

Universidad de las Ciencias Informáticas

Facultad 2



**Título: “Almacenamiento Distribuido de los Datos en el
Ministerio de Auditoría y Control”**

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autores:

Alexis Yacel Del Castillo Peña
Erick Fidel Hernández Cedeño

Tutor:

Lic. Iván Hernández Ferrero

Asesor:

Ing. Daniel Ernesto Vargas Allegue

Consultantes:

Ing. Norge Fajardo Vega
Ing. Alain Fernández Del Toro

Ciudad de la Habana, Julio de 2008.

EN LA VEJEZ LA CIENCIA ES PARA NOSOTROS UN
CÓMODO REFUGIO; Y SI NO LA PLANTAMOS DE
JÓVENES, NO NOS DARÁ SOMBRA CUANDO SEAMOS
VIEJOS.

CHESTERFIELD

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los 2 días del mes de Julio de 2008.

Alexis Yacel Del Castillo Peña

Erick Fidel Hernández Cedeño

Lic. Iván Hernández Ferrero



DATOS DE CONTACTO

Alexis Yacel Del Castillo Peña

[aypena\(at\)estudiantes\(dot\)uci\(dot\)cu](mailto:aypena(at)estudiantes(dot)uci(dot)cu)

Erick Fidel Hernández Cedeño

[ehernandezc\(at\)estudiantes.uci\(dot\)cu](mailto:ehernandezc(at)estudiantes.uci(dot)cu)

Lic. Iván Hernández Ferrero

[ivanhf\(at\)uci\(dot\)cu](mailto:ivanhf(at)uci(dot)cu)

Ing. Daniel Ernesto Vargas Allegue

[dvargas\(at\)uci\(dot\)cu](mailto:dvargas(at)uci(dot)cu)

Ing. Norge Fajardo Vega

[norge\(at\)uci\(dot\)cu](mailto:norge(at)uci(dot)cu)

Ing. Alain Fernandez Del Toro

[afernandez\(at\)uci\(dot\)cu](mailto:afernandez(at)uci(dot)cu)

AGRADECIMIENTOS

Primero que a todo, le agradecemos a la Revolución por habernos dado la oportunidad de formarnos como profesionales, a nosotros, hijos de campesinos humildes.

Le agradecemos especialmente a Daniel Ernesto Vargas Allegue, por su dedicación y presencia permanente, con las mejores ideas para afrontar el trabajo que desarrollábamos.

Alexis Yacel

Sin todas las personas que han estado pendientes de nuestro desempeño, hubiera sido una quimera llegar a cumplir este sueño.

Especialmente a mi familia, por la confianza mostrada en mí, por lo orgulloso que siempre me hicieron sentir por los logros que iba obteniendo a lo largo de mi vida estudiantil, y por haberme dado ánimo cuando la lucha se ponía difícil.

También a mis amigos que estuvieron para momentos de sosiego, y para ayudarme siempre que los necesité; son otros que merecen mi eterno agradecimiento: Islandy, Angel, Iyugnis.

A mis compañeros de aula; sin su ayuda y preocupación, hubiera sido más difícil. Gracias, Marcos, Liane, Yeilin, Joan, Félix, Rigo, Tobías, Dairon, Yoandys.

Erick Fidel

Agradecer a mi familia en especial a mi madre Marisbel. Gracias, muchas gracias por haber confiado siempre en mí. Gracias por los sacrificios que has hecho, por la dedicación y por la eterna paciencia.

Agradecimientos a mi abuela Reina, siempre dispuesta a realizar sacrificios y siempre alentándome a no rendirme.

A mis familiares que siempre me brindan apoyo.

A los amigos y colegas que me han ayudado en momentos de crisis. Rigo (Tra), Felipón, José Antonio Pla.



DEDICATORIA

Alexis Yacel

Esta tesis va dedicada a todos los que han mostrado interés en que yo me haga ingeniero, a todos los que lo han deseado, y han soñado con este día tanto como yo. A todos los que se han ocupado de que así sea.

Muy especialmente a mi familia, en la que cada miembro ha jugado un papel súper importante:

A mi hermano Yay, que ha sido siempre el faro en que debo mirarme.

A mi mamá Ana Ibis, que ha hecho todo para que llegara este día

A mi abuela Librada, por quien nunca tuve excusas de no lograrlo, sabiendo que yo estudié más que lo que a ella las circunstancias de la vida le habían permitido.

A mi abuelo "Chino", que siempre ha visto la parte práctica del estudio en la vida diaria. Todo un luchador.

A mis padres Pedrito y Chávez, quienes nunca dejaron de señalarme el camino.

A todos mis tíos y primos, que siempre estuvieron pendientes de mí.

A todos los que ya no están físicamente, y a todos los que no cumplieron este sueño.

Erick Fidel

Dedico los resultados de esta tesis a quienes me han apoyado en todo momento. A las personas que me han alentado a no desistir, a seguir siempre adelante, a todos aquellos que de alguna forma han hecho posible que me gradúe como Ingeniero.

Dedicado especial para mi familia, sin su apoyo este sueño no hubiese sido posible.

A mi madre (mami).

A mi abuela (mi otra mami).

Dedicado a mi hermano.

Todos ustedes quienes siempre esperan un buen resultado de mí, espero no defraudarles.



RESUMEN

Este trabajo se propone lograr un sistema para el almacenamiento distribuido de datos en el lenguaje libre PostgreSQL, que pueda gestionar la información que se usará en el Ministerio de Auditoría y Control (MAC) de la República de Cuba.

Resulta muy necesario y actual este trabajo, lo que se evidencia en que un software de las características de uno que gestione la actividad de un ministerio, debe tener una BD que almacene dicha información, y no otro tipo de almacenamiento, dígase, por ejemplo, cualquier tipo de ficheros, y no solo por la capacidad almacenadora del mismo, sino por las posibilidades de interacción que ofrece una BD, que despunta casi siempre como la más óptima solución, por complejo o sencillo que sea un problema.

La informatización de este ministerio tendrá a su disposición la gestión de sus datos con uno de los gestores que más impulso ha tenido en los últimos años, lo que garantiza además que no va a quedar obsoleto tan cerca en el tiempo, dadas las ventajas que presenta, además.

El equipo de investigadores, al comenzar la solución del problema, se trazó varias metas, entre las que se pueden mencionar lograr la propuesta de solución al tema de almacenamiento distribuido de datos que soportará el negocio del MAC.

Este trabajo no ha sido presentado a la comunidad científica, por lo que se descarta que hasta ahora haya obtenido resultados, sin embargo, aunque los resultados que ha obtenido no han sido a nivel competitivo, si lo han sido en cuanto a cumplimiento de los objetivos, lo que para el equipo indica un cumplimiento de las expectativas.



ABSTRACT

In this work, we take care of developing an investigation to achieve a system of distributed storage of data in the free language PostgreSQL that can negotiate the information that will be used in the Ministry of Audit and Control (MAC) of the Republic of Cuba.

This topic its an important current concern of anyone that works with databases, what is evidenced in that a software of the characteristics of one that negotiates the activity of a ministry, should have a BD that stores it's information, and not another storage type, for example, any type of files, and not alone for the storing capacity of the same one, but for the interaction possibilities that a BD offers that it blunts almost always as the best solution, for complex or simple that is a problem.

The informatization of this ministry will have the administration of its data at your disposal with one of the agents that more impulse has had in the last years, what also guarantees that it won't be obsolete so close in the time, given the advantages that it presents, also.

We, when beginning the solution of the problem, trace ourselves several goals, among those that we can mention to achieve the solution proposal to the topic of distributed storage of data that it supported the business of the MAC.

This work has not been presented to the scientific community, for what is discarded that up to now it has obtained results, however, although the results that he/she has obtained they have not been at competitive level, if they have been it as for execution of the objectives, what indicates an execution of the expectations for us.

TABLA DE CONTENIDOS

AGRADECIMIENTOS.....	I
DEDICATORIA.....	II
RESUMEN.....	III
INTRODUCCIÓN.....	1
CAPÍTULO 1: SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.....	5
1.1 Introducción.....	5
1.2 Consideraciones Generales.....	5
1.2.1 Sistemas de bases de datos distribuidas.....	6
1.2.2 Tipos de Sistemas Gestores de Bases de Datos.....	8
1.2.3 Ambientes con múltiples procesadores.....	9
1.2.4 Estructura de Base de Datos Distribuidas.....	10
1.2.5 Consideraciones al distribuir la base de datos.....	11
1.2.5.1 Ventajas.....	11
1.2.5.2 Desventajas.....	12
1.3 Aspectos importantes de los SMBD distribuidos.....	13
1.4 Estado del Arte.....	15
1.5 Diseño de SBDD.....	17
1.6 Diseño de la distribución.....	21
1.7 Fragmentación.....	22
1.7.1 Tipos de Fragmentación.....	24
1.7.1.1 Fragmentación Horizontal.....	25
1.7.1.2 Fragmentación Vertical.....	25
1.7.1.3 Fragmentación mixta o híbrida.....	26
1.7.2 Grado de Fragmentación.....	26
1.8 Transparencia y Autonomía.....	26
1.9 Procesamiento distribuido de consultas.....	27
1.10 Manejo de Transacciones.....	27
1.10.1 Condiciones de terminación de una transacción.....	28

1.11	Control de Concurrencia.....	29
1.12	Manejo de Bloqueos.....	29
1.13	Confiabilidad.....	31
1.13.1	Tipos de fallas en SMBDD.....	31
1.14	Selección del Coordinador.....	32
1.15	Conclusiones.....	32
CAPÍTULO 2: SISTEMAS DE BASES DE DATOS REPLICADAS.....		34
2.1.	Introducción.....	34
2.2.	Conceptos.....	34
2.3.	Objetivos.....	36
2.4.	Ventajas y desventajas.....	37
2.5.	¿Para qué usar replicación?.....	37
2.5.1.	Distribución de datos a otras ubicaciones.....	38
2.5.2.	Consolidación de datos desde otras ubicaciones.....	39
2.5.3.	Intercambio bidireccional de datos con otras ubicaciones.....	40
2.5.4.	Otros requerimientos.....	40
2.6.	Entornos de réplica.....	41
2.7.	Clasificación de los algoritmos de replicación.....	41
2.7.1.	Modelo funcional de la replicación.....	42
2.7.2.	Replicación en Bases de Datos.....	42
2.7.2.1.	Comunicación en Grupo.....	42
2.8.	Modelos de Consistencia.....	43
2.8.1.	Linealidad y consistencia secuencial.....	44
2.9.	Modelos de Distribución.....	44
2.10.	¿Cómo se implementa la replicación?.....	45
2.10.1.	Replicación sincrónica.....	45
2.10.2.	Replicación asincrónica.....	45
2.10.2.1.	Replicación asincrónica entre base de datos.....	46
2.10.2.2.	Replicación asincrónica entre procesos.....	46
2.11.	Técnicas de Replicación.....	47
2.11.1.	Sincrónica.....	47
2.11.1.1.	Confirmación en dos Fases(Two Phases Commit).....	47

2.11.2. Asincrónica	47
2.11.2.1. Descarga y Recarga (Dump and Reload)	47
2.11.2.2. Instantánea (Snapshot)	48
2.11.2.3. Disparadores(Triggers)	48
2.12. Servicios de Alta Disponibilidad.....	49
2.12.1. Arquitectura Cotilla.....	49
2.12.2. Arquitectura de Bayou.....	51
2.12.3. Arquitectura Coda.....	51
2.13. Recuperación en protocolos de consistencia.	51
2.14. Conflictos de la replicación de datos.	52
2.14.1. Conflicto de Actualización	52
2.14.2. Conflicto de unicidad.....	53
2.14.3. Conflicto de supresión	53
2.14.4. Conflicto de orden	54
2.15. Estado del arte y análisis de productos existentes	54
2.15.1. Solución a los Servicios Autónomos de Registros y Notarias (SAREN).	54
2.15.1.1. Escenario	54
2.15.1.2. Modelo de Replicación	55
2.15.1.3. Solución	55
2.15.1.4. Alcance.....	56
2.15.2. Ministerio de las Fuerzas Armadas.	56
2.15.2.1. Escenario	56
2.15.2.2. Modelo de Replicación	57
2.15.2.3. Solución	57
2.15.2.4. Alcance.....	60
2.16. Conclusiones	60
CAPÍTULO 3: COMPARACIONES Y ELECCION DE ESTRATEGIA DE ALMACENAMIENTO.....	62
3.1 Introducción.	62
3.2 Comparación entre variantes del Almacenamiento Distribuido de Datos	62
3.2.1 Tipología de la red.	64
3.2.1.1. Velocidad de la Red.....	64
3.2.2. Ventajas y desventajas.....	64

3.2.3.	Aspectos importantes	65
3.2.4.	Estado del arte	66
3.2.5.	Diseño de la distribución y la replicación	66
3.2.6.	Procesamiento de Consultas	66
3.2.7.	Control de Concurrencia y Consistencia	67
3.2.8.	Confiabilidad	68
3.2.9.	Selección del Coordinador.....	69
3.3.	Características del MAC	70
3.4.	Software a emplear en el entorno de replicación	71
3.4.1.	Sistema operativo Linux.....	71
3.4.1.1.	Características.....	73
3.4.2.	PostgreSQL.....	74
3.4.2.1.	Características.....	74
3.4.3.	Slony I.....	76
3.4.3.1.	Características.....	77
3.5.	Elección del tipo de Almacenamiento Distribuido para el MAC	77
3.6.	Conclusiones	78
CAPITULO 4: PROPUESTA		79
4.1.	INTRODUCCIÓN.....	79
4.2.	Diseño.....	79
4.2.1.	Jerarquía.....	79
4.2.2.	Nodos Implicados	81
4.2.3.	Multiusuario	82
4.2.4.	Conexiones de red.....	83
4.3.	Bases de Datos	84
4.3.1.	Estructura de la Información en los Servidores	84
4.3.2.	Consideraciones generales.....	85
4.3.3.	Manejo de datos de forma local.....	86
4.3.4.	Recuperación de las BD.....	86
4.3.4.1.	Utilizando Archivos LOG.....	86
4.3.4.2.	Salvas (Backup).....	87
4.3.4.2.1.	Salvas en Frío (Cold Backup).....	87

4.3.4.2.2. Salvas en Caliente (Hot Backup)	87
4.4. Definiendo Maestros y Esclavos.....	88
4.4.1. Flujo de Replicación	88
4.4.2. Alternativa	89
4.4.2.1. Pérdida del Servicio	89
En el análisis de este tema, como pérdida del servicio se asumirá en primero lugar, la perdida de conexión entre la localidad y el resto del sistema, además de la caída del nodo, relacionadas con fallos no referidos a la conexión.	90
4.4.2.1.1. Pérdida del Servicio del Nodo Provincial	90
4.4.2.1.2. Pérdida del Servicio del Nodo Regional.....	90
4.4.2.1.3. Pérdida del Servicio del Nodo del Ministerio	90
4.4.2.1.4. Recomendación para el caso de caída de nodos	91
4.4.3. Alternativa de Comunicación usando la Red de Fibra Óptica Nacional	91
4.5. Aspectos Importantes	92
4.6. Conclusiones	93
CONCLUSIONES	94
RECOMENDACIONES	95
REFERENCIAS BIBLIOGRÁFICAS	96
BIBLIOGRAFÍA.....	97
ANEXOS	100
GLOSARIO Y SIGLAS	101

INTRODUCCIÓN

El Ministerio de Auditoría y Control, en lo adelante MAC, se encarga de centralizar la gestión auditora y controladora de la nación en un solo organismo de administración central del estado (OACE), que responda a las necesidades actuales, se ha propuesto informatizarse, tarea encomendada a la Universidad de las Ciencias Informáticas, cuyos resultados servirán como experiencia para los otros OACE que se informaticen en el futuro.

El MAC tiene su sede en la capital, y tiene delegaciones en cada una del resto de las provincias, por lo que se plantea el reto de lograr que la información fluya correctamente de un lado a otro del país, lo que a nivel de informática es solucionable con un ambiente de almacenamiento distribuido de datos, que garantice la eficiente gestión de la información generada en sus procesos habituales, en término de lograr la comunicación física, hacer que la información, efectivamente, viaje por canales seguros, y que el tiempo de respuesta de la interacciones con la base de datos, sea lo menor posible.

En este trabajo, se asumió la investigación de un problema muy común en el ámbito actual de bases de datos, y es el de lograr alguna variante de Almacenamiento Distribuido de Datos, que pueda cumplir las funcionalidades que se piden actualmente, cumpliendo con los requisitos de usabilidad, rendimiento, soporte, portabilidad, seguridad, y confiabilidad, concretamente en PostgreSQL,

Este tema tiene la importancia que su actualidad tiene, pues justo ahora no se había logrado responder a la inquietud expresada anteriormente, y al estar presente en el proyecto de informatización del MAC, pues tiene una significación sin igual, amparada en su actualidad, por lo que era altamente priorizado que esta investigación llegara a buen término, ya que su apremiante utilización así lo requería.

