----	-----	-----
sis_com:informix.scmae_fact_din	637278	3750
sis_com:informix.scdet_fact	641028	6250
sis_com:informix.scmae_fact_din	647278	3750
sis_com:informix.system-rowid	651028	1766
sis_com:informix.scmae_saldo_docto	652794	2000
sis_com:informix.system-rowid	654794	1766
sis_com:informix.scmae_fact_din	656560	3750
sis_com:informix.system-rowid	660310	1766
sis_com:informix.scmae_fact_din	662076	3750
sis_com:informix.scdet_mov_notas_emb	665826	2500
sis_com:informix.scdet_fact	668326	6250
sis_com:informix.system-rowid	674576	1766
sis_com:informix.scmae_saldo_docto	676342	2000
sis_com:informix.scmae_fact_din	678342	3750
sis_com:informix.system-rowid	682092	13203
sis_com:informix.system-rowid	695295	1766
sis_com:informix.scmae_fact_din	697061	3750
sis_com:informix.system-rowid	700811	1766
FREE	702577	2649
sis_com:informix.scmae_fact_din	705226	3750
sis_com:informix.scmae_saldo_docto	708976	2000
sis_com:informix.system-rowid	710976	3532
FREE	714508	800
sis_com:informix.scmae_fact_din	715308	3750
FREE	719058	1506
sis_com:informix.scmae_fact_din	720564	7500
sis_com:informix.system-rowid	728064	3532
FREE	731596	1506
sis_com:informix.scmae_saldo_docto	733102	2000
FREE	735102	1891
sis_com:informix.system-rowid	736993	3532
sis_com:informix.scmae_fact_din	740525	7500
FREE	748025	2399
sis_com:informix.system-rowid	750424	454
FREE	750878	800
sis_com:informix.scdet_fact	751678	15000
FREE	766678	706
sis_com:informix.scmae_saldo_docto	767384	2000
sis_com:informix.jun03_00	769384	13203
sis_com:informix.system-rowid	782587	4239
FREE	786826	1570
sis_com:informix.scmae_fact_din	788396	7500
FREE	795896	706
sis_com:informix.system-rowid	796602	454
FREE	797056	2814
sis_com:informix.scdet_fact	799870	15000
sis_com:informix.scmae_saldo_docto	814870	2000
sis_com:informix.system-rowid	816870	4239
sis_com:informix.system-rowid	821109	454
sis_com:informix.scmae_fact_din	821563	7500
FREE	829063	2212
sis_com:informix.fkc04abr01	831275	7423
FREE	838698	1570
sis_com:informix.system-rowid	840268	13203
sis_com:informix.system-rowid	853471	8478
sis_com:informix.scdet_acum_sc4	861949	15000
sis_com:informix.scdet_mov_saldo	876949	15000
sis_com:informix.system-rowid	891949	13203
sis_com:informix.system-rowid	905152	4239
sis_com:informix.scmae_notas_emb_din	909391	3250
sis_com:informix.system-rowid	912641	8478
Disk usage for Chunk 25		
	Start	Length
-----	-----	-----
FREE	921119	78881

DBspace Usage Report: rltab4

Owner: informix Created: 08/29/1999

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

---

Chunk: 26	/dev/vgtabdbbs/rltab4	Size	Used	Free
		1000000	740359	259641

Ya teniendo las tablas que están contenidas en el dbspace rltab3, y auxiliándonos en las graficas del monitoreo de tablas mas frecuentemente accesadas como se muestra en la grafica 3.2 y por el monitoreo de las tablas con mas de 100,000 registros, los administradores de la base de datos, podrán decidir que tablas tienen que redistribuirse en otros dbspaces, o crearles índices mas adecuados, para facilitar su acceso.

Para generar graficas del comportamiento de las tablas más accesadas a la base de datos central, se tiene programado la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
**** /siic_onbar2/monitoreo/locks_tbl.sh 1>>/tmp/locks_tbl.txt 2>&1
```

El programa Shell `locks_tbl.sh`, monitorea el acceso a las tablas y el tipo de bloqueo que se aplica en las mismas, el usuario, el servidor remoto, etc. Se ejecuta cada minuto todo el tiempo, este monitoreo únicamente se activa, cuando se tienen problemas de desempeño, para contar con información para rastrear las tablas con problemas de acceso.

El código del programa shell `locks_tbl.sh` es el siguiente:

```
<informix@siic1> </siic_onbar2/monitoreo/locks_tbl>
==> cat locks_tbl.sh
export DBDATE=dmy4/
export INFORMIXDIR=/informix
export INFORMIXSERVER=prod_300shm
export INFORMIXSQLHOSTS=/informix/etc/sqlhosts
export INFORMIXTERMCAP=TERMCAP
export ONCONFIG=prod_300shm
export PATH=$INFORMIXDIR/bin:/usr/bin
cd /siic_onbar2/monitoreo/locks_tbl
dbaccess sysmaster <<! 2>locks_tbl.ins
set isolation to dirty read;
unload to "locks_tbl.lst" DELIMITER "|"
select pid,sid,current,tabname,type,username,hostname
from syslocks, sysessions
where tabname not matches "sys*"
  and tabname matches "sc*"
  and syslocks.owner=sysessions.sid
order by tabname
!
echo "----- `date` -----">>locks_tbl.bit
sed s/%/\ /g locks_tbl.lst > locks_tbl.tmp
cat locks_tbl.tmp >> locks_tbl.bit
#sed s/%/\ /g locks_tbl.bit > locks_tbl.tmp2
#grep -i siic1 locks_tbl.tmp2 > locks_tbl.bat
#chmod 744 locks_tbl.bat
#./locks_tbl.bat 1>> locks_tbl_pid.bit 2>&1
#echo "----- `date` -----" 1>> locks_tbl_pid.bit 2>&1
```

El programa anterior genera un archivo con la siguiente información, se puede activar por 1 día, o por una semana, o por un mes:

```
3918|1603811|2006-08-14 17:24:00.000|scdet_mov_saldo|IX|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scdet_saldo_docto|X|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scdet_saldo_docto|IX|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmae_acum_sc4|X|cdr|siic1|
```



**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

---

```

13918|1603811|2006-08-14 17:24:00.000|scmae_acum_sc4|IX|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmae_fact_din|X|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmae_fact_din|IX|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmae_rec_fon|U|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmae_rec_fon|IX|cdr|siic1|
18876|1603935|2006-08-14 17:24:00.000|scmae_saldo_cte|IX|pctmo300|siic1|
18722|1603927|2006-08-14 17:24:00.000|scmae_saldo_cte|IX|pctmo300|siic1|
18876|1603935|2006-08-14 17:24:00.000|scmae_saldo_cte|X|pctmo300|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmae_saldo_cte|X|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmae_saldo_cte|IX|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmae_saldo_docto|X|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmae_saldo_docto|X|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmae_saldo_docto|IX|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmc_fact_din|X|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmc_fact_din|IX|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmt_fact_din|X|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|scmt_fact_din|IX|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|screl_cto_depto|IS|cdr|siic1|
13918|1603811|2006-08-14 17:24:00.000|screl_prod_pres|IS|cdr|siic1|
----- Wed Aug 14 17:34:02 CDT 2006 -----
1312|1600381|2006-08-14 17:34:00.000|scdet_mov_saldo|IX|gtos3602|cadereyta|
1312|1600381|2006-08-14 17:34:00.000|scdet_mov_saldo|X|gtos3602|cadereyta|
1312|1600381|2006-08-14 17:34:00.000|scmae_reg_veh_for|X|gtos3602|cadereyta|
1312|1600381|2006-08-14 17:34:00.000|scmae_reg_veh_for|IX|gtos3602|cadereyta|
1312|1600381|2006-08-14 17:34:00.000|scmae_saldo_cte|X|gtos3602|cadereyta|
1312|1600381|2006-08-14 17:34:00.000|scmae_saldo_cte|IX|gtos3602|cadereyta|

```

Como mencionamos en el punto 3.2, estos son los tipos de bloqueo que se aplican a las tablas:

B	Byte lock
S	Shared
X	Exclusive
I	Intent
U	Update
IX	Intent exclusive
IS	Intent shared
SIX	Shared, intent exclusive

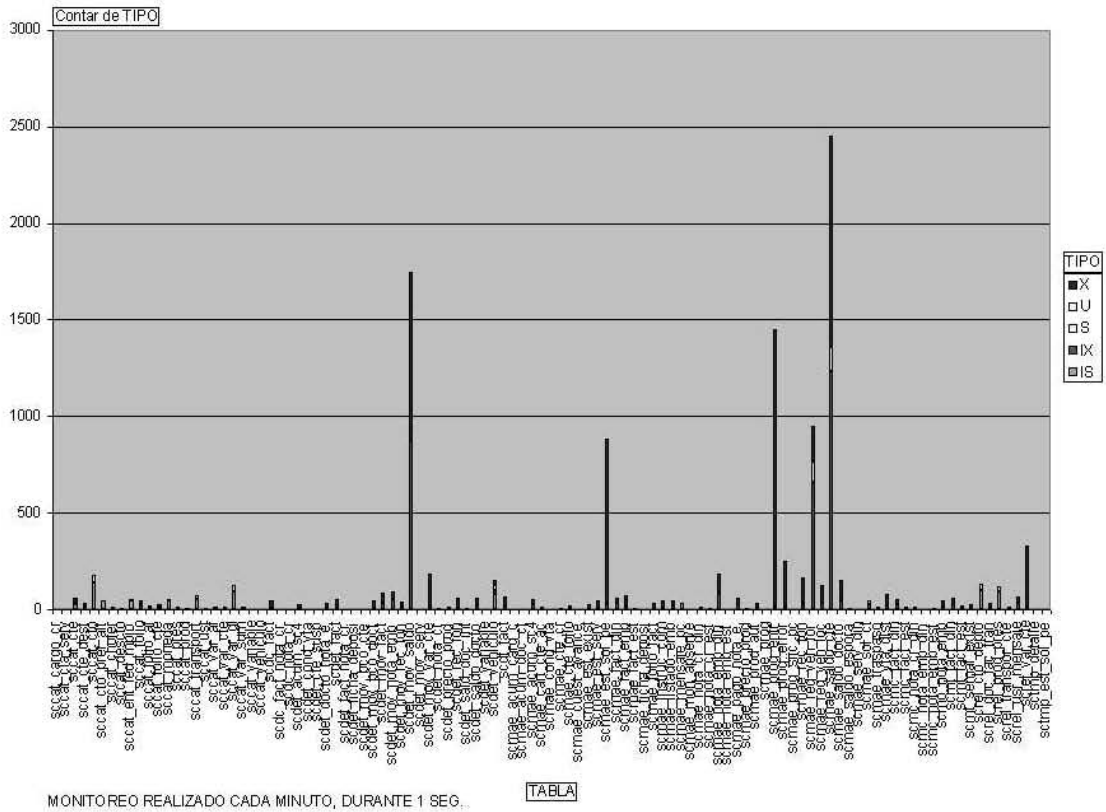
Toda la información recolectada se exporta a la Hoja de Excel, de la cual se generan las siguientes graficas, que sirven de guía, para auxiliar en la identificación de las tablas con problemas de acceso. En el capítulo 5, punto 5 se mencionará como realizar los cambios necesarios a las tablas con problemas de acceso.

# MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

## CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

FECHA (Todas)

TABLAS MAS ACCESADAS EN EL SERVIDOR CENTRAL Y QUE TIPO DE BLOQUEO SE APLICO



GRAFICA 3.2.1

Del grafico anterior se puede comentar que las siguientes tablas: `scdet_mov_saldo`, `scmae_est_sol_pe`, `scmae_prog_entr`, `scmae_reg_veh_for`, `scmae_saldo_cte`, son las tablas que más accesos simultáneos tienen, en un instante dado.

Con estas tablas se tienen que tener especial atención, que estén bien distribuidas, que tengan una estrategia de fragmentación adecuada, que no tengan un número excesivo de extents, que tengan un bloqueo por registro, etc.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

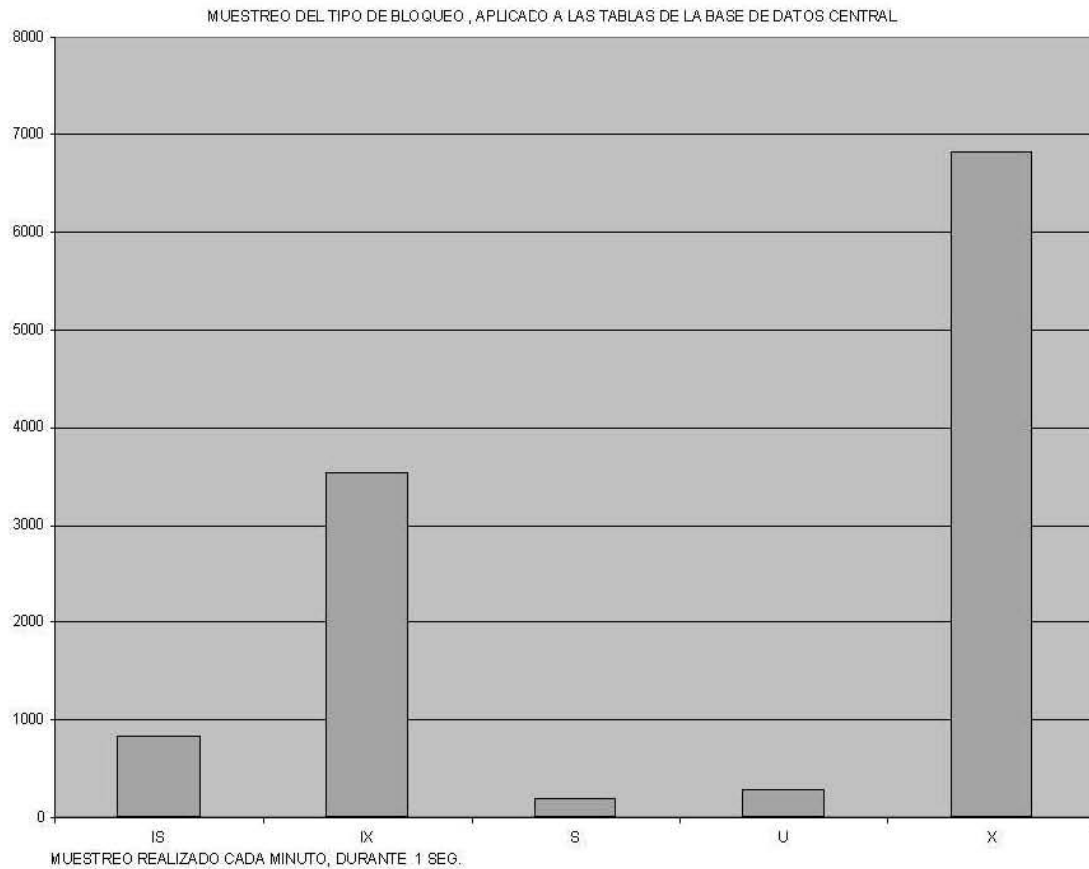
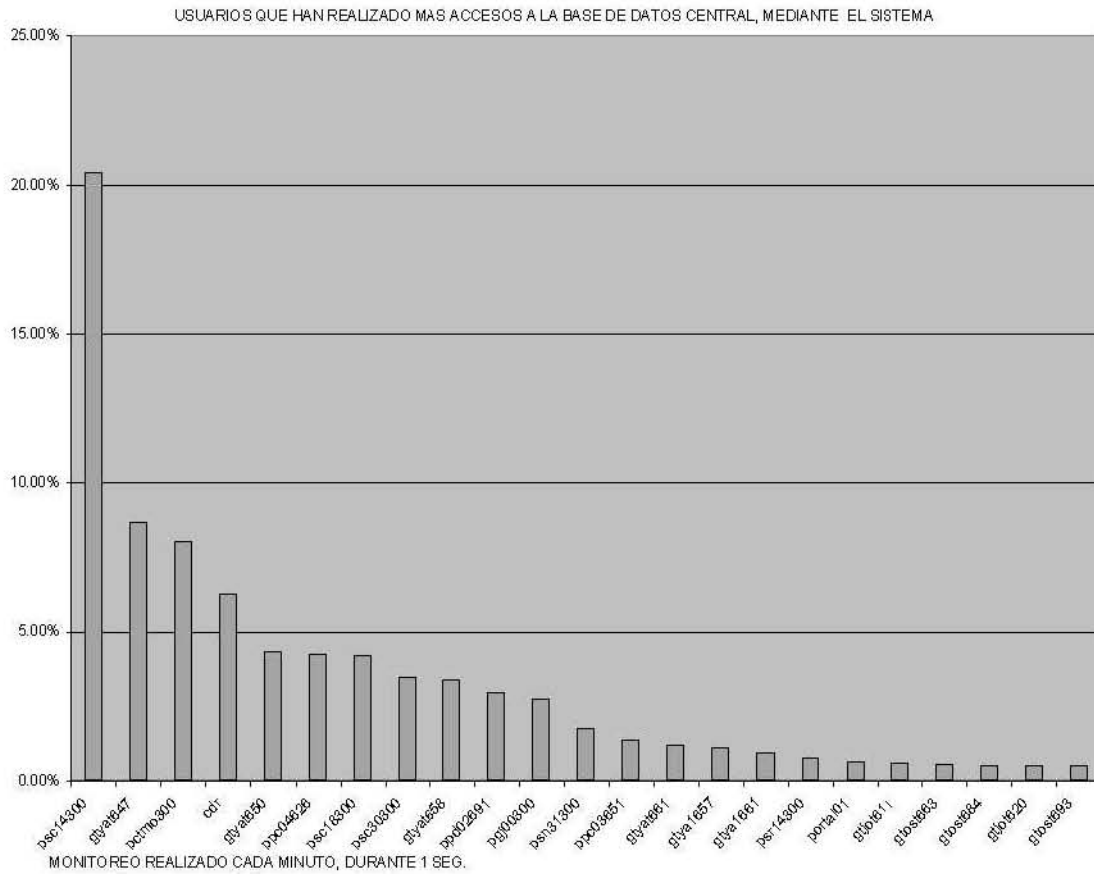


Grafico 3.2.2

Del grafico anterior podemos comentar que el tipo principal de bloqueo es el de tipo "X", es decir, el registro a modificar o eliminar, se bloquea exclusivamente por la sesión generadora de la transacción, lo importante es que los programas responsables de la transacción, no mantengan el bloqueo al recurso, más del tiempo necesario.

Pues de lo contrario, esto podría afectar a los demás usuarios, que están esperando por ese mismo recurso.

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**



**Grafico 3.2.3**

Del grafico anterior podemos comentar que el usuario psc14300, es el que más recursos ha consumido en este periodo de monitoreo, se tiene que investigar que es lo que ha estado realizando, platicar con los desarrolladores de los programas que el usuario tienen autorizados y finalmente platicar con el usuario, para orientarlo en el uso optimo y correcto de el sistema.

Es muy común que por falta de capacitación los usuarios generen consultas masivas a la base de datos, sin un propósito definido. En vez de consultar su información en un rango específico de tiempo, consultan toda la información almacenada lo cual consume una cantidad impresionante de recursos, afectando a los demás usuarios.

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

FECHA: (Todas) TERMINALES DE ALMACENAMIENTO Y DISTRIBUCION QUE REALIZARON MAS ACCESOS A LA BASE DE DATOS CENTRAL

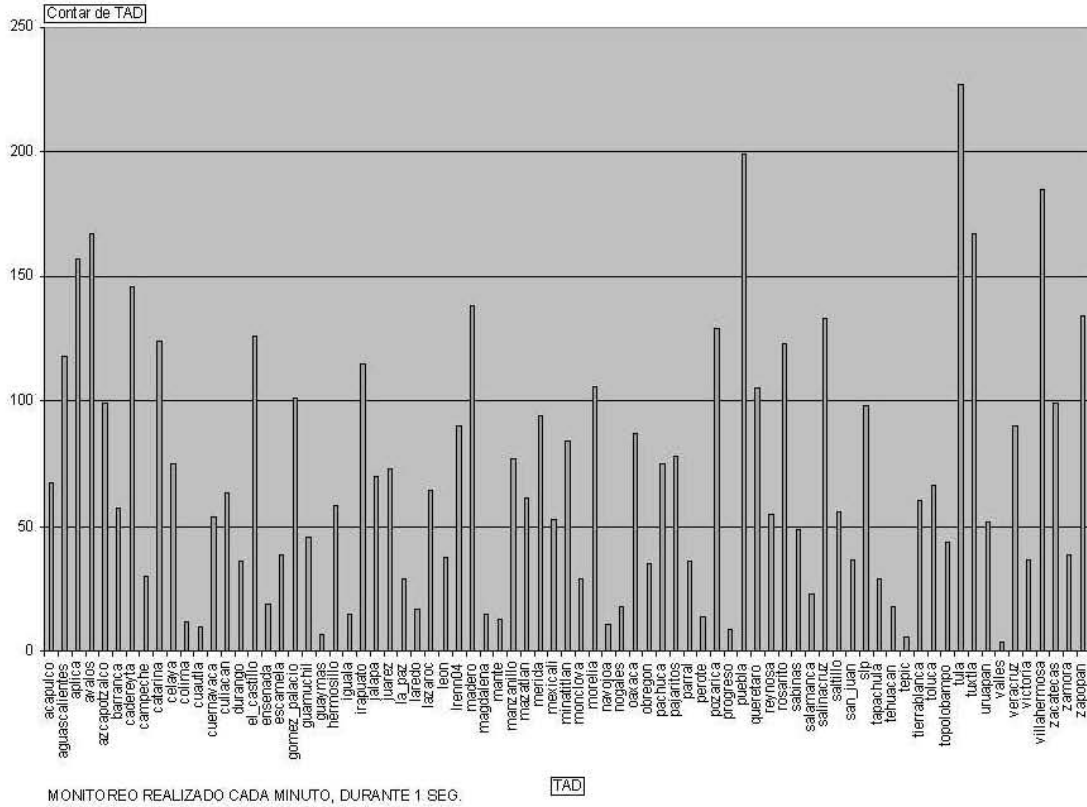


Gráfico 3.2.4

Del gráfico anterior podemos comentar que la Terminal de Almacenamiento y Distribución (TAD) de Tula, es la que mas accesos a la base de datos central tiene registrados, es lógico por la parte de que es el 2do centro facturador en el país, pero la TAD del "El CASTILLO" en Guadalajara es el 1er centro facturador y registra un 40 % menos de accesos a la base de datos.

# MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

## CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

MUESTREO DE EN QUE INSTANTE SE REALIZAN MAS ACCESOS A LA BASE DE DATOS CENTRAL

Contar de HORA

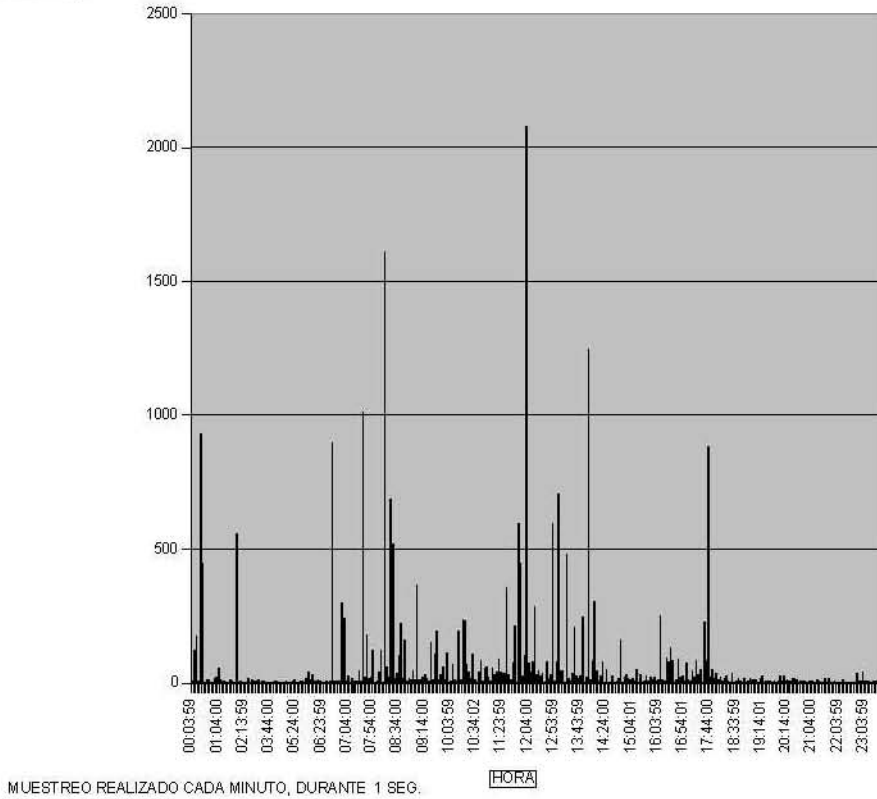


Grafico 3.2.5

Del grafico anterior obtenemos información de, en que instantes el servidor de base de datos tiene mas trabajos por atender, esta información nos ayuda a planear la programación de procesos centrales y que estos se realicen cuando el servidor esta con menos carga de trabajo.

# MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

## CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

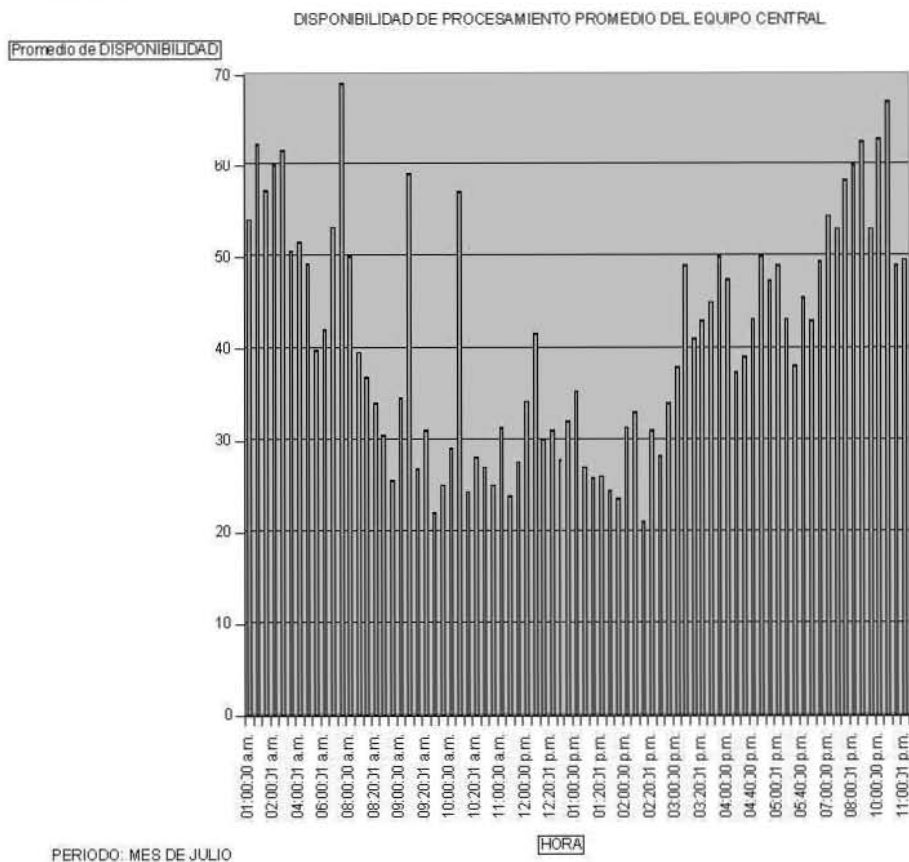


Grafico 3.2.6

Otro de los monitoreos importantes, es el de la disponibilidad de proceso en el equipo central, pues el mismo nos indica que tan saturados están los recursos del equipo:

- Memoria
- Discos
- Procesadores

Como se ilustra en la grafica anterior, se observa claramente que las horas "pico" son aproximadamente entre las 8:00 AM y las 14:00 PM.

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

---

### 3.3 Ocupación de cada dbspace y por chunk

Al crear la estructura de la base de datos se tiene que realizar una proyección de la distribución de los datos, para ello se debe comentar al respecto con los desarrolladores del sistema, para que hagan una estimación del número inicial de registros por cada tabla, del incremento mensual de registros en cada tabla, y el periodo de tiempo que se van a conservar la información, en el caso de este sistema, es de 6 meses hacia atrás.

Con toda esa información el administrador de la base de datos establece las estructuras de las tablas, con el tamaño del extent inicial adecuado y con la política de fragmentación adecuada en caso de ser requerida y la distribución en cada uno de los dbspaces disponibles.

Pasado este periodo de tiempo, se debe de depurar las tablas y pasarlas a otra base de datos "histórica" en otro equipo. En el cual los usuarios podrán consultar todos los movimientos que se han realizado en el sistema, para generar reportes de tendencias, auditorias, etc.

Como regularmente sucede, las proyecciones de crecimiento de las tablas, no son exactas; Por lo cual el dimensionamiento para ciertas tablas puede llegar a ser insuficiente y por lo tanto los dbspaces pueden llegar a llenarse, lo cual es un problema muy serio, pues la actividad del sistema se detendría.

Por lo cual se tiene un programa shell que monitorea el porcentaje de llenado de los dbspaces, manualmente se ejecuta con el siguiente comando:

```
> /informix/envia_pagers/pager_dbs_lleno.sh
```

Este mismo monitoreo de este evento, se tiene automatizado, programándose por medio de la utilería crontab del sistema operativo:

```
0,30 * * * * /informix/envia_pagers/pager_dbs_lleno.sh 1>/tmp/pager_dbs_lleno.bit 2>&1
```

Mediante el cual se ejecuta cada 30 minutos todo el tiempo, puede ser un evento crítico, se dispara la alarma cuando un dbspace llega a más del 95 % de su capacidad, en esos casos se debe de agregar inmediatamente un chunk de 2 GB al dbspaces lleno.

Una notificación típica es la siguiente:

```
SIIC OK DBSPACE's CAPACIDAD
```

Otra notificación típica es la siguiente:

```
DBSPACE r1temp4   ESTA LLENO AL 99.0 % 01/August-17:30  
DBSPACE r1temp1   ESTA LLENO AL 98.0 % 01/August-17:30
```



## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

---

```
DBSPACE rlttemp3   ESTA LLENO AL 95.0 % 01/August-17:30
DBSPACE rlttemp2   ESTA LLENO AL 95.0 % 01/August-17:30
DBSPACE rltab5     ESTA LLENO AL 95.0 % 09/April-16:00
DBSPACE rootdbs    ESTA LLENO AL 99.0 % 02/July-17:30
```

El código del programa shell pager\_dbs\_lleno.sh es el siguiente:

```
==> cat /informix/envia_pagers/pager_dbs_lleno.sh
#####
# 1-AGO-00, 08-mar-2006
#
#####
export DBDATE=dmy4/
export INFORMIXDIR=/informix
export INFORMIXSERVER=prod_300shm
export INFORMIXSQLHOSTS=/informix/etc/sqlhosts
export INFORMIXTERMCAP=TERMCAP
export ONCONFIG=prod_300shm
export PATH=$INFORMIXDIR/bin:/usr/bin
#
#ESTE SHELL INFORMA DE CHUNKS DE LA BD LLENOS
#
cd /tmp
#
> /tmp/pager_dbs_lleno_asunto.txt
> /tmp/pager_dbs_lleno_mensaje.txt
#
# -----
#
echo '
select name DBSPACE,
       round((sum(chksize)-sum(nfree))/sum(chksize)*100) PORC_USADO
  from sysdbspaces d , syschunks c
 where d.dbsnum = c.dbsnum
       and
       name not in ("rllog1","rllog2","rllog3","rllog4","rllog5")
 group by name
 into temp jvv_fkrr ;

unload to "/tmp/dbs_lleno.lst"
select dbspace, porc_usado
  from jvv_fkrr
 where porc_usado >= 95
 order by porc_usado desc ; ' > /tmp/dbs_lleno.sql

dbaccess sysmaster - < /tmp/dbs_lleno.sql
#
full=`grep -c r /tmp/dbs_lleno.lst`
if [ $full -gt 0 ]
 then
   for i in `cat /tmp/dbs_lleno.lst`
   do
    chk=`echo "$i"|cut -d "|" -f 1`
    spa=`echo "$i"|cut -d "|" -f 2`
    echo "SIIC CHUNKS CASI LLENOS" > /tmp/pager_dbs_lleno_asunto.txt
    echo "\nDBSPACE $chk \tESTA LLENO AL $spa % `date +%d/%B-%H:%M` \n" >> /t
mp/pager_dbs_lleno_mensaje.txt
   done
   rcp /tmp/pager_dbs_lleno_asunto.txt prueba:/tmp
   rcp /tmp/pager_dbs_lleno_mensaje.txt prueba:/tmp
   exit
 fi
#
DIA=`date +%u`
export DIA
HORA=`date +%H`
export HORA
MINUTOS=`date +%M`
export MINUTOS
#
if [ $HORA -eq 10 ] && [ $MINUTOS -eq 30 ]
 then
  echo "SIIC OK DBSPACE's CAPACIDAD" > /tmp/pager_dbs_lleno_asunto.txt
  echo "SIIC OK DBSPACE's CAPACIDAD" > /tmp/pager_dbs_lleno_mensaje.txt
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

```
rcp /tmp/pager_dbs_lleno_asunto.txt prueba:/tmp
rcp /tmp/pager_dbs_lleno_mensaje.txt prueba:/tmp
fi
```

Para anticiparse a que se llene un dbspace se cuenta con otro monitoreo automatizado, que nos genera un reporte de los dbspaces con los que cuenta la base de datos, revisando el mismo podemos ver la variación de la capacidad disponible día con día, el cual se ejecuta una vez al día, por medio de lo siguiente:

```
40 15 * * 1,4,5 /informix/dbs_rpt/dbs_rpt.sh 1>/tmp/dbs_rpt.bit 2>&1
```

Una notificación vía correo es la siguiente:

PAGINA : 1 08/09/2006 17:40:00

REPORTE DE LOS DBSPACES DEL SERVIDOR : prod\_300shm EN EL EQUIPO : siicl

DBSPACE	TAMANO_MB	LIBRE_MB	% USADO	#_CHK's
rllog5	2,048	16	99	1
rllog1	2,048	16	99	1
rllog2	2,048	16	99	1
rllog3	2,048	102	95	1
rllog4	2,048	102	95	1
rlgran7	2,048	141	93	1
rltab1	18,432	1,633	91	9
rlgran3	2,048	175	91	1
rlgran10	2,048	215	89	1
rltab3	2,048	249	88	1
rlgran5	2,048	272	87	1
rlgran6	2,048	404	80	1
rlidx1	2,048	403	80	1
rlidx9	2,048	445	78	1
rlidx3	2,048	455	78	1
rlidx8	2,048	458	78	1
rlidx5	2,048	463	77	1
rlidx6	2,048	496	76	1
rltab4	2,048	527	74	1
rlidx2	2,048	609	70	1
rlidx7	2,048	616	70	1
rlgran11	2,048	637	69	1
rlgran19	2,048	634	69	1
rlgran1	2,048	647	68	1
rlgran20	2,048	683	67	1
rlgran17	2,048	699	66	1
rlidx4	2,048	718	65	1
rlgran9	3,072	1,085	65	2
rltab6	2,048	729	64	1
rlgran15	3,072	1,138	63	2
rlgran14	4,096	1,611	61	2
rlgran13	2,048	805	61	1
rlgran4	2,048	808	61	1
rlgran12	2,048	818	60	1
rlgran23	4,096	1,626	60	2
rlgran18	2,048	866	58	1
rlgran16	4,096	1,718	58	2
rlgran2	2,048	968	53	1
rlgran8	2,048	992	52	1
rltab5	4,096	2,035	50	2
rlgran27	2,048	1,252	39	1
rlgran26	2,048	1,349	34	1
rlgran25	2,048	1,409	31	1
rltab2	2,048	1,461	29	1
rlcdr	2,048	1,543	25	1
rootdbs	4,096	3,870	06	2

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

rltemp3	2,048	1,970	04	1
rltemp2	2,048	1,986	03	1
rltemp1	2,048	1,985	03	1
rltemp4	2,048	1,985	03	1
ESPACIO RESERVADO	:	131,072 MB		
ESPACIO LIBRE	:	45,840 MB		
PROMEDIO DE OCUPACION	:	63 %		

En el reporte anterior podemos observar que el dbspace rlgran7 esta al 93 % de ocupación, por lo cual se debe de revisar, si algunas tablas se pueden mover a otros dbspaces, en caso contrario se debe de tener listo un chunk para anexárselo.

El código del reporte dbs\_rpt.sh es el siguiente:

```
==> cat *.sh
#####
#
#
#####
EQUIPO=`uname -a | awk -F" " ' { printf "%s",$2 } ' `
export EQUIPO
#
rm /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
#
FECHA=`date +%d_%m_%Y`
export FECHA
#
#
SERVIDORES="prod_300shm"
export SERVIDORES
#
cd /informix/dbs_rpt
#
for i in `echo $SERVIDORES`
do
#
#
INFORMIXSERVER=$i
export INFORMIXSERVER
#
INFORMIXDIR=/informix
export INFORMIXDIR
INFORMIXSQLHOSTS=$INFORMIXDIR/etc/sqlhosts
export INFORMIXSQLHOSTS
PATH=$INFORMIXDIR/bin:/usr/bin
export PATH
echo $INFORMIXSERVER
echo $INFORMIXDIR
echo $INFORMIXSQLHOSTS
echo $PATH
#
/informix/bin/sacego dbs_rpt $i $EQUIPO
echo "1 DBSPACES RPT $EQUIPO " > /tmp/pager_asunto_chk-rpt-$EQUIPO.lst
cat dbs_rpt.lst >> /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
#
done
#
chmod 764 /tmp/pager_*chk*-rpt-$EQUIPO.lst
rcp /tmp/pager_asunto_chk-rpt-$EQUIPO.lst sun_correo:/tmp &
rcp /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst sun_correo:/tmp &
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

---

El código del reporte `db_s_rpt.ace` es el siguiente:

```
==> cat db_s_rpt.ace
database sysmaster end

define

param[1] SERVER char(20)
param[2] EQ_UNIX char(20)

end

output report to "db_s_rpt.lst"
top margin 2
end

select name dspace,
       round((sum(chksize)*2048)/1000000) TAMANO_MB ,
       round((sum(nfree)*2048/1000000)) LIBRE_MB,
       round((sum(chksize)-sum(nfree))/sum(chksize)*100) pcused,
       nchunks
from sysdbspaces d , syschunks c
where d.dbsnum = c.dbsnum
-- and name not matches "*log*"
group by name, nchunks
order by 4 desc

end

FORMAT

PAGE HEADER
print "PAGINA : ",pageno using "##", " ",today," ", time
skip 1 lines
print "REPORTE DE LOS DBSPACES DEL SERVIDOR : ",SERVER clipped,
      " EN EL EQUIPO : ",EQ_UNIX clipped
skip 1 lines
print "DBSPACE          TAMANO_MB  LIBRE_MB      % USADO #_CHK's"
skip 1 lines

ON EVERY ROW
print dspace[1,18], " ",
      tamano_mb using "###,##&"," ",
      libre_mb using "###,##&"," ",
      pcused using "&&"," ",
      nchunks using "##"," "

ON LAST ROW
skip 1 lines
print "ESPACIO RESERVADO      : ", total of tamano_mb using "#,###,###"," MB"
print "ESPACIO LIBRE          : ", total of libre_mb using "#,###,###"," MB"
print "PROMEDIO DE OCUPACION : ", avg of pcused using "&&"," %"
print " _____"
end
<informix@siic1> </informix/db_s_rpt>
```

También se tiene automatizado, la generación de otro reporte con información más detallada, esto es al nivel de chunk, mediante el `crontab` del sistema operativo, de acuerdo a lo siguiente:

```
50 15 * * 1,4,5 /informix/chk_rpt/chk_rpt.sh 1>/tmp/chk_rpt.bit 2>&1
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

---

El programa shell chk\_rpt.sh se ejecuta una vez al día, y envía por correo electrónico el siguiente reporte:

```

PAGINA : 1    24/04/2006 17:50:02

CHUNCK's DEL SERVIDOR : prod_300shm EN EL EQUIPO : siicl

DBSPACE          CHUNK                TAMANO_MB  LIBRE_MB  %_USADO  NEXT_CHK
-----
28 rlcdr         29 vgrootdbs/rlcdr           2,048     1,543     25
 2 rlgran1       2 vggrandbs/rlgran1         2,048         647     68
11 rlgran10     11 vggrandbs/rlgran10       2,048         215     89
12 rlgran11     12 vggrandbs/rlgran11       2,048         637     69
13 rlgran12     13 vggrandbs/rlgran12       2,048         818     60
14 rlgran13     14 vggrandbs/rlgran13       2,048         805     61
15 rlgran14     15 vggrandbs/rlgran14       2,048          2    100     42
15 rlgran14     42 vggrandbs/rlgran21       2,048     1,609     21

ESPACIO RESERVADO :    4,096 MB
ESPACIO LIBRE     :    1,611 MB
PROMEDIO DE OCUPACION :    61 %

16 rlgran15     16 vggrandbs/rlgran15       2,048         114     94     62
16 rlgran15     62 vggrandbs/rlgran28       1,024     1,024      0

ESPACIO RESERVADO :    3,072 MB
ESPACIO LIBRE     :    1,138 MB
PROMEDIO DE OCUPACION :    47 %

17 rlgran16     17 vggrandbs/rlgran16       2,048          7    100     49
17 rlgran16     49 vggrandbs/rlgran22       2,048     1,711     16

ESPACIO RESERVADO :    4,096 MB
ESPACIO LIBRE     :    1,718 MB
PROMEDIO DE OCUPACION :    58 %

18 rlgran17     18 vggrandbs/rlgran17       2,048         699     66
19 rlgran18     19 vggrandbs/rlgran18       2,048         866     58
20 rlgran19     20 vggrandbs/rlgran19       2,048         634     69
 3 rlgran2       3 vggrandbs/rlgran2         2,048         968     53
21 rlgran20     21 vggrandbs/rlgran20       2,048         683     67
41 rlgran23     50 vggrandbs/rlgran23       2,048         192     91     51
41 rlgran23     51 vggrandbs/rlgran24       2,048     1,434     30

ESPACIO RESERVADO :    4,096 MB
ESPACIO LIBRE     :    1,626 MB
PROMEDIO DE OCUPACION :    61 %

42 rlgran25     52 vggrandbs/rlgran25       2,048     1,409     31
47 rlgran26     58 vggrandbs/rlgran26       2,048     1,349     34
48 rlgran27     59 vggrandbs/rlgran27       2,048     1,252     39

```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE**

---

PAGINA : 2 24/04/2006 17:50:02

CHUNCK's DEL SERVIDOR : prod\_300shm EN EL EQUIPO : siicl

DBSPACE	CHUNK	TAMANO_MB	LIBRE_MB	%_USADO	NEXT_CHK		
4	rlgran3	4	vggrandbs/rlgran3	2,048	175	91	
5	rlgran4	5	vggrandbs/rlgran4	2,048	808	61	
6	rlgran5	6	vggrandbs/rlgran5	2,048	272	87	
7	rlgran6	7	vggrandbs/rlgran6	2,048	404	80	
8	rlgran7	8	vggrandbs/rlgran7	2,048	141	93	
9	rlgran8	9	vggrandbs/rlgran8	2,048	992	52	
10	rlgran9	10	vggrandbs/rlgran9	2,048	61	97	61
10	rlgran9	61	vggrandbs/rlgran28	1,024	1,024	0	
ESPACIO RESERVADO : 3,072 MB							
ESPACIO LIBRE : 1,085 MB							
PROMEDIO DE OCUPACION : 49 %							
36	rlidx1	43	vgidxdb/rlidx1	2,048	403	80	
37	rlidx2	44	vgidxdb/rlidx2	2,048	609	70	
38	rlidx3	45	vgidxdb/rlidx3	2,048	455	78	
39	rlidx4	46	vgidxdb/rlidx4	2,048	718	65	
40	rlidx5	47	vgidxdb/rlidx5	2,048	463	77	
43	rlidx6	53	vgidxdb/rlidx6	2,048	496	76	
44	rlidx7	54	vgidxdb/rlidx7	2,048	616	70	
45	rlidx8	55	vgidxdb/rlidx8	2,048	458	78	
46	rlidx9	56	vgidxdb/rlidx9	2,048	445	78	
29	rllog1	30	vgrootdb/rllog1	2,048	16	99	
30	rllog2	31	vgrootdb/rllog2	2,048	16	99	
33	rllog3	35	vgrootdb/rllog3	2,048	102	95	
34	rllog4	36	vgrootdb/rllog4	2,048	102	95	
35	rllog5	37	vgrootdb/rllog5	2,048	16	99	
22	rltab1	22	vgtabdb/rltab1	2,048	3	100	23
22	rltab1	23	vgrootdb/rllee1	2,048	20	99	34
22	rltab1	34	vgtabdb/rltab7	2,048	37	98	38
22	rltab1	38	vgtabdb/rltab8	2,048	76	96	39
22	rltab1	39	vgtabdb/rltab9	2,048	31	98	40
22	rltab1	40	vgtabdb/rltab10	2,048	4	100	41
22	rltab1	41	vgtabdb/rltab11	2,048	4	100	48

PAGINA : 3 24/04/2006 17:50:02

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

CHUNCK's DEL SERVIDOR : prod\_300shm EN EL EQUIPO : siic1

DBSPACE	CHUNK	TAMANO_MB	LIBRE_MB	%_USADO	NEXT_CHK
22 rltab1	48 vgtabdb/rltab12	2,048	2	100	57
22 rltab1	57 vgtabdb/rltab13	2,048	1,455	29	
ESPACIO RESERVADO : 18,432 MB ESPACIO LIBRE : 1,632 MB PROMEDIO DE OCUPACION : 91 %					
23 rltab2	24 vgtabdb/rltab2	2,048	1,461	29	
24 rltab3	25 vgtabdb/rltab3	2,048	249	88	
25 rltab4	26 vgtabdb/rltab4	2,048	527	74	
26 rltab5	27 vgtabdb/rltab5	2,048	107	95	60
26 rltab5	60 vgtabdb/rltab14	2,048	1,928	6	
ESPACIO RESERVADO : 4,096 MB ESPACIO LIBRE : 2,035 MB PROMEDIO DE OCUPACION : 51 %					
27 rltab6	28 vgtabdb/rltab6	2,048	729	64	
31 rltemp1	32 vgtabdb/rltemp1	2,048	1,985	3	
32 rltemp2	33 vgtabdb/rltemp2	2,048	1,986	3	
49 rltemp3	64 vgtabdb/rltemp3	2,048	1,970	4	
50 rltemp4	65 vgtabdb/rltemp4	2,048	1,985	3	
1 rootdb	1 vgtabdb/rlrootdb	2,048	1,822	11	63
1 rootdb	63 vgtabdb/rlrootdb2	2,048	2,048	0	
ESPACIO RESERVADO : 4,096 MB ESPACIO LIBRE : 3,870 MB PROMEDIO DE OCUPACION : 6 %					
<b>TOTALES</b> ESPACIO RESERVADO : 131,072 MB ESPACIO LIBRE : 45,839 MB PROMEDIO DE OCUPACION : 64 %					

En el reporte anterior se obtiene información a más detalle de los dbspaces, es decir cuantos chunks tienen. Se debe de evitar dbspaces con demasiados chunks, pues se pueden generar dbspaces con demasiados accesos al mismo. En caso de que no se pueda evitar lo anterior, se deben de generar los chunks en discos diferentes.