La actualidad de este tema está justificada por el hecho de que hasta ahora no se había desarrollado investigación alguna para resolver el problema de datos en ambiente distribuido, que como se ha ido esbozando, es la respuesta a las necesidades del MAC. Esta actualidad le confiere una importancia mayor, puesto que la urgencia de una solución fue también una de las fuerzas motrices de la investigación.

Como antecedente a esta investigación, se pueden encontrar otras que fueron en su mayoría, realizadas en el exterior, según la documentación a la que tuvimos acceso: Tesis de Replicación de Datos para el MINFAR, sitio de desarrollo de PostgreSQL, donde está la documentación de la versión

en desarrollo, sitios que se dedican a la difusión del conocimiento, además de las listas de correo de PostgreSQL, pero a las que tenemos el deber de catalogar como insuficientes, incompletas, y muchas veces de nivel científico no muy comprometido, característica esta relacionada principalmente con las listas de correo de Internet. Además de las ya mencionadas, hay que agregar otras que fueron desarrolladas en un entorno académico, universidades de Latinoamérica y España, que fueron de gran apoyo.

Con este trabajo, el equipo de investigadores afronta el problema científico de plantear la arquitectura que solucione el tema de Almacenamiento Distribuido de Datos, que forma parte del proyecto de informatizar el MAC, el primero que se informatizará en el país, por lo que reviste importancia vital, teniendo en cuenta que las experiencias de la informatización de este ministerio, serán recopiladas en un expediente con el objetivo de usarlas en los próximos ministerios a informatizar.

El objeto de estudio de esta investigación son los ambientes distribuidos de información usando los gestores de bases de datos, donde podemos encontrar los Sistemas de Bases de Datos Distribuidos (SBDD) y los Sistemas de Bases de Datos Replicados (SBDR).

El campo de acción está constituido por el conjunto opciones de Ambientes Distribuidos de Datos que se le puede dar a la información del MAC, dentro de lo que enunciamos que PostgreSQL no soporta la distribución/replicación nativamente, solo con soporte de terceros, además de que hay escollos como la pérdida de información, cuando se usan métodos asíncronos de interacción, y otro es la magnitud de tiempo en que se ejecuta una acción y es propagada a los otros servidores, cuando se usan métodos sincrónicos.

Este trabajo se ha encaminado sobre la hipótesis de que un sistema distribuido de bases de datos servirá para dar una solución óptima a las necesidades informáticas que surjan de informatizar el MAC.

El equipo, al comenzar la solución del problema, tenía el objetivo de encontrar una solución científicamente fundamentada que permitiera obtener una arquitectura apropiada para el almacenamiento distribuido de datos MAC.

Para cumplir nuestro objetivo específico, nos centramos en ciertas tareas que nos permitirían lograr su alcance, dentro de las que mencionamos de forma general:

- Elaboración del Diseño Teórico
- Elaboración del Diseño metodológico

- Confección del Prediseño de la Investigación
- Confección del Diseño de la Investigación (Definitivo)

Siendo más específicos, podemos enunciar como tareas:

- Realizar la búsqueda bibliográfica sobre almacenamiento distribuido de datos, así como de los sistemas informáticos existentes para la ayuda al mismo
- Comparar de las últimas tendencias y tecnologías para construir una aplicación distribuida.
- Detallar la estructura organizacional - tanto jerárquica como geográfica- y necesidades del MAC.
- Realizar el diseño de la arquitectura del sistema de almacenamiento distribuido de datos.

Los métodos científicos que fueron usados son de dos tipos:

- Métodos empíricos: observación y entrevista
- Métodos teóricos: Analítico-Sintético, Inductivo-Deductivo, Hipotético-Deductivo y Modelación

La observación se ampara en la revisión documental que realizó el equipo en el estudio del estado del arte, haciendo registro visual de lo que se presentaba en el tema del almacenamiento distribuido de los datos.

Dentro de las entrevistas, hay que mencionar las que se le realizaron a los consultantes, de las que no se hicieron grabaciones, pero si aportaron claridad en los temas por los que se preguntó.

El método Analítico-Sintético fue usado en la división, mediante el análisis de la realidad presente en el tema analizado por el equipo, dividiéndola en partes para su mejor estudio, y que es la estructura del trabajo por capítulos, secciones; mientras que la síntesis, permitía unificar lo los fragmentos.

El Inductivo-Deductivo se usó para llegar de casos particulares presentes en el ámbito del almacenamiento distribuido, a elementos generales, mediante la inducción, mientras que la deducción es el procedimiento de llegar a especificidades desde generalidades.

El método Hipotético-Deductivo se usó en el caso de que a partir de la hipótesis que se planteó, hacer especificidades.

La Modelación se basa en que hubo que hacer esbozos de cómo se debían hacer los diseños propuestos.

Este trabajo, como principal resultado obtenido, tiene el de haber propuesto la solución al tema de Almacenamiento Distribuido de Datos que se utilizará en el proyecto de informatizar en MAC, que como hemos mencionado antes, sería la primera experiencia para informatizar otros ministerios, y punto en el que radica su novedad científica. Esta aportación teórica que hemos hecho, tiene la misma significación práctica, al ser utilizada inmediatamente en el objetivo mayor de informatizar este ministerio.

La novedad de este trabajo no radica en se hayan descubierto metodologías, o modos de interpretar la realidad de almacenamiento de datos en ambientes distribuidos, sino que se ha utilizado lo que se conoce sobre el tema, a la realidad del primer ministerio del país en informatizarse, y con este trabajo tampoco el equipo se plantea la búsqueda de soluciones que requirieran nueva infraestructura, solamente la ya existente en el país.

También es posible imaginar que el aporte teórico de este trabajo consiste en haber planteado la solución al tema de bases de datos distribuidas, que en la práctica, se usará muy pronto.

Como el trabajo versa sobre Almacenamiento Distribuido de Datos, la estructura en que se desarrolla es la que se muestra a continuación:

Capítulo 1: Sistemas de Bases de Datos Distribuidos

En esta sección se analizan las principales características de esta variante de almacenamiento distribuido de datos, haciendo énfasis en el diseño de la distribución, manejo distribuido de consultas, transacciones, y otros elementos que dependen de los anteriores.

Capítulo 2: Sistemas de Bases de Datos Replicados

En esta sección se analiza la teoría que soporta este tipo de ambiente, y se hace especial énfasis, al final del capítulo, en estudiar sistemas que brindan servicio en la actualidad.

Capítulo 3: Comparaciones y elección de estrategia de almacenamiento

En este acápite se realiza una comparación de ambos sistemas de distribución con el objetivo de definir cual de ellos resulta más apropiado para proponer su aplicación en el MAC.

Capítulo: Propuesta

En este tema se define la manera en que el equipo pretende que se ejecute lo propuesto como resultado de la investigación.

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

CAPÍTULO 1: SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

1.1 Introducción

En este capítulo, se ofrece una visión general sobre los aspectos referidos a los Sistemas de Bases de Datos Distribuidas, pensando en la manera de aplicar estos elementos al modelado de lo que será la base de datos del MAC, por lo que se pasa a definir las categorías presentes en nuestra investigación, definiendo el concepto de base de datos distribuida, de sistemas de bases de datos distribuidas, sistema de manejo de bases de datos distribuidas, y el resto de la teoría relacionada.

1.2 Consideraciones Generales.

En un sistema de base de datos distribuida, los datos se almacenan en varias computadoras. Las computadoras de un sistema distribuido se comunican entre sí a través de diversos medios de comunicación, tales como cables de alta velocidad o líneas telefónicas. No comparten la memoria principal ni el reloj.

Los procesadores de un sistema distribuido pueden variar en cuanto a su tamaño y función. Pueden incluir computadoras de baja capacidad, estaciones de trabajo y sistemas de computadoras grandes de aplicación general. Estos procesadores reciben diferentes nombres, tales como localidades, nodos o computadores. (Santos, 2005)

Un sistema distribuido de bases de datos es un conjunto de localidades, cada uno de las cuales puede participar en la ejecución de transacciones que accedan a datos de una o varias localidades.

Cabe destacar que en un sistema de BDD tiene las siguientes características:

- Cada sitio es un sistema de base de datos completo por derecho propio, es decir, cada sitio tiene bases de datos reales, usuarios locales, Sistema Gestor de Bases de Datos (SGBD) local y software de administración de transacciones y administrador de comunicación de datos local propios. De esta forma, un usuario determinado puede realizar operaciones sobre los datos desde su propio sitio local, tal como si ese sitio no participara en el sistema distribuido.
- Los sitios han acordado trabajar juntos, a fin de que un usuario cualquiera pueda acceder a los datos desde cualquier lugar de la red, exactamente como si los datos estuvieran guardado en el propio sitio del usuario. (Vega, 2007)

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

1.2.1 Sistemas de bases de datos distribuidas¹.

Los sistemas de bases de datos distribuidas son un caso particular de los **sistemas de cómputo distribuido** en los que un conjunto de elementos de procesamiento autónomos (no necesariamente homogéneos) se interconectan por una red de comunicaciones y cooperan entre ellos para realizar sus tareas asignadas. Históricamente, el cómputo distribuido se ha estudiado desde muchos puntos de vista. Así, es común encontrar en la literatura un gran número de términos que se han usado para identificarlo. Entre los términos más comunes que se utilizan para referirse al cómputo distribuido podemos encontrar: funciones distribuidas, procesamiento distribuido de datos, multiprocesadores, multicomputadoras, procesamiento satelital, procesamiento tipo "backend", computadoras dedicadas y de propósito específico, sistemas de tiempo compartido, sistemas funcionalmente modulares.

Existen muchas componentes a distribuir para realizar una tarea. En computación distribuida los elementos que se pueden distribuir son Control, Datos, Funciones, y Procesamiento lógico.

Una **base de datos distribuida** (BDD) es un conjunto de múltiples bases de datos *lógicamente relacionadas* las cuales se encuentran distribuidas entre diferentes sitios interconectados por una red de comunicaciones.

Un **sistema de bases de datos distribuidos** (SBDD) es un sistema en el que múltiples sitios de bases de datos están ligados por un sistema de comunicaciones, de forma que, un usuario en cualquier sitio puede acceder a los datos en cualquier parte de la red exactamente como si los datos estuvieran almacenados en su sitio propio.

Un **sistema de manejo de bases de datos** (SMBD, también conocido como Sistema Gestor de Bases de Datos (SGBD)) es un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. (Wikipedia, 2008)

Un **sistema de manejo de bases de datos distribuidas** (SMBDD) es aquel que se encarga del manejo de la BDD y proporciona un mecanismo de acceso que hace que la distribución sea

¹ Información obtenida en http://www.cs.cinvestav.mx/SC/prof_personal/adiaz/Disdb/

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

transparente a los usuarios. El término transparente significa que la aplicación trabajaría, desde un punto de vista lógico, como si un solo SMBD ejecutado en una sola máquina, administrara esos datos.

Un **sistema de base de datos distribuida** (SBDD) es entonces el resultado de la integración de una base de datos distribuida con un sistema para su manejo.

El medio ambiente típico de un SMBDD consiste de un conjunto de sitios o nodos los cuales tiene un sistema de procesamiento de datos completo que incluye una base de datos local, un sistema de manejo de bases de datos y facilidades de comunicaciones. Si los diferentes sitios pueden estar geográficamente dispersos, entonces, ellos están interconectados por una red de tipo WAN. Por otro lado, si los sitios están localizados en diferentes edificios o departamentos de una misma organización pero geográficamente en la misma ubicación, entonces, están conectados por una red local (LAN)

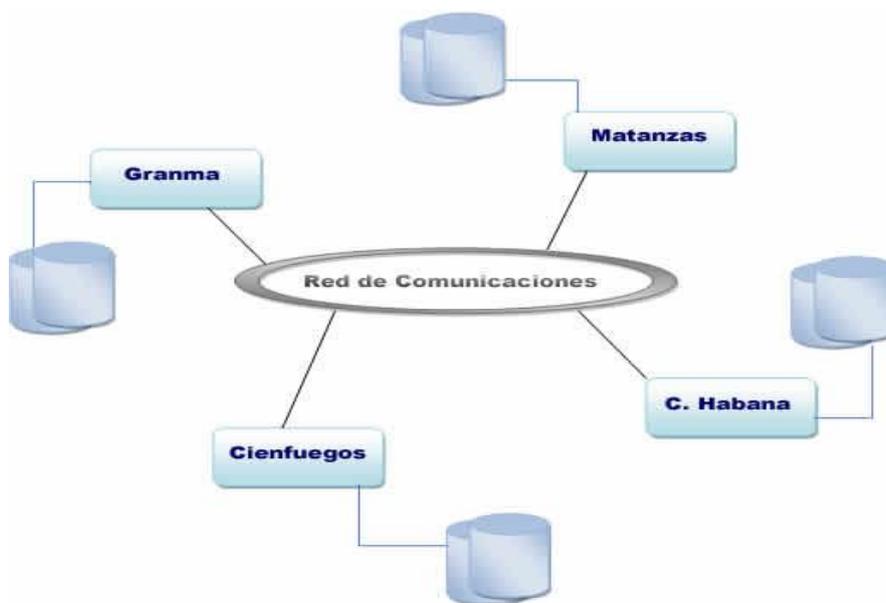


Figura 1.1. Un medio ambiente distribuido para bases de datos

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

1.2.2 Tipos de Sistemas Gestores de Bases de Datos².

A lo largo del desarrollo histórico del tema de BDD, se han tenido en cuenta algunos criterios para su clasificación y estudio, y se mencionan a continuación:

- Autonomía
- Distribución
- Heterogeneidad

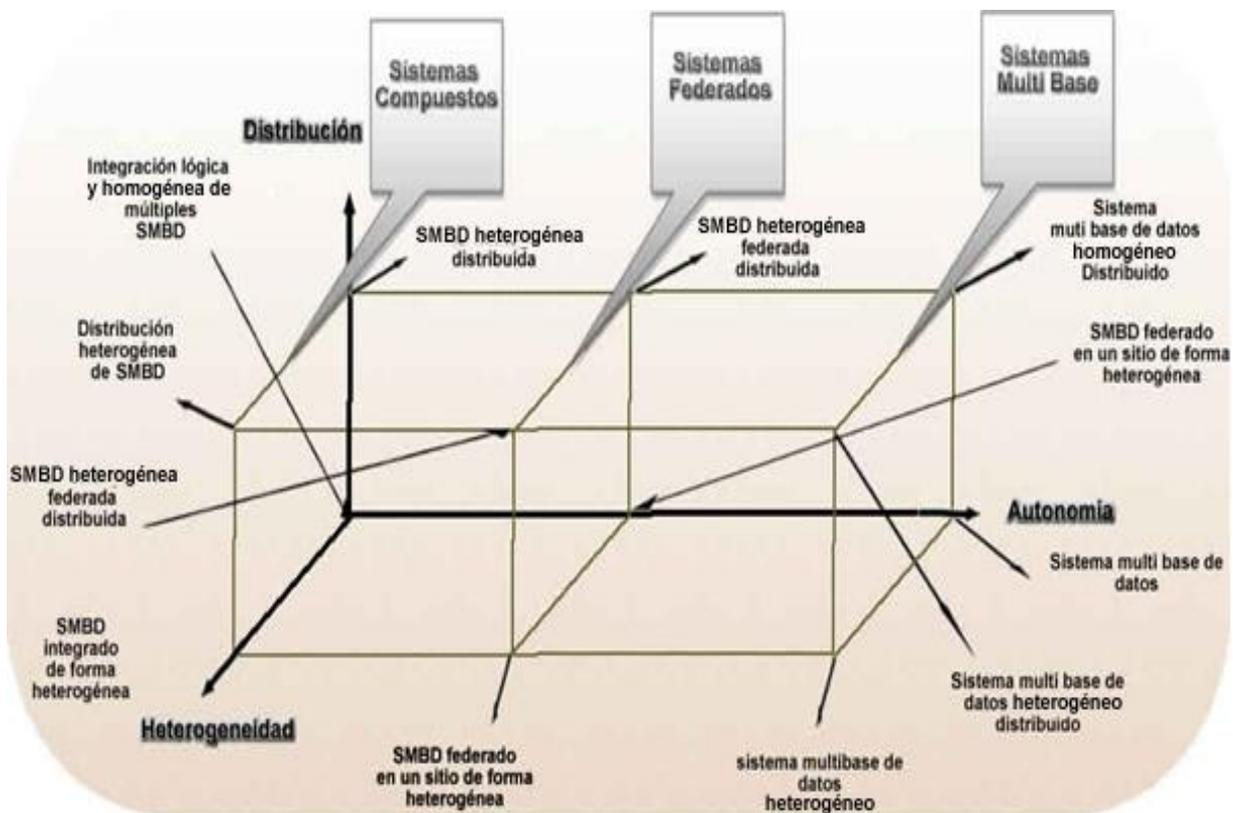


Figura 1.2. Alternativas de implementación de sistema de manejo de bases de datos.