El código del programa shell chk\_rpt.sh es el siguiente:

```

==> cat /informix/chk_rpt/chk_rpt.sh
#####
#
#
#
#####
EQUIPO=`uname -a | awk -F" " ' { printf "%s",$2 } '`
export EQUIPO
#
FECHA=`date +%d_%m_%Y`

```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

---

```
export FECHA
#
SERVIDORES="prod_300shm" # UN SOLO SERVIDOR
export SERVIDORES
#
cd /informix/chk_rpt
#
rm pager*.lst
> chk_rpt_$EQUIPO.lst
#
for i in `echo $SERVIDORES`
do
#
#
INFORMIXSERVER=$i
export INFORMIXSERVER
#
INFORMIXDIR=/informix
export INFORMIXDIR
INFORMIXSQLHOSTS=$INFORMIXDIR/etc/sqlhosts
export INFORMIXSQLHOSTS
PATH=$INFORMIXDIR/bin:/usr/bin
export PATH
echo $INFORMIXSERVER
echo $INFORMIXDIR
echo $INFORMIXSQLHOSTS
echo $PATH
#
/informix/bin/saceprep chk_rpt $i
/informix/bin/sacego chk_rpt $i $EQUIPO
echo "1 CHUNKS RPT $EQUIPO" > pager_asunto_chk-rpt-$EQUIPO.lst
cat chk_rpt.lst >> pager_mensaje_chk-rpt-$EQUIPO.lst
#
done
#
rcp pager_asunto_chk-rpt-$EQUIPO.lst sun_correo:/tmp &
rcp pager_mensaje_chk-rpt-$EQUIPO.lst sun_correo:/tmp &
#
```

El código del reporte **chk\_rpt.ace** es el siguiente:

```
<informix@siic1> </informix/chk_rpt>
==> cat chk_rpt.ace
database sysmaster end

define

param[1] SERVER char(20)
param[2] EQ_UNIX char(20)

end

output report to "chk_rpt.lst"
top margin 2
end

select d.dbsnum,
       d.name ,
       c.chknum,
       c.fname,
       c.nxchknum,
       round((sum(c.chksize)*2048)/1000000) TAMANO_MB ,
       round((sum(c.nfree)*2048/1000000)) LIBRE_MB,
       round((sum(c.chksize)-sum(c.nfree))/sum(c.chksize)*100) pcused
from sysdbspaces d , syschunks c
```



## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 3 - MONITOREO DEL I/O POR DBSPACE

---

```
where d.dbsnum = c.dbsnum
-- and name not matches "*log*"
group by d.dbsnum, d.name, c.chknum, c.fname,c.nxchknum
order by name, chknum
-- order by dbsnum, chknum

end

FORMAT

PAGE HEADER
print "PAGINA : ",pageno using "##", " ",today," ", time
skip 1 lines
print "CHUNCK's DEL SERVIDOR : ",SERVER clipped,
      " EN EL EQUIPO : ",EQ_UNIX clipped
skip 1 lines
print "DBSPACE          CHUNK          TAMANO_MB  ",
      "LIBRE_MB  %_USADO NEXT_CHK"
skip 1 lines

ON EVERY ROW
print dbsnum      using "##", " ",
      name[1,10], " ",
      chknum      using "##", " ",
      fname[6,28], " ",
      tamaño_mb   using "###,##&"," ",
      libre_mb    using "###,##&"," ",
      pcused      using "##&"," ",
      nxchknum    using "##"

AFTER GROUP OF name
-- AFTER GROUP OF dbsnum
skip 1 lines
if (group count > 1) then
begin
print "ESPACIO RESERVADO      : ", group total of tamaño_mb using "#,###,###"," M
B"
print "ESPACIO LIBRE          : ", group total of libre_mb   using "#,###,###"," M
B"
print "PROMEDIO DE OCUPACION :   ", group avg of pcused using "##&"," %"
skip 1 lines
end

ON LAST ROW
skip 1 lines
print "TOTALES                    "
print "ESPACIO RESERVADO      : ", total of tamaño_mb using "#,###,###"," MB"
print "ESPACIO LIBRE          : ", total of libre_mb   using "#,###,###"," MB"
print "PROMEDIO DE OCUPACION :   ", avg of pcused using "##&"," %"
print " _____ "
end
```

## CAPITULO 4 MONITOREO DE LAS TABLAS

### 4.1 La seguridad en el acceso a las tablas

La confiabilidad y la integridad de la información en cualquier sistema de información, deben de estar garantizadas en todo momento, una de las actividades para lograr lo anterior es el establecimiento de las políticas de acceso a las tablas de la base de datos.

En el sistema SIIC se tienen 4 niveles de seguridad: El 1ro es al nivel de la aplicación, mediante los perfiles de usuario, se otorgan los permisos al usuario de los módulos a los cuales únicamente tendrá derecho de ingresar, en este perfil sé específica a detalle, si el usuario podrá consultar, actualizar, añadir o borrar información.

El 2do nivel de seguridad es a través del monitoreo de las transacciones no autorizadas, para usuarios que intenten acceder la información de la base de datos central, desde equipos remotos, que no pertenezcan al sistema SIIC, el cual fue comentado a detalle en el punto 3.7 'Transacciones no autorizadas' del Capítulo 3 'Monitoreo de las Transacciones'.

El 3er nivel esta dirigido a la integridad de la información mediante el uso de los constrains referenciales o llaves foráneas. Las cuales se implementan al momento de crear las tablas, el sistema SIIC actualmente cuenta con 400 tablas y 694 constrains referenciales.

El 4to nivel es la implementación de los permisos de acceso por cada tabla, para cada usuario, el siguiente listado contiene las instrucciones SQL, para implementar estas políticas de acceso en 3 tablas:

```
{ TABLE "informix".scmae_saldo_cte row size = 73 number of columns = 12 index
size = 34 }
create table "informix".scmae_saldo_cte
(
  icte char(10) not null ,
  xforma_pago char(2) not null ,
  imoneda char(1) not null ,
  msaldo money(13,2),
  mcomp money(11,2),
  msaldo_pemex money(11,2),
  mcomp_pemex money(11,2),
  mint_moratorio money(11,2),
  iprog char(8) not null ,
  falta date not null ,
  halta datetime hour to second not null ,
  iusr_alta char(8) not null ,
  check ((xforma_pago = 'CO' ) OR (xforma_pago = 'CR' ) ),
  primary key (icte,xforma_pago,imoneda)
) extent size 50000 next size 25000 lock mode page;
revoke all on "informix".scmae_saldo_cte from "public";
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 4 - MONITOREO DE LAS TABLAS**

---

```
create index "informix".abr04_01_jvv on "informix".scmae_saldo_cte
(xforma_pago);

grant update on "informix".scmae_saldo_cte to "pcmo0300" as "informix";
grant insert on "informix".scmae_saldo_cte to "pcmo0300" as "informix";
grant delete on "informix".scmae_saldo_cte to "pcmo0300" as "informix";
grant index on "informix".scmae_saldo_cte to "pcmo0300" as "informix";
grant select on "informix".scmae_saldo_cte to "public" as "informix";

{ TABLE "informix".scdet_fact row size = 46 number of columns = 11 index size =
39 }
create table "informix".scdet_fact
(
  iconc_cat_docto char(3) not null ,
  icto_entr smallint not null ,
  idepto_entr integer not null ,
  fentr date not null ,
  kfolio_rem integer not null ,
  icargo_cr char(5) not null ,
  icto_fact smallint,
  unitario_cargo_cr decimal(12,5),
  mcargo_cr money(11,2),
  kdesg_fact smallint,
  xreplica char(1) not null ,

  check (xreplica IN ('I' , 'U' , 'D' , 'E' , 'A' ))
) with rowids
fragment by expression
((icto_entr >= 600 ) AND (icto_entr <= 650 ) ) in rlgran21
,
((icto_entr >= 651 ) AND (icto_entr <= 699 ) ) in rlgran22
,
remainder in rlgran19
extent size 16 next size 50000 lock mode page;
revoke all on "informix".scdet_fact from "public";

create unique index "informix".may0801a on "informix".scdet_fact
(iconc_cat_docto, icto_entr, idepto_entr, fentr, kfolio_rem, icargo_cr);

grant update on "informix".scdet_fact to "pcmo0300" as "informix";
grant insert on "informix".scdet_fact to "pcmo0300" as "informix";
grant delete on "informix".scdet_fact to "pcmo0300" as "informix";
grant index on "informix".scdet_fact to "pcmo0300" as "informix";
grant select on "informix".scdet_fact to "public" as "informix";
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 4 - MONITOREO DE LAS TABLAS**

---

```
{ TABLE "informix".sccat_cto row size = 194 number of columns = 9 index size =
9 }
create table "informix".sccat_cto
(
  icto smallint not null constraint "informix".n146_452,
  ecorta varchar(15) not null constraint "informix".n146_453,
  elarga varchar(50) not null constraint "informix".n146_454,
  ient_fed smallint not null constraint "informix".n146_455,
  impio smallint not null constraint "informix".n146_456,
  isiglas varchar(2) not null constraint "informix".n146_457,
  dcto varchar(100) not null constraint "informix".n146_458,
  xzona char(2) not null constraint "informix".n146_459,
  kdir_ip char(15) not null ,

  check (xzona IN ('TO' , 'TV' , 'TS' , 'TN' , 'TC' , 'GO' , 'GV' , 'GS' , 'GN' , 'GC'
, 'OF' , 'DT' , 'DO' , 'AT' , 'AO' , 'PT' , 'PO' , 'MT' , 'MO' )),
  primary key (icto) constraint "informix".u146_451
) extent size 16 next size 16 lock mode page;
revoke all on "informix".sccat_cto from "public";

grant update on "informix".sccat_cto to "pcmo0300" as "informix";
grant insert on "informix".sccat_cto to "pcmo0300" as "informix";
grant delete on "informix".sccat_cto to "pcmo0300" as "informix";
grant index on "informix".sccat_cto to "pcmo0300" as "informix";
grant select on "informix".sccat_cto to "public" as "informix";
```

## 4.2 Actualización de las estadísticas de las tablas

En una base de datos que soporta un sistema de información, con cientos de usuarios, generando miles de transacciones cada día, las cuales afectan la información contenida en las tablas, es frecuente que el número de registros en las tablas aumente día con día, en el caso del sistema SIIC, existe una tabla en la cual se agregan 100,000 registros por día.

Todas las instrucciones SQL que mandan a ejecutar las aplicaciones de los usuarios a través del sistema, al ejecutarlas la base de datos, previamente pasan por un optimizador de instrucciones SQL, la función de este optimizador es elegir el camino más óptimo de acceder la información requerida por la instrucción SQL, la cual forma parte de una transacción.

Las alternativas posibles para acceder los datos en una tabla pueden ser por un índice en particular, por la llave primaria, por un índice compuesto y finalmente por acceso secuencial, todo depende de cómo se estructure la instrucción SQL en particular.

El optimizador se basa fundamentalmente en el histograma de la(s) tabla(s) involucradas por la instrucción SQL y en la estructura del mismo. Por lo cual es muy importante, pero muy importante el tener las estadísticas de las tablas siempre actualizadas.

Podemos conocer las estadísticas y/o histograma de una tabla en particular con el siguiente comando:

```
dbschema -d sis_com -t scmae_cond_vta -hd all
```

El tener desactualizadas los histogramas de las tablas, en una base de datos muy dinámica y con cientos de usuarios accediéndola, generan inmediatamente problemas de desempeño muy importantes. El resultado de actualizar los histogramas de las tablas es casi mágico, pues los tiempos de respuesta disminuyen dramáticamente.

Por ejemplo, en ocasiones ha pasado que un proceso tarda 10 horas, se actualizan los histogramas de las tablas, se vuelve a ejecutar el proceso y termina en 1 hora.

Por lo anterior se tiene automatizado la actualización de los histogramas de las tablas de la base de datos mediante lo siguiente:

```
1 23 * * 0,4,6 /informix/bin/updstat.sh 1> /tmp/updstat.bit 2>&1
```

El programa shell se programa para que se ejecute a las 11:00 PM de los días: domingo, miércoles y viernes. El proceso genera una bitácora como la siguiente:

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 4 - MONITOREO DE LAS TABLAS**

---

23/04/2006 07:04:54 CDT

```
Table Name      :Index Name      :Column Name
-----:-----:-----
```

```
<< Columns heading an index. >>
scmt_nota_emb_est : 233_1412      :icto_emb
<< Columns NOT heading an index. >>
scmt_nota_emb_est : 233_1412      :femb
scmt_nota_emb_est : 233_1412      :idepto_emb
scmt_nota_emb_est : 233_1412      :kfolio_emb
```

```
update statistics low for table scmt_nota_emb_est drop distributions;
update statistics medium for table scmt_nota_emb_est distributions only;
update statistics high for table scmt_nota_emb_est(icto_emb) distributions
only
;
```

23/04/2006 07:04:54 CDT

```
Table Name      :Index Name      :Column Name
-----:-----:-----
```

```
<< Columns heading an index. >>
scmh_mov_inv    : 180_709      :icto
<< Columns NOT heading an index. >>
scmh_mov_inv    : 180_709      :fbalance
scmh_mov_inv    : 180_709      :iconsecutivo
scmh_mov_inv    : 180_709      :ipres
scmh_mov_inv    : 180_709      :iproducto
scmh_mov_inv    : 180_709      :itipo_mov
```

```
update statistics low for table scmh_mov_inv drop distributions;
update statistics medium for table scmh_mov_inv distributions only;
update statistics high for table scmh_mov_inv(icto) distributions only;
```

El codigo del programa shell updstat.sh es el siguiente:

```
export PDQPRIORITY=50
export INFORMIXDIR=/informix
export INFORMIXSERVER=prod_300shm
export INFORMIXSQLHOSTS=/informix/etc/sqlhosts
export ONCONFIG=prod_300shm
export PSORT_NPROCS=2
export PATH=$INFORMIXDIR/bin:$PATH
env | sort | grep -i informix
$INFORMIXDIR/bin/dbs_updstats sis_com
```

El codigo del programa shell dbs\_updstat es el siguiente:

```
==> cat dbs_updstats
#!/bin/ksh
#
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 4 - MONITOREO DE LAS TABLAS**

---

```
#
# Este script llama a su vez al tbl_updstats script para cada una de las tables
# de la bade de datos
#
# Uso:  dbs_updstats database_name

unset dbserver
if [ $# -eq 1 ]
then
  dbserver=$1; export dbserver
fi
until [ ${dbserver} ]
do
  read dbserver?"Tecla nombre de la base : "
  if [ -z "${dbserver}" ]
  then
    echo '>> Debe teclear el nombre de la base de datos. <<'
    exit 1
  fi
done

#
sqlfile=sql$$
result=rs$$
tablelist=tbl$$
#
echo '
select tablename,":" ,nrows from systables
  where tabid > 99
  order by nrows desc; ' >/tmp/${sqlfile}
dbaccess ${dbserver} - </tmp/${sqlfile} >/tmp/${result} 2>/dev/null
grep ':' /tmp/${result} | cut -f1 -d: >/tmp/${tablelist}
#
# Start database report
cat /dev/null >/tmp/dbupdstats.rpt

# loop through the list of tables.
for tblname in `cat /tmp/${tablelist}`
do
echo 'Actualizando Estadisticas para la Tabla: '${tblname}
$INFORMIXDIR/bin/tbl_updstats ${dbserver} ${tblname}
cat /tmp/tblupdstats.rpt >>/tmp/dbupdstats.rpt
echo '=====
' >>/tmp/dbupdstats.rpt
echo ' ' >>/tmp/dbupdstats.rpt
done
#
rm /tmp/${tablelist}
rm /tmp/${sqlfile}
rm /tmp/${result}

exit 0
```

El código del programa shell tbl\_updstats es el siguiente:

```
==> cat tbl_updstats
#!/bin/ksh
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 4 - MONITOREO DE LAS TABLAS

---

```
#
# Script para crear la actualización de estadísticas como se describe en el
# document Guide to 7.1 Feature Enhancements Página 3-133
# el cual asume que todas variables de ambiente estan atualizadas.
#
# Uso: tbl_updstats database_name table_name

unset dbserver
unset tabname
if [ $# -eq 2 ]
then
    dbserver=$1
    tabname=$2
else
    if [ $# -eq 1 ]
    then
        dbserver=$1
        tabname=
    fi
fi

until [ ${dbserver} ]
do
    read dbserver?"Teclee el nombre de la Base de Datos: "
done

until [ ${tabname} ]
do
    read tabname?"Teclee el nombre de la Tabla: "
done

# Para el caso que el nombre se de en minusculas
tabname2=`echo ${tabname} | tr "[A-Z]" "[a-z]"`
tabname=${tabname2}

sqlfile=sql$$
result=r$$

cat /dev/null >/tmp/${result}
echo "select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid
and t.tabid = c.tabid
and colno = part1
order by 1;" >/tmp/${sqlfile}
dbaccess ${dbserver} - </tmp/${sqlfile} >/tmp/${result} 2>/dev/null
result1=r1$$
grep "\:" /tmp/${result} >/tmp/${result1}

echo "
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part2 is not null and colno = part2
union
select tabname||':'||idxname||':'||colname
```



**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 4 - MONITOREO DE LAS TABLAS**

---

```
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part3 is not null and colno = part3
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part4 is not null and colno = part4
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part5 is not null and colno = part5
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part6 is not null and colno = part6
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part7 is not null and colno = part7
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part8 is not null and colno = part8
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part9 is not null and colno = part9
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part10 is not null and colno = part10
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part11 is not null and colno = part11
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part12 is not null and colno = part12
union
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 4 - MONITOREO DE LAS TABLAS**

---

```

select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part13 is not null and colno = part13
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part14 is not null and colno = part14
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part15 is not null and colno = part15
union
select tabname||':'||idxname||':'||colname
from systables t, sysindexes i, syscolumns c
where tabname = '${tabname}'
and t.tabid = i.tabid and t.tabid = c.tabid
and part16 is not null and colno = part16
;" >/tmp/${sqlfile}
dbaccess ${dbserver} - </tmp/${sqlfile} >/tmp/${result} 2>/dev/null
result2=r2$$
grep "\:" /tmp/${result} >/tmp/${result2}
cp /tmp/${result} /tmp/x

# Reporte de índices
rpt=rpt$$
date >/tmp/${rpt}
echo '
Table Name          :Index Name          :Column Name
-----:-----:-----
' >>/tmp/${rpt}
echo '<< Columns heading an index. >> ' >>/tmp/${rpt}
cat /tmp/${result1} >>/tmp/${rpt}
echo '<< Columns NOT heading an index. >> ' >>/tmp/${rpt}
cat /tmp/${result2} >>/tmp/${rpt}
echo ' ' >>/tmp/${rpt}

# Crea declaracion sql para minusculas
us=us$$
col_low=coll$$
cut -f3 -d: /tmp/${result2} | sort | uniq >/tmp/${result}
comm -13 /tmp/${col_high} /tmp/${result} >/tmp/${col_low}
echo ' update statistics low for table '${tabname}' drop distributions;'
>/tmp/${us}

echo ' update statistics medium for table '${tabname}' distributions only; '
>>/tmp/${us}

col_high=colh$$
cut -f3 -d: /tmp/${result1} | sort | uniq >/tmp/${col_high}

for colname in `cat /tmp/${col_high}`

```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
***CAPITULO 4 - MONITOREO DE LAS TABLAS***

---

```
do
echo ' update statistics high for table '${tabname}'('${colname}')
distributions
  only;' >>/tmp/${us}
done

dbaccess ${dbserver} - </tmp/${us}
cat /tmp/${us} >>/tmp/${rpt}

mv /tmp/${rpt} /tmp/tblupdstats.rpt
rm /tmp/${us}
rm /tmp/${sqlfile}
rm /tmp/${result}
rm /tmp/${result1}
rm /tmp/${result2}
rm /tmp/${col_high}
rm /tmp/${col_low}

exit 0
```

### 4.3 Verificación de la estructura interna de las tablas

La corrupción de la estructura interna de las tablas, es un peligro latente en toda base de datos, las causas pueden ser físicas y lógicas. Físicamente se puede dañar un disco, lo que causaría que cuando un proceso trate de acceder la información en ese sector dañado del disco, la base de datos pase al estado "off-line" o se "caiga" como comúnmente se dice.

En el caso del sistema SIIC se tiene implementado todos los discos externos con RAID- 1, por lo cual todos los dbspaces de la base de datos, tienen "mirror" vía el hardware de marca EMC, modelo DMX 1000 con capacidad de 1.2 Tb cada uno, son equipos de alta disponibilidad (99.9%), con lo anterior se previene el daño físico de las tablas de la base de datos del sistema SIIC.

Para prevenir el daño lógico de la estructura interna de alguna tabla, lo que también provocaría que la base de datos de "caiga", se debe de revisar periódicamente la integridad de las tablas y sus respectivos índices y paginas, mediante el siguiente comando:

```
> oncheck -cID sis_com
```

Por lo anterior, se tiene automatizado la verificación interna de las tablas de la base de datos mediante lo siguiente:

```
1 1 * * 0 /informix/bin/revisa_tablas.sh 1> /tmp/revisa_tablas.bit 2>&1
```

El programa shell se programa para que se ejecute a los 1:00 AM cada domingo. Se debe de ejecutar cuando casi nadie esta usando el sistema, pues es un proceso que consume un buen % de los CPU'S. El proceso genera una bitácora por cada tabla como la siguiente:

```
==> oncheck -cI sis_com:scmae_saldo_cte

Validating indexes for sis_com:informix.scmae_saldo_cte...
      Index 288_1830
            Index abr04_01_jvv

==> oncheck -cI sis_com:scmae_saldo_cte

Validating indexes for sis_com:informix.scmae_saldo_cte...
      Index 368_2875
      Index jun03wo_01
      Index jun03fq_01
      Index jun03nw_01
      Index jun03zi_01

==> oncheck -cI sis_com:scdet_fact

Validating indexes for sis_com:informix.scdet_fact...
      Index abr2201
```

El código del programa shell `revisa_tablas.sh` es el siguiente:

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
***CAPITULO 4 - MONITOREO DE LAS TABLAS***

---

```
export PDQPRIORITY=50
export INFORMIXDIR=/informix
export INFORMIXSERVER=prod_300shm
export INFORMIXSQLHOSTS=/informix/etc/sqlhosts
export ONCONFIG=prod_300shm
export PSORT_NPROCS=2
export PATH=$INFORMIXDIR/bin:$PATH
env | sort | grep -i informix
$INFORMIXDIR/bin/oncheck -cI sis_com
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 4 - MONITOREO DE LAS TABLAS**

---

#### 4.4 Acceso secuencial a las tablas

El acceso a la información contenida en las tablas debe ser preferentemente vía los índices, ya sea por la llave única o "primary key", por índices compuestos, por las llaves foráneas, la última opción será que la tabla se lea o acceda secuencialmente.

El acceder las tablas más grandes o más demandadas secuencialmente, genera problemas de desempeño en el sistema inmediatamente. Al identificar las tablas con accesos secuenciales de deben de generar nuevos índices y se debe de acordar con el desarrollador de la aplicación la optimización de las instrucciones SQL que están accedando las tablas mediante el acceso secuencial, esto puede significar cambios en uno o varios programas del sistema.