² Información obtenida en <http://basesdatos.uc3m.es/fileadmin/Docencia/FuBD/Teoria/TemaVII0607.pdf>

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

1.2.3 Ambientes con múltiples procesadores³.

Desde el punto de vista de las bases de datos, conceptualmente existen tres tipos de ambientes que se integran con múltiples procesadores:

1. Arquitecturas de memoria compartida:

Consisten de diversos procesadores los cuales acceden una misma memoria y una misma unidad de almacenamiento (uno o varios discos). Algunos ejemplos de este tipo son las computadoras Sequent Encore y los mainframes IBM4090 y Bull DPS8 (Figura 1.3).

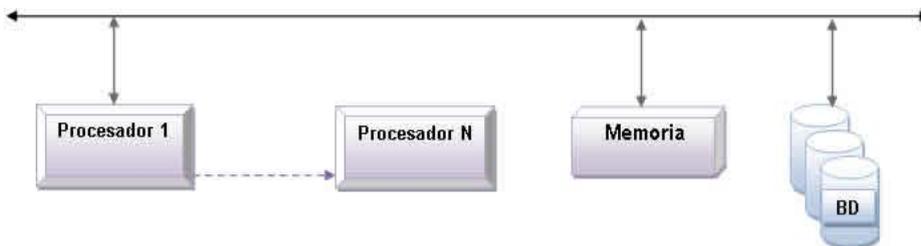


Figura 1.3. Arquitectura de memoria compartida.

2. Arquitecturas de disco compartido:

Consiste de diversos procesadores cada uno de ellos con su memoria local pero compartiendo una misma unidad de almacenamiento (uno o varios discos). Ejemplos de estas arquitecturas son los cluster de Digital, y los modelos IMS/VS Data Sharing de IBM (Figura 1.4).

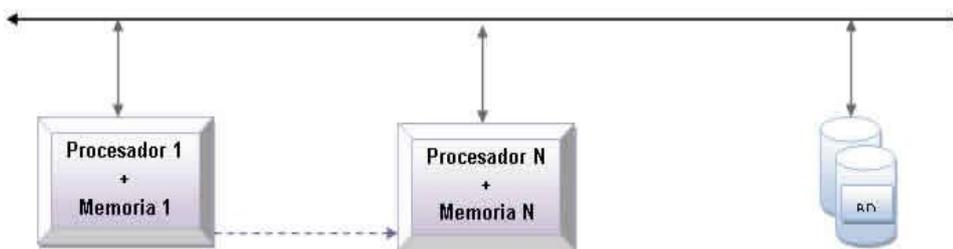


Figura 1.4. Arquitectura de disco compartido.

3. Arquitecturas nada compartido:

³ Información obtenida en http://www.cs.cinvestav.mx/SC/prof_personal/adiaz/Disdb/

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

Consiste de diversos procesadores cada uno con su propia memoria y su propia unidad de almacenamiento. Aquí se tienen los clusters de estaciones de trabajo, las computadoras Intel Paragon, NCR 3600 y 3700 e IBM SP2 (Figura 1.5).

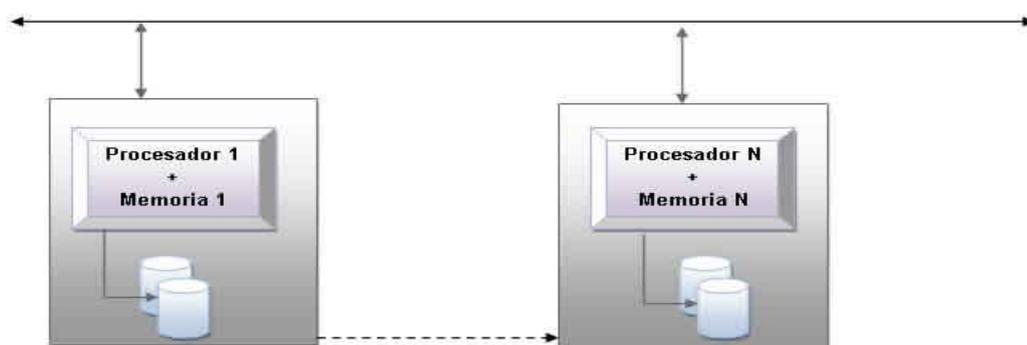


Figura 1.5. Arquitectura nada compartido.

1.2.4 Estructura de Base de Datos Distribuidas⁴.

Un sistema distribuido de base de datos consiste en un conjunto de localidades, cada una de las cuales mantiene un sistema de base de datos local. Cada localidad puede procesar transacciones locales, o bien transacciones globales entre varias localidades, requiriendo para ello comunicación entre ellas.

Las localidades pueden conectarse físicamente de diversas formas, las principales son:

- Red totalmente conectada
- Red prácticamente conectada
- Red con estructura de árbol
- Red de estrella
- Red de anillo

Las diferencias principales entre estas configuraciones son:

- Coste de instalación: El coste de conectar físicamente las localidades del sistema

⁴ Información obtenida en http://www.cs.cinvestav.mx/SC/prof_personal/adiaz/Disdb/

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

- Coste de comunicación: El coste en tiempo y dinero que implica enviar un mensaje desde la localidad A hasta la B.
- Fiabilidad: La frecuencia con que falla una línea de comunicación o una localidad.
- Disponibilidad: La posibilidad de acceder a información a pesar de fallos en algunas localidades o líneas de comunicación.

Las localidades pueden estar dispersas, ya sea por un área geográfica extensa (a lo largo de un país), llamadas redes de larga distancia; o en un área reducida (en un mismo edificio), llamadas redes de área local. Para las primeras se utilizan en la comunicación líneas telefónicas, conexiones de microondas y canales de satélites; mientras que para las segundas se utiliza cables coaxiales de banda base o banda ancha y fibra óptica.

1.2.5 Consideraciones al distribuir la base de datos⁵.

Existen varias razones para construir sistemas distribuidos de bases de datos, dentro de las que se incluyen compartir la información, fiabilidad y disponibilidad y agilizar el procesamiento de las consultas. Pero también presenta entre sus desventajas, desarrollo de software más costoso, mayor posibilidad de errores y costos extras de procesamiento.

1.2.5.1 Ventajas

La principal ventaja de los sistemas distribuidos es la capacidad de compartir y acceder a la información de una forma fiable y eficaz.

- **Utilización compartida de los datos y distribución del control**

⁵ Información obtenida en http://html.rincondelvago.com/bases-de-datos-distribuidas_1.html

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

La ventaja principal de compartir los datos por medio de la distribución es que cada localidad pueda controlar hasta cierto punto los datos almacenados localmente. En un sistema distribuido existe un administrador global de la base de datos que se encarga de todo el sistema. Parte de esta responsabilidad se delega al administrador de base de datos de cada localidad. Dependiendo del diseño del sistema distribuido, cada administrador local podrá tener un grado de autonomía diferente, que se conoce como autonomía local. La posibilidad de contar con autonomía local es en muchos casos una ventaja importante de las bases de datos distribuidas.

➤ **Fiabilidad y disponibilidad**

Si se produce un fallo en una localidad, es posible que las demás localidades puedan seguir trabajando. En particular, si los datos se repiten en varias localidades, una transacción que requiere un dato específico puede encontrarlo en más de una localidad. Así, el fallo de una localidad no implica necesariamente la desactivación del sistema.

El sistema debe detectar cuando falla una localidad y tomar las medidas necesarias para recuperarse del fallo. El sistema no debe seguir utilizando la localidad que falló. Por último, cuando se recupere o repare esta localidad, debe contarse con mecanismos para reintegrarla al sistema con el mínimo de complicaciones.

La disponibilidad es fundamental para los sistemas de bases de datos que se utilizan en aplicaciones de tiempo real.

➤ **Agilización del procesamiento de consultas**

Si una consulta comprende datos de varias localidades, puede ser posible dividir la consulta en varias subconsultas que se ejecuten en paralelo en distintas localidades. Sin embargo, en un sistema distribuido no se comparte la memoria principal, así que no todas las estrategias de intersección se pueden aplicar en estos sistemas. En los casos en que hay repetición de los datos, el sistema puede pasar la consulta a las localidades más ligeras de carga.

1.2.5.2 Desventajas.

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

La desventaja principal de los sistemas distribuidos es la mayor complejidad que se requiere para garantizar una coordinación adecuada entre localidades.

El aumento de la complejidad se refleja en:

- Coste del desarrollo de software: es más difícil estructurar un sistema de bases de datos distribuidos y por tanto su coste es menor
- Mayor posibilidad de errores: puesto que las localidades del sistema distribuido operan en paralelo, es más difícil garantizar que los algoritmos sean correctos.
- Mayor tiempo extra de procesamiento: el intercambio de mensajes y los cálculos adicionales son una forma de tiempo extra que no existe en los sistemas centralizados.

1.3 Aspectos importantes de los SDBD distribuidos⁶.

Existen varios factores relacionados a la construcción de bases de datos distribuidas que no se presentan en bases de datos centralizadas. Entre los más importantes se encuentran los siguientes:

1. Diseño de la base de datos distribuida. En el diseño de bases de datos distribuidas se debe considerar el problema de como distribuir la información entre diferentes sitios. Existen razones organizacionales las cuales determinan en gran medida lo anterior. Sin embargo, cuando se busca eficiencia en el acceso a la información, se deben abordar dos problemas relacionados. Primero, como fragmentar la información. Segundo, como asignar cada fragmento entre los diferentes sitios de la red. En el diseño de la BDD también es importante considerar si la información está replicada, es decir, si existen copias múltiples del mismo dato y, en este caso, como mantener la consistencia de la información. Finalmente, una parte importante en el diseño de una BDD se refiere al manejo del directorio. Si existen únicamente usuarios globales, se debe manejar un solo directorio global. Sin embargo, si existen también usuarios locales, el directorio combina información local con información global.

2. Procesamiento de consultas. El procesamiento de consultas es de suma importancia en bases de datos centralizadas. Sin embargo, en BDD éste adquiere una relevancia mayor. El objetivo

⁶ Información obtenida en http://www.cs.cinvestav.mx/SC/prof_personal/adiaz/Disdb/

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

es convertir transacciones de usuario en instrucciones para manipulación de datos. No obstante, el orden en que se realizan las transacciones afecta grandemente la velocidad de respuesta del sistema. Así, el procesamiento de consultas presenta un problema de optimización en el cual se determina el orden en el cual se hace la menor cantidad de operaciones. Este problema de optimización es NP-difícil, por lo que en tiempos razonables solo se pueden obtener soluciones aproximadas. En BDD se tiene que considerar el procesamiento local de una consulta junto con el costo de transmisión de información al lugar en donde se solicitó la consulta.

3. Control de concurrencia. El control de concurrencia es la actividad de coordinar accesos concurrentes a la base de datos. El control de concurrencia permite a los usuarios acceder a la base de datos en una forma multiprogramada mientras se preserva la ilusión de que cada usuario está utilizándola solo en un sistema dedicado. El control de concurrencia asegura que transacciones múltiples sometidas por usuarios diferentes no interfieran unas con otras de forma que se produzcan resultados incorrectos. En BDD el control de concurrencia es aún más complejo que en sistemas centralizados. Los algoritmos más utilizados son variaciones de aquellos usados en sistemas centralizados: candados de dos fases, ordenamiento por estampas de tiempo, ordenamiento por estampas de tiempo múltiples y control de concurrencia optimista. Un aspecto interesante del control de concurrencia es el manejo de interbloqueos. El sistema no debe permitir que dos o más transacciones se bloqueen entre ellas.

4. Confiabilidad. En cualquier sistema de bases de datos, centralizado o distribuido, se debe ofrecer garantías de que la información es confiable. Así cada consulta o actualización de la información se realiza mediante transacciones, las cuales tienen un inicio y fin. En sistemas distribuidos, el manejo de la atomicidad y durabilidad de las transacciones es aún más complejo, ya que una sola transacción puede involucrar dos o más sitios de la red. Así, el control de recuperación en sistemas distribuidos debe asegurar que el conjunto de agentes que participan en una transacción realicen todos, un compromiso (commit) al unísono o todos al mismo tiempo restablezcan la información anterior (roll-back).

En la figura 1.8 se presenta un diagrama con las relaciones entre los aspectos relevantes sobre las BDD.

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

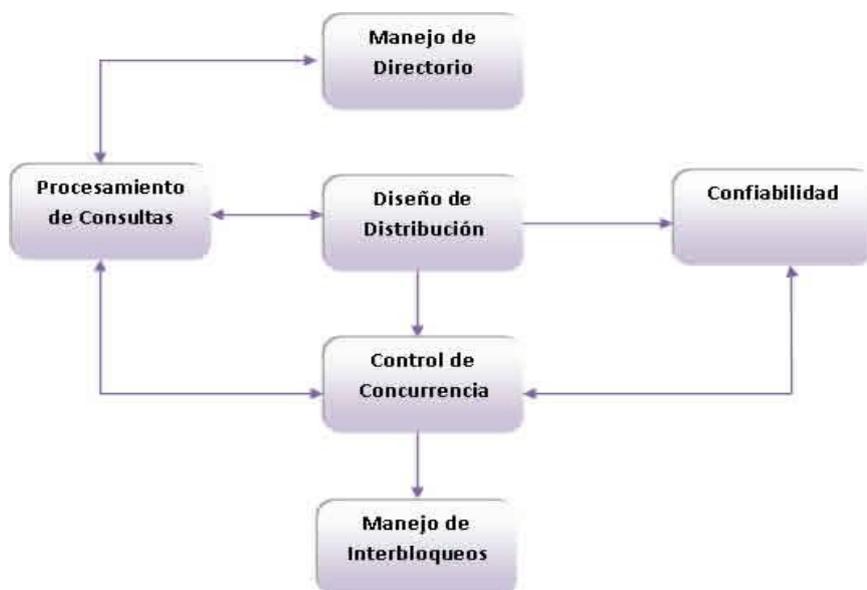


Figura 1.6. Factores importantes en BDD.

1.4 Estado del Arte⁷.

El tema de las BDD está bastante estudiado/investigado pero no existen muchos productos comerciales aún (aunque esto es sólo cuestión de tiempo).

Aun cuando los beneficios del uso de BDD son claramente perceptibles, en la actualidad muchos de los desarrollos se encuentran únicamente en sistemas experimentales (de investigación). A continuación se discute el estado actual de las bases de datos comerciales respecto de cuatro logros potenciales asequibles en BDD.

1. Manejo transparente de datos distribuidos, fragmentados y replicados. Comercialmente aún no se soporta la replicación de información. La fragmentación utilizada es únicamente de tipo horizontal. La distribución de información no se realiza aún con la transparencia requerida. Por ejemplo, el usuario debe indicar la localización de un objeto y el acceso a los datos es mediante sesiones remotas a bases de datos locales. La mayoría de los sistemas comerciales utilizan el modelo múltiples clientes-un solo servidor.

⁷ Información obtenida en http://www.cs.cinvestav.mx/SC/prof_personal/adiaz/Disdb/

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

2. Mejoramiento de la confiabilidad y disponibilidad de la información mediante transacciones distribuidas. Algunos sistemas como Ingres, NonStop SQL y Oracle V 7.x ofrecen el soporte de transacciones distribuidas. En Sybase, por ejemplo, es posible tener transacciones distribuidas pero éstas deber ser implementadas en las aplicaciones mediante primitivas dadas. Respecto del soporte para replicación de información o no se ofrece o se hace a través de la regla one-lee-todos-escriben.

3. Mejoramiento de la eficiencia. Una mayor eficiencia es una de las grandes promesas de los SMBDD. Existen varias partes en donde esto se puede lograr. En primer lugar, la ubicación de los datos a lugares próximos a donde se usan puede mejorar la eficiencia en el acceso a la información. Sin embargo, para lograrlo es necesario tener un buen soporte para fragmentación y replicación de información. Otro punto en donde se puede incrementar la eficiencia es mediante la explotación del paralelismo entre operaciones. Especialmente en el caso de varias consultas independientes, éstas se pueden procesar por sitios diferentes. Más aún, el procesamiento de una sola consulta puede involucrar varios sitios y así procesarse de manera más rápida. Sin embargo, la explotación del paralelismo requiere que se tenga tanta información requerida por cada aplicación en el sitio donde la aplicación se utiliza, lo cual conduciría a una replicación completa, esto es, tener toda la información en cada sitio de la red. El manejo de réplicas es complicado dado que las actualizaciones a este tipo de datos involucran a todos los sitios teniendo copias del dato. Los sistemas comerciales ofrecen únicamente aproximaciones a este requisito. Por ejemplo, en los bancos se destina usualmente el horario de oficina para hacer lecturas y las horas no hábiles para hacer actualizaciones. Otra estrategia es tener dos bases de datos, una para consultas y otra para actualizaciones.

4. Mejor escalabilidad de las BD. El tener sistemas escalables de manera fácil y económica se ha logrado por el desarrollo de la tecnología de microprocesadores y estaciones de trabajo. Sin embargo, respecto de la escalabilidad, la comunicación de la información tiene un costo el cual no se ha estudiado con suficiente profundidad.

1.5 Diseño de SBDD⁸.

Diseñar un Sistema de Bases de Datos Distribuido implica tomar decisiones sobre la ubicación de los programas que accederán a la base de datos y sobre los propios datos que la constituyen, a lo largo de los diferentes puestos que configuren una red de ordenadores. La ubicación de los programas no debería suponer un excesivo problema dado que se puede tener una copia de ellos en cada máquina de la red (de hecho, en este capítulo se asumirá que así es). Sin embargo, cuál es la mejor opción para colocar los datos: en una gran máquina que albergue a todos ellos, encargada de responder a todas las peticiones del resto de las estaciones – sistema de base de datos centralizado –, o podríamos pensar en repartir las relaciones, las tablas, por toda la red. En el supuesto que nos decantásemos por esta segunda opción, ¿qué criterios se deberían seguir para llevar a cabo tal distribución? En los párrafos sucesivos se tratará de responder a estas cuestiones.

Tradicionalmente se ha clasificado la organización de los sistemas de bases de datos distribuidos sobre tres dimensiones:

- el nivel de compartición,
- las características de acceso a los datos
- el nivel de conocimiento de esas características de acceso.

⁸ Información de Jorge Rodríguez Santos, Diseño de Bases de Datos Distribuidas

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

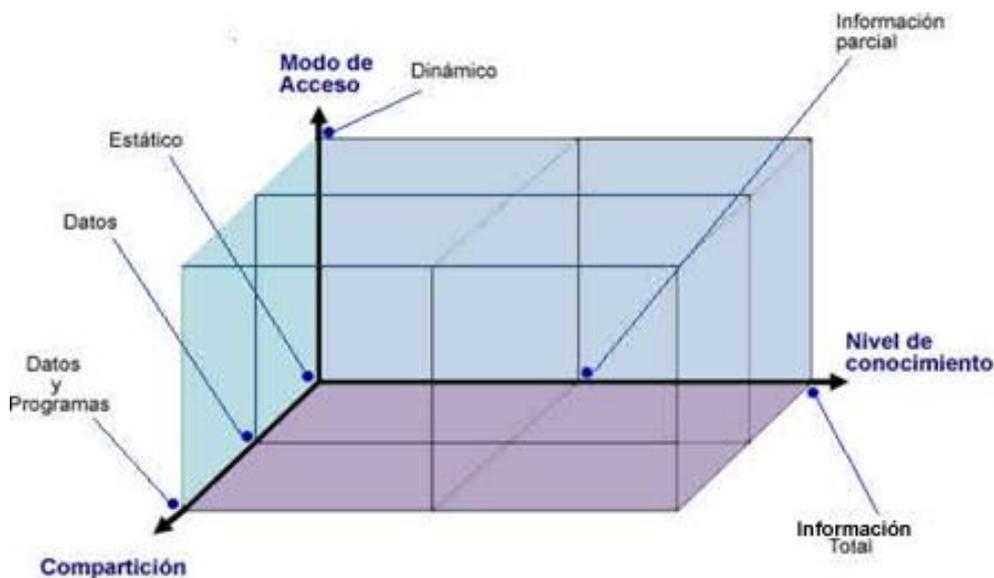


Figura 1.7. Enfoque de la distribución.

Respecto a las características de acceso a los datos existen principalmente dos posibilidades: el modo de acceso a los datos que solicitan los usuarios puede ser estático, es decir, no cambiará a lo largo del tiempo, o bien, dinámico. Se puede comprender fácilmente la dificultad de encontrar sistemas distribuidos reales que puedan clasificarse como estáticos. Sin embargo, lo realmente importante radica, estableciendo el dinamismo como base, cómo de dinámico es, cuántas variaciones sufre a lo largo del tiempo. Esta dimensión establece la relación entre el diseño de bases de datos distribuidas y el procesamiento de consultas.

El segundo aspecto está referido a la compartición del sistema, y pueden aparecer dos posibilidades, una en que se asume la compartición de los datos, y otra en que también son compartidos los programas.

La tercera clasificación es el nivel de conocimiento de las características de acceso. Una posibilidad es que los diseñadores carezcan de información alguna sobre cómo los usuarios acceden a la base de datos. Es una posibilidad teórica, pero sería muy engorroso abordar el diseño de la base de datos con tal ausencia de información. Lo más práctico sería conocer con detenimiento la forma de acceso de los usuarios o, en el caso de su imposibilidad, conformarnos con una información parcial de ésta.

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

El problema del diseño de bases de datos distribuidas podría enfocarse a través de esta trama de opciones. En todos los casos, excepto aquel en el que no existe compartición, aparecerán una serie de nuevos problemas que son irrelevantes en el caso centralizado.

A la hora de abordar el diseño de una base de datos distribuida podremos optar principalmente por dos tipos de estrategias: ascendente y descendente. Ambos tipos no son excluyentes, y no resultaría extraño a la hora de abordar un trabajo de diseño de una base de datos que se pudiesen emplear en diferentes etapas del proyecto una u otra estrategia. La estrategia ascendente podría aplicarse en el caso donde haya que proceder a diseñar a partir de un número de pequeñas bases de datos existentes, con el fin de integrarlas en una sola. En este caso se partiría de los esquemas conceptuales locales y se trabajaría para llegar a conseguir el esquema conceptual global. Aunque este caso se pueda presentar con facilidad en la vida real, se prefiere pensar en el caso donde se parte de cero y se avanza en el desarrollo del trabajo siguiendo la estrategia descendente. La estrategia descendente (Fig. 1.8) debería resultar familiar a la persona que posea conocimientos sobre el diseño de bases de datos, exceptuando la fase del diseño de la distribución.

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

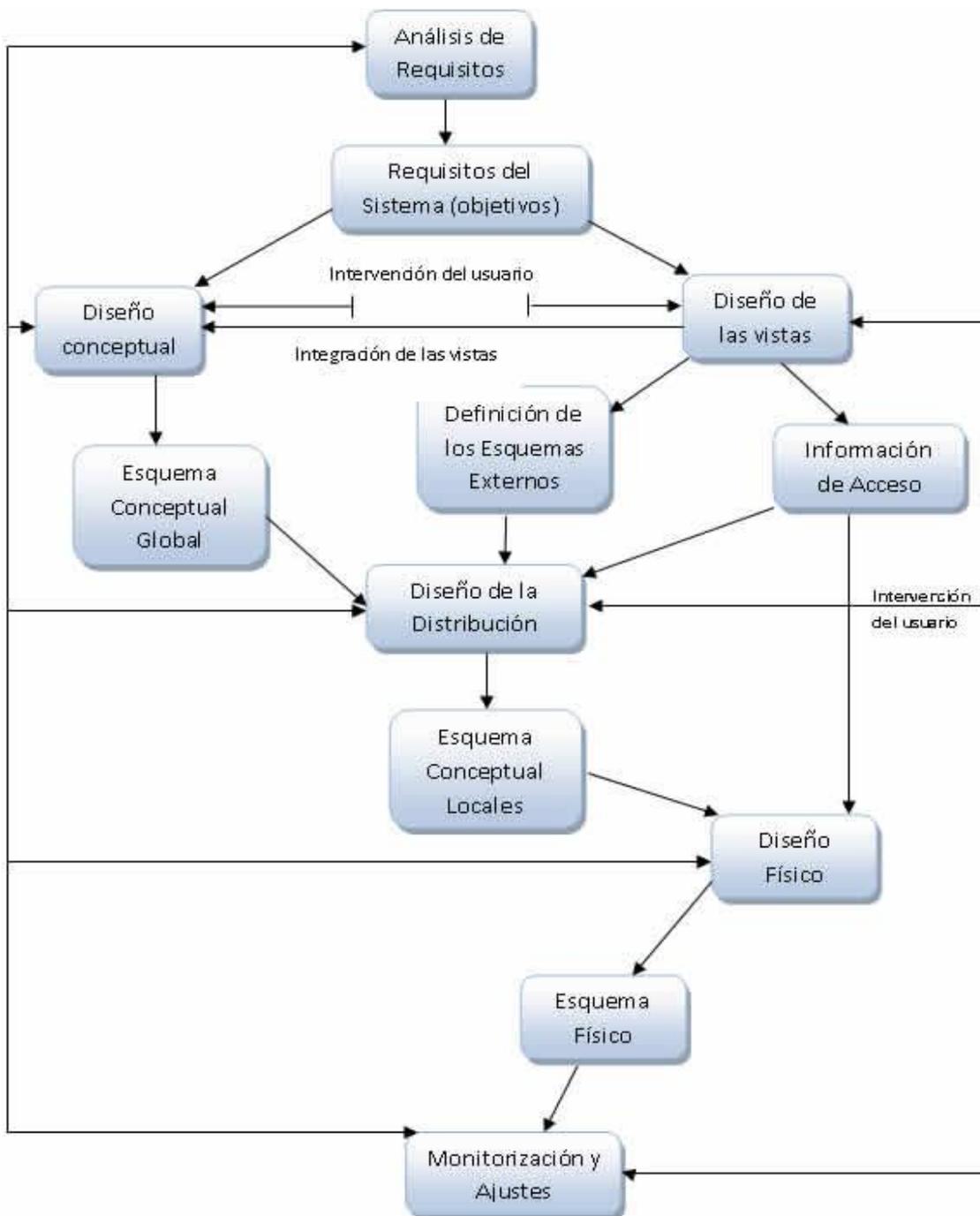


Figura 1.8. Estrategia descendente

1.6 Diseño de la distribución⁹.

Existen diversas formas de afrontar el problema del diseño de la distribución. Las más usuales se muestran en la figura 1.9.

- Caso A, los dos procesos fundamentales, la fragmentación y la asignación, se abordan de forma simultánea. Esta metodología se encuentra en desuso, sustituida por el enfoque en dos fases
- Caso B: la realización primeramente de la partición para luego asignar los fragmentos generados. El resto de los casos se comentan en la sección referente a los distintos tipos de la fragmentación.

Un aspecto importante en el diseño de la distribución es la cantidad de factores que contribuyen a un diseño óptimo. La organización lógica de la base de datos, la localización de las aplicaciones, las características de acceso de las aplicaciones a la base de datos y las características del sistema en cada sitio, tienen una decisiva influencia sobre la distribución. La información necesaria para el diseño de la distribución puede dividirse en cuatro categorías: la información del banco de datos, la información de la aplicación, la información sobre la red de ordenadores y la información sobre los ordenadores en sí. Las dos últimas son de carácter cuantitativo y servirán, principalmente, para desarrollar el proceso de asignación. Se entrará en detalle sobre la información empleada cuando se aborden los distintos algoritmos de fragmentación y asignación.

⁹ Información de Jorge Rodríguez Santos, Diseño de Bases de Datos Distribuidas

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

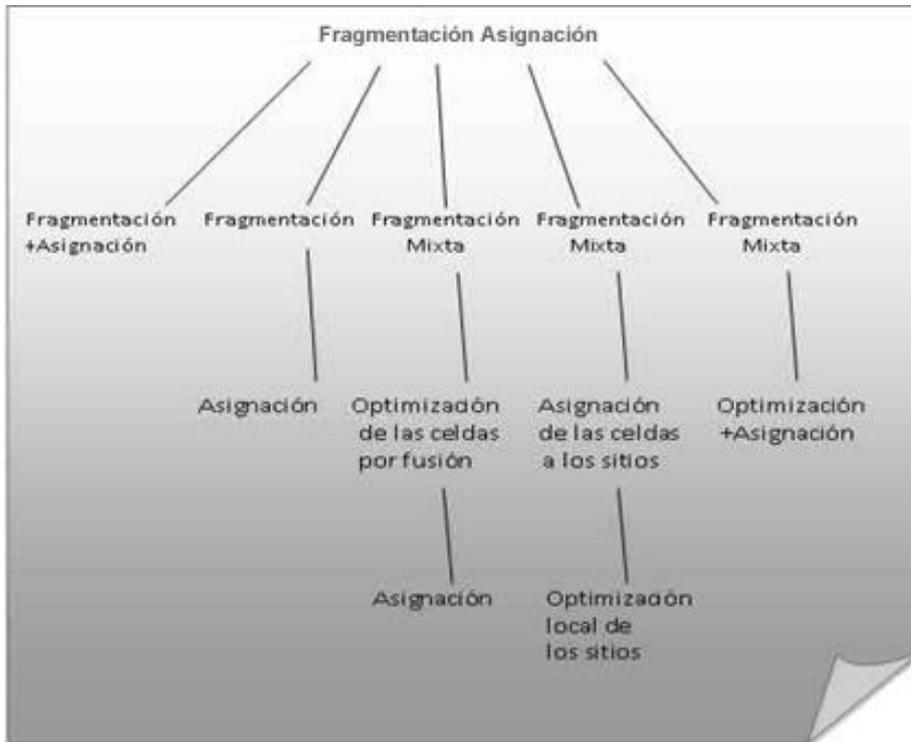


Figura 1.9. Enfoques para realizar el diseño distributivo.

1.7 Fragmentación¹⁰.

Una base de datos fragmentada es aquella donde los fragmentos se alojan en sitios donde únicamente existe una copia de cada uno de ellos a lo largo de toda la red. En caso de réplica, puede considerarse una base de datos replicada, donde existe una copia de todo el banco de datos en cada sitio, o considerar una base de datos parcialmente replicada donde existan copias de los fragmentos ubicados en diferentes sitios. El número de copias de un fragmento será una de las posibles entradas a los algoritmos de asignación, o una variable de decisión cuyo valor lo determine el algoritmo. A continuación se muestra la comparación de las tres alternativas de réplica con respecto a distintas funciones de un sistema de base de datos distribuido.

¹⁰ Información de Jorge Rodríguez Santos, Diseño de Bases de Datos Distribuidas

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

	Réplica total	Réplica parcial	Partición
Procesamiento de consultas	Fácil	Dificultad similar	Dificultad similar
Gestión del directorio	Fácil o inexistente	Dificultad similar	Dificultad similar
Control de concurrencia	Moderado	Difícil	Fácil
Seguridad	Muy alta	Alta	Baja
Realidad	Posible aplicación	Realista	Posible aplicación

El principal problema de la fragmentación radica en encontrar la unidad apropiada de distribución. Una relación no es una buena unidad por muchas razones. Primero, las vistas de la aplicación normalmente son subconjuntos de relaciones. Además, la localidad de los accesos de las aplicaciones no está definida sobre relaciones enteras pero sí sobre subconjuntos de las mismas. Por ello, sería normal considerar como unidad de distribución a estos subconjuntos de relaciones.

Segundo, si las aplicaciones tienen vistas definidas sobre una determinada relación (considerándola ahora una unidad de distribución) que reside en varios sitios de la red, se puede optar por dos alternativas. Por un lado, la relación no estará replicada y se almacena en un único sitio, o existe réplica en todos o algunos de los sitios en los cuales reside la aplicación. Las consecuencias de esta estrategia son la generación de un volumen de accesos remotos innecesario. Además, se pueden realizar réplicas innecesarias que causen problemas en la ejecución de las actualizaciones y puede no ser deseable si el espacio de almacenamiento está limitado.

Tercero, la descomposición de una relación en fragmentos, tratados cada uno de ellos como una unidad de distribución, permite el proceso concurrente de las transacciones. También la relación de estas relaciones, normalmente, provocará la ejecución paralela de una consulta al dividirla en una serie de subconsultas que operará sobre los fragmentos.

Pero la fragmentación también acarrea inconvenientes. Si las aplicaciones tienen requisitos tales que prevengan la descomposición de la relación en fragmentos mutuamente exclusivos, estas aplicaciones

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

cuyas vistas estén definidas sobre más de un fragmento pueden sufrir una degradación en el rendimiento. Por tanto, puede ser necesario recuperar los datos de dos fragmentos y llevar a cabo sobre ellos operación de unión y yunto, lo cual es costoso.

Un segundo problema se refiere al control semántico. Como resultado de la fragmentación los atributos implicados en una dependencia se descomponen en diferentes fragmentos los cuales pueden destinarse a sitios diferentes. En este caso, la sencilla tarea de verificar las dependencias puede resultar una tarea de búsqueda de los datos implicados en un gran número de sitios.

1.7.1 Tipos de Fragmentación

Dado que una relación se corresponde esencialmente con una tabla y la cuestión consiste en dividirla en fragmentos menores, inmediatamente surgen dos alternativas lógicas para llevar a cabo el proceso: la división horizontal y la división vertical. La división o fragmentación horizontal trabaja sobre las tuplas, dividiendo la relación en subrelaciones que contienen un subconjunto de las tuplas que alberga la primera. La fragmentación vertical, en cambio, se basa en los atributos de la relación para efectuar la división. Existen otras alternativas, fundamentalmente se habla de fragmentación mixta o híbrida cuando el proceso de partición hace uso de los dos tipos anteriores.