Por lo anterior se tiene automatizado el proceso de identificar las tablas accedadas secuencialmente en la base de datos mediante lo siguiente:

```
59 23 * * * /informix/seqscans/seqscans.sh 1> /tmp/seqscans.bit 2>&1
```

El programa shell se programa para que se ejecute a las 23:59 PM de todos los días. El proceso genera una bitácora como la siguiente, la cual es enviada a los administradores vía correo electrónico:

```
PAGINA : 1 24/04/006 17:20:01
```

```
TABLAS CON EL MAYOR # DE LECTURAS SECUENCIALES, SERVIDOR: prod_300shm EN  
EL EQUIPO: siicl
```

TABLA	# DE LECTURAS SECUENCIALES
scmae_mensaje_pc	29,772
scmae_cte_folio	10,335
scmae_est_serv	8,756
scdet_mov_saldo	7,865
screl_cte_cob_lin	5,065
scdet_mov_est_pe	4,142
scmae_id_cto	2,407
sccat_cte	1,764
scmae_deposito_cte	1,436
avisos	1,431
scmc_fact_din	858
scmc_nota_emb_din	858
scmc_nota_emb_est	858
scmt_fact_din	852
scmae_cte_cr	715
scmae_saldo_cte	715
sccat_cto	596
scmae_nota_emb_din	521
sccat_titulo_rep	416
sccat_apl	363
scmae_acum_inv	295
scmae_vol_tanque	288
scmt_serv_est	286
scmt_serv_din	286

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 4 - MONITOREO DE LAS TABLAS

---

scdt_serv	286
scmc_serv_din	286
scmt_semaf_exist	147
scmae_fact_din	111

El código del programa shell seqscans.sh es el siguiente:

```
==> cat /informix/seqscans/seqscans.sh
#####
#
# 23-ago-04
#
#####
#
EQUIPO=`uname -a | awk -F" " ' { printf "%s", $2 } ' `
export EQUIPO
#
HORA=`date +%H`
export HORA
#
SERVIDORES="prod_300shm" # UN SOLO SERVIDOR
export SERVIDORES
#
cd /informix/seqscans
#
rm /tmp/pager*_chk-rpt-$EQUIPO.lst
> seqscans.lst
#
for i in `echo $SERVIDORES`
do
#
INFORMIXSERVER=$i
export INFORMIXSERVER
#
INFORMIXDIR=/informix
export INFORMIXDIR
INFORMIXSQLHOSTS=$INFORMIXDIR/etc/sqlhosts
export INFORMIXSQLHOSTS
PATH=$INFORMIXDIR/bin:/usr/bin
export PATH
#
$INFORMIXDIR/bin/saceprep seqscans
$INFORMIXDIR/bin/sacego seqscans $i $EQUIPO
echo "$EQUIPO SEQSCANS LST " > /tmp/pager_asunto_chk-rpt-$EQUIPO.lst
cat seqscans.lst > /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
cat seqscans.lst >> /tmp/seqscans.bit
#
done
#
rcp /tmp/pager_asunto_chk-rpt-$EQUIPO.lst sun_correo:/tmp
rcp /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst sun_correo:/tmp
#
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 4 - MONITOREO DE LAS TABLAS**

---

El código del programa seqscans.ace es el siguiente:

```
==> cat /informix/seqscans/seqscans.ace
database sysmaster end

define

param[1] SERVER char(20)
param[2] EQ_UNIX char(20)

end

output report to "seqscans.lst"
top margin 2
end

select tabname, sum(seqscans) num_lect_secuenc
  from sysptprof
  where tabname not like "sys%"
     and dbsname not like "sys%"
     and seqscans > 0
group by tabname
order by 2 desc

end

FORMAT

PAGE HEADER
print "PAGINA : ",pageno using "##", " ",today using "ddd dd/mm/yyyy", " ",
time
skip 1 lines
print "TABLAS CON EL MAYOR # DE LECTURAS SECUENCIALES, SERVIDOR : ",SERVER
clipp
ed,
  " EN EL EQUIPO : ",EQ_UNIX clipped
skip 1 lines
print "TABLA          # DE LECTURAS SECUENCIALES"
skip 1 lines

ON EVERY ROW
print tabname,"          ",num_lect_secuenc using "###,###,###"

ON LAST ROW
skip 1 lines
print "TOTAL DE LECTURAS SECUENCIALES: ", total of num_lect_secuenc using "###,
###,##&"
end
```



#### 4.5 Los extents de las tablas

Las bases de datos están contenidas en los dbspaces, los dbspaces están contenidos en los chunks, los chunks albergan a las tablas, las tablas están formadas por páginas de 2048 bytes, esta es la unidad mínima de almacenamiento en cualquier base de datos, un conjunto de páginas contiguas se le denomina '**extent**'.

Las grandes tablas de una base de datos mientras menos extent's las conformen, menor será el tiempo de accesos a los datos contenidos en las mismas, pues los datos estarán agrupados en paginas contiguas, Uno de los atributos de las tablas es el numero de extents que la conforman, otros son: tamaño del registro, numero de registros, nivel de bloqueo, etc.

El número de extents óptimo que conforman a una tabla, debe de ser menor a 50 extents, las tablas que rebasan este parámetro pudieran generar problemas de desempeño al accederlas, pues los datos estarán más dispersos, por lo cual los procesos tendrán que esperar más tiempo para procesar su información.

Con un sistema como el SIIC, que cuenta con varias tablas que tienen mas de 10 millones de registros cada una, se debe de mantener el número de extents en el limite optimo. Para lo cual se debe de realizar una proyección del crecimiento de las tablas, desde el momento de su creación.

El administrador de la base de datos al recibir las peticiones para la creación de nuevas tablas por parte de los desarrolladores, le deben de notificar el numero inicial de registros de la tabla, el incremento mensual de registros, y cuantos meses o años la información estará en el sistema en producción, antes de ser depurada y enviada al sistema SIIC Histórico, el cual esta en otro equipo, en otra base de datos.

En base a los datos proporcionados por los usuarios, en el tamaño del registro, en el tamaño de los índices y principalmente en el número de registros, el administrador de la base de datos calculara el tamaño del extent inicial de la tabla y el tamaño del siguiente extent.

La tabla `scdet_fact`, tendrá inicialmente 10 millones de registros, cada día crecerá en 100,000 registros, se conservaran los últimos 5 meses, el tamaño del registro es de 46 bytes, el tamaño del índice es de 79 bytes, como es una tabla muy grande se decide separar los datos y los índices, también se decide fragmentar la tabla en varios dbspaces, por lo cual la instrucción SQL para crear la tabla es la siguiente:

```
{ TABLE "informix".scdet_fact row size = 46
number of columns = 11 index size = 79 }
create table "informix".scdet_fact
(
  iconc_cat_docto char(3) not null ,
  icto_entr smallint not null ,
  idepto_entr integer not null ,
  fentr date not null ,
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 4 - MONITOREO DE LAS TABLAS**

---

```
kfolio_rem integer not null ,
icargo_cr char(5) not null ,
icto_fact smallint,
munitario_cargo_cr decimal(12,5),
mcargo_cr money(11,2),
kdesg_fact smallint,
xreplica char(1) not null ,

check (xreplica IN ('I' , 'U' , 'D' , 'E' , 'A' ))
) with rowids
fragment by expression
((icto_entr >= 600 ) AND (icto_entr <= 610 ) ) in rlgran3 ,
((icto_entr >= 611 ) AND (icto_entr <= 620 ) ) in rltab3 ,
((icto_entr >= 621 ) AND (icto_entr <= 630 ) ) in rlgran4 ,
((icto_entr >= 631 ) AND (icto_entr <= 640 ) ) in rltab4 ,
((icto_entr >= 641 ) AND (icto_entr <= 650 ) ) in rlgran5 ,
((icto_entr >= 651 ) AND (icto_entr <= 660 ) ) in rltab5 ,
((icto_entr >= 661 ) AND (icto_entr <= 670 ) ) in rlgran6 ,
((icto_entr >= 671 ) AND (icto_entr <= 680 ) ) in rltab6 ,
((icto_entr >= 681 ) AND (icto_entr <= 690 ) ) in rlgran7 ,
((icto_entr >= 691 ) AND (icto_entr <= 695 ) ) in rlgran8 ,
((icto_entr >= 696 ) AND (icto_entr <= 699 ) ) in rlgran9 ,
remainder in rltab2
extent size 92000 next size 30000 lock mode row;

revoke all on "informix".scdet_fact from "public";

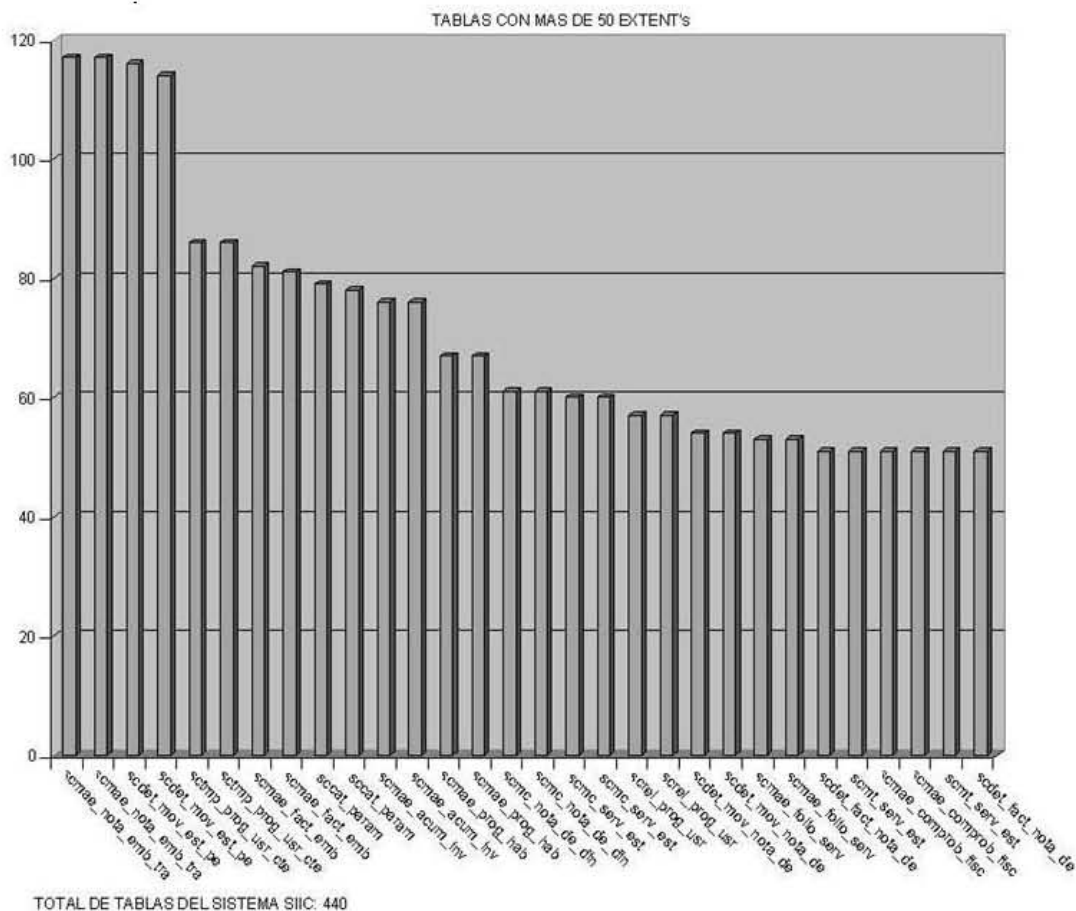
create unique index "informix".abr2201 on "informix".scdet_fact
(iconc_cat_docto,icto_entr,idepto_entr,fentr,kfolio_rem,icargo_cr)

fragment by expression
((icto_entr >= 600 ) AND (icto_entr <= 620 ) ) in rlidx1,
((icto_entr >= 621 ) AND (icto_entr <= 640 ) ) in rlidx2,
((icto_entr >= 641 ) AND (icto_entr <= 660 ) ) in rlidx3,
((icto_entr >= 661 ) AND (icto_entr <= 680 ) ) in rlidx4,
((icto_entr >= 681 ) AND (icto_entr <= 699 ) ) in rlidx5,
remainder in rlgran8 ;
```

De la instrucción anterior el extent size sé definió de 92000 paginas, que son 188 MB (92000\*2048) en cada dbspace, la tabla se fragmento en 11 dbspaces diferentes, por lo tanto se reservan 2068 MB (188 Mb \*11), capacidad suficiente para almacenar la información de esta tabla, con la finalidad de mantener un numero reducido de extents.

Este análisis se debe de realizar con las tablas mas grandes del sistema, a la creación de cada una, cualquier desviación en las proyecciones iniciales de cada tabla se deben de estar monitoreando, para lo cual se deben de revisar el numero total de extents por cada tabla periódicamente.

MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX  
CAPITULO 4 - MONITOREO DE LAS TABLAS



Gráfica 4.5.a

La grafica anterior identifica las tablas que tienen demasiados extent's, el administrador deberá gradualmente reorganizar esas tablas, primeramente ajustando el tamaño del extent, y posteriormente cambiando la fragmentación o creando índices tipo cluster, para forzar a la recreación de la tabla, todo lo anterior primero se debe de probar en el ambiente de desarrollo, el cual contiene una copia exacta y reciente de la base de datos en producción.

Para finalmente aplicar los cambios en el ambiente de producción, únicamente los días domingo, ya que estos cambios bloquean las tablas y no permiten a los usuarios el acceso a la información, durante el proceso que pueden tardar varias horas.

#### 4.6 La fragmentación de las tablas

La fragmentación se aplica únicamente en las tablas que contendrán más de 1 millón de registros y que sean tablas bastante accedidas por el sistema, es decir por las aplicaciones y/o usuarios.

Existen 2 técnicas de fragmentación: Por "expresión" y por "Round Robin", en sistemas OLTP (On Lline Transaction Processing) como el SIIC se debe de usar la fragmentación por "expresión", la fragmentación por "Round Robin" es para sistemas de información del tipo "Data Ware House" o para sistemas de información "Histórica". La fragmentación se puede aplicar desde la creación de la tabla, o en cualquier momento.

La razón principal para aplicarla en una tabla es evitar la contención de acceso por la gran cantidad de registros contenidos en la misma y por la gran cantidad de procesos de los usuarios que tratan de acceder simultáneamente esa misma tabla, lo cual puede generar problemas de desempeño al sistema en general, cuando se tienen problemas de ese tipo, la fragmentación es una forma eficiente de solucionar el problema de desempeño.

En la fragmentación por "Round Robin" la tabla se divide, por ejemplo en 5 dbspaces, si la tabla tiene 10 millones de registros, cada dbspace contendrá 2 millones de registros cada una. Al realizar una búsqueda en esa tabla, la instrucción SQL podría buscar en los 5 dbspaces, hasta encontrar el registro deseado, esto podría generar problemas de desempeño en sistemas tipo OLTP como en el caso del sistema SIIC.

En la fragmentación del tipo "expresión", la tabla se divide por ejemplo en 5 dbspaces, y a cada dbpace se le asigna un rango, en base al valor del contenido de un campo de la tabla. Al realizar una búsqueda en esa tabla, la instrucción SQL buscará únicamente en el dbpace que contenga el registro que se desea, basándose en el rango que se tiene definido, en este método de fragmentación el número de registros por dbpace no es el mismo.

En el sistema SIIC, que es un típico sistema OLTP, únicamente se usa la fragmentación por "expresión", 53 de las 440 tablas del sistema SIIC están fragmentadas por "expresión".

La siguiente instrucción SQL aplica la fragmentación por "expresión" al momento de crear la tabla, para evitar el problema de la contención:

```
{ TABLE "informix".scdet_acum_sc4 row size = 43 number of columns = 11 index
size = 90 }
create table "informix".scdet_acum_sc4
(
  icto_entr smallint not null ,
  iprod integer not null ,
  ipres smallint not null ,
  fentr date not null ,
  ient_fed smallint not null ,
  impio smallint not null ,
  icargo_cr char(3) not null ,
  munitario_cargo_cr decimal(12,5) not null ,
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 4 - MONITOREO DE LAS TABLAS

---

```
        itipo_cons smallint not null ,
        iconc_tipo_cons char(3) not null ,
        mcargo_cr money(11,2),
        primary key
(icta_entr,iproducto,ipres,fentr,ient_fed,impio,icargo_cr,itipo_cons
,iconc_tipo_cons)

) with rowids
fragment by expression
  ((icta_entr >= 600 ) AND (icta_entr <= 620 ) ) in rltab2,
  ((icta_entr >= 621 ) AND (icta_entr <= 640 ) ) in rltab3,
  ((icta_entr >= 641 ) AND (icta_entr <= 660 ) ) in rltab4,
  ((icta_entr >= 661 ) AND (icta_entr <= 680 ) ) in rltab5,
  ((icta_entr >= 681 ) AND (icta_entr <= 699 ) ) in rltab6,
  remainder in rlgran9
extent size 5700 next size 30000 lock mode row;

revoke all on "informix".scdet_acum_sc4 from "public";

alter table "informix".scdet_acum_sc4 add constraint (foreign
  key (icargo_cr) references "informix".sccat_cargo_cr );

alter table "informix".scdet_acum_sc4 add constraint (foreign
  key (icta_entr,iproducto,ipres,fentr,itipo_cons,iconc_tipo_cons,
  ient_fed,impio) references "informix".scmae_acum_sc4 );
```

La siguiente instrucción SQL aplica la fragmentación por "expresión" en una tabla ya existente para resolver un grave problema de contención de tiempos de respuesta al acceder la tabla en cuestión:

```
set pdqpriority 90;

alter table scdet_mov_bco_cte modify next size 11000;

alter fragment on table scdet_mov_bco_cte
init with rowids
  fragment by expression
    iconc_cat_bco = "003" in rlgran9 ,
    iconc_cat_bco = "006" in rlgran10 ,
    iconc_cat_bco = "010" in rlgran11 ,
    remainder in rlgran15 ;
```

La primera línea 'set pdqpriority' hace uso de la funcionalidad del paralelismo del servidor de la base de datos, al usarlo la instrucción SQL se ejecutara en un lapso de tiempo mas corto, por ejemplo de 10 horas a 1 hora, únicamente lo debe de usar el administrador de la base de datos y cuando no hay usuarios en el sistema, pues acapara todos los recursos del servidor de base de datos.

## CAPITULO 5 MONITOREO DE LAS TRANSACCIONES

### 5.1 Transacciones en espera debido a registros y/o tablas bloqueadas

La concurrencia es un evento crítico en una base de datos, y más aun cuando la misma soporta un sistema de información con cientos de usuarios, realizando todo tipo de **transacciones**: almacenando más información, modificando información ya existente, eliminando información obsoleta, generando reportes y/o consultando información.

En este tipo de ambientes operativos, el acceso a la información y la administración de los recursos se vuelve crítica.

En los ambientes orientados a **OLTP** (On Line Transaction Processing), que son los sistemas de información que soportan la operación diaria de todo tipo de empresas (90 %), las bases de datos están diseñadas y construidas tomando el modelo OLTP, el cual debe de permitir a cientos de usuarios concurrentemente realizar las actividades típicas de agregar, modificar, borrar y consultar su información a través del sistema de información con que cuentan los usuarios, ya sea por su red local, por su Intranet o por la Internet.

Hay otro tipo de bases de datos orientadas a DataWare House, en las cuales los sistemas de información están orientados a consultar grandes volúmenes de información **histórica**, estos sistemas de información están dirigidas a pocos usuarios, generalmente de alta dirección en la empresa, únicamente consultan la información, la agrupan y a través de herramientas especializadas de explotación de información tipo "Data Minig" y "Metacubos" pueden manipular la información para generar información muy especializada, tales como "comportamientos, tendencias", etc.

El sistema SIIC es un sistema típico OLTP, por lo tanto uno de los eventos críticos a monitorear es la concurrencia, pues cuando se presentan problemas de este tipo de evento en una tabla muy usada, importante, crítica del sistema, puede llegar a detener a decenas de usuarios que traten de acceder esa misma tabla o registro.

Es decir se puede llegar a detener la operación del sistema, esto es, se puede detener el servicio principal que proporciona el sistema de información, en el caso del sistema SIIC, se detendría la facturación de los productos petrolíferos y afectar el reparto oportuno de los combustibles hacia las 5000 estaciones de servicio o gasolineras en todo el País.

Uno de los casos típicos, es cuando un usuario desde alguna TAD (Terminal de Almacenamiento y Distribución), por ejemplo la TAD de Rosarito, esta realizando un cargo por la cantidad de \$XXXX al cliente YYYY porque se le surtió 20,000 litros de gasolina "Premium", el sistema SIIC tiene que actualizar el registro del cliente en la tabla `scmae_cte_saldo` en el servidor central, si por problemas de red su sesión se corta o interrumpe, la transacción en el servidor central se queda "activa" y el registro afectado

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

permanece "bloqueado" o con un ("candado","lock"), mientras la transacción en el servidor central permanezca "activa", los usuarios de la TAD de Rosarito, no podrán generar la Factura del producto surtido al cliente YYYY.

Si a lo anterior, por un error de diseño en la base de datos, el nivel de bloqueo de la tabla mencionada, `scmae_cte_saldo` esta a nivel de "Table", todo el sistema se **detendría**.

Por todo lo anterior, este tipo de evento de bloqueo de registros, paginas y tablas de la base de datos, en un evento muy importante a monitorear, los administradores se deben enterar de inmediato para terminar con el bloqueo, ya que de lo contrario, se enteraran por las llamadas de los usuarios quejándose de que el sistema no responde normalmente.

El monitoreo manual de este tipo de evento es mediante la siguiente instrucción:

```
> onstat -u | egrep "\-1"
```

```
Informix Dynamic Server Version 7.31.UD7W4 -- On-Line -- Up 2 days 22:11:12 --
1265688 Kbytes
```

```
Userthreads
address flags sessid user tty wait tout locks nreads nwrites
d0543550 L--PR-- 4700 ppr05601 tLc 838c8e48 -1 1 172 0
d4b2e638 L--PR-- 4715 ppr05601 tWd 838c8e48 -1 2 7 0
d4b2fa58 L--PR-- 4696 psu01300 tVd 838c8e48 -1 2 19 0
d4b36908 L--PR-- 4731 psc25300 tHd 8072943c -1 2 19452 232
d4b37d28 L--PR-- 4804 ppr05601 tIe 838c8e48 -1 2 10 0
d4b3d2b0 L--PR-- 4768 cdr - 838c2bf8 -1 1 2565 0
```

El primer indicio de que se tienen problemas de bloqueo y/o concurrencia es cuando algunas transacciones en el servidor de la base de datos en el parámetro **tout** (time out) tienen el valor (-1), lo anterior nos indica que esas sesiones han excedido el tiempo de espera máximo predeterminado (**30 seg.**), para acceder los registros en los cuales la transacción se aplicara.

Para automatizar el monitoreo de este tipo de eventos y ser notificados oportunamente los administradores, ya sea por correo electrónico ó teléfono móvil, sé decidió realizar programas en lenguaje "Shell", los cuales se muestran a continuación:

Se programo la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
5,10,15,20,25,30,35,40,45,50,55,59 * * * * informix/envia_pagers/pager_uso_locks.sh 1>
/tmp/pager_uso_locks.bit 2>&1
```

El programa Shell `pager_uso_locks.sh`, monitorea si hay transacciones con el parámetro **tout** con el valor -1, si ese es el caso enviara un aviso mediante correo electrónico

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

a los administradores del sistema SIIC, el mismo se ejecuta cada 5 minutos, todo el tiempo, su contenido es el siguiente:

```
==> cat /siic_onbar2/monitoreo/logs_lock_to/logs_lock_to.sh
#####
#
#
#####
#
export DBDATE=dmy4/
export INFORMIXDIR=/informix
export INFORMIXSERVER=prod_300shm
export INFORMIXSQLHOSTS=/informix/etc/sqlhosts
export INFORMIXTERMCAP=TERMCAP
export ONCONFIG=prod_300shm
export PATH=$INFORMIXDIR/bin:/usr/bin

cd /siic_onbar2/monitoreo/logs_lock_to
onstat -u | egrep " \-1" 1> /tmp/logs_lock_to.bit 2>&1

tout=`grep -ci "\-1" /tmp/logs_lock_to.bit`
if [ $tout -gt 2 ]
then
    echo "CHECAR LOCKS `date`" > /tmp/pager_logs_lock_asunto.txt
    echo "Hay $tout transacciones con Time Out Revisar" >
/tmp/pager_logs_lock_mensaje.t
xt
    echo "Lock con Time Out `date`">>logs_lock_to.bit
    echo "Locks con Time Out `date`" >>
/tmp/pager_logs_lock_mensaje.txt
    cat /tmp/logs_lock_to.bit|tee -a /tmp/pager_logs_lock_mensaje.txt
1>>logs_l
ock_to.bit

    rcp /tmp/pager_logs_lock_asunto.txt prueba:/tmp
    rcp /tmp/pager_logs_lock_mensaje.txt prueba:/tmp
fi
```



**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES**

---

**5.2 Transacción(es) que esta(n) bloqueando registros y/o tablas**

El detectar las sesiones que están bloqueando registros y/o tablas, y que están deteniendo las transacciones de otros usuarios, no es un proceso directo, por lo cual este proceso no esta automatizado.