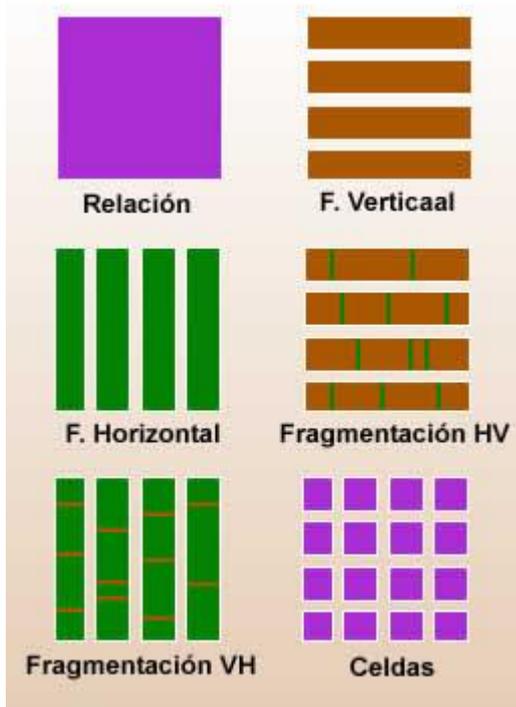


Figura 1.10. Tipos de Fragmentación

1.7.1.1 Fragmentación Horizontal

Como se ha explicado anteriormente, la fragmentación horizontal se realiza sobre las tuplas de la relación. Cada fragmento será un subconjunto de las tuplas de la relación. Existen dos variantes de la fragmentación horizontal: la primaria y la derivada. La fragmentación horizontal primaria de una relación se desarrolla empleando los predicados definidos en esa relación. Por el contrario, la fragmentación horizontal derivada consiste en dividir una relación partiendo de los predicados definidos sobre alguna otra.

1.7.1.2 Fragmentación Vertical

Recordemos que la fragmentación vertical de una relación R produce una serie de fragmentos R_1, R_2, \dots, R_r , cada uno de los cuales contiene un subconjunto de los atributos de R así como la clave primaria

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

de [R](#). El objetivo de la fragmentación vertical consiste en dividir la relación en un conjunto de relaciones más pequeñas tal que algunas de las aplicaciones de usuario sólo hagan uso de un fragmento. Sobre este marco, una fragmentación óptima es aquella que produce un esquema de división que minimiza el tiempo de ejecución de las aplicaciones que emplean esos fragmentos.

1.7.1.3 Fragmentación mixta o híbrida

En muchos casos la fragmentación vertical u horizontal del esquema de la base de datos no será suficiente para satisfacer los requisitos de las aplicaciones. Como ya se citó al comienzo de este documento podemos combinar ambas, utilizando por ello la denominada fragmentación mixta. Cuando al proceso de fragmentación vertical le sigue una horizontal, es decir, se fragmentan horizontalmente los fragmentos verticales resultantes, se habla de la fragmentación mixta HV. En el caso contrario, estaremos ante una fragmentación VH. Una característica común a ambas es la generación de árboles que representan la estructura de fragmentación.

1.7.2 Grado de Fragmentación

Cuando se va a fragmentar una base de datos deberíamos sopesar qué grado de fragmentación va a alcanzar, ya que éste será un factor que influirá notablemente en el desarrollo de la ejecución de las consultas. El grado de fragmentación puede variar desde una ausencia de la división, considerando a las relaciones unidades de fragmentación; o bien, fragmentar a un grado en el cada tupla o atributo forme un fragmento. Ante estos dos casos extremos, evidentemente se ha de buscar un compromiso intermedio, el cual debería establecerse sobre las características de las aplicaciones que hacen uso de la base de datos. Dichas características se podrán formalizar en una serie de parámetros. De acuerdo con sus valores, se podrá establecer el grado de fragmentación del banco de datos.

1.8 Transparencia y Autonomía¹¹.

¹¹ Información de Jorge Rodríguez Santos, Diseño de Bases de Datos Distribuidas.

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

Una relación puede almacenarse de varias formas en un sistema de base de datos distribuida. El sistema debe reducir al mínimo la necesidad de que el usuario sepa cómo está almacenada una relación. La transparencia de la red es el grado hasta el cual los usuarios del sistema pueden ignorar los detalles del diseño distribuido. La autonomía local es el grado hasta el cual el diseñador o administrador de una localidad pueden ser independientes del resto del sistema distribuido. Los temas de transparencia y autonomía serán considerados desde los siguientes puntos de vista:

- Nombre de los datos.
- Repetición de los datos.
- Fragmentación de los datos.
- Localización de los fragmentos y copias.

1.9 Procesamiento distribuido de consultas¹².

El objetivo del procesamiento de consultas en un ambiente distribuido es transformar una consulta sobre una base de datos distribuida en una especificación de alto nivel a una estrategia de ejecución eficiente expresada en un lenguaje de bajo nivel sobre bases de datos locales.

Así, el problema de optimización de consultas es minimizar una función de costo tal que

Función de costo total = Costo de Input/Output + Costo de CPU + Costo de comunicación

1.10 Manejo de Transacciones

En caso de ocurrir una falla de una consulta, que puede ser de lectura o actualización, a veces no se puede simplemente reactivar la ejecución de una consulta, puesto que algunos datos pueden haber sido modificados antes la falla. El no tomar en cuenta esos factores puede conducir a que la información en la base de datos contenga datos incorrectos.

¹² Información obtenida en http://www.cs.cinvestav.mx/SC/prof_personal/adiaz/Disdb/

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

Los sistemas distribuidos son muy confiables debido a la posibilidad de brindar redundancia y autonomía de recursos en diferentes nodos, esto posibilita detectar y localizar fallas, sin embargo tenemos varios aspectos que representan problemas para la integridad de los recursos y que a su vez motivan el uso de transacciones

- Dificultad para mantener consistencia en los datos
- Requerimientos de procesamientos en paralelo
- Manejo interactivo de uno o más usuarios

Las transacciones son mecanismos que ayudan a simplificar la construcción de sistemas confiables mediante procesos que proporcionan soporte para invocar y sincronizar operaciones como:

- Operaciones de comparación de datos
- Aseguramiento de la seriabilidad de las transacciones con otras
- Atomicidad en su comportamiento
- Recuperación de fallas

1.10.1 Condiciones de terminación de una transacción¹³

Una transacción siempre termina, aun en la presencia de fallas. Si una transacción termina de manera exitosa se dice que la transacción hace un compromiso. Si la transacción se detiene sin terminar su tarea, se dice que la transacción aborta. Cuando la transacción es abortada, puede ser por distintas razones relacionadas con la naturaleza de la transacción misma, o por conflicto con otras transacciones o por fallo de un proceso o computador, entonces su ejecución es detenida y todas las acciones ejecutadas hasta el momento son deshechas regresando a la base de datos al estado antes de su ejecución. A esta operación también se la conoce como rollback.

¹³ Información obtenida en

<http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/MonogSO/TRANS02.htm>

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

1.11 Control de Concurrencia¹⁴.

El control de concurrencia trata sobre los problemas de acceso concurrente de las transacciones a los datos. El control de concurrencia distribuido en sistema de manejo de bases de datos distribuidas (DDBMS) asegura que la consistencia de la base de datos se mantiene, en un ambiente distribuido multiusuario.

Si no se lleva a cabo un adecuado control de concurrencia, se podrían llegar a presentar dos anomalías. En primer lugar, se pueden perder actualizaciones provocando que los efectos de algunas transacciones no se reflejen en la base de datos. En segundo lugar, pueden presentarse recuperaciones de información inconsistentes.

Resumiendo, se encuentran los siguientes problemas de concurrencia específicos de las bases de datos distribuidas:

- Consistencia de copia múltiple (se produce cuando un mismo dato está en varias localizaciones).
- Además se siguen dando los mismos problemas para bases de datos centralizadas (pérdida de actualizaciones, dependencia neutral (uncommitted dependency) y análisis inconsistentes).

La finalidad del control de concurrencia es asegurar la consistencia de los datos al ejecutar transacciones, y que cada acción atómica sea completada en un tiempo finito.

1.12 Manejo de Bloqueos¹⁵.

La prevención de bloqueos puede hacer que algunas transacciones esperen y otras retrocedan. En todos estos esquemas se requiere que cada localidad mantenga un grafo de espera local. Los nodos del grafo corresponden a todas las transacciones (locales o no) que en un momento dado utilizan o solicitan algunos de los datos que pertenecen a esa localidad.

¹⁴ Información obtenida en

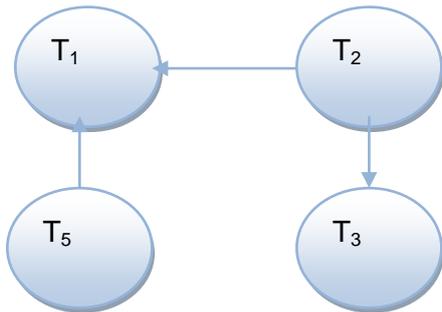
http://asignaturas.inf.udec.cl/~basedato/trabajos/estructura_debasededatosdistribuidas.doc

¹⁵ Información obtenida en

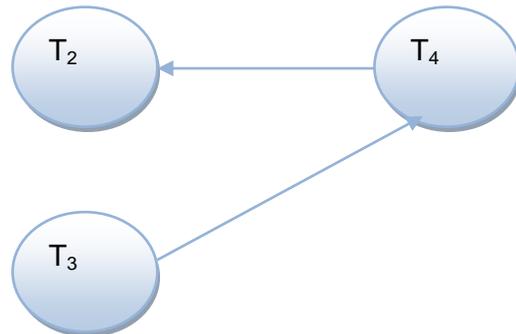
http://asignaturas.inf.udec.cl/~basedato/trabajos/estructura_debasededatosdistribuidas.doc

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

Observemos que las transacciones T1 y T2 que aparecen en ambos grafos, lo que indica que estas transacciones han solicitado datos de ambas localidades.



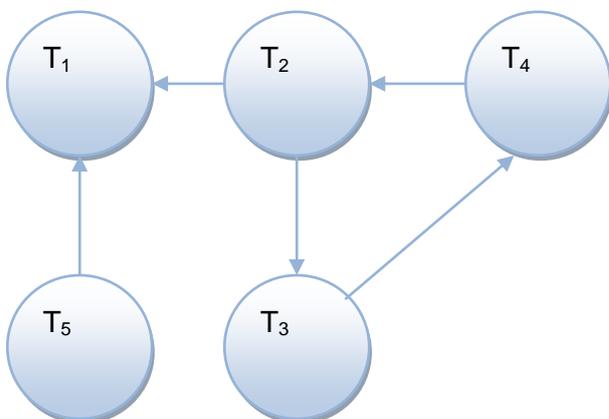
Localidad 1



Localidad 2

Grafos de espera local

Naturalmente, si cualquier grafo de espera local contiene un ciclo, se habrá presentado un bloqueo. Sin embargo, el hecho de que ninguno de los grafos locales de espera contenga un ciclo no significa que no existan bloqueos. Para ilustrar este problema, consideramos los grafos de espera locales de la figura anterior. Ambos grafos son acíclicos; no obstante, existe un bloqueo en el sistema. El bloqueo existe porque la unión de los grafos locales de espera contiene un ciclo. Este grafo se muestra en la figura siguiente.



Grafo de espera global para la figura anterior

Algunos esquemas de uso común para organizar el grafo de espera en un sistema distribuido son el *enfoque centralizado* y el *enfoque de distribución total*, existen otros, pero nos abocaremos a describir éstos.

1.13 Confiabilidad¹⁶.

Un sistema de manejo de bases de datos confiable es aquel que puede continuar procesando las solicitudes de usuario aún cuando el sistema sobre el que opera no es confiable. En otras palabras, aun cuando los componentes de un sistema distribuido fallen, un DDMBS confiable debe seguir ejecutando las solicitudes de usuario sin violar la consistencia de la base de datos.

1.13.1 Tipos de fallas en SMBDD

➤ **Fallas de transacciones.** Las fallas en transacciones se pueden deber a un error debido a datos de entrada incorrectos así como a la detección de un interbloqueo. La forma usual de enfrentar las fallas en transacciones es abortarlas.

¹⁶ Información obtenida en http://www.cs.cinvestav.mx/SC/prof_personal/adiaz/Disdb/

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

- **Fallas del sistema.** En un sistema distribuido se pueden presentar fallas en el procesador, la memoria principal o la fuente de energía de un nodo. En este tipo de fallas se asume que el contenido de la memoria principal se pierde, pero el contenido del almacenamiento secundario es seguro.
- **Fallas del medio de almacenamiento.** Se refieren a las fallas que se pueden presentar en los dispositivos de almacenamiento secundario que almacenan las bases de datos. Esas fallas se pueden presentar por errores del sistema operativo, por errores del controlador del disco, o del disco mismo.
- **Fallas de comunicación.** Las fallas de comunicación en un sistema distribuido son frecuentes. Estas se pueden manifestar como pérdida de mensajes lo que lleva en un caso extremo a dividir la red en varias subredes separadas.

1.14 Selección del Coordinador¹⁷.

Algunos algoritmos requieren de un coordinador. Si el coordinador falla al quedar fuera de servicio, la localidad donde reside, el sistema podrá continuar con la ejecución solo si se activa un coordinador en alguna otra localidad. Esto puede hacerse si se mantiene una copia de seguridad del coordinador que este preparado para asumir sus responsabilidades si el coordinador de otra localidad falla. Otra técnica consiste en elegir el nuevo coordinador después de que haya fallado el coordinador original. Los algoritmos que determinan el sitio donde debe activarse una nueva copia del coordinador se denominan algoritmos de elección.

1.15 Conclusiones

En este capítulo se ha realizado un estudio y observación de las características que poseen los Sistemas de Bases de Datos Distribuidas:

Las BDD han permitido que los SGBD manejen una cantidad mayor de información ya que permiten aunar los datos de diferentes fuentes (integrar sistemas de diferentes computadoras y/o redes posiblemente en diferentes localizaciones geográficas).

¹⁷ Información obtenida en

http://asignaturas.inf.udec.cl/~basedato/trabajos/estructura_debasededatosdistribuidas.doc

CAPÍTULO 1 SISTEMAS DE BASES DE DATOS DISTRIBUIDAS.

Propician el aumento de rendimiento de los SGBD al poderse dividir las consultas y transacciones, lo cual permite su procesamiento en diferentes maquinas.

Existe el problema de que el diseño de un SGBDD es costoso y su diseño y administración complejos.

El tema de las BDD está bastante estudiado/investigado pero no existen muchos productos comerciales aún

En un futuro cercano, es previsible que los SGBDD se extiendan hasta alcanzar un amplio sector del mercado, sobre todo en las grandes multinacionales debido a que les permite acceder remotamente a los datos de todas sus filiales.

Por esto, en las PYMES no es de esperar que ocurra un fenómeno similar.

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

CAPÍTULO 2: SISTEMAS DE BASES DE DATOS REPLICADAS.

2.1. Introducción

En este capítulo, se hace una exposición general sobre los aspectos referidos a los Sistemas de Bases de Datos Replicados (SBDR, SMDB Sistemas de Memoria Compartida Distribuida, por sus siglas en inglés), pensando desde el inicio en la manera de aplicar estos elementos al modelado de lo que será la base de datos del MAC, por lo que se pasa a definir las categorías presentes en muestra investigación, definiendo el concepto de base de datos replicada y otros elementos que son considerados. Además, se intentará dar respuesta a la inquietud expresada en el capítulo anterior, en que el equipo se preguntaba si existirían alternativas al diseño de SBDD.

2.2. Conceptos

Una réplica se define como una copia de una base de datos que es integrante de un conjunto de réplicas y que se puede sincronizar con otras réplicas del conjunto. Los cambios realizados en los datos de una tabla replicada se envían y aplican a las demás réplicas. (Diaz)

La réplica de la base de datos es la creación y el mantenimiento de copias múltiples de la misma base de datos. Un servidor de la base de datos mantiene la copia principal de la base de datos y los servidores adicionales de la base de datos mantienen las copias auxiliares de la base de datos. La base de datos escribe, se envía al servidor principal de la base de datos y después es replegada por los servidores auxiliares de la base de datos. Esta se divide entre todos los servidores de la base de datos, que da lugar a una ventaja grande del funcionamiento debido a compartir carga. (Date, 2003)

Una réplica es una copia de la base de datos, contiene todos los documentos y elementos de diseño de la base de datos (tales como los formularios y los agentes). (Soledad Aravena, 2003)

La réplica de datos es una técnica que permite copiar y distribuir idénticamente las tablas de una base de datos en múltiples bases de datos ubicadas en diferentes nodos de la red. La replicación asegura que los datos correctos estén siempre disponibles en el momento y en el lugar necesario. (Vega, 2007)

La réplica en la base de datos consiste en el transporte de datos entre dos o más servidores,

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

permitiendo que ciertos datos de la base de datos estén almacenados en más de un sitio, y así aumentar la disponibilidad de los datos y mejorar el funcionamiento de las consultas globales a las bases de datos. (Gabriela Basconcel, 2002)

La replicación es el acto o resultado de reproducir una copia. Es el acto de reproducir, de modo que es más que una simple copia, **es un servicio de administración de copias**. Desde el punto de vista técnico, replicación **es el mantenimiento automático de réplicas de tablas relacionadas en un ambiente distribuido**.

La replicación de datos es mucho más que simples copias entre datos almacenados, está involucrado análisis, diseño, implementación, administración y monitoreo de un servicio que garantice la consistencia de datos a través de la administración de múltiples recursos en un ambiente distribuido.