Los administradores de sistema, una vez notificados por el evento descrito en el punto 3.1, inicialmente deben de monitorear que sesiones tienen "retenidos, bloqueados" registros y/o tablas en modo exclusivo "HDR+IX", por mas de 1 minuto con el siguiente comando:

**> onstat -kr**

```
Informix Dynamic Server Version 7.31.UD7W4 -- On-Line -- Up 8 days 22:13:49 --
1665688 Kbytes
```

```
Locks
address wtlst  owner   lklist  type    tblsnum rowid   key#/bsiz
80738ab8 0        d309ce10 82379850 IS      1600019 0       0
8154c018 0        cd143958 0       HDR+S   100002 203     0
8154d5a0 0        c34e09d8 0       HDR+S   100002 204     0
ce02bdf8 0        d30991b0 80745508 HDR+IX 1600019 305     0
831b1bc4 0        c9fa6298 85c37170 IS      1600019 0       0
831bb964 0        c9f925a0 831c4d44 IS      1600019 0       0
83fcf370 0        c7da4f10 0       HDR+S   100002 20e     0
85c2ce20 0        d1579480 83ff2958 IS      1600019 0       0
566 active, 2000000 total, 262144 hash buckets
```

**DEFINICIONES DE LAS COLUMNAS**

Encabezado de la Columna	Descripción
Address	La dirección en memoria del candado o "lock".
Wtlst	La dirección del primer "userthread" esperando por este candado o lock.
Owner	La dirección del userthread que tiene el candado o lock.
Lklist	La dirección del próximo candado retenido por esta sesión o userthread. Un cero indica el fin de la lista de candados.
type	Describe el tipo de candado o lock aplicado, puede ser uno o mas de los siguientes valores:  HDR    Header B      Byte lock - S      Shared - Compartido X      Exclusive - Exclusivo I      Intent – Intento de U      Update – Para Modificar IX     Intent exclusive – Intento exclusivo IS     Intent shared – Intento compartido SIX    Shared, intent exclusive
tblsnum	El numero de tablespace donde se intenta el bloqueo.
rowid	El numero de rowid, pagina o 0 (para bloquear toda la tabla).
key#/bsiz	La llave del índice para campos tipo VARCHAR.

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES**

---

**Resumen de los bloqueos y/o locks**

Columna	Descripción
<i>active</i>	El numero de bloqueos, candados, locks presentes en ese momento en el servidor de BD.
total	El numero máximo de candados simultáneos soportados en el servidor de BD, parámetro configurable, máximo 8,000,000.
hash buckets	El numero de "hash buckets" creados para rastrear y administrar los bloqueos.

Un rowid de 0 indica un bloqueo al nivel de toda la tabla, un rowid terminando en 0x00 y 500, representa un bloqueo al nivel de pagina (2.48 Kb) Un rowid con cualquier otro valor, representa un bloqueo a nivel registro, este tipo de bloqueo debe de ser el mas usado.

Por lo menos cada sesión de usuario conectado a la base de datos, debe de tener un bloqueo compartido, en el rowid correspondiente al la tabla tablespace (0x100002).

Modificando el valor del parámetro LOCKS en el archivo de configuración de la base de datos "\$ONCONFIG", se controla el numero máximo de candados en la instancia o servidor de base de datos.

Si un recurso (registro, tabla) esta bloqueado mas del tiempo permitido, el comando onstat -k nos puede proporcionar la dirección de **userthread** que esta bloqueando el recurso. Para conocer el usuario dueño del bloqueo debemos tomar la dirección del userthread y buscarlo en el listado generado por el comando onstat -u, de esta manera conoceremos el usuario y el numero de sesión responsable del bloqueo.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

Informix Dynamic Server Version 7.31.UD7W4 -- On-Line -- Up 8 days 22:11:48 --  
1665688 Kbytes

```
Userthreads
address flags  sessid  user      tty      wait      tout locks nreads  nwrites
c34cf018 ---P--D 1      informix -        0         0    0    319    532
c34de198 ---P--B 28     informix -        0         0    0    121    0
c34df0b0 ---P--D 32     informix -        0         0    0    0      0
cd13b180 B--PR-- 868125 cdr      -        8803b4b8 0    2    4728  0
cd147fc8 ---PR-- 867870 ppc03626 tm      0         0    1    191284 304
ce02bdf8 --BPX-- 867637 psc08300 t4e    0         0    1    349600 217461
598 active, 1408 total, 702 maximum concurrent
```

El administrador de la base de datos deberá tomar la decisión de terminar la sesión determinada, para liberar los recursos bloqueados, y con esto permitir a los otros usuarios terminar sus transacciones. Cuando se bloquean tablas importantes del sistema y/o procesos (ejemplo: Proceso de Facturación) la resolución del problema debe de ser inmediata por parte de los administradores. Para terminar o matar la sesión que esta bloqueando a otros usuarios se usa el siguiente comando:

```
onmode -z #_de_sesion
```

```
onmode -z 867637
```

Para complicar el problema, en muchas ocasiones la tabla supuestamente afectada por el bloqueo no es tal, si no que indirectamente puede ser otra tabla, lo anterior debido a los **constraints referenciales**, también conocidas como **llaves foráneas**, las cuales sirven para que la información en la base de datos sea consistente, esta característica no debe de faltar en las grandes bases de datos actuales. Pues de ello depende en un buen porcentaje, la calidad de la información contenida y administrada por el sistema de información.

Pues bien, una tabla que sirve de catalogo puede tener de 1 a "n" constraints referenciales desde las otras tablas. El sistema SIIC cuenta con 400 tablas, las cuales van aumentando en número, en la medida que se le incorporan más módulos al mismo. También cuenta con 694 llaves foráneas o constraints referenciales, este número nos indica la interrelación e interdependencia existente entre las tablas del sistema.

Por lo anterior, se debe de monitorear a todas las sesiones sospechosas de estar causando contención o problemas de bloqueo por varios minutos y rastrearlas una por una, verificar que instrucciones SQL están realizando, verificar que usuario y con que programa se están generando y basándose en todo lo anterior el administrador de la base de datos podrá decidir que sesión o sesiones deberá terminar.

En sistemas de gran demanda, la contención de los bloqueos puede significar un gran problema, por lo que debe de mantenerse al mínimo por varios razones. Uno de los principales métodos es a través de **isolation levels**, o niveles de aislamiento. Asignado el

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
***CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES***

---

adecuado nivel a una transacción y evitando el uso innecesario de bloqueos o candados o locks.

En sistemas OLTP como el SIIC, se deben de establecer niveles de bloqueo al nivel de registro, esto al construirse la tabla. En caso de que alguna no tenga este nivel, se puede corregir, cuando nadie la este usando.

Los administradores del sistema deben de estar en constante comunicación con los desarrolladores del sistema, para establecer las políticas relacionadas con todo este tema, para que al crear las aplicaciones, las mismas no bloqueen recursos innecesariamente, lo cual puede provocar problemas severos a la operación de un sistema de información cuando se tiene una gran de manda de usuarios, para sistemas de información con muy pocos usuarios todo lo anterior nunca es un problema.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

#### 5.3 Transacciones en proceso

En el servidor de base de datos del sistema SIIC se tienen en promedio 1000 sesiones conectadas a la base de datos, pero únicamente de 20 a 30 sesiones están en estado de "ejecución", es decir "trabajando al mismo tiempo".

Esto varía de segundo a segundo, las restantes sesiones pueden estar esperando por otros recursos administrados por el servidor.

El cambio de estado de las sesiones de "ejecución" a "espera" lo decide el servidor de base de datos de acuerdo a la disponibilidad de los recursos, que cada sesión pretende acceder, modificar, insertar, borrar, consultar.

Por lo que es importante para los administradores de la base de datos, identificar las sesiones que están en estado de "ejecución" en un instante dado, pues estas sesiones pueden ser las que están consumiendo la mayor cantidad de recursos, en caso de problemas de desempeño o "performace" o tiempo de respuesta del sistema SIIC a los usuarios, se necesita identificar que sesiones están ocasionando este evento. Para que los administradores analicen las sesiones sospechosas y en su caso, terminarlas, para que el servidor de la base de datos nuevamente proporcione los tiempos de respuesta adecuados o aceptables a los usuarios.

Una sesión ejecutando una instrucción SQL, mal estructurada puede provocar problemas de desempeño, afectando a todo el sistema por lo cual los administradores del sistema deben de estarlas monitoreado para identificar la sesión, programa, usuario y/o programador, para platicar con ellos y averiguar que es lo que desean realizar en el servidor de la base de datos.

Lo anterior puede generar una modificación en un programa, en un proceso, recomendar la ejecución del mismo en horas de pocos usuarios, por ejemplo las tardes o en las madrugadas, etc.

El monitoreo manual de este tipo de evento es con el siguiente programa shell, el cual se ejecuta continuamente hasta que se interrumpe, a intervalos de cada 5 segundos:

> **vlocks.sh**

```
Informix Dynamic Server Version 7.31.UD7W4 -- On-Line -- Up 10 days 22:01:39 --
1665688 Kbytes
```

Userthreads

address	flags	sessid	user	tty	wait	tout	locks	nreads	nwrites
c34cf018	---P--D	1	informix	-	0	0	0	202	575
c34de198	---P--B	28	informix	-	0	0	0	170	33
c34df0b0	---P--D	32	informix	-	0	0	0	0	0
c7da4f10	---PR--	258	informix	-	0	0	1	202	0

872 active, 1408 total, 974 maximum concurrent

```
Informix Dynamic Server Version 7.31.UD7W4 -- On-Line -- Up 10 days 22:01:44 --
1665688 Kbytes
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

```
Userthreads
address flags sessid user tty wait tout locks nreads nwrites
c34cf018 ---P--D 1 informix - 0 0 0 202 575
c34de198 ---P--B 28 informix - 0 0 0 170 33
c34df0b0 ---P--D 32 informix - 0 0 0 0 0
872 active, 1408 total, 974 maximum concurrent
```

También se pueden identificar las sesiones que realizan más lecturas a la base de datos en un instante dado, mediante la ejecución manual del programa shell **vqml.sh**:

LECTURAS				ESCRITURAS			
04694242	ce01fec8	Y--P---	941141	portal01	-	dae90070	0 1 1853
03712866	d309bef8	Y--P---	936487	portal01	-	d98d6388	0 1 1592
03643905	cd14da58	Y--P---	936486	portal01	-	dd5601d0	0 1 1741
03612021	c34f0a70	Y--P---	936459	portal01	-	d6822018	0 1 1679
02942823	d1575d28	Y--P---	942469	portal01	-	dcd218d8	0 1 951
02009075	d157daf0	Y--P---	948457	pst01671	tnc	dc2641d0	0 1 0
01938851	cd144368	Y--P---	852761	portal01	-	dbe8e280	0 2 47259
01897660	dadcfc40	Y--P---	852758	portal01	-	e021a280	0 1 54909
01829530	cd13ac78	Y--P---	855639	portal01	-	de882cd0	0 1 52768
01749760	ce024538	Y--P---	948587	portal01	-	cd06f3e0	0 1 1028
01675308	dadb0a28	Y--P---	825064	portal01	-	ddd299e0	0 1 52502
01614771	d156b218	Y--P---	950180	portal01	-	d707a018	0 1 1300
01613120	df0c0b28	Y--P---	859348	portal01	-	d6e30a68	0 2 43870
01601516	dadd60e0	Y--P---	855192	portal01	-	ddcec018	0 1 29631
01561478	cf818940	Y--P---	822677	portal01	-	d40faeb0	0 1 54891
01527411	c34e7380	B--PR--	948594	portal01	-	87f1cb2c	0 1 1092
01519647	c9fb3af0	Y--P---	825742	portal01	-	d2df7a10	0 1 49860
01361365	d1565280	Y--P---	950643	psc10300	tte	dd3c9fb8	0 1 15165
01336228	d156d550	Y--P---	853664	portal01	-	dded47a8	0 1 65376
01303278	c7d9db58	Y--P---	950693	portal01	-	d7d847a8	0 1 503

La primera columna es el numero de lecturas realizadas por la sesión, la 4ta columna es el # de sesión, la 5ta columna es el usuario. El código del programa shell es el siguiente:

```
==> cat vqml.sh
#####
# quien_es_el_que_mas_lee.sh
# Este programa muestra que sesiones son las que estan leyendo MAS
# y que estan haciendo ....
# 22-nov-01
#####
#
FECHA=`date +%d_%b_%Y_%H_%M_%S`
export FECHA
#
echo "\n\n PROCESANDO ... \n\n"
#
echo "\nLECTURAS                                ESCRITURAS
\n" > /tmp/tmp_proc_que_mas_leen_$FECHA.lst
onstat -u | egrep -v "Informix|address|active|userthreads" | awk -F" " '{printf
"%08d %s %s %8s %8s %3s %8s %3s %3s %8s\n", $9, $1, $2, $3, $4, $5, $6, $7, $8, $10 }' |
sort -r >> /tmp/tmp_proc_que_mas_leen_$FECHA.lst
#
cat /tmp/tmp_proc_que_mas_leen_$FECHA.lst | awk -F" " '{ printf "%8d\n", $4 }' >
/tmp/tmp_01.lst
#
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES**

---

```
for i in `cat /tmp/tmp_01.lst | head -60`
do
#
echo " PROCESANDO ... $i "
echo " PROCESANDO ... $i " >> /tmp/tmp_proc_que_mas_leen_QUE_$FECHA.lst
onstat -g ses $i >> /tmp/tmp_proc_que_mas_leen_QUE_$FECHA.lst
done
#
echo "\n\n"
cat /tmp/tmp_proc_que_mas_leen_$FECHA.lst | head -60 | more
#
rm /tmp/tmp_???.lst
```

También se pueden identificar las sesiones que realizan mas escrituras a la base de datos en un instante dado, mediante la ejecución manual del programa shell **vqme.sh**:

ESCRITURAS				LECTURAS					
00065098	d156d550	Y--P---	853664	portal01	- dded47a8	0	1	1334606	
00063216	df0c0620	Y--P---	887243	portal01	- dd712070	0	1	1300840	
00055865	d3088708	Y--P---	857465	portal01	- df445670	0	1	1288580	
00054891	cf818940	Y--P---	822677	portal01	- d40faeb0	0	1	1559797	
00054814	c34d63d0	---P--F	0	informix	-	0	0	0	
00054356	dadcfc40	---PR--	852758	portal01	-	0	0	2	1891659
00052502	dadb0a28	Y--P---	825064	portal01	- ddd299e0	0	2	1673436	
00052427	cd13ac78	Y--P---	855639	portal01	- de882cd0	0	1	1825583	
00049584	c9fb3af0	Y--P---	825742	portal01	- d2df7a10	0	1	1516130	
00048587	c34d9118	---P--F	0	informix	-	0	0	0	
00047259	cd144368	Y--P---	852761	portal01	- dbe8e280	0	2	1938851	
00043870	df0c0b28	Y--P---	859348	portal01	- d6e30a68	0	2	1613120	
00032939	c34d4fb0	---P--F	0	informix	-	0	0	0	
00032656	c34d8708	---P--F	0	informix	-	0	0	0	
00032450	df0b8d60	Y--P---	887253	portal01	- df07b6c8	0	2	1048406	
00030186	c34d8c10	---P--F	0	informix	-	0	0	0	
00029599	dadd60e0	Y--P---	855192	portal01	- ddcec018	0	1	1598274	
00029252	d157ea08	Y--P---	855353	portal01	- daf08e88	0	1	634357	
00028304	c34d5ec8	---P--F	0	informix	-	0	0	0	
00026739	c34d59c0	---P--F	0	informix	-	0	0	0	

La primera columna es el numero de escrituras realizadas por la sesión, la 4ta columna es el # de sesión, la 5ta columna es el usuario. El código del programa shell es el siguiente:

```
==> cat vqme.sh
#####
# Este programa muestra que sesiones son las que estan escribiendo MAS
# y que estan haciendo ....
# 22-nov-01
#####
#
FECHA=`date +%d_%b_%Y_%H_%M_%S`
export FECHA
#
echo "\n\n PROCESANDO ... \n\n"
#
echo "\nESCRITURAS                                LECTURAS \
n          " > /tmp/tmp_proc_que_mas_escriben_$FECHA.lst
#
onstat -u | egrep -v "Informix|address|active|userthreads" | awk -F" " '{printf
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

```
"%08d %s %s %8s %8s %3s %8s %3s %3s %8s \n", $10, $1, $2, $3, $4, $5, $6, $7, $8, $9 }' |
sort -r >> /tmp/tmp_proc_que_mas_escriben_$FECHA.lst
#
cat /tmp/tmp_proc_que_mas_escriben_$FECHA.lst | awk -F " " '{ printf "%8d\n", $4
}' > /tmp/tmp_01.lst
#
#
for i in `cat /tmp/tmp_01.lst | head -120`
do
#
echo " PROCESANDO ... $i "
echo " PROCESANDO ... $i " >> /tmp/tmp_proc_que_mas_escriben_QUE_$FECHA.lst
onstat -g ses $i >> /tmp/tmp_proc_que_mas_escriben_QUE_$FECHA.lst
done
#
echo "\n\n"
cat /tmp/tmp_proc_que_mas_escriben_$FECHA.lst | head -120 | more
#
rm /tmp/tmp_???.lst
```

La información obtenida por los comandos anteriores, proporcionan información a los administradores del sistema, de que sesiones puedan estar ocasionando problemas de desempeño en el servidor de base de datos.



## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

#### 5.4 Transacciones con código de error

Las sesiones en ejecución pueden tener problemas al estar ejecutándose en el servidor de base de datos por múltiples razones, los administradores del sistema deben de ser notificados inmediatamente de estos eventos, para resolver el problema.

El comando manual para monitorear este tipo de eventos, es mediante el siguiente comando:

```
> vqtssce.sh
```

```
PROCESANDO ...
```

```
NO EXISTEN SESIONES SQL CON CODIGO DE ERROR
```

Para automatizar el monitoreo de este tipo de eventos y ser notificados oportunamente los administradores, ya sea por correo electrónico ó teléfono móvil, sé decidió realizar programas en lenguaje "Shell", los cuales se muestran a continuación:

Se programo la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
5,10,15,20,25,30,35,40,45,50,55 * * * * /informix/errores_sql_isam/errores_sql_i  
sam.sh 1> /tmp/errores_sql_isam.bit 2>&1
```

El programa Shell `errores_sql_isam.sh`, monitorea si hay transacciones en ejecución con código de error (valor diferente de **0**), si ese es el caso enviara un aviso mediante correo electrónico a los administradores del sistema SIIC, el mismo se ejecuta cada 5 minutos, todo el tiempo, una notificación típica es la siguiente:

```
PAGINA : 1 Tue 24/04/2006 11:35:02
```

```
SESIONES CON ERROR SQL Y/O ISAM, VERIFICAR: SIS_COM EN EL EQUIPO: siic1
```

```
USUARIO EQUIPO          SESION  SQL  ISAM  
INSTRUCCION SQL
```

```
felipe siic1           24631  317  0
```

```
select iusr_c|te,gu_val_portal(kpassword),elarga_raz_soc,icto_cont_cr  
from sccat_usr_cte,sccat_cte  
where iusr_cte matches '00013*'  
and iusr_cte=icte
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

```
cdr      siic1                26491  256    0

select fglgname, fpath, db4glname, dpath from 'valeria '.runner where progname = ?

TOTAL :      2
```

La primera sesión tiene el código de error 317 y la segunda 256, con el comando **finderr**, los administradores del sistema obtienen más información del error, para rastrear las causas y su posible solución:

#### **finderr 317**

-317 Must have the same number of selected columns in each UNION element.

All rows that are produced in a union must have the same format, so each SELECT statement in the union must select the same number of columns. In this union, one of the second or subsequent SELECT statements does not list the same number of columns as the preceding one. Review the entire union, and check that all select lists are alike in number and data type. If no appropriate column exists for one of the statements, specify a literal value of the appropriate type at that position. For example, where you need to match a numeric column, specify a literal zero.

#### **finderr 256**

-256 Transaction not available.

You cannot begin a transaction in this database because it does not have a transaction log. In order to support transactions, you must start a transaction log. Refer to your Administrator's Guide for information on how to start a transaction log.

El código del programa Shell errores\_sql\_isam.sh es el siguiente:

```
<informix@siic1> </informix>
==> cat /informix/errores_sql_isam/errores_sql_isam.sh
#####
#
# 12-JUL-05
#
#####
EQUIPO=`uname -a | awk -F" " ' { printf "%s", $2 } ' `
export EQUIPO
#
rm /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
#
HORA=`date +%H`
export HORA
#
MINUTOS=`date +%M`
export MINUTOS
#
SERVIDORES="prod_300shm"
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

```
export SERVIDORES
#
cd /informix/errores_sql_isam
#
for i in `echo $SERVIDORES`
do
#
INFORMIXSERVER=$i
export INFORMIXSERVER
#
INFORMIXDIR=/informix
export INFORMIXDIR
INFORMIXSQLHOSTS=$INFORMIXDIR/etc/sqlhosts
export INFORMIXSQLHOSTS
PATH=$INFORMIXDIR/bin:/usr/bin
export PATH
echo $INFORMIXSERVER
echo $INFORMIXDIR
echo $INFORMIXSQLHOSTS
echo $PATH
done
#
echo "$EQUIPO ERRORES_SQL_ISAM OK " > /tmp/pager_asunto_chk-rpt-$EQUIPO.lst
echo "$EQUIPO ERRORES_SQL_ISAM OK " > /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
#
echo '
unload to /tmp/error_sql_isam_00.txt
select
    sqs_sessionid,sqs_sqlerror,sqs_isamerror, sqs_statement
from syssqlstat
where sqs_sqlerror <> 0
' > /tmp/error_sql_isam.sql
#
dbaccess sysmaster - < /tmp/error_sql_isam.sql
#
ERRORES_SQL_ISAM=`wc -l /tmp/error_sql_isam_00.txt | cut -d" " -f1`
export ERRORES_SQL_ISAM
#
if [ $ERRORES_SQL_ISAM -gt 0 ]
then
echo "$EQUIPO errores_SQL_ISAM ($ERRORES_SQL_ISAM SESIONES) " > /tmp/pager_asu
nto_chk-rpt-$EQUIPO.lst
echo "$EQUIPO ERRORES SQL_ISAM " > /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
saceprep errsqliam
sacego errsqliam SIS_COM $EQUIPO
cat /tmp/errsqliam.lst >> /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
fi
#
if [ $ERRORES_SQL_ISAM -gt 0 ] || [ "${HORA}" = "10" ] && [ "${MINUTOS}" = "35"
]
then
rcp /tmp/pager_asunto_chk-rpt-$EQUIPO.lst prueba:/tmp &
rcp /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst prueba:/tmp &
fi
#
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

#### 5.5 Transacciones sin terminar del día anterior

El sistema SIIC es un sistema de misión crítica, esto es, debe de estar en operación las 24 horas del día, los 365 días del año, por lo cual las actividades programadas de mantenimiento a cualquier componente del sistema (HW,SW,BD) deben de programarse con anticipación, pues por el servicio proporcionado por el sistema, siempre debe de estar disponible.

Por todo lo anterior la base de datos, no se da de baja, no se apaga, lo que implica que algunas sesiones de los usuarios, permanezcan por más de varios días, consumiendo recursos dentro del servidor, la razón más común es que se haya cortado la sesión de emulación de terminal del usuario y el proceso permanezca en el servidor de base de datos.

Esto es un problema, pues al paso de los días y semanas nos podemos encontrar con decenas de sesiones que ya no deben de estar en el servidor, pues solo están consumiendo recursos innecesariamente.

Para automatizar el monitoreo de este tipo de eventos y ser notificados oportunamente los administradores, ya sea por correo electrónico ó teléfono móvil, sé decidió realizar programas en lenguaje "Shell", los cuales se muestran a continuación:

Se programo la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
==> crontab -l | grep agi
10 12 * * * /informix/login_aging/login_aging.sh 1> /tmp/login_aging.bit 2>&1
```

El programa Shell `login_aging.sh`, monitorea si hay transacciones que sean de días anteriores al actual, si ese es el caso, las terminara y enviara un aviso mediante correo electrónico a los administradores del sistema SIIC, el mismo se ejecuta cada a las 12:00 hrs. de cada día, una notificación típica es la siguiente:

#### SESIONES COLGADAS

```
onmode -z 673466 # gtjo1624 siicl Fri Apr 2 08:18:16 2006
onmode -z 674942 # psr03300 siicl Fri Apr 2 08:35:35 2006
onmode -z 679977 # gtjot637 siicl Fri Apr 2 09:33:53 2006
onmode -z 682214 # gtyal620 madero Fri Apr 2 09:54:00 2006
onmode -z 684385 # gtact627 siicl Fri Apr 2 10:12:01 2006
onmode -z 684444 # psr03300 siicl Fri Apr 2 10:12:28 2006
onmode -z 689809 # ppr05601 siicl Fri Apr 2 10:59:33 2006
onmode -z 690176 # ppr05601 siicl Fri Apr 2 11:02:50 2006
onmode -z 693609 # gtac2602 siicl Fri Apr 2 11:46:52 2006
onmode -z 694533 # gtact631 siicl Fri Apr 2 12:23:08 2006
onmode -z 695609 # gtact627 siicl Fri Apr 2 13:03:22 2006
onmode -z 697605 # gtact647 siicl Fri Apr 2 14:13:17 2006
onmode -z 706055 # gtost624 zacatecas Fri Apr 2 15:56:03 2006
onmode -z 713163 # gtjo1624 siicl Fri Apr 2 17:49:53 2006
onmode -z 713910 # psr04300 siicl Fri Apr 2 18:37:49 2006
onmode -z 713947 # psr04300 siicl Fri Apr 2 18:39:18 2006
onmode -z 717480 # gtjot619 siicl Fri Apr 2 23:45:09 2006
onmode -z 718623 # gtjot619 siicl Sat Apr 3 03:16:35 2006
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES**

```

onmode -z 721491 # gtost647 culiacan Sat Apr 3 08:03:45 2006
onmode -z 722961 # gtjot650 siicl Sat Apr 3 09:26:35 2006
onmode -z 722972 # gtjot650 siicl Sat Apr 3 09:26:58 2006
onmode -z 723457 # ppl02626 siicl Sat Apr 3 09:54:00 2006
onmode -z 723470 # ppl02626 siicl Sat Apr 3 09:54:11 2006
onmode -z 725428 # gtact639 siicl Sat Apr 3 11:54:10 2006
onmode -z 726117 # gtjot674 siicl Sat Apr 3 12:44:12 2006
onmode -z 726115 # gtjot674 siicl Sat Apr 3 12:44:33 2006
onmode -z 726125 # gtact647 culiacan Sat Apr 3 12:45:34 2006
onmode -z 727018 # gtyat639 ensenada Sat Apr 3 13:42:01 2006
onmode -z 727806 # gtac2602 siicl Sat Apr 3 14:26:46 2006
onmode -z 728007 # gtjot624 siicl Sat Apr 3 14:46:01 2006
onmode -z 728468 # gtyat684 siicl Sat Apr 3 14:59:27 2006
onmode -z 728642 # gtyat684 siicl Sat Apr 3 15:10:14 2006
onmode -z 728945 # gtact647 culiacan Sat Apr 3 15:33:48 2006
onmode -z 729611 # gtyat624 zacatecas Sat Apr 3 16:23:35 2006
onmode -z 729826 # gtjot602 siicl Sat Apr 3 16:45:41 2006
onmode -z 730240 # gtos2617 aguascalientes Sat Apr 3 17:09:37 2006
onmode -z 730488 # gtac1672 siicl Sat Apr 3 17:29:17 2006
onmode -z 730817 # gtjot620 madero Sat Apr 3 17:52:33 2006
onmode -z 731702 # gtya1697 barranca Sat Apr 3 19:13:03 2006
onmode -z 732190 # gtya1617 aguascalientes Sat Apr 3 19:56:40 2006
onmode -z 732360 # gtost637 hermosillo Sat Apr 3 20:12:12 2006
onmode -z 741285 # gtact630 zapopan Sat Apr 3 21:40:49 2006
onmode -z 755278 # gtjot602 cadereyta Sat Apr 3 23:36:30 2006
onmode -z 820492 # gtyat693 tula Sun Apr 4 09:30:51 2006
onmode -z 829931 # gtos5602 cadereyta Sun Apr 4 10:31:57 2006
onmode -z 831172 # gtyat602 cadereyta Sun Apr 4 10:40:19 2006
onmode -z 837175 # gtyat689 siicl Sun Apr 4 11:27:49 2006
onmode -z 838615 # ppl01601 siicl Sun Apr 4 11:39:27 2006
onmode -z 838636 # ppl01601 siicl Sun Apr 4 11:39:37 2006
onmode -z 839867 # gtact627 siicl Sun Apr 4 11:49:22 2006
onmode -z 840331 # gtact627 siicl Sun Apr 4 11:53:06 2006
onmode -z 841224 # gtjot602 cadereyta Sun Apr 4 12:02:58 2006
onmode -z 842476 # gtac1672 siicl Sun Apr 4 12:09:58 2006
onmode -z 842482 # gtac1672 siicl Sun Apr 4 12:09:59 2006
onmode -z 842480 # gtac1672 siicl Sun Apr 4 12:09:59 2006
onmode -z 842479 # gtac1672 siicl Sun Apr 4 12:10:00 2006
onmode -z 842483 # gtac1672 siicl Sun Apr 4 12:10:00 2006
onmode -z 842484 # root siicl Sun Apr 4 12:10:00 2006

```

ESTO ES LO QUE ESTABAN HACIENDO ESTAS SESIONES COLGADAS (solo se muestra la primera)

Informix Dynamic Server Version 7.31.UD7W4 -- On-Line -- Up 7 days 14:44:48 -- 1665688 Kbytes

```

session          #RSAM    total    used
id      user    tty      pid      hostname threads memory  memory
673466  gtjo1624 t9b      12554    siicl    1         73728  54360

```

```

tid      name      rstcb  flags  curstk  status
780850  sqlexec  c9f9b788 Y--P--- 39088  cond wait(sm_read)

```

```

Memory pools      count 1
name      class addr      totalsize freesize #allocfrag #freefrag
673466    V      e64dc018 73728    19368    330      14

```

```

name      free      used      name      free      used
overhead  0         128      scb        0         96
opentable 0         4200     filetable 0         856
log        0         2152     temprec    0         1608
keys       0         408      ralloc     0         19096
gentcb     0         8536     ostcb      0         2656
sort       0         56       sqscb      0         8728
rdahead   0         1504     hashfiletab 0         280
osenv      0         1616     sqtcdb     0         2048
fragman    0         392

```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

Sess	SQL	Current	Iso Lock	SQL	ISAM	F.E.
Id	Stmt type	Database	Lvl Mode	ERR	ERR	Vers
673466	-	sis_com	CR Not Wait	0	0	7.20

Last parsed SQL statement :

```
select unique iprog, xdeshabilita, xdias, hini_deshabilita,
hfin_deshabilita from scmae_prog where iprog =?
```

El contenido de programa shell login\_aging.sh es el siguiente:

```
==> cat login_aging*.sh
#####
#
# 22-AGO-01 SESIONES x ANTIGUEDAD
# CUIDADO CON ESTE SHELL PUES MATA SESIONES EN EL SERVIDOR
#
#####
export DBDATE=dmy4/
export INFORMIXDIR=/informix
export INFORMIXSERVER=prod_300shm
export INFORMIXSQLHOSTS=/informix/etc/sqlhosts
export INFORMIXTERMCAP=TERMCAP
export ONCONFIG=prod_300shm
export PATH=$INFORMIXDIR/bin:/usr/bin
#
rm /tmp/pager*chk*
#
EQUIPO=`uname -a | awk -F " " '{ printf "%s", $2 }'`
export EQUIPO
#
HORA=`date +%H`
export HORA
#
MINUTOS=`date +%M`
export MINUTOS
#
cd /informix/login_aging
rm /tmp/login_*
#
echo '
database sysmaster ;
select sid, username, hostname, connected login_time, hex(state) edo
  from sysessions
  where username not in ("informix","portal01","cdr")
order by 4 ;
' > /tmp/login_aging.sql
dbaccess sysmaster - < /tmp/login_aging.sql > /tmp/login_aging.lst
grep x /tmp/login_aging.lst > /tmp/login_aging_x.lst
cat /tmp/login_aging_x.lst | awk -F " " '{ printf "/informix/login_aging/login_aging %s\n", $4 }' \
  >> /tmp/login_aging_fecha.sh
chmod 700 /tmp/login_aging_fecha.sh
/tmp/login_aging_fecha.sh > /tmp/login_aging_fecha.lst
paste -d"\t" /tmp/login_aging_x.lst /tmp/login_aging_fecha.lst > /tmp/login_aging_final_00.lst
HOY=`date +%-%h %02d`
export HOY
echo "\n[$HOY]\n"
grep -v "$HOY" /tmp/login_aging_final_00.lst > /tmp/login_aging_final_01.lst
#
#
cat /tmp/login_aging_final_01.lst|awk -F " " '{printf "onstat -g ses %ld >> /tmp/login_aging_ses_det.lst \n ", $1 }' > /tmp/login_aging_ses.sh
chmod 700 /tmp/login_aging_ses.sh
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

```
/tmp/login_aging_ses.sh
#
cat /tmp/login_aging_final_01.lst|awk -F" " '{printf "onmode -z %8ld # %s %8s %1
0s %8s %s %s %s \n", $1, $2, $3, $6, $7, $8, $9, $10 }' \
> /tmp/login_aging_ses_onmode_z.sh
#
#
SI_HAY=`wc -l /tmp/login_aging_ses_onmode_z.sh | cut -d" " -f1`
#
echo "$EQUIPO OK NO HAY SESIONES COLGADAS DE AYER ">/tmp/pager_asunto_chk-rpt-$E
QUIPO.lst
echo "\n\nNO HAY SESIONES COLGADAS DE AYER [ $SI_HAY ] \n\n" > /tmp/pager_mensaj
e_chk-rpt-$EQUIPO.lst
#
if [ $SI_HAY -gt 0 ] && [ $SI_HAY -lt 100 ]
then
echo "$EQUIPO SESIONES COLGADAS RIP " > /tmp/pager_asunto_chk-rpt-$EQUIPO.lst
echo "\n\nSESIONES COLGADAS\n\n" > /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
cat /tmp/login_aging_ses_onmode_z.sh >> /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
echo "\n\nESTO ES LO QUE ESTABAN HACIENDO ESTAS SESIONES COLGADAS\n\n" \
>> /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
cat /tmp/login_aging_ses_det.lst >> /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst
rcp /tmp/pager_asunto_chk-rpt-$EQUIPO.lst prueba:/tmp
rcp /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst prueba:/tmp
if [ "${HORA}" = "12" ] && [ "${MINUTOS}" = "10" ]
then
chmod 700 /tmp/login_aging_ses_onmode_z.sh
sh -x /tmp/login_aging_ses_onmode_z.sh # ESTE SHELL LAS MATA 30may02
else
chmod 700 /tmp/login_aging_ses_onmode_z.sh
sh -x /tmp/login_aging_ses_onmode_z.sh # ESTE SHELL LAS MATA
fi
fi
#
else
rcp /tmp/pager_asunto_chk-rpt-$EQUIPO.lst prueba:/tmp
rcp /tmp/pager_mensaje_chk-rpt-$EQUIPO.lst prueba:/tmp
fi
```

El contenido del programa login\_aging.c es el siguiente:

```
#include <stdio.h>
#include <time.h>

main (int argc, char *argv[])
{
    FechaCambio (atoi (argv[1]));
}

int FechaCambio (time_t Segundos)
{
    char *fecha;
    time_t t;
    struct tm *tm;

    t = Segundos;
    if (t == 0)
        return (1);
    tm = localtime (&t);
    fecha = asctime (tm);
    printf ("%s", fecha);
    return (0);
}
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

#### 5.6 Transacciones sin terminar en log's anteriores

Toda la actividad de las transacciones se registran en el área de LOG, divididos en 152 secciones de de 250 Mb cada una, es una lista circular, por lo que es importante que ningún log contenga transacciones pendientes.

Pues si el "current" log, trata de ocupar un área de log con alguna transacción pendiente de hace varios días, la Base de datos se detiene y por lo tanto la operación del sistema SIIC también.

En ese tipo de evento es crítico, pues se tiene que llamar al soporte técnico de Informix en los E.U. para que resuelvan el problema.

Para monitorearlo manualmente es con la ejecución del siguiente programa shell:

```
> logs_tran_atra.sh
```

```
Database selected.
```

number	uniqid	size	actual	resp	usado
106	22032	25000	1	0	19.8240000000000

```
1 row(s) retrieved.
```

```
Database closed.
```

Del listado anterior nos indica que únicamente un log de los 152 disponibles tiene transacciones pendientes, este es el estado normal para este tipo de evento.

Para automatizar el monitoreo de este tipo de eventos y ser notificados oportunamente los administradores, ya sea por correo electrónico ó teléfono móvil, sé decidió realizar programas en lenguaje "Shell", los cuales se muestran a continuación:

Se programo la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
18,48 * * * * /siic_onbar2/monitoreo/logs_tran_atra/logs_tran_atra.sh
```

El programa Shell logs\_tran\_atra.sh, monitorea si hay transacciones atrasadas en cualquiera de las 152 áreas de log disponibles. Si hay mas de 20 áreas de log con transacciones atrasadas enviara un aviso mediante correo electrónico a los administradores del sistema SIIC, el mismo se ejecuta en 2 ocasiones, cada hora.

Una notificación típica de este evento es la siguiente:

```
Transacciones Atrasadas Fri Apr 22 03:29:40 CST 2006
```



**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES**

---

85	19311	25000	0	1	100.000000000000
86	19312	25000	0	1	100.000000000000
87	19313	25000	0	1	100.000000000000
88	19314	25000	0	1	100.000000000000
89	19315	25000	0	1	100.000000000000
90	19316	25000	0	1	100.000000000000
91	19317	25000	0	1	100.000000000000
92	19318	25000	0	1	100.000000000000
93	19319	25000	0	1	100.000000000000
94	19320	25000	0	1	100.000000000000
95	19321	25000	0	1	100.000000000000
96	19322	25000	0	1	100.000000000000
97	19323	25000	0	1	100.000000000000
98	19324	25000	1	0	69.324000000000

22 row(s) retrieved.

Database closed.

El código del programa shell logs\_tran\_atra.sh es el siguiente:

```
==> cat /siic_onbar2/monitoreo/logs_tran_atra/logs_tran_atra.sh
export DBDATE=dmy4/
export INFORMIXDIR=/informix
export INFORMIXSERVER=prod_300shm
export INFORMIXSQLHOSTS=/informix/etc/sqlhosts
export INFORMIXTERMCAP=TERMCAP
export ONCONFIG=prod_300shm
export PATH=$INFORMIXDIR/bin:/usr/bin

cd /siic_onbar2/monitoreo/logs_tran_atra
dbaccess sysmaster logs_tran_atra.sql 1> /tmp/logs_tran_atra.bit 2>&1

atra=`egrep -ci " 25000 | 124000 " /tmp/logs_tran_atra.bit`
if [ $atra -gt 20 ]
then
    echo "ERROR $atra LOGs en TRANS. LARGA `date` " 1> /tmp/pager_tran_atr
a_asunto.txt
    echo "Transaccion larga, ha llenado $atra Logs `date` " 1> /tmp/pager_t
ran_atra_mensaje.txt
    echo " Transacciones Atrasadas `date`">>logs_tran_atra.bit
    tail -20 /tmp/logs_tran_atra.bit|tee -a /tmp/pager_tran_atra_mensaje.txt 1>>
logs_tran_atra.bit
    onstat -m >>/tmp/pager_tran_atra_mensaje.txt

    rcp /tmp/pager_tran_atra_asunto.txt prueba:/tmp
    rcp /tmp/pager_tran_atra_mensaje.txt prueba:/tmp

fi
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

#### 5.7 Transacciones no autorizadas

El sistema SIIC cuenta con un servidor central y 77 servidores remotos instalados en las terminales de Almacenamiento y Distribución, en las principales ciudades del País, el servidor central "ve" a todos los servidores remotos, un servidor remoto únicamente "ve" al servidor central.

De esta forma la seguridad de accesos al sistema SIIC esta controlada, pero la red de comunicaciones interna de Pemex Refinación es muy amplia, es enorme, con miles de usuarios y las posibilidades de que alguien intente acceder indebidamente a la base de datos es probable.

Estadísticamente los ataques a los sistemas de información, provienen en un 90 % de personas de la misma compañía, la integridad de la información debe de garantizarse siempre, únicamente deben de acceder los usuarios autorizados, de acuerdo a su perfil y privilegios permitidos.

Para evitar accesos indebidos, no autorizados, a la base de datos se tiene un programa corriendo continuamente en el servidor UNIX, monitoreando el **origen** de todas las sesiones que entran al servidor de base de datos, el programa detecta.sh revisa el nombre del equipo origen, y si no esta autorizado para acceder el servidor central termina la sesión, generando una bitácora del evento.

Para automatizar el monitoreo de este tipo de eventos y ser notificados oportunamente los administradores, ya sea por correo electrónico ó teléfono móvil, sé decidió realizar programas en lenguaje "Shell", los cuales se muestran a continuación:

Se programo la ejecución del programa Shell mediante el comando **crontab** del sistema operativo UNIX, de acuerdo a lo siguiente:

```
1,11,21,31,41,51 * * * * /siic_onbar1/detecta/ses_origen.sh >>/tmp/ses_origen.bit2>&1
```

El programa Shell centinela.sh, monitorea que el proceso este permanentemente en ejecución o "residente", en caso contrario lo "levanta" nuevamente, también enviara un aviso mediante correo electrónico a los administradores del sistema SIIC, se ejecuta cada 1 minuto.

Una notificación típica de este evento es la siguiente:

Sesiones terminadas, provenientes de un equipo no autorizado

```
informix|1036892|sqlexec|27834|2006-04-25 14:05:54.00|
informix|1038370|y|sqlexec|3873|2006-04-25 14:07:53.00|
ppo0a601|1038358|rodu|sqlexec|3839|2006-04-25 14:07:53.00|
informix|1099433|sqlexec|15657|2006-04-25 15:03:17.00|
ppo0a601|1099441|]vy|sqlexec|15686|2006-04-25 15:03:17.00|
informix|1118900|sqlexec|15939|2006-04-25 15:20:26.00|
ppo0a601|1118885|sqlexec|15976|2006-04-25 15:20:26.00|
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

```
informix|1146673|H|sqllexec|20693|2006-04-25 15:44:28.00|
ppo0a601|1146703||sqllexec|20776|2006-04-25 15:44:28.00|
informix|1339836||sqllexec|28158|2006-04-25 17:40:09.00|
ppo0a601|1339821|b1|sqllexec|28079|2006-04-25 17:40:09.00|
```

El código del programa shell es el siguiente:

```
#####
#
#####
#
cd /informix/monito/centinela
#
export EQUIPO=`uname -a | awk -F" " ' { printf "%s",$2 } ' `
export ALEATORIO=`echo "$RANDOM\c"`
export FECHA=`date +%d_%m_%Y`
export HORA=`date +%H_%M`
export INFORMIXSERVER=prod_300shm
export INFORMIXDIR=/informix
export INFORMIXSQLHOSTS=$INFORMIXDIR/etc/sqlhosts
export PATH=$INFORMIXDIR/bin:/usr/bin
#
ya_esta_corriendo=` ps -ef | grep "centinela.sh" | grep -v "grep" | wc -l `
#
if [ $ya_esta_corriendo -gt 2 ]
then
echo "Ya esta corriendo [$ya_esta_corriendo] `date` "
exit
else
echo "Arrancando [$ya_esta_corriendo] `date` "
echo "$EQUIPO - CENTINELA ARRANQUE - $FECHA-$HORA" > /tmp/pa-1-cen
tinela-$ALEATORIO-$EQUIPO.lst
ps -ef | grep centinela|grep -v grep > /tmp/pm-1-cen
tinela-$ALEATORIO-$EQUIPO.lst
cat /informix/monito/emails_gpo_soporte.NB > /tmp/pe-1-cen
tinela-$ALEATORIO-$EQUIPO.lst
rcp /tmp/p?-1-centinela-$ALEATORIO-$EQUIPO.lst correo:/tmp
##### CICLO #####
while (true)
do

export ALEATORIO=`echo "$RANDOM\c"`
export FECHA=`date +%d_%m_%Y`
export HORA=`date +%H_%M`

terminar=` grep -ci informix /informix/monito/centinela/centinela.bit | cut -d"
" -f1 `
if [ $terminar -gt 0 ]
then
echo "$EQUIPO - CENTINELA TERMINA SESION - $FECHA-$HORA" > /tmp/pa-1-centinela
-$ALEATORIO-$EQUIPO.lst
echo "Sesiones terminadas ..." > /tmp/pm-1-centinela-$ALEA
TORIO-$EQUIPO.lst
strings centinela.bit > /tmp/pm-1-centinela-$ALEA
TORIO-$EQUIPO.lst
cat /informix/monito/centinela/centinela.bit >> /informix/monito/centinela/ce
ntinela_acum.bit
cat /informix/monito/emails_gpo_soporte.NB > /tmp/pe-1-centinela-$ALEA
TORIO-$EQUIPO.lst
chmod 700 /tmp/*centinela*
rcp /tmp/p?-1-centinela-$ALEATORIO-$EQUIPO.lst correo:/tmp
fi
fglgo centinela
sleep 1
done
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES

---

```
##### CICLO #####  
fi  
#  
# #####
```

El programa en Informix-4gl es el siguiente, el cual se compila en lenguaje C:

```
#####  
# Programa para detectar las sesiones remotas y termina las no autorizadas ...  
# JVV  
# 5-jul-00, 18-sep-02 28nov04 16dic04 22mar05-(siic1sd) 18mar06 27mar06  
#####  
DATABASE sysmaster
```

MAIN

DEFINE

```
reporte                CHAR(18),  
elimina_sesion         CHAR(200),  
elimina_proceso       CHAR(200),  
  
reg_tmp RECORD  
sesion                INTEGER,  
proceso               INTEGER,  
usuario               CHAR(18),  
equipo                CHAR(18),  
tread                 CHAR(18),  
fecha                 DATETIME YEAR TO FRACTION(2)  
END RECORD
```

SET ISOLATION TO DIRTY READ

```
DECLARE cur_centinela CURSOR FOR SELECT  
sid,  
pid,  
username,  
hostname,  
th_name,  
current fecha  
FROM syssessions,sysuserthreads,systhreads  
WHERE  
    th_name      = "sqlexec" AND  
    ( hostname != "siic1sd" AND  
      hostname != "siic2sd" AND  
      hostname != "aplica" AND  
      hostname != "aplica2" AND  
      hostname != "siic1v"  
    )  
    AND sid      = us_sid  
    AND us_tid   = th_id  
ORDER BY 3,1
```

```
LET reporte="centinela.bit"  
START REPORT rpt_centinela TO reporte
```

```
FOREACH cur_centinela into  
    reg_tmp.sesion,  
    reg_tmp.proceso,  
    reg_tmp.usuario,  
    reg_tmp.equipo,  
    reg_tmp.tread,  
    reg_tmp.fecha
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 5 - MONITOREO DE LAS TRANSACCIONES**

---

```
OUTPUT TO REPORT rpt_centinela(reg_tmp.*)

LET elimina_sesion = "onmode -z ",reg_tmp.sesion
DISPLAY elimina_sesion

IF reg_tmp.usuario = "informix" THEN
  RUN (elimina_sesion) WITHOUT WAITING
END IF

END FOREACH

CLOSE cur_centinela

FINISH REPORT rpt_centinela

END MAIN
# ##### #
report rpt_centinela(reg_tmp)
define
reg_tmp RECORD
  sesion                INTEGER,
  proceso               INTEGER,
  usuario               CHAR(18),
  equipo                CHAR(18),
  tread                CHAR(18),
  fecha                 DATETIME YEAR TO FRACTION(2)
END RECORD

output
page length 1
top margin 0
left margin 0
bottom margin 0

format

on every row

print
  reg_tmp.usuario clipped,"|",
  reg_tmp.sesion using "<<<<<<<&","|",
  reg_tmp.equipo clipped,"|",
  reg_tmp.tread clipped,"|",
  reg_tmp.proceso using "<<<<<<<&","|",
  reg_tmp.fecha,"|"

END REPORT
# ##### #
```

## CAPITULO 6 LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS

### 6.1 Para los dbspaces

La definición de las áreas de almacenamiento o dbspaces se definen al inicio de la creación de la base de datos y conforme se vayan requiriendo a lo largo de la operación del sistema, en el caso del sistema SIIC, se migro la información existente de la plataforma anterior, un Main Frame IBM 9121-521.