Un servicio de replicación de datos debe proveer las siguientes funcionalidades:

- Ser escalable para replicar volúmenes grandes o pequeños de datos
- Proveer servicios de mapeo y transformación de datos: esto es copias idénticas o semánticamente equivalentes.
- Soportar los mecanismos de replicación sincrónica y asincrónica
- Proveer mecanismos de descripción de datos y objetos a ser replicados
- Proveer mecanismos de suscripción de datos y objetos a ser replicados
- Proveer mecanismos de seguridad
- Proveer mecanismos de recuperación automática

Es una técnica para la mejora de los servicios que ofrecen los sistemas distribuidos porque proporciona una mejora del rendimiento de los servicios e incrementa su disponibilidad y lo hace tolerante a fallos. (Gabriela Basconcel, 2002)

Una **Base de Datos Replicada** (BDR) es un conjunto de múltiples bases de datos *lógicamente relacionadas* que se encuentran replicadas en diferentes sitios interconectados por una red de comunicaciones.

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

Un **Sistema de Bases de Datos Replicados** (SBDR) es un conjunto de bases de datos están ligados por un sistema de comunicaciones, en el que tributan su información a un sitio central a través de los mecanismos de réplica.

Un **Sistema de Manejo de Bases de Datos** (SMBD) es aquel que se encarga del manejo de la BDR y proporciona un mecanismo de acceso que hace que la replicación sea transparente a los usuarios. El término transparente significa que la aplicación trabajaría, desde un punto de vista lógico, como si un solo SMBR ejecutado en una sola máquina, administrara esos datos.

El modelo de replicación está formado por: publicador, distribuidor, suscriptor, publicación, artículo y suscripción; y varios agentes responsabilizados de copiar los datos entre el publicador y el suscriptor. Estos agentes son los procesos responsabilizados de copiar los datos entre el publicador y el suscriptor. Estos agentes son: agente de instantáneas, agente de distribución, agente del lector del registro, agente del lector de cola y agente de mezcla. Posee las siguientes características:

- *Efectividad*: depende de la forma en la que los datos sean distribuidos y almacenados. A mayor efectividad, mayor será la disponibilidad de datos para ejecutar procesos paralelos.
- *Alta Disponibilidad*: es la razón de tiempo prudente en la que un servicio puede ser accedido. En el mejor de los casos puede ser de un 100%, a pesar de los fallos que se puedan presentar en el servidor, ya que debe existir un servidor adicional que posea alguna técnica de replicación que lo pueda suplantar en caso de ser necesario.
- *Tolerancia a fallos*: garantiza un comportamiento correcto, donde efectivamente pueden existir un número finito de fallos y tipos de fallos.
- *Coordinación*: cada una de las partes que conforman la base de datos distribuida, acuerda un consenso para realizar las invocaciones de los servicios a los objetos, que al final de la transacción debe realizarse tal y como fue solicitada, para lo cual debe utilizar algún tipo de ordenamiento. (Vega, 2007)

2.3. Objetivos

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

Su objetivo es que las operaciones de los clientes sobre las réplicas se realicen de forma consistente y con un tiempo de respuesta y un caudal satisfactorio. La consistencia no se logra algunas veces debido a que las réplicas de un cierto objeto no son necesariamente idénticas, al menos no en cada instante de tiempo particular. Algunas réplicas pueden haber recibido actualizaciones que otras no hayan recibido. (Gabriela Basconcel, 2002)

2.4. Ventajas y desventajas

Las principales ventajas de los SBDR consiste en:

➤ **Disponibilidad**

Ya que con un buen diseño de la red, puede caer una localidad, y la información seguir disponible en otras; y no perderse esta, ni tiempo de trabajo, ya que se interactuará con otros servidores, que después de restablecido el servicio, replican al que perdió la disponibilidad

➤ **Paralelismo Incrementado**

Lo que se explica por el hecho de que

Según Elías (Elías, 2003), las ventajas están referidas a la alta disponibilidad, la mejora del desempeño, la tolerancia a fallas, la esclabilidad, y la autonomía.

Mientras que la principal desventaja de este tipo de sistemas se basa en:

Sobrecarga incrementada en la actualización:

Pues durante el proceso de réplica de las actualizaciones después de cada transacción a las locaciones que almacenan los mismos datos que la que sobre se trabajó, es generalmente alta.

2.5. ¿Para qué usar replicación?¹⁸

Las instituciones utilizan la replicación en sus aplicaciones por varias razones, las cuáles pueden ser categorizadas de la siguiente forma:

¹⁸ Información obtenida en

<http://www.fing.edu.uy/inco/cursos/tagSI/Trabajos/2003/TAGSI03-TareaTecnSI-grupo4.pdf>

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

- Distribución de datos a otras ubicaciones
- Consolidación de datos desde otras ubicaciones
- Intercambio bidireccional de datos con otras ubicaciones
- Alguna variante o combinación de los anteriores

2.5.1. Distribución de datos a otras ubicaciones

La distribución de datos involucra el movimiento de todos o un subconjunto de datos de una o más ubicaciones.

La distribución es utilizada para proveer datos a aplicaciones desarrolladas en plataformas similares o diferentes. Esto puede ser tan simple como mantener una copia de los datos de producción en otro sistema similar, así como requerir una transformación compleja para adaptarse a los requerimientos de una nueva aplicación. Los datos que se copien pueden necesitar ser filtrados o transformados para la nueva aplicación. La distribución también puede ser usada para proveer coexistencia de dos sistemas, en la instancia de migración de uno al otro.

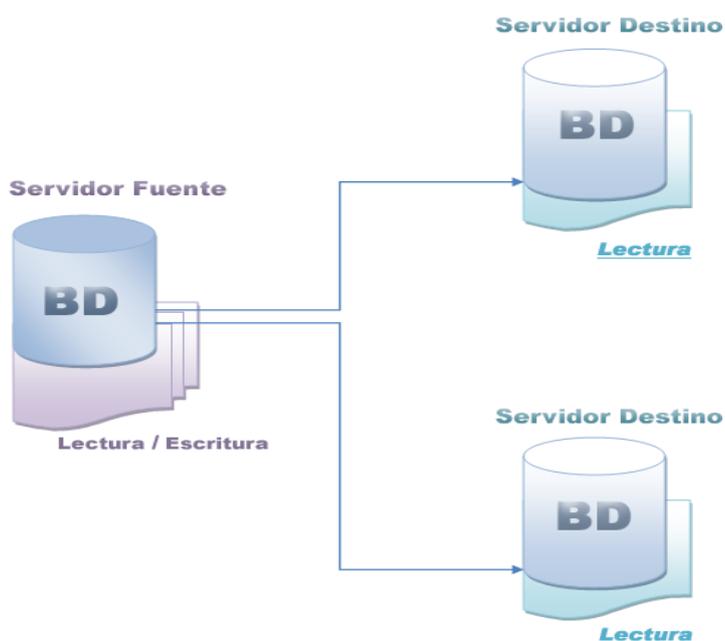


Figura 2.1. Replicación desde un servidor fuente hacia dos servidores destinos.

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

Los servidores destino podrían mantener un subconjunto diferente de los datos de acuerdo a las necesidades propias.

2.5.2. Consolidación de datos desde otras ubicaciones

Una institución puede tener sus datos en distintos sistemas distribuidos. A modo de ejemplo, una empresa de ventas puede tener sus datos en cada local, una empresa de manufactura los puede tener en cada planta de procesamiento, así como una compañía aseguradora podría tenerlos en cada una de sus oficinas o llegar incluso a cada computador personal de sus corredores. La replicación puede copiar los cambios de cada sitio distribuido al sitio central, para analizar, hacer reportes, o para el procesamiento central de los datos.

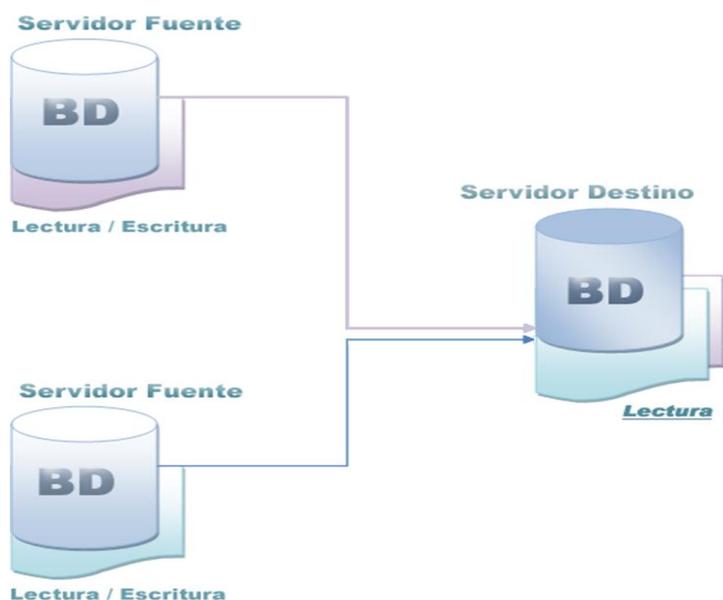


Figura 2.2 Replicación desde dos servidores fuentes, hacia un servidor destino.

La estructura de los datos en cada fuente es la misma. Los datos del destino son la unión de los datos de las distintas fuentes.

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

2.5.3. Intercambio bidireccional de datos con otras ubicaciones

Si los datos se pueden modificar en múltiples ubicaciones, entonces la replicación debe procesar los cambios realizados en cada uno de los sitios de forma coordinada. Uno de los servidores, es visto como el servidor maestro, quien se encarga de distribuir los cambios a todos los sitios. Los cambios realizados en los destinos fluyen hacia los otros sitios a través del servidor maestro.

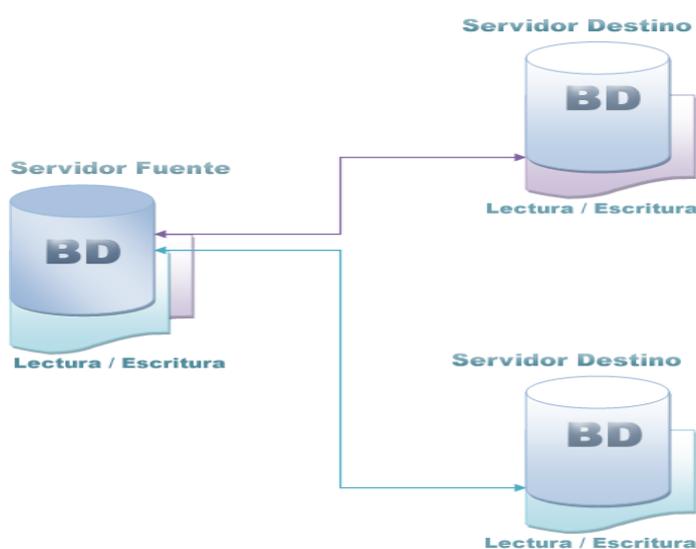


Figura 2.3. Réplica bidireccional, en la que tenemos un servidor maestro designado.

Otro tipo de replicación bidireccional, es aquella que no tiene designada un servidor maestro.

Cada ubicación copia los cambios desde todos los otros sitios directamente. Esto habitualmente sucede en los entornos “multi-maestro” o “par-a-par”.

2.5.4. Otros requerimientos

Algunas aplicaciones podrían no requerir una copia completa de los datos pero si necesitar un registro de todos los cambios que han ocurrido. Esto es útil para mantener una auditoría del sistema y también puede ser útil para proveer los cambios para procesos especiales. Por ejemplo, los cambios realizados un sistema fuente pueden ser almacenados en una tabla intermedia, donde se analicen y validen, antes de procesarlos y enviarlos a los sitios destino.

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

Esta es una variante de la técnica de distribución de datos, en la cual el destino sólo recibe los cambios.

Los cambios también pueden ser utilizados en escenarios de replicación multi-capas, donde son copiados desde una fuente central hacia un área de almacenamiento en otro sistema y finalmente desde esta área a los sitios destino. Esto es muy útil en caso de existir muchos sitios destino, ya que minimiza el impacto sobre el sistema fuente.

2.6. Entornos de réplica

Los tipos de entornos de réplica son los siguientes:

- *Maestro-Esclavo (master-slave)*: o de solo lectura, permite a un solo maestro recibir consultas de lectura/escritura, mientras los esclavos solo pueden aceptar consultas de lectura.
- *Multi-Maestro (multi-master)*: también llamada par-a-par o la réplica de camino de n, permite múltiples sitios, actuando como pares iguales. Cada sitio en un ambiente de réplica de multimaestro es un sitio de maestro, y cada sitio se comunica con otros sitios maestros. Esta capacidad tiene también un severo impacto en el desempeño debido a la necesidad de sincronizar los cambios entre los servidores.

Este tipo de entorno puede ser usado para mantener sitios recuperables ante posibles desastres o caídas, así como para proveer sistemas con alta disponibilidad y para balancear la carga de consultas a través de las distintas ubicaciones. (Vega, 2007)

2.7. Clasificación de los algoritmos de replicación¹⁹.

En el mundo se ha abordado la replicación sobre determinadas tendencias, condicionadas por el ámbito en que se usará la solución, lo que ha originado gran cantidad de clasificaciones y algoritmos. A continuación serán descritos algunos de los más importantes.

¹⁹ Información obtenida en <http://personals.ac.upc.edu/leandro/pubs/tesis-jacosta.pdf>

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

2.7.1. Modelo funcional de la replicación

Un protocolo de replicación puede ser descrito usando cinco fases genéricas. Sin embargo con algunas técnicas de replicación se pueden saltar fases o alterar el orden. En estas fases se omiten detalles, por ejemplo algunos investigadores interponen entre el cliente y las réplicas un front end o middleware.

Las fases son:

1. Petición
2. Coordinación
3. Ejecución
4. Acuerdo
5. Respuesta

2.7.2. Replicación en Bases de Datos

Se han clasificado los protocolos de replicación usando dos parámetros, uno es cuando se hace la propagación (eager vs lazy), y el otro es quien realiza la actualización (primary vs everywhere), y se clasifica como sigue:

- Replicación eager
- Replicación lazy
- Primaria (Primary o master)
- Grupo (everywhere) (Elías, 2003)

2.7.2.1. Comunicación en Grupo

La comunicación por multidifusión se entiende como comunicación en grupo ya que los grupos de procesos son los destinatarios de los mensajes de multidifusión. Los grupos sirven para manejar datos

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

replicados y sistemas donde los procesos operan en conjunto para recibir y procesar el mismo conjunto de mensajes de multidifusión.

La pertenencia a los grupos estaba definida de manera estática a pesar de que los miembros del grupo fallen. No obstante, en la práctica los sistemas requieren pertenencia dinámica, esto significa que los procesos se unen y dejan el grupo según el sistema trabaja. Este servicio de pertenencia al grupo y la comunicación por multidifusión se incluyen en una implementación completa de la comunicación en grupo.

La gestión de la comunicación por multidifusión y la de pertenencia a grupos están fuertemente relacionadas

El **papel del servicio de pertenencia** a grupos tiene cuatro tareas principales:

- Proporcionar una interfaz para los cambios en la pertenencia a grupos
- Implementar un detector de fallos
- Notificar a los miembros del grupo los cambios en la pertenencia
- Expandir las direcciones del grupo

La comunicación en grupos tiene en cuenta el estudio de:

- Entrega de Vistas
- Comunicación en grupos con vistas Síncronas

2.8. Modelos de Consistencia²⁰

El modelo de consistencia es el encargado de que los datos que se encuentran distribuidos sean similares, en el sentido de que cada tupla los mismos valores en cada nodo que se encuentre, y estén las mismas tuplas por tablas similares que existan en la red. Estos dos elementos garantizan la consistencia de los datos, que entre otras técnicas, se basa en la Linealidad y la Consistencia Secuencial.

²⁰ Información obtenida en

<http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/MonogSO/REPLIC02.html>

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

2.8.1. Linealidad y consistencia secuencial

Los datos compartidos replicados se dicen linealizables si para cualquier ejecución existe algún entrelazamiento de las series de operaciones emprendidas por cada cliente que satisface los siguientes criterios:

- La secuencia entrelazada de operaciones cumple la especificación de una única copia correcta de los objetos.
- El orden de las operaciones del entrelazamiento es consistente con los tiempos reales en los cuales ocurrieron las operaciones en la ejecución real.

Los datos replicados compartidos son secuencialmente consistentes si para cualquier ejecución existe algún entrelazamiento de las series de operaciones realizadas por los clientes que satisfacen los dos siguientes criterios:

- La secuencia de operaciones intercaladas cumple la especificación de una única copia correcta de los objetos.
- El orden de las operaciones en el entrelazamiento es consistente con el orden en el programa con el cual cada cliente individual ejecutó dichas operaciones.

2.9. Modelos de Distribución

Existen dos modelos de distribución de datos esencialmente aplicados a cada uno de los entornos antes vistos:

- *Asincrónica*: a menudo llamada almacena-y-reenvía, captura cualquier cambio local, los almacena en una cola, y, a intervalos regulares, propaga y aplica estos cambios en sitios remotos. Con esta forma de réplica, hay un período de tiempo antes de que todos los sitios alcancen la convergencia de datos.
- *Sincrónica*: también conocida como la réplica en tiempo real, aplica cualquier cambio o ejecuta cualquier procedimiento reproducido en todos los sitios que participan en el ambiente de réplica como parte de una sola transacción. Si el procedimiento falla en cualquier sitio, entonces la transacción

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

entera se anula. La réplica sincrónica asegura la consistencia de datos en todos los sitios en tiempo real. (Vega, 2007)

2.10. ¿Cómo se implementa la replicación?

Generalmente se clasifica a la replicación por el grado de latencia, esto es, cuanto tiempo están desactualizados los datos replicados respecto de los originales. Por otro lado, hay dos tipos de replicación:

- Replicación sincrónica
- Replicación asincrónica

2.10.1. Replicación sincrónica

Se sincroniza la copia con el original con tiempo de latencia cero, se la denomina también de consistencia hermética. Cuando se usa replicación sincrónica debe tenerse en cuenta:

- Asegurar que el diseño de transacción refleje la unidad fundamental de recuperación
- Asegurar una infraestructura robusta y eficiente con un ambiente de red con alto ancho de banda y posibilidad de ruteo.
- Definir en forma “razonable” la longitud de la transacción de modo de monopolizar un conjunto de recursos “razonable”.

2.10.2. Replicación asincrónica.

No existe una sincronización entre copia y original sino que transcurre un tiempo (latencia) medido en segundos en que hay inconsistencia entre ambas.

La replicación asincrónica puede implementarse de varias maneras:

- **Refresco completo:** se extraen los datos de la fuente primaria y se propagan eliminando del destino lo que hubiera antes.

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

- **Refresco incremental:** se extraen los datos de la fuente primaria tomando solo los cambios ocurridos desde la última extracción.
- **Propagación delta (en triángulo) de eventos:** implica un procesamiento en tiempo real y puede darse de base de datos a base de datos o de proceso a proceso.

Cuando se usa propagación delta de eventos para replicación asincrónica, los destinos deben primero ser inicializados para ser llevados a un estado consistente con las fuentes de datos primarias. Luego, solamente los eventos que ocurren en la fuente de datos primaria son llevados a los destinos replicados. Cada evento preserva esta identidad discreta durante el proceso de replicación.

2.10.2.1. Replicación asincrónica entre base de datos

Este tipo de replicación puede hacerse a través de tres componentes separadas y distintas: colección de datos de la fuente primaria, distribución y aplicación del proceso en los destinos designados, se puede incorporar una cuarta de monitoreo.

Esta técnica suele llevar consigo una mejora de la disponibilidad, pero presenta a menudo inconvenientes en cuanto a las prestaciones que se obtienen del sistema. Sin embargo, por su simplicidad (relativamente, ya que requiere la implementación de un control de concurrencia adecuado) es bastante usada en los sistemas comerciales.

2.10.2.2. Replicación asincrónica entre procesos

Se basa en general en el paradigma de mensajes publicadores y suscriptores. El proceso en si se separa en cuatro componentes: proceso de publicación, proceso de distribución (store and forward), proceso de suscripción y proceso de aplicación.

Las técnicas de replicación de información suelen ofrecer un mejor comportamiento en cuanto a prestaciones, sin perjuicio de la disponibilidad. Como contrapartida, al poner en práctica estas técnicas aparece invariablemente la necesidad de mantener consistencia de la información que se replica, lo que en muchos casos obliga a establecer un compromiso entre prestaciones (escalabilidad), Disponibilidad y Consistencia. (Briz)

2.11. Técnicas de Replicación²¹.

En esta parte hay que mencionar que se encuentran diversas técnicas de replicación de la información, que además, están indisolublemente ligadas a los entornos que se mencionaron anteriormente.

2.11.1. Sincrónica

2.11.1.1. Confirmación en dos Fases(Two Phases Commit)

Esta tecnología surge a mediados de los años 80. Permite la sincronización de datos distribuidos. Cada transacción solamente es aceptada si todos los sistemas implicados en la réplica están conectados y listos para recibirla, si al menos uno falla, todo el proceso es anulado, como se muestra en la Fig. 4.

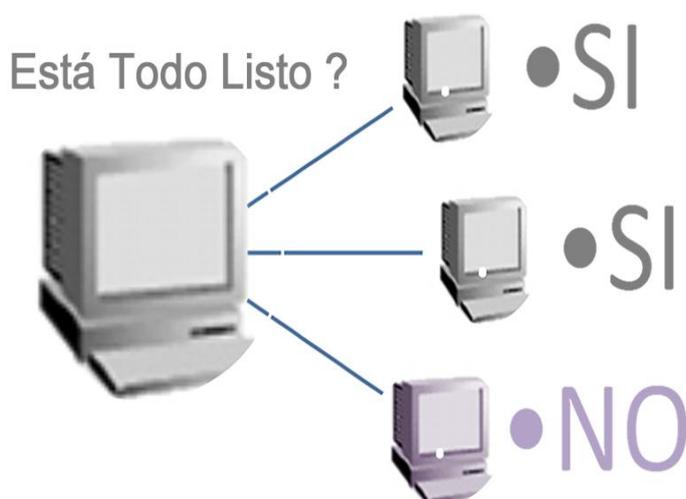


Fig. 2.4. Confirmación en dos fases.

2.11.2. Asíncrona

2.11.2.1. Descarga y Recarga (Dump and Reload)

²¹ Información obtenida en http://www.sybase.com/content/1028711/6143_whitepaper_v2.pdf

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

Consiste en hacer un volcado de los datos, copiar la salva para un dispositivo de almacenamiento para luego distribuir la salva por los demás servidores. Esta técnica presenta el inconveniente de que en la mayoría de las ocasiones se consultaban datos que tenían semanas de desactualización, además de que el proceso se realizaba de forma manual.

2.11.2.2. Instantánea (Snapshot)

Consiste en hacer una instantánea de una tabla en un momento dado y esta “foto” es la que se replica en las demás bases de datos. Aunque mucho más avanzado que la técnica de Descarga y Recarga esta presenta el inconveniente de que las tablas esclavas son tablas de solo lectura.

Se recomienda utilizar cuando: la mayoría de los datos no cambian con frecuencia; se replican pequeñas cantidades de datos; los sitios con frecuencia están desconectados y es aceptable un periodo de latencia largo (la cantidad de tiempo que transcurre entre la actualización de los datos en un sitio y en otro).

2.11.2.3. Disparadores(Triggers)

La función del disparador es ejecutar una acción cuando ocurre un evento en la base de datos, los eventos pueden ser de inserción, actualización o eliminación.

Conjuntamente con las instantáneas, los disparadores son otro de los mecanismos asincrónicos que proporciona la base de datos como una manera de replicación de datos.

Para mayor comodidad serán llamados triggers, ya que es el término por el cuál son conocidos.

Todo lo descrito en los epígrafes 2.5 y 2.6 se puede resumir en la figura:

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

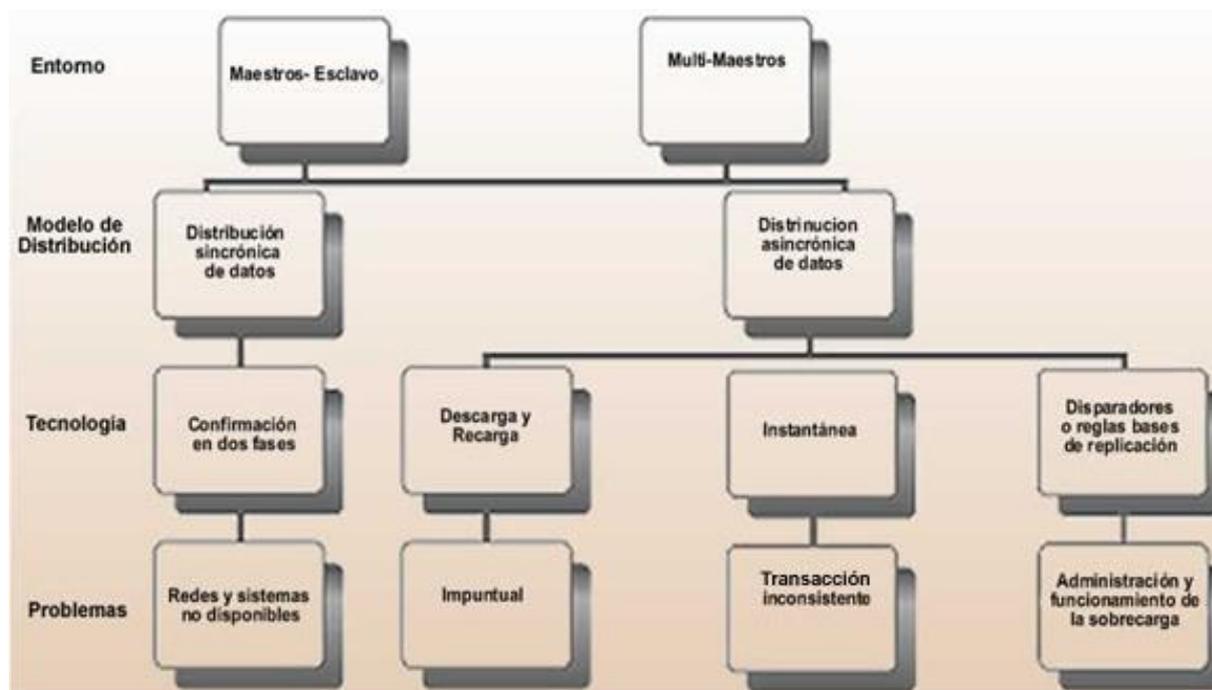


Figura 2.5 Esquema de Réplica de Datos

2.12. Servicios de Alta Disponibilidad²².

En este tipo de sistemas se busca un nivel de servicio aceptable utilizando un conjunto mínimo de gestores de réplicas conectados al cliente. Y debería minimizar el tiempo que el cliente espera hasta que los gestores de réplicas procesen una respuesta.

A continuación se detallan tres sistemas (Cotilleo, Bayou y Coda) que proporcionan servicios con alta disponibilidad.

2.12.1. Arquitectura Cotilla

²² Información obtenida en

<http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/MonogSO/REPLIC02.html>

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

En ésta arquitectura los clientes hacen las solicitudes a su frontal, cada uno de estos se comunica con un gestor de réplicas (no fijo), y estos, a su vez, se comunican entre si mediante mensajes "**cotillas**".

Soporta tres esquemas de ordenación:

- **Causal:** esta se implementa para proporcionar la consistencia relajada, esto significa que todos los gestores de réplicas en un momento reciben todas las actualizaciones y las aplican de forma ordenada para hacer a las réplicas lo suficientemente parecidas, para cumplir con las necesidades de la aplicación.
- **Forzado:** aporta garantías de ordenación más fuertes.
- **Inmediato:** las actualizaciones inmediatamente ordenadas se aplican en un orden consistente en relación a otras actualizaciones en todos los gestores de réplicas.

Proporciona dos tipos de operaciones:

- **Consulta:** son operaciones de solo lectura.
- **Actualización:** solo modifica y no lee el estado.

En un modelo básico de replicación las operaciones de pregunta y ordenación en un sistema de cotillas se ordenan como sigue:

- **Petición:** el frontal envía sus peticiones a un solo gestor de réplicas normalmente, si este falla o se vuelve inaccesible el frontal se comunicara con otro gestor de réplicas.
- **Respuesta a una actualización:** si la petición es una actualización el gestor responde tan pronto como la recibe.
- **Coordinación:** el gestor que recibe la petición no la procesa hasta que no coordine con las restricciones de ordenación que se requieran.
- **Ejecución:** el gestor de réplicas ejecuta la petición.
- **Respuesta:** si la petición es una pregunta, el gestor de réplicas responde en el momento.
- **Acuerdo:** los gestores de réplicas se actualizan entre ellos mediante mensajes de cotilleo que contiene las más recientes actualizaciones.

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

2.12.2. Arquitectura de Bayou

Este sistema proporciona replicación de datos con alta disponibilidad, con garantías más débiles que la arquitectura de cotilla. Al final cada gestor de réplicas recibe el mismo conjunto de actualizaciones y aplica estas de modo que las bases de datos de los gestores de réplicas sean iguales.

El sistema Bayou se diferencia de los otros en que la replicación no es transparente para la aplicación. Una de las desventajas es el aumento de la complejidad para el programador, otra es el aumento de la complejidad para el usuario.

2.12.3. Arquitectura Coda

Este sistema de archivos proporciona alta disponibilidad a pesar de que se realizan operaciones sin conexión. El sistema Coda intenta cumplir algunos requisitos para gestionar los datos con disponibilidad constante para que los usuarios se beneficien de un repositor de archivos compartidos y a la vez que puedan depender totalmente de los recursos locales cuando este repositor este inaccesible.

El diseño de Coda se basa en la replicación de volúmenes de archivos para conseguir una mayor capacidad de procesamiento de las operaciones de acceso a archivos y un mayor grado de tolerancia a fallos. Además, se basa en la ampliación de un mecanismo para el mantenimiento en caché de copias de los archivos dentro de las computadoras de los clientes, para permitirles operar cuando no están conectados a la red.

2.13. Recuperación en protocolos de consistencia²³.

Las acciones atómicas permiten recuperación ante fallos consiguiendo consistencia de la información. Estos fallos pueden clasificarse en tres grupos:

- **Fallos de Transacción:** Son abortos explícitos. La información volátil está disponible.

²³ Información obtenida en <http://gsyc.escet.urjc.es/moodle/file.php/33/Teoria/T6-ConsistenciaDatos.pdf>

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

- **Fallos del sistema** paran el procesador y borran la memoria volátil.
- **Fallos de disco:** Son enmascarados con memoria estable.

Existen varios métodos de recuperación, dentro de los que sobresalen:

Deshacer\Rehacer

Usando el log

Haciendo guarda de estado

2.14. Conflictos de la replicación de datos²⁴.

Existen varios tipos de conflictos a tener en cuenta, al tratar la replicación de datos. Los conflictos pueden ocurrir cuando estamos trabajando en un ambiente de replicación que permite actualizaciones concurrentes sobre los mismos datos en múltiples sitios.

2.14.1. Conflicto de Actualización

Un conflicto de actualización ocurre cuando se produce la replicación de una actualización (*update*) sobre un registro con otra actualización (*update*) sobre el mismo registro. Este conflicto ocurre cuando dos transacciones originadas desde distintos sitios actualizan el mismo registro, en forma cercana en el tiempo.

Resolución

- **Prioridad:** Cada servidor obtiene una prioridad única, y el servidor de mayor prioridad “gana”, respecto de aquellos con prioridad menor.
- **Timestamp:** La más nueva o la más antigua de las modificaciones es la considerada correcta, y por defecto, sino se eligió ninguno de los criterios “gana” la más nueva.

²⁴ Información obtenida en

<http://www.fing.edu.uy/inco/cursos/tagsi/Trabajos/2003/TAGSI03-TareaTecnSI-grupo4.pdf>

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

- Particionamiento de datos: Se garantiza que cada registro sea manipulado por un único servidor, lo que simplifica la arquitectura.

2.14.2. Conflicto de unicidad

El conflicto de unicidad sucede cuando la replicación de un registro intenta violar una restricción de integridad, ya sea por llave primaria o única (*Primary Key* o *Unique*). Por ejemplo considere lo que sucede, cuando dos transacciones originadas de dos sitios diferentes, cada una inserta un registro, en su respectiva tabla replicada, con el mismo valor de clave primaria. En ese caso sucede un conflicto de unicidad.

Resolución

- Para cada servidor brindar un rango distinto de números para los generadores de clave (secuencias).
- Agregar el identificador del servidor a la clave primaria.
- Replicar en tablas separadas, y acceder a los datos a través de una vista formada por la unión de ellas. Para resolver el conflicto de potenciales claves duplicadas en la unión se usará una pseudo columna que representa la base de datos fuente.

2.14.3. Conflicto de supresión

Un conflicto de supresión ocurre cuando dos transacciones originadas de sitios diferentes, una de ellas intenta borrar un registro, y la otra actualizar o borrar el mismo registro, ya que en este caso el registro no existe, tanto para ser actualizado como borrado.

Resolución

Para evitar este tipo de conflictos, una posible solución es que los sitios marquen lógicamente los registros a ser borrados y que periódicamente el sitio maestro corra un proceso que realice el borrado (“delete”) físico de los datos, es decir desde los sitios replicados no se puede ejecutar una sentencia para hacer el borrado de los datos (delete).

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

2.14.4. Conflicto de orden

Los conflictos de orden pueden ocurrir en ambientes de replicación con tres o más sitios maestros. Si la propagación al sitio maestro X, está bloqueada por alguna razón, entonces la replicación de modificaciones en datos puede seguir siendo propagada a través de los otros sitios maestros; al finalizar la propagación estas modificaciones debieron ser propagadas al sitio X en un orden diferente a como ocurrieron en los otros sitios maestros, pudiendo producirse un conflicto.