Se migraron aproximadamente 50 millones de registros del ambiente Main Frame a la nueva versión del sistema SIIC en UNIX en agosto de 1999, actualmente el sistema SIIC a más de 5 años de operación tiene 800 millones de registros distribuidos en 580 tablas, en 55 dbspaces con 128 chunks de 2 Giga bytes cada uno.

Se depura la base de datos periódicamente, para conservar únicamente la información de los últimos 4 meses, la información histórica se traslada al sistema SIIC HISTÓRICO, ubicado en otro servidor, en otra base de datos.

Los lineamientos que se han seguido, al definir los dbspaces en el sistema SIIC son las siguientes:

El tamaño de cada chunk es de 2 GB, en Informix Dynamic Server Versión 7.31.UD7W4 el tamaño máximo de un chunk es de 2 GB, un dspace puede estar conformado por varios chunks.

Reservar un dspace dedicado únicamente a las tablas internas del manejador de la base de datos, este es el dspace "rootdbs", (rlrootdbs).

Reservar un grupo de dbspaces dedicados a las áreas de LOG, para uso exclusivo del manejador de la base de datos, evitando que se ubiquen el dspace "rootdbs". El área de LOG debe contar con el espacio suficiente para guardar la actividad de 5 días, (rllogX).

Reservar un grupo de dbspaces dedicados a las áreas de trabajo TEMPORALES para uso exclusivo del manejador de la base de datos. Inicialmente sé tenían 2 de 2 GB cada uno, debido a la actividad de cada vez más usuarios simultáneos se agregaron 2 mas, lo cual ayudo a mejorar el tiempo de respuesta de los procesos de los usuarios, (rltempX).

Reservar un grupo de dbspaces dedicados para contener o almacenar las tablas medianas y pequeñas del sistema, (rltabX).

Reservar un grupo de dbspaces dedicados para contener o almacenar las grandes tablas del sistema, (rlgranXX)

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS**

---

Reservar un dbspace exclusivo para el modulo de replicación continua de datos (CDR), pues en el mismo se almacenan temporalmente las transacciones entrantes y salientes a los 77 servidores remotos que conforman el sistema SIIC, en este dbspace en especial no debe de tener contención pues están involucradas transacciones distribuidas o remotas que tienen que viajar por la WAN de la empresa, (rlcdr).

Reservar un grupo de dbspaces dedicados para contener o almacenar los índices de las tablas. Pues no es conveniente tener datos e índices en un mismo dbspace, pues esto genera contención en el dbspace, en el disco, (rlidx).

Los dbspaces que conforman un grupo, no deben de estar en el mismo disco físico, para esto al formar los "volumen groups" a nivel de Sistema Operativo, se debe de tener a la mano el mapa de discos del equipo de discos EMC DMX 1000.

En el sistema SIIC se tienen definidos los siguientes grupos de discos:

Uso	Nombre
Base de Datos	/dev/vggrandbs
Base de Datos	/dev/vgidxdbs
File System	/dev/vgiintranet
File System	/dev/vgrespaldo
Base de Datos	/dev/vgrootdb
File System	/dev/vgsoporte
Base de Datos	/dev/vgtabdb
File System	/dev/vgutil

El comando `vgdisplay` nos proporciona información de cómo están conformados los grupos de discos, únicamente se muestra el grupo `vgrootdb`:

```
=> vgdisplay -v
```

```
VG Name                /dev/vgrootdb
VG Write Access        read/write
VG Status              available, exclusive
Max LV                 255
Cur LV                14

Open LV                14
Max PV                 30
Cur PV                4
Act PV                 4
Max PE per PV         2157
VGDA                   8
PE Size (Mbytes)      4
Total PE               8628
Alloc PE               7000
Free PE                1628
Total PVG              0
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS**

---

Total Spare PVs                   0  
Total Spare PVs in use           0

--- Logical volumes ---

LV Name                           /dev/vgrootdbs/lcdr  
LV Status                         available/syncd  
LV Size (Mbytes)                 2000  
Current LE                        500  
Allocated PE                      500  
Used PV                           1

LV Name                           /dev/vgrootdbs/llog2  
LV Status                         available/syncd  
LV Size (Mbytes)                 2000  
Current LE                        500  
Allocated PE                      500  
Used PV                           1

LV Name                           /dev/vgrootdbs/llog1  
LV Status                         available/syncd  
LV Size (Mbytes)                 2000  
Current LE                        500  
Allocated PE                      500  
Used PV                           1

LV Name                           /dev/vgrootdbs/llee1  
LV Status                         available/syncd  
LV Size (Mbytes)                 2000  
Current LE                        500  
Allocated PE                      500  
Used PV                           1

LV Name                           /dev/vgrootdbs/llog3  
LV Status                         available/syncd  
LV Size (Mbytes)                 2000  
Current LE                        500  
Allocated PE                      500  
Used PV                           1

LV Name                           /dev/vgrootdbs/llog4  
LV Status                         available/syncd  
LV Size (Mbytes)                 2000  
Current LE                        500  
Allocated PE                      500  
Used PV                           1

LV Name                           /dev/vgrootdbs/llog5  
LV Status                         available/syncd  
LV Size (Mbytes)                 2000  
Current LE                        500  
Allocated PE                      500  
Used PV                           1

LV Name                           /dev/vgrootdbs/lrootdbs  
LV Status                         available/syncd  
LV Size (Mbytes)                 2000  
Current LE                        500



**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS**

---

```
Allocated PE          500
Used PV              1

LV Name              /dev/vgrootdbs/ltemp1
LV Status            available/syncd
LV Size (Mbytes)     2000
Current LE           500
Allocated PE         500
Used PV              1

LV Name              /dev/vgrootdbs/ltemp2
LV Status            available/syncd
LV Size (Mbytes)     2000
Current LE           500
Allocated PE         500
Used PV              1

LV Name              /dev/vgrootdbs/llee2
LV Status            available/syncd
LV Size (Mbytes)     2000
Current LE           500
Allocated PE         500
Used PV              1

LV Name              /dev/vgrootdbs/lrootdbs2
LV Status            available/syncd
LV Size (Mbytes)     2000
Current LE           500
Allocated PE         500
Used PV              1

LV Name              /dev/vgrootdbs/ltemp3
LV Status            available/syncd
LV Size (Mbytes)     2000
Current LE           500
Allocated PE         500
Used PV              1

LV Name              /dev/vgrootdbs/ltemp4
LV Status            available/syncd
LV Size (Mbytes)     2000
Current LE           500
Allocated PE         500
Used PV              1

--- Physical volumes ---
PV Name              /dev/dsk/c9t1d7
PV Name              /dev/dsk/c5t1d7  Alternate Link
PV Name              /dev/dsk/c7t1d7  Alternate Link
PV Name              /dev/dsk/c11t1d7  Alternate Link
PV Status            available
Total PE             2157
Free PE              157
Autoswitch           On

PV Name              /dev/dsk/c11t2d0
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
***CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS***

---

PV Name /dev/dsk/c5t2d0 Alternate Link  
PV Name /dev/dsk/c7t2d0 Alternate Link  
PV Name /dev/dsk/c9t2d0 Alternate Link  
PV Status available  
Total PE 2157  
Free PE 657  
Autoswitch On

PV Name /dev/dsk/c5t2d1  
PV Name /dev/dsk/c7t2d1 Alternate Link  
PV Name /dev/dsk/c9t2d1 Alternate Link  
PV Name /dev/dsk/c11t2d1 Alternate Link  
PV Status available  
Total PE 2157  
Free PE 157  
Autoswitch On

PV Name /dev/dsk/c7t2d2  
PV Name /dev/dsk/c5t2d2 Alternate Link  
PV Name /dev/dsk/c9t2d2 Alternate Link  
PV Name /dev/dsk/c11t2d2 Alternate Link  
PV Status available  
Total PE 2157  
Free PE 657  
Autoswitch On

## 6.2 Para los índices, Foreign Key y Primary Key

Los índices de las tablas mas accesadas y con mas de un millón de registros se deben de fragmentar, se deben de alojar en los dbspaces dedicados exclusivamente a índices. Con ello se ayuda de manera significativa al servidor de base de datos a realizar transacciones sobre las tablas y/o datos correspondientes, con un tiempo de respuesta mas optimo.

Para los constraints referenciales o llaves foráneas (Foreign Key), tienen la desventaja de que se alojan por default en el dbspace inicial donde sé creo la base de datos, en este caso el dbspace rltab1, con lo cual se consume una gran cantidad de espacio en disco, para un solo dbspace (rltab1). En la base de datos existen 690 Foreign Key.

Para evitar la concentración excesiva en el dbspace inicial, por cada FK, se debe de borrar, crear un índice similar pero fragmentado y alojado en los dbspaces exclusivos de índices, finalmente crear nuevamente el FK. El servidor de base de datos detecta que ya existe un índice idéntico fragmentado y lo usa como pivote para el FK en cuestión.

En el caso de los Primary Key o Llave primaria, se alojan por default en el dbspace inicial de la base de datos, rlgran7, son índices compuestos generalmente con varios campos, en el sistema SIIC se tienen 363 PK.

Por esta razón los PK demandan una gran cantidad de espacio en el dbspace inicial, para evitarlo, para cada PK se debe de borrar, crear un índice único similar, con los mismos campos, en el mismo orden, fragmentado en los dbspaces dedicados a índices. Lo anterior es únicamente posible para tablas que no cuenten con FK.

El observar los 3 lineamientos anteriores ayuda a optimizar el desempeño del servidor de la base de datos, en sistemas de alta demanda, con millones de registros, con cientos de usuarios simultáneos, como es el caso del sistema SIIC. En sistemas de baja demanda no se observa una mejora significativa en el desempeño global del sistema.

### 6.3 Para las tablas

Uno de los lineamientos principales es no modificar la estructura de cualquier tabla en días de operación, es decir cuando están cientos de usuarios trabajando en el sistema, pues se corre el gran riesgo de que la tabla modificada y anexas, se corrompan, ocasionando un gran problema, por lo anterior los cambios a la estructura de las tablas únicamente se realiza los días Domingo, sacando previamente a los usuarios. Si además esa tabla tiene replicación del modulo CDR (Continuos Data Replication) también se tiene que detener y deshabilitar previamente.

Lo más conveniente es planear desde el inicio del proyecto la distribución adecuada de las todas las tablas del sistema, en todos los dbspaces disponibles, en el caso del sistema SIIC, se realizo una proyección inicial para las tablas más importantes, del numero de registros iniciales, del crecimiento de registros esperado por mes, el tiempo de permanecía de la información 1,2,3 o 4 meses, antes de depurar la información y enviarla al sistema SIIC HISTÓRICO, totalmente independiente en otro equipo UNIX, en otra base de datos.

Frecuentemente las proyecciones para cada tabla no fueron la mas adecuado, lo cual origina que el espacio requerido aumenta considerablemente, lo que puede llegar a llenar alguno de los dbspaces, por otra parte el numero de extents para cierta tabla se puede disparar mas de lo conveniente, con lo cual se tiene dispersión en los datos de la misma.

Lo anterior obliga a cambiar de ubicación la tabla en particular y ajustar sus parámetros básicos de la misma, estas operaciones únicamente se realizan los DIA Domingo.

También es frecuente la necesidad de crear nuevos índices, para reducir notablemente los tiempos de recuperación de un registro, todo esto se debe de hacer tabla por tabla y en base a un proceso o programa en particular, en conjunto con el programador y los usuarios.

También se debe de revisar periódicamente que todas las tablas del sistema, tengan el nivel de bloqueo a nivel de registro, las tablas que estén fragmentadas deben de contar con el índice rowid.

Se debe de realizar un oncheck -cID a toda la base de datos, aproximadamente cada 2 meses para verificar el estado interno de las tablas de datos, los índices PK y FK. Esto es muy importante para prevenir corrupción interna en las paginas de índices y datos, que en caso de presentarse es causa suficiente para "tirar" el servidor de base de datos y por consiguiente todo el sistema.

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS**

---

**6.4 Para los parámetros de configuración de la Base de Datos**

Estos parámetros de la Base de Datos se encuentran en el archivo de configuración localizado en la ruta \$INFORMIXDIR/etc/\$ONCONFIG, el archivo contiene aproximadamente 100 parámetros, vamos a mencionar los más importantes, de acuerdo a nuestra experiencia en la administración de la base de datos INFORMIX. En el anexo 3 se muestra el archivo de configuración completo.

Parámetro : ROOTSIZE	# KB
Valor: 4000000	
Lineamiento: El espacio asignado al dbspace principal ROOTDBS de la base de datos debe de ser suficientemente grande para contener las tablas internas del servidor. Debe de estar ubicada en un disco exclusivo, pues este dbspace es muy accesado, por los procesos internos del servidor. El administrador y los programadores deben de asegurarse que todas las tablas temporales incluyan la cláusula "with no log", si a alguien se le olvida este lineamiento, esta área debe soportar la creación de tablas temporales en ROOTDBS, aunque debe de ser la excepción.	

Parámetro : LOGSIZE	# Logical log size (Kbytes)
Valor: 250000	
Lineamiento: El tamaño de cada LOG, debe de ser el suficiente para contener por lo menos una hora de actividad, en las horas pico. Este valor se va ajustando conforme los usuarios del sistema se van incrementando, pues los mismos van generando mas transacciones y por ello la demanda de LOGs aumenta. Se debe de contar con el numero de LOGs suficientes para almacenar la actividad de una semana.	

Parámetro : NUMCPUVPS	# Number of user (cpu) vps
Valor: 7	
Lineamiento: Este tipo de cpu-vps son los procesos que más trabajan en una base de datos, son los procesadores virtuales del servidor de la base de datos, se ligan 1 a 1, a los CPU's físicos con que cuenta un equipo, excepto 1 que se dedica completamente a las tareas del Sistema Operativo	

Parámetro : LOCKS	# Maximum number of locks
Valor: 6000000	
Lineamiento: Este parámetro debe de ser suficiente, para que todos los procesos de los usuarios obtengan locks o candados, pues cuando se terminan los mismos en periodos de máxima actividad en el servidor, el mismo puede volverse inestable, volver inestable al sistema operativo y puede provocar que todo el sistema en su conjunto falle, pasando a un estado critico.	

Parámetro : BUFFERS	# Maximum number of shared buffers
---------------------	------------------------------------

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS**

Valor: 250000
Lineamiento: Este parámetro debe de ser suficiente, para que todos los procesos de los usuarios obtengan paginas en la memoria compartida de la base de datos, de lo contrario los procesos tendrán que esperar por el recurso y lo que es peor el sistema tendrá que realizar I/O a disco por la falta de Buffers, lo que degradaría el desempeño del sistema en una forma notable.

Parámetro : CLEANERS	# Number of buffer cleaner processes
Valor: 45	
Lineamiento: Se debe de contar con un CLEANER por cada DBSPACE	

Parámetro : SHMVIRTSIZE	# initial virtual shared memory segment size KB
Valor: 300000	
Lineamiento: Se debe de contar con máximo 2 áreas de esta clase, pues de lo contrario en el HPUX, se tiene una degradación del desempeño notable, por lo cual se debe de ajustar periódicamente este parámetro.	

Parámetro :	
LRU_MAX_DIRTY 3	# LRU percent dirty begin cleaning limit
LRU_MIN_DIRTY 2	# LRU percent dirty end cleaning limit
Valor: 3 % y 2 % respectivamente	
Lineamiento: El valor en estos 2 parámetros va ligado directamente a la duración de un CHECK-POINT, que es cuando se actualizan a disco todas las transacciones de los procesos de los usuarios, se deben de ajustar periódicamente estos valores, para que la duración del CHECK-POINT no exceda de los 10 seg., pues en este lapso todo se detiene y los usuarios podrían pensar que el sistema a fallado, detenido, etc.	

Parámetro :	
LTXHWM	# Long transaction high water mark percentage
LTXEHWM	# Long transaction high water mark (exclusive)
Valor: 40 % y 70 % respectivamente	
Lineamiento: El valor del segundo parámetro, detecta si una transacción ha ocupado el 70 % del total de las áreas de LOG (106), si es el caso detiene todos los procesos y cancela/termina la sesión que esta causando el problema, es un estado critico del servidor de base de datos. Se debe de ajustar estos 2 parámetros de acuerdo a las condiciones particulares del sistema y principalmente por estar usando el modulo de CDR, Continuos Data Replication.	

Parámetro :	
BAR_MAX_BACKUP	
Valor: 10	

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS**

---

Lineamiento:

Al integrar el software de respaldo OMNI-BACK, para respaldar la base de datos, simultáneamente por varios drives, se requiere ajustar este parámetro al valor mencionado.

Parámetro :

RA\_PAGES # Number of pages to attempt to read ahead  
RA\_THRESHOLD # Number of pages left before next group

Valor: 120 y 100 respectivamente

Lineamiento:

Ajustar este valor, para reducir el número de accesos a disco, pues en lugar de traerse 10 paginas contiguas de disco, se trae 120, con lo anterior el servidor trata de tener ya en memoria, la posible información que requiera un proceso.

Parámetro :

DBSPACETEMP # Default temp dbspaces

Valor: rltemp1,rltemp2,rltemp3,rltemp4

Lineamiento:

Inicialmente se inicio con dbspaces dedicados a las áreas temporales de 2 GB cada uno, con el incremento de los usuarios, procesos, transacciones y numero de registros almacenados en la base de datos, se agregaron 2 dbspaces mas, con excelentes resultados en el desempeño general del sistema.

Parámetro :

MAX\_PDQPRIORITY

Valor: 50

Lineamiento:

Para la construcción de índices en paralelo en tablas con millones de registros, reduce en un 80 % el tiempo de proceso, de 10 horas lo reduce a 2 horas, pero consume la gran mayoría de los recursos del servidor, por lo cual no deben de usarlo los usuarios comunes, únicamente el administrador y cuando no haya usuarios accesando el sistema.

Parámetro :

ONDBSPACEDOWN # Dbspace down option: 0 = CONTINUE, 1 = ABORT, 2 = WAIT

Valor: 1

Lineamiento:

Si se llegara a dañar un dbspace, que las operaciones en el servidor de base de datos se detengan, hasta arreglar el problema y evitar inconsistencias en la información.

Parámetro :

NETTYPE ipcshm,5,200,NET # Configure poll thread(s) for nettype  
NETTYPE soctcp,2,200,NET # Configure poll thread(s) for nettype

Valor: 5,200 y 2,200

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS**

---

**Lineamiento:**

Se reserva la cantidad suficiente de recursos en el servidor, para poder recibir a todos los usuarios que desean acceder el sistema, en este caso se pueden recibir hasta 1000 conexiones locales por la conexión de tipo ipcshm o memoria compartida.

También se pueden recibir hasta 400 conexiones remotas del tipo soctcp o TCP

**Parámetro :**

CDR\_NIFRETRY

# Connection retry (seconds)

Valor: 60

**Lineamiento:**

El ajuste de 300 segundos a 60 segundos para reintentar reconectarse a un servidor remota, que perdió la conexión por cualquier razón, ayudo a evitar en gran medida que la conexión quedara en estado de "connecting", lo cual obliga a tirar/levantar el servidor central, tal como se comenta el capitulo 2.1



## 6.5 Para el equipo UNIX

Los lineamientos para contar con un sistema operativo, seguro, estable son las siguientes:

Se deben de ajustar **periódicamente** (cada 3 meses), los principales parámetros del Kernel, se mencionan los siguientes, que afectan a otros tantos parámetros secundarios relacionados:

**NFILES** - Numero máximo de archivos abiertos simultáneamente, en UNIX, todo es finalmente un tipo de archivo, de bloque o carácter, especial. Por lo cual se debe de tener un número adecuado, pues de lo contrario el sistema operativo puede tener problemas y comportarse inestablemente.

**NPROCS** - Numero máximo de procesos simultáneos en el sistema, es importante ajustar periódicamente este parámetro, pues de lo contrario se puede llegar a llenar la tabla de procesos residente en el Kernel, con lo cual es seguro que el sistema operativo se comporte inestablemente.

**SEMMAX** - Numero máximo de semáforos simultáneos en el sistema operativo, este parámetro va directamente ligado con el arrea de memoria compartida (shared memory), en la cual trabaja principalmente la Base de Datos, por todo lo anterior el valor de este parámetro debe de ser suficiente.

Además de lo anterior, se debe de actualizar al sistema operativo cada 6 meses, con el último conjunto de parches liberados para la versión en particular, lo que se conoce técnicamente como pro-patch.

El fabricante o desarrollador del sistema operativo con lo anterior garantiza que se corrigen los bugs (errores) de todo tipo detectados en el sistema operativo, es muy importante realizar esta actividad, particularmente por los errores en el área de seguridad, pues los Hackers o intrusos indeseables usan estas "debilidades" del sistema operativo para dañar los sistemas.

El equipo debe de estar completamente redundante es todos y cada una de sus elementos, es decir deberá contar con:

Doble tarjeta de ATM.

Doble tarjeta de Ethernet 10/100.

Los discos donde reside el sistema operativo deberán estar en espejo (Mirror), por si se daña el disco primario, entra en operación el secundario o de respaldo

El sistema SIIC es un sistema de misión crítica, no debe de fallar, por lo cual se tienen 2 equipos en CLUSTER, lo anterior significa que en caso de falla del servidor UNIX Primario, toda la aplicación, base de datos, calendarizador, impresoras, passwords, se pasan en 15 minutos aproximadamente al servidor UNIX Secundario.

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS**

---

Lo anterior se implemento con el Software Service Guard (SG) de la compañía HP, el cual esta monitoreando continuamente que ambos equipos se "vean" entre si, el SG mantiene una sola IP para el sistema, no importando que equipo este activo, Primario o Secundario, de tal manera que los usuarios y sistemas externos, no se enteran que en ocasiones pueden estar trabajando en el equipo secundario.

El equipo Secundario únicamente se activa en caso de falla o mantenimiento del equipo Primario, es un equipo con menor capacidad de procesamiento el cual solo se debe de usar por periodos cortos de tiempo.

El nivel de seguridad se eleva a Trusted System, esto es mas haya del default, que normalmente manejan los equipos, con lo cual se implementan la política de cambio de password cada 28 días, bloqueo de la cuenta del usuario a los 15 días de no usarla, bloqueo de la cuenta del usuario al fallar en mas de 5 ocasiones al proporcionar el password.

Realizar cada 6 meses pruebas de vulnerabilidad del sistema, por compañías externas dedicadas a este tipo de pruebas y aplicar las recomendaciones que se generen.

Revisar las bitácoras del sistema operativo, periódicamente para tomar las medidas preventivas que procedan, es decir ser proactivos y no reactivos, es decir adelantarse a los problemas.

Actualizar periódicamente todos los procedimientos, para facilitar la integración de nuevo personal al área de soporte técnico, y también en caso de ausencia de personal, poder realizar las actividades que el mismo realizaba. Además el aplicar y cumplir los procedimientos establecidos reducen el margen de error en este tipo de actividades críticas para el sistema.

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS**

---

**6.6 Para los discos externos**

La interconexión de la caja de discos externos (EMC DMX 1000 con 1.2 Terabytes de disco configurado, con capacidad de crecer a mas de 5 Terabytes) y el equipo UNIX (HP-SUPER DOME) se realiza a través de "Fibre Channel", esto es mediante cables de fibra óptica.

Lo anterior proporciona un excelente canal de comunicación entre ambos equipos, garantizando que no habrá cuellos de botella entre ambos equipos. En los equipos EMC se tiene la facilidad de tener más de 1 copia de la información total, que estén almacenando en los mismos.

Estas copias son con el propósito de contar con respaldos de la base de datos, las cuales las puedes separar antes de realizar operaciones de mantenimiento a la base de datos. Para que en caso de error, pueda uno dejar la base de datos como estaba originalmente, esto da gran seguridad, y rapidez para recuperarse de situaciones criticas en caso de daño a la base de datos.

Existen 2 alternativas de realizar las copias de seguridad: SRF y Time-Finder, ambas copias se pueden separa y/o resincronizar en el momento que se requiera.

Para verificar el estatus de la copia de la base de datos, por el método de SRDF manualmente ejecutar el siguiente comando:

**V2600 /# symrdf -g GPOSIIC query**

```
Device Group (DG) Name: GPOSIIC
DG's Type           : RDF1
DG's Symmetrix ID   : 000184700363
```

Source (R1) View					Target (R2) View					M O D E S				
Standard	Logical	Device	Dev	E	ST	LI	ST	LI	ST	LI	Mode	ACp	STATE	
A	T R1 Inv	R2 Inv	Tracks	Tracks	N	K	T R1 Inv	R2 Inv	Tracks	Tracks	e	Dom		
DEV001	001	RW	0	0	RW	001	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV002	002	RW	0	0	RW	002	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV003	003	RW	0	0	RW	003	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV004	004	RW	0	0	RW	004	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV005	005	RW	0	0	RW	005	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV006	006	RW	0	0	RW	006	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV007	007	RW	0	0	RW	007	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV008	008	RW	0	0	RW	008	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV009	009	RW	0	0	RW	009	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV010	00A	RW	0	0	RW	00A	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV011	00B	RW	0	0	RW	00B	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV012	00C	RW	0	0	RW	00C	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV013	00D	RW	0	0	RW	00D	WD	0	0	0	SYN	DIS	OFF	Synchronized
DEV014	00E	RW	0	0	RW	00E	WD	0	0	0	SYN	DIS	OFF	Synchronized

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS**

DEV015	00F	RW	0	0	RW	00F	WD	0	0	SYN	DIS	OFF	Synchronized
DEV016	010	RW	0	0	RW	010	WD	0	0	SYN	DIS	OFF	Synchronized
DEV017	011	RW	0	0	RW	011	WD	0	0	SYN	DIS	OFF	Synchronized
DEV018	012	RW	0	0	RW	012	WD	0	0	SYN	DIS	OFF	Synchronized
DEV019	013	RW	0	0	RW	013	WD	0	0	SYN	DIS	OFF	Synchronized
DEV020	014	RW	0	0	RW	014	WD	0	0	SYN	DIS	OFF	Synchronized
DEV021	015	RW	0	0	RW	015	WD	0	0	SYN	DIS	OFF	Synchronized
DEV022	016	RW	0	0	RW	016	WD	0	0	SYN	DIS	OFF	Synchronized
DEV023	017	RW	0	0	RW	017	WD	0	0	SYN	DIS	OFF	Synchronized
DEV024	018	RW	0	0	RW	018	WD	0	0	SYN	DIS	OFF	Synchronized

Total	-----	-----	-----	-----
Track(s)	0	0	0	0
MB(s)	0.0	0.0	0.0	0.0

Para verificar el estatus de las copias de la base de datos, por el metodo de Time-Finder manualmente ejecutar el siguiente comando:

**V2600 /# symmir -g GPOSIIC query**

Device Group (DG) Name: GPOSIIC  
 DG's Type : RDF1  
 DG's Symmetrix ID : 000184700363

Standard Device			BCV Device		State
Logical	Inv. Sym Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV
DEV001	001	0 BCV001	043	*	0 Synchronized
DEV002	002	0 BCV002	044	*	0 Synchronized
DEV003	003	0 BCV003	045	*	0 Synchronized
DEV004	004	0 BCV004	046	*	0 Synchronized
DEV005	005	0 BCV005	047	*	0 Synchronized
DEV006	006	0 BCV006	048	*	0 Synchronized
DEV007	007	0 BCV007	049	*	0 Synchronized
DEV008	008	0 BCV008	04A	*	0 Synchronized
DEV009	009	0 BCV009	04B	*	0 Synchronized
DEV010	00A	0 BCV010	04C	*	0 Synchronized
DEV011	00B	0 BCV011	04D	*	0 Synchronized
DEV012	00C	0 BCV012	04E	*	0 Synchronized
DEV013	00D	0 BCV013	04F	*	0 Synchronized
DEV014	00E	0 BCV014	050	*	0 Synchronized
DEV015	00F	0 BCV015	051	*	0 Synchronized
DEV016	010	0 BCV016	052	*	0 Synchronized
DEV017	011	0 BCV017	053	*	0 Synchronized
DEV018	012	0 BCV018	054	*	0 Synchronized
DEV019	013	0 BCV019	055	*	0 Synchronized
DEV020	014	0 BCV020	056	*	0 Synchronized
DEV021	015	0 BCV021	057	*	0 Synchronized
DEV022	016	0 BCV022	058	*	0 Synchronized
DEV023	017	0 BCV023	059	*	0 Synchronized
DEV024	018	0 BCV024	05A	*	0 Synchronized

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**CAPITULO 6 - LINEAMIENTOS PARA CONSTRUIR Y AFINAR UNA BASE DE DATOS**

---

Total	-----	-----
Track(s)	0	0
MB(s)	0.0	0.0

Legend:

(\*): The paired BCV device is associated with this group.

Para separar la copia de Time-Finder antes de realizar actividades que pongan en riesgo la integridad de la base de datos manualmente ejecutar el siguiente comando:

**symmir -g GPOSIIC -full split**

Para separar la copia de SRDF antes de realizar actividades que pongan en riesgo la integridad de la base de datos manualmente ejecutar el siguiente comando:

**symrdf -g GPOSIIC -full split**

Para sincronizar la copia de Time-Finder después de realizar algún mantenimiento exitoso a la base de datos o cuando se tenga necesidad de **recuperarse** de algún daño severo donde la única alternativa sea recuperarse del respaldo, manualmente ejecutar el siguiente comando:

**symmir -g GPOSIIC -full establish**

Para sincronizar la copia de SRDF después de realizar algún mantenimiento exitoso a la base de datos o cuando se tenga necesidad de **recuperarse** de algún daño severo donde la única alternativa sea recuperarse del respaldo, manualmente ejecutar el siguiente comando:

**symrdf -g GPOSIIC -full establish**

Este tipo de recuperación de la base de datos, se usa en casos extremos, únicamente cuando sea la última alternativa, el tiempo de restauración es de aproximadamente 25 minutos en los cuales se transfieren aproximadamente 250 Giga bites. El tiempo de recuperación de una cinta de respaldo o drive DLT es de 8 horas como mínimo.

### **6.7 Para el respaldo de la BD y el SO por medio de una librería**

Los respaldos son una importante medida de precaución, para prevenir la pérdida total de la información, que es el bien más valioso de cualquier sistema de información. En este caso se usa el Software Data Protector Ver. 5.5 de la compañía HP.

Para realizar los respaldos, el software mencionado tiene la capacidad de enviar los respaldos a disco EMC CENTERA (respaldos recientes) o a cintas tipo DLT-70000 (respaldos históricos).

Se tiene un servidor UNIX N4000 dedicado para la administración de los respaldos de todos los equipos que se encuentran en el centro de cómputo central del SIIC, conocido como el CELL SERVER.

Se tienen dos tipos de respaldo, el de la "Base de Datos" y "File Systems" del sistema operativo. El respaldo de la base de datos tiene 2 componentes: El respaldo total de la Base de Datos a nivel 0 y el respaldo de los LOGS.

El respaldo de total de la base de datos a nivel 0, se programa para ejecutarse todos los días a las 3:00 AM, se realiza a esta hora en que se tiene un nivel mínimo de uso por parte los usuarios, para no degradar el rendimiento del sistema SIIC.

El respaldo de los LOGS, se dispara cada vez que se llena un área de LOG, el promedio de LOGs usados al día es de 20, aunque en días con alto nivel de transacciones puede llegar a 40 LOGs.

El respaldo de los File Systems más importantes del Sistema Operativo, se realiza a las 23:00 horas todos los días, también se tiene otro respaldo de 2 File Systems incremental a las 1:00 AM, el cual únicamente respalda los nuevos archivos generados en el día.

Los respaldos de la Base de Datos, por Ley se tienen que conservar por lo menos 5 años, con lo cual se tiene una biblioteca de cintas DLT-7000 de 2000 cintas históricas y 45 TB de disco para respaldos.

Los respaldos de los File Systems se conservan 6 meses.

## **CONCLUSIONES**

**a. Riesgos de no efectuar un monitoreo adecuado**

Los riesgos de no contar con un monitoreo adecuado de un base de datos, tal como la que soporta al sistema SIIC, sistema de información de misión crítica, el cual controla la venta de todos los productos petrolíferos de Pemex Refinación en todo el País, son múltiples y potencialmente peligrosos.

Este sistema debe de estar operando las 24 horas del día, durante todo el año, pues por el mismo se registran los ingresos de 1,200 millones de pesos por día, y se deben evitar los riesgos y costos inherentes de que el sistema salga de operación por periodos de tiempo no programados.

Para evitar una situación crítica se toman todas las medidas de precaución disponibles, redundancia en los servidores, redundancia en los enlaces de comunicación, redundancia en los equipos externos de almacenamiento de la información, y adicionalmente se debe de contar procesos de monitoreo y afinación de base de datos adecuados, los cuales permiten conocer los parámetros fundamentales de la operación diaria del sistema mencionado, y tomar acciones proactivas para evitar problemas potenciales.

**b. Ventajas y beneficios al efectuar un monitoreo adecuado**

Los beneficios de mantener controlados todos los procesos de operación de un sistema de alta disponibilidad, permiten proporcionar un servicio adecuado a los requerimientos de operación de aproximadamente 1,200 usuarios internos y 7,000 externos del sistema SIIC. El servicio proporcionado por el sistema SIIC en los últimos 3 años es mayor al 99.9 % de disponibilidad, esto es unas 8 horas de falla al año, las cuales no han sido consecutivas.

Lo anterior se logra, mediante un adecuado monitoreo y afinación de la base de datos, el cual debe de ser un proceso de mejora permanente, y por supuesto en coordinación con las áreas de desarrollo, de control de la producción, de redes y de la mesa de ayuda.

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX *CONCLUSIONES*

---

### c. **Monitoreo, una actividad permanente y cambiante**

El monitoreo y afinación de las bases de datos, es una actividad permanente y cambiante, pues las condiciones en la cual esta implementado un sistema de información, siempre están cambiando, tales como:

- Numero de usuarios
- Numero de programas
- I/O en discos
- Procesos batch
- Nuevos módulos en la aplicación,
- Nuevas reglas de replicación distribuida de datos
- Cambios en las reglas de operación del negocio
- Actualizaciones al sistema operativo,
- Actualizaciones del manejador de base de datos,
- Cambios de servidores
- Cambios de los equipos externos de almacenamiento
- etc.

El monitoreo y afinación de la base de datos, debe de ir ajustándose a la evolución propia del sistema de información a través del tiempo, pues generalmente la demanda de los diversos recursos va creciendo, por lo que los administradores de la base de datos, deben de estar pendientes de mantener un adecuado tiempo de respuesta y de proporcionar el nivel de disponibilidad requerido, más aún en ambientes que requieren de alta disponibilidad como es el caso del sistema que se ha descrito en este documento.



## BIBLIOGRAFÍA

Administrator's guide for Informix dynamic server, version 7.3  
Archive and backup guide for Informix dynamic server, version 7.3  
Backup and restore guide for Informix dynamic server, version 7.3  
Dbaccess user manual  
Error messages  
Getting started with Informix dynamic server, version 7.3  
Guide to Informix enterprise replication, version 7.3  
Informix guide to database design and implementation  
Informix guide to sql: reference  
Informix guide to sql: syntax  
Informix server administrator  
Informix storage manager administrator's guide  
Installation guide for Informix dynamic server version 7.3  
On-archive quick start guide, version 7.3  
Performance guide for Informix dynamic server, version 7.3  
Trusted facility manual for Informix dynamic server, version 7.3

Ingeniería del software  
Un enfoque practico  
Roger S. Pressman  
MC GRAW HILL  
Segunda edición

Tecnologías de interconectividad de redes.  
Merilee Ford  
PRENTICE-HALL  
México, 1998

## ANEXO 1

### Definición de una replica de datos hacia los servidores remotos

#### Creación de servidor central de replicación CDR:

```
cdr def serv -c prod_300 -i 0 -s rlcdR -r rlcdR -I -A /siic_onbar2/ats  
-R siic_onbar2/ris prod_300
```

#### Creación de servidor remoto de replicación CDR:

```
cdr def serv -c prod_$1 -i 0 -s cdrdbs -r cdrdbs -A /informix/ats  
-R /informix/ris -I -S prod_300 -L prod_$1
```

#### Definición de una replica de datos, del servidor central hacia todos los servidores remotos:

```
cdr define repl -C ignore -A -R -S trans r_centro$1 \  
"P sis_com@prod_300:informix.sccat_cto" "select * from sccat_cto" \  
"R sis_com@prod_$1:informix.sccat_cto" "select * from sccat_cto"
```

#### Definición de una replica de datos, del servidor central hacia un servidor remoto:

```
cdr define repl -C ignore -A -R -S trans r_serv_est_aso$1 \  
"P sis_com@prod_300:informix.scmc_serv_est" "select * from scmc_serv_est  
where iconc_cat_docto='RS' and icto_entr <> icto_serv and icto_serv=$1" \  
"R sis_com@prod_$1:informix.scmc_serv_est" "select * from scmc_serv_est  
where iconc_cat_docto='RS' and icto_entr <> icto_serv and icto_serv=$1"
```

#### Definición de una replica de datos, donde el secundario es el servidor central:

```
cdr define repl -C ignore -A -R -S trans r_param_cal_$1 \  
"P sis_com@prod_$1:informix.sccat_param_cal" "select * from  
sccat_param_cal" \  
"R sis_com@prod_300:informix.sccat_param_cal" "select * from  
sccat_param_cal"
```

#### Definición de una replica de datos, bidireccional:

```
cdr define repl -C timestamp -A -R -S trans r_prog_entr$1 \  
"sis_com@prod_300:informix.scmae_prog_entr" "select * from scmae_prog_entr  
where icto=$1" \  
"sis_com@prod_$1:informix.scmae_prog_entr" "select * from scmae_prog_entr  
where icto=$1"
```

**MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX**  
**ANEXO 2 PARÁMETROS CONFIGURABLES DEL SERVIDOR DE BASE DE DATOS**

---

**ANEXO 2**

**Parámetros configurables del servidor de Base de datos**

Informix Dynamic Server Version 7.31.UD7W4 -- On-Line -- Up 3 days 06:35:35 -- 1475984  
Kbytes

```
Configuration File: /informix/etc/prod_300shm
#*****
#
#           INFORMIX SOFTWARE, INC.
#
# Title:      onconfig.std
# Description: Informix Dynamic Server Configuration Parameters
#
#*****

# Root Dbspace Configuration

ROOTNAME      rootdbs      # Root dbspace name
ROOTPATH      /dev/vgrootdbs/rlrootdbs
              # Path for device containing root dbspace
ROOTOFFSET    0           # Offset of root dbspace into device (Kbytes)
ROOTSIZE      2000000     # Size of root dbspace (Kbytes)

# Disk Mirroring Configuration Parameters

MIRROR        0           # Mirroring flag (Yes = 1, No = 0)
MIRRORPATH    # Path for device containing mirrored root
MIRROROFFSET  0           # Offset into mirrored device (Kbytes)

# Physical Log Configuration

PHYSDBS       rootdbs     # Location (dbspace) of physical log
PHYSFILE      200000     # Physical log file size (Kbytes)

# Logical Log Configuration

LOGFILES      100        # Number of logical log files
LOGSIZE       50000     # Logical log size (Kbytes)

# Diagnostics

MSGPATH       /informix/prod_300shm.log
              # System message log file path
CONSOLE       /dev/console # System console message path
ALARMPROGRAM  /informix/etc/log_full.sh # Alarm program path
SYSALARMPROGRAM /informix/etc/evidence.sh # System Alarm program path
TBLSPACE_STATS 1

# System Archive Tape Device

TAPEDEV       /informix/etc/CINTA_FICTICIA_NO_BORRAR # Tape device path
#TAPEDEV      /dev/null # Tape device path
TAPEBLK       32         # Tape block size (Kbytes)
TAPESIZE      23500000   # Maximum amount of data to put on tape (Kbytes)

# Log Archive Tape Device

LTAPEDEV      /informix/etc/CINTA_FICTICIA_NO_BORRAR # Log tape device path
# LTAPEDEV    /dev/rmt/0m # Log tape device path
LTAPEBLK      32         # Log tape block size (Kbytes)
LTAPESIZE     23500000   # Max amount of data to put on log tape (Kbytes)

# Optical
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### ANEXO 2 PARÁMETROS CONFIGURABLES DEL SERVIDOR DE BASE DE DATOS

---

```

STAGEBLOB                # Informix Dynamic Server/Optical staging area

# System Configuration

SERVERNUM                1                # Unique id corresponding to a Dynamic Server in
stance
DBSERVERNAME            prod_300shm        # Name of default database server
DBSERVERALIASES         prod_300tcp        # List of alternate dbservernames
DEADLOCK_TIMEOUT        60                # Max time to wait of lock in distributed env.
RESIDENT                 0                # Forced residency flag (Yes = 1, No = 0)

MULTIPROCESSOR          1                # 0 for single-processor, 1 for multi-processor
NUMCPUVPS               10               # Number of user (cpu) vps
SINGLE_CPU_VP            0                # If non-zero, limit number of cpu vps to one

NOAGE                   1                # Process aging 1 es el bueno, 0 te quita priori
AFF_SPROC               2                # Affinity start processor
AFF_NPROCS              10               # Affinity number of processors

# Shared Memory Parameters

LOCKS                   6000000          # Maximum number of locks
BUFFERS                 300000           # Maximum number of shared buffers
NUMAIOVPS               3                # Number of IO vps
PHYSBUFF                64               # Physical log buffer size (Kbytes)
LOGBUFF                 64               # Logical log buffer size (Kbytes)
LOGSMAX                 2000             # Maximum number of logical log files
CLEANERS                 60              # Number of buffer cleaner processes
SHMBASE                 0x0              # Shared memory base address
SHMVIRTSIZE             300000           # initial virtual shared memory segment size
SHMADD                  200000           # Size of new shared memory segments (Kbytes)
SHMTOTAL                0                # Total shared memory (Kbytes). 0=>unlimited
CKPTINTVL               300             # Check point interval (in sec)
LRUS                     16             # Number of LRU queues
LRU_MAX_DIRTY           3                # LRU percent dirty begin cleaning limit
LRU_MIN_DIRTY           2                # LRU percent dirty end cleaning limit
LTXHWM                  40              # Long transaction high water mark percentage
LTXEHWM                 70              # Long transaction high water mark (exclusive)
TXTIMEOUT               0x12c           # Transaction timeout (in sec)
STACKSIZE               32              # Stack size (Kbytes)

# System Page Size
# BUFFSIZE - Dynamic Server no longer supports this configuration parameter.
#           To determine the page size used by Dynamic Server on your platform
#           see the last line of output from the command, 'onstat -b'.

# Recovery Variables
# OFF_RECVRY_THREADS:
# Number of parallel worker threads during fast recovery or an offline restore.
# ON_RECVRY_THREADS:
# Number of parallel worker threads during an online restore.

OFF_RECVRY_THREADS 10          # Default number of offline worker threads
ON_RECVRY_THREADS  1          # Default number of online worker threads

# Data Replication Variables
# DRAUTO: 0 manual, 1 retain type, 2 reverse type
DRAUTO                  0                # DR automatic switchover
DRINTERVAL              30              # DR max time between DR buffer flushes (in sec)
DRTIMEOUT               30              # DR network timeout (in sec)
DRLOSTFOUND              /informix/dr.lostfound # DR lost+found file path

# CDR Variables
CDR_LOGBUFFERS          2048            # size of log reading buffer pool (Kbytes)
CDR_EVALTHREADS         1,2            # evaluator threads (per-cpu-vp,additional)
CDR_DSLOCKWAIT          5                # DS lockwait timeout (seconds)
CDR_QUEUEMEM            6144            # Maximum amount of memory for any CDR queue (Kb
ytes) (Default 2048 AL)

```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### ANEXO 2 PARÁMETROS CONFIGURABLES DEL SERVIDOR DE BASE DE DATOS

---

```
# Backup/Restore variables
BAR_ACT_LOG      /tmp/bar_act.log
BAR_MAX_BACKUP  10
BAR_RETRY       1
BAR_NB_XPORT_COUNT 10
BAR_XFER_BUF_SIZE 31

# Informix Storage Manager variables
ISM_DATA_POOL   ISMData      # If the data pool name is changed, be sure to
                        # update $INFORMIXDIR/bin/onbar. Change to
ISM_LOG_POOL    ISMLogs      # ism_catalog -create_bootstrap -pool <new name>

# Read Ahead Variables
RA_PAGES        120          # Number of pages to attempt to read ahead
RA_THRESHOLD    100         # Number of pages left before next group

# DBSPACETEMP:
# Dynamic Server equivalent of DBTEMP for SE. This is the list of dbspaces
# that the Dynamic Server SQL Engine will use to create temp tables etc.
# If specified it must be a colon separated list of dbspaces that exist
# when the Dynamic Server system is brought online. If not specified, or if
# all dbspaces specified are invalid, various ad hoc queries will create
# temporary files in /tmp instead.

DBSPACETEMP     rltemp1,rltemp3,rltemp4    # Default temp dbspaces

# DUMP*:
# The following parameters control the type of diagnostics information which
# is preserved when an unanticipated error condition (assertion failure) occurs
# during Dynamic Server operations.
# For DUMPSHMEM, DUMPGCORE and DUMPCORE 1 means Yes, 0 means No.

DUMPDIR        /siic_onbar2    # Preserve diagnostics in this directory
DUMPSHMEM      0               # Dump a copy of shared memory
DUMPGCORE      0               # Dump a core image using 'gcore'
DUMPCORE       0               # Dump a core image (Warning:this aborts Dynamic
Server)
DUMPCNT        1               # Number of shared memory or gcore dumps for
# a single user's session

FILLFACTOR     90              # Fill factor for building indexes

# method for Dynamic Server to use when determining current time
USEOSTIME      0               # 0: use internal time(fast), 1: get time from O
S(slow)

# Parallel Database Queries (pdq)
MAX_PDQPRIORITY 75            # Maximum allowed pdqpriority
DS_MAX_QUERIES  30             # Maximum number of decision support queries
DS_TOTAL_MEMORY 3840          # Decision support memory (Kbytes)
DS_MAX_SCANS    1048576       # Maximum number of decision support scans
DATASKIP        off           # List of dbspaces to skip

# OPTCOMPIND
# 0 => Nested loop joins will be preferred (where
# possible) over sortmerge joins and hash joins.
# 1 => If the transaction isolation mode is not
# "repeatable read", optimizer behaves as in (2)
# below. Otherwise it behaves as in (0) above.
# 2 => Use costs regardless of the transaction isolation
# mode. Nested loop joins are not necessarily
# preferred. Optimizer bases its decision purely
# on costs.
OPTCOMPIND     2               # To hint the optimizer

ONDBSPACEDOWN  1               # Dbspace down option: 0 = CONTINUE, 1 = ABORT,
2 = WAIT
LBU_PRESERVE   0               # Preserve last log for log backup
OPCACHEMAX     0               # Maximum optical cache size (Kbytes)
```

## MONITOREO Y AFINACIÓN DE BASES DE DATOS INFORMIX

### ANEXO 2 PARÁMETROS CONFIGURABLES DEL SERVIDOR DE BASE DE DATOS

---

```
# HETERO_COMMIT (Gateway participation in distributed transactions)
# 1 => Heterogeneous Commit is enabled
# 0 (or any other value) => Heterogeneous Commit is disabled
HETERO_COMMIT    0

# Optimization goal: -1 = ALL_ROWS(Default), 0 = FIRST_ROWS
OPT_GOAL        -1

# Optimizer DIRECTIVES ON (1/Default) or OFF (0)
DIRECTIVES      1

# Status of restartable restore
RESTARTABLE_RESTORE off

NETTYPE         ipcshm,5,200,NET # Configure poll thread(s) for nettype
NETTYPE         soctcp,4,200,NET # Configure poll thread(s) for nettype

CDR_LOGDELTA    30                # % of log space allowed in queue memory
CDR_NUMCONNECT  16                # Expected connections per server
CDR_NIFRETRY    60                # Connection retry (seconds)
CDR_NIFCOMPRESS 0                # Link level compression (-1 never, 0 none, 9 max)
BAR_BSALIB_PATH /opt/omni/lib/libob2informix.sl
```