Resolución

Este tipo de conflictos suelen resolverse asignándole distintas prioridades a los sitios maestros, de forma de ordenar las transacciones de acuerdo a esta. (Vega, 2007)

2.15. Estado del arte y análisis de productos existentes

Este método es uno de los más sencillos que se puede usar para lograr el objetivo de tener información almacenada en una red, y así lo evidencia en hecho de que ha sido usado exitosamente en distintas partes del mundo, dentro de las que se incluye nuestro país, donde son pioneros en su uso, como respuesta a sus propias necesidades y recursos, causantes de que se generen sus propias soluciones.

Empresas y organismos como la Empresa de Telecomunicaciones de Cuba (ETECSA), la empresa porcina, el Ministerio de las Fuerzas Armadas Revolucionarias se han beneficiado de esta tecnología.

Una solución que se usa en Venezuela es la de Servicios Autónomos de Registros y Notarias (SAREN). También, a nivel internacional existen empresas que se dedican a brindar soluciones, que han sido diseminadas a lo largo del mundo, como la argentina IPCorpSoftware, que entre sus servicios oferta Bases de datos con Replicación usando Slony I, además de servicio de Tuneo.

2.15.1. Solución a los Servicios Autónomos de Registros y Notarias (SAREN).

2.15.1.1. Escenario

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

Solución sobre ambiente de Almacenamiento Distribuido de Datos, cuenta con:

- Un servidor central para su Centro de Datos sobre el gestor Oracle DataBase Enterprise Edition 10g y la plataforma RetHat Enterprise Edition Linux Advanced Server (RHEELAS) v 4.0.
- Numerosos servidores de datos locales, sobre el gestor Oracle DataBase 10g Standard Edition One, establecidos en oficinas clientes, sobre la plataforma Windows Small Business Server 2003, y poseen conexión completa con el servidor de datos central.
- Se usa la replicación bidireccional
- Existen dos tipos de oficinas locales: oficinas mercantiles y oficinas públicas, por lo que la estructura de las BD locales no será idéntica en todas.
- La estructura del servidor central contiene a todas las anteriores, con particionamiento horizontal y vertical.

2.15.1.2. Modelo de Replicación

En este entorno, pueden ser necesarias las réplicas que se mencionan a continuación:

- Replicar hacia todas las BD locales la información del Centro de Datos
- Replicar hacia las BD locales solo la información correspondiente.
- Replicar desde una oficina información hacia el Centro de Datos
- Replicar desde una oficina información hacia el Centro de Datos, que deba actualizarse en todos los servidores locales restantes.
- Replicar desde una oficina información hacia el Centro de Datos, la cual debe actualizarse en todos los servidores locales restantes. Replicar hacia todas las oficinas información agregada en el Centro de Datos
- Replicar en ambos sentidos.

2.15.1.3. Solución

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

En este caso se dio respuesta a las necesidades del cliente con un mecanismo que se encarga de sincronizar los datos entre los servidores y que se compone de dos procesos principales:

- Captura o detección de cambios; y
- Aplicación de los cambios en el destino para lograr la sincronización

2.15.1.4. Alcance

Éste mecanismo de replicación es independiente del gestor de bases de Datos que se use. Debido a la simplicidad de sus procesos, para adaptarlo a cualquier otro gestor solo es necesario modificar la forma en que se crean los objetos de la BD (tablas de control, disparadores y procedimientos almacenados) y modificar la cadena de conexión al servidor. Esta solución soporta la replicación entre servidores con cualquier gestor, e incluso entre servidores con gestores distintos.

2.15.2. Ministerio de las Fuerzas Armadas²⁵.

2.15.2.1. Escenario

En este caso, se tienen las siguientes características:

Un servidor central a nivel ministerial

Tres servidores a nivel regional que conectan, igualmente, las regiones militares con el servidor nacional

Servidores a nivel de región militar

El gestor de bases de datos en cada nivel es PostgreSQL

Teniendo en cuenta la compartimentación que se necesita en los datos de este ministerio, dado el carácter jerárquico

²⁵ Información obtenida en la tesis “Sistema de réplica para bases de datos distribuidas en PostgreSQL”.

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

2.15.2.2. Modelo de Replicación

En este ministerio, se hacen las réplicas que se mencionan a continuación:

Replicación en 2 sentidos entre el servidor nacional y los regionales.

Replicación en 2 sentidos entre los servidores regionales y los de las regiones militares bajo su jerarquía.

Lo expuesto en los dos epígrafes anteriores, se puede resumir en la figura:

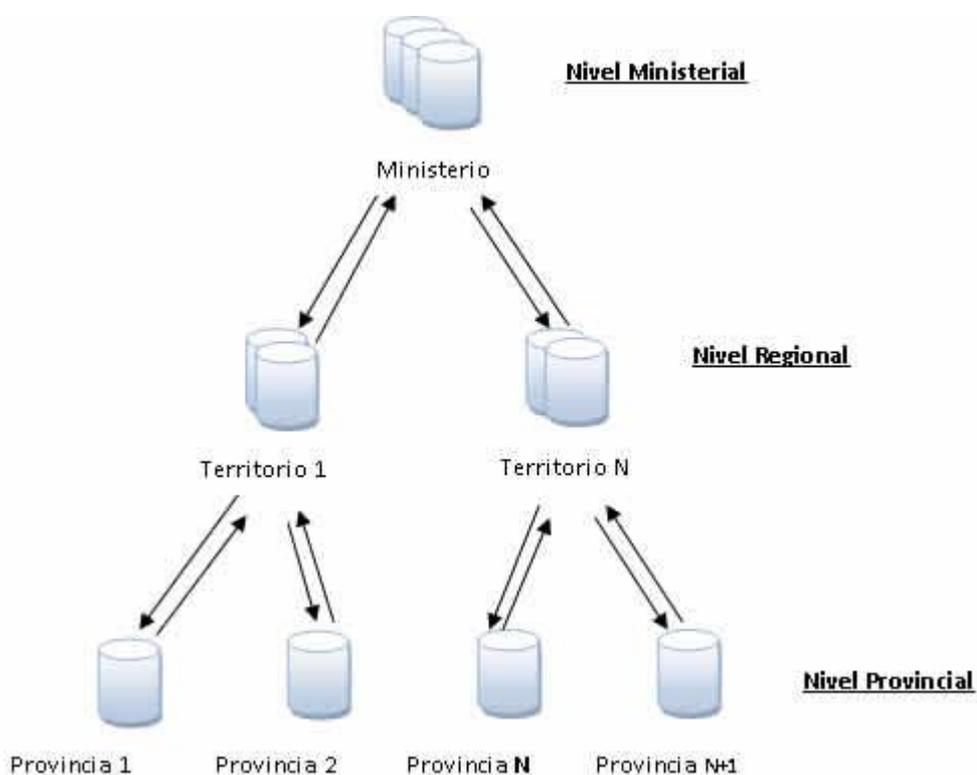


Figura 2.6. Estructura jerárquica de la información en el MAC

2.15.2.3. Solución

La solución que se le dio al problema de informatizar el MINFAR, con el Slony I como herramienta para lograr la replicación de los datos, se muestra a continuación.

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

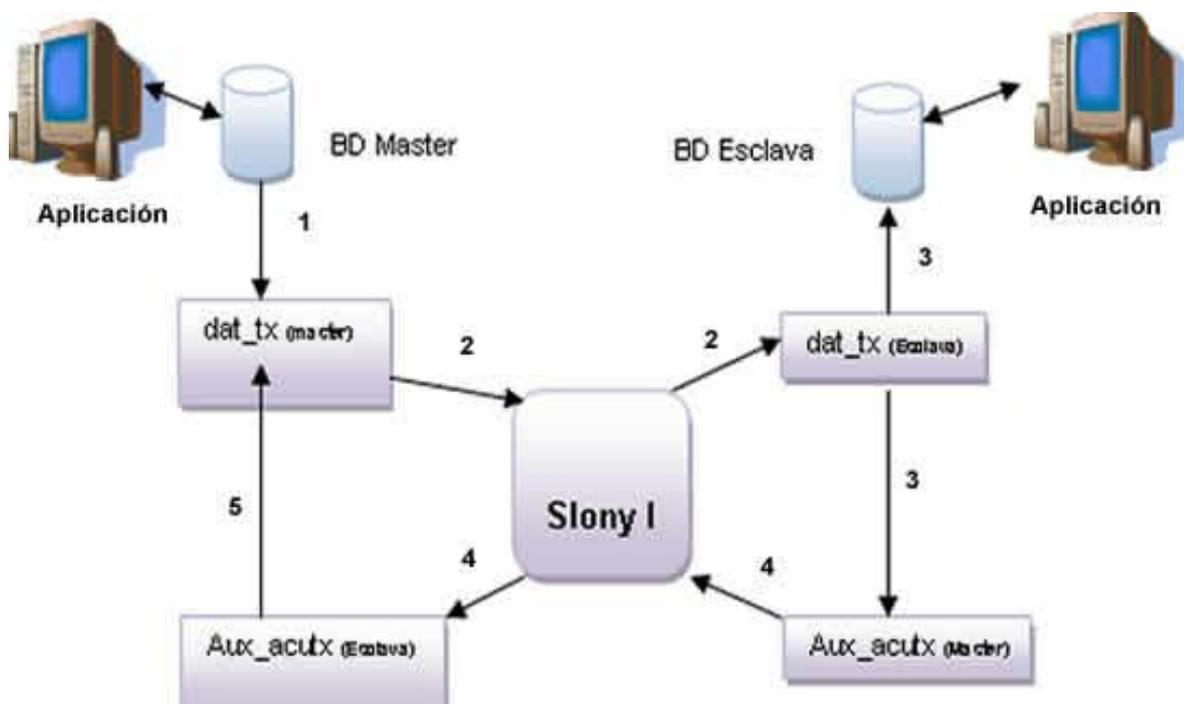


Figura 2.7 Esquema general de replicación de BD para base de datos fragmentadas

Flujo 1

- Será un trigger escrito en pl/pgsql que se dispare siempre luego de uno de los eventos de inserción, borrado o actualización (INSERT, DELETE o UPDATE), reconstruirá la sentencia ejecutada y la insertará en la entidad `dat_tx (master)`.
- Según las reglas definidas decidirá si la sentencia SQL es o no propagada al nodo esclavo.

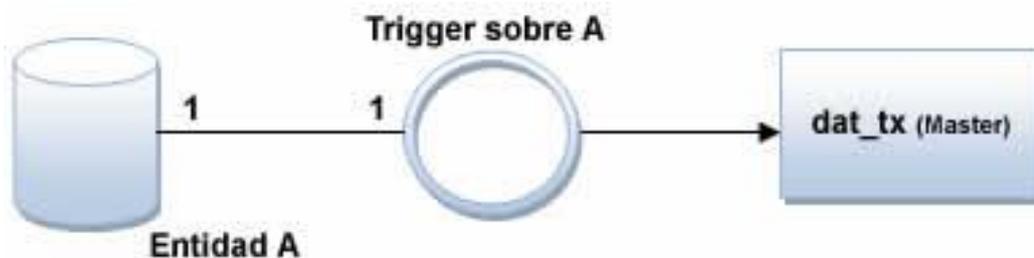


Figura 2.8. Descripción del flujo 1 de la propuesta

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

Flujo 2 Descripción del Flujo 1

Slony I con su capacidad para crear espejos replicará cada tupla que pongamos en dat_tx (master) en dat_tx (esclava o espejo). Aún cuando se haya perdido la conexión, éste en cuanto se restablezca realizará la actualización de las mismas.



Figura 2.9. Descripción del flujo 2 de la propuesta

Flujo 3

Será un trigger que se dispare siempre luego de una inserción (after INSERT) sobre la entidad dat_tx (esclava). Su cuerpo estará implementado en pl/pgsql, posibilitando así la conexión a la “BD esclava” y la ejecución de las instrucciones sql recibidas. De esta última haberse ejecutado satisfactoriamente será insertado su identificador (llave) en la entidad dat_acutx (master).

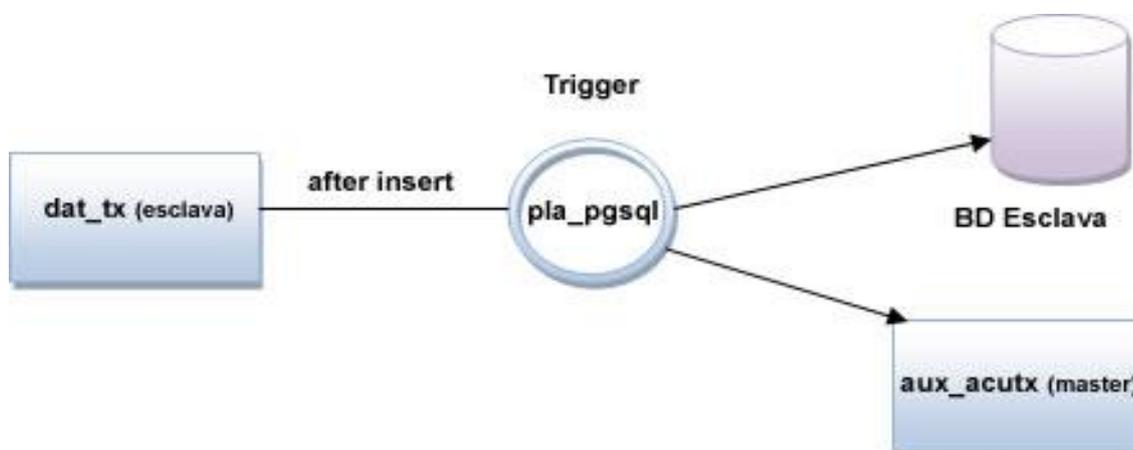


Figura 2.10. Descripción del flujo 3 de la propuesta

Flujo 4

Slony I replicará cada tupla que pongamos en dat_acutx (master) en dat_acutx (esclava o espejo). Esta es la propuesta para mantener la entidad dat_tx (master) vacía, evitando así la lentitud por sobrecarga del sistema.

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.



Figura 2.11. Descripción del flujo 4 de la propuesta

Flujo 5

Será un trigger (pl/pgsql) que se dispare siempre luego de una inserción (after INSERT) sobre la entidad aux_acutx (esclava), este tendrá la responsabilidad de ejecutar el borrado de todas las tuplas que estén en dat_tx (master) en la medida en que lleguen a él sus identificadores (así mismo se vacía dat_tx (esclava)).

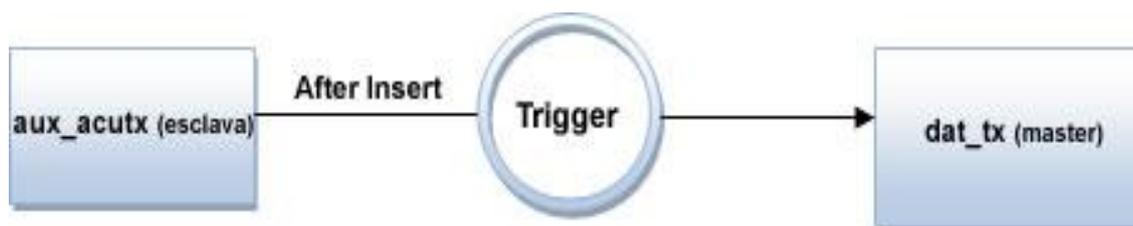


Figura 2.12. Descripción del flujo 5 de la propuesta

2.15.2.4. Alcance

Esta solución se encuentra en uso en el MINFAR hace poco más de un año y no se han reportado fallas, lo que evidencia la robustez y la fiabilidad que la caracterizan.

2.16. Conclusiones

En este capítulo se hizo patente que las Bases de Datos Replicadas cuentan con ventajas que las convierten en una alternativa interesante a la hora de enfrentar la solución de problemas en que se necesiten ambientes distribuidos de datos.

En este capítulo se ha realizado un estudio y observación de las características que poseen los Sistemas de Bases de Datos Distribuidas:

CAPÍTULO 2. SISTEMAS DE BASES DE DATOS REPLICADAS.

Las BDR hacen que los SGBD manejen una cantidad mayor de información ya que deben responder por la consistencia de todos los datos que se encuentran dispersos en la red (integrar sistemas de diferentes computadoras y/o redes posiblemente en diferentes localizaciones geográficas).

El diseño de un SGBDR es menos costoso y su diseño y administración más simples que el del SGBDD.

El tema de las BDR ha sido estudiado/investigado profundamente, y la presencia de soluciones usando sus fundamentos, es bastante elevada.

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

CAPÍTULO 3: COMPARACIONES Y ELECCION DE ESTRATEGIA DE ALMACENAMIENTO.

3.1 Introducción.

En este capítulo, el equipo realizan las comparaciones pertinentes, en el proceso que le permitirá al equipo decidirse por alguna variante entre la gama que ofrecen los ambientes de almacenamiento distribuido de datos, teniendo en cuenta las características y necesidades del MAC; así como enunciar características de herramientas de replicación existentes, específicamente el Slony I.

3.2 Comparación entre variantes del Almacenamiento Distribuido de Datos

Estos métodos persiguen un objetivo común, y es el de manejar datos en ambientes distribuidos, y las principales características se encuentran reflejadas en la tabla que se muestra a continuación:

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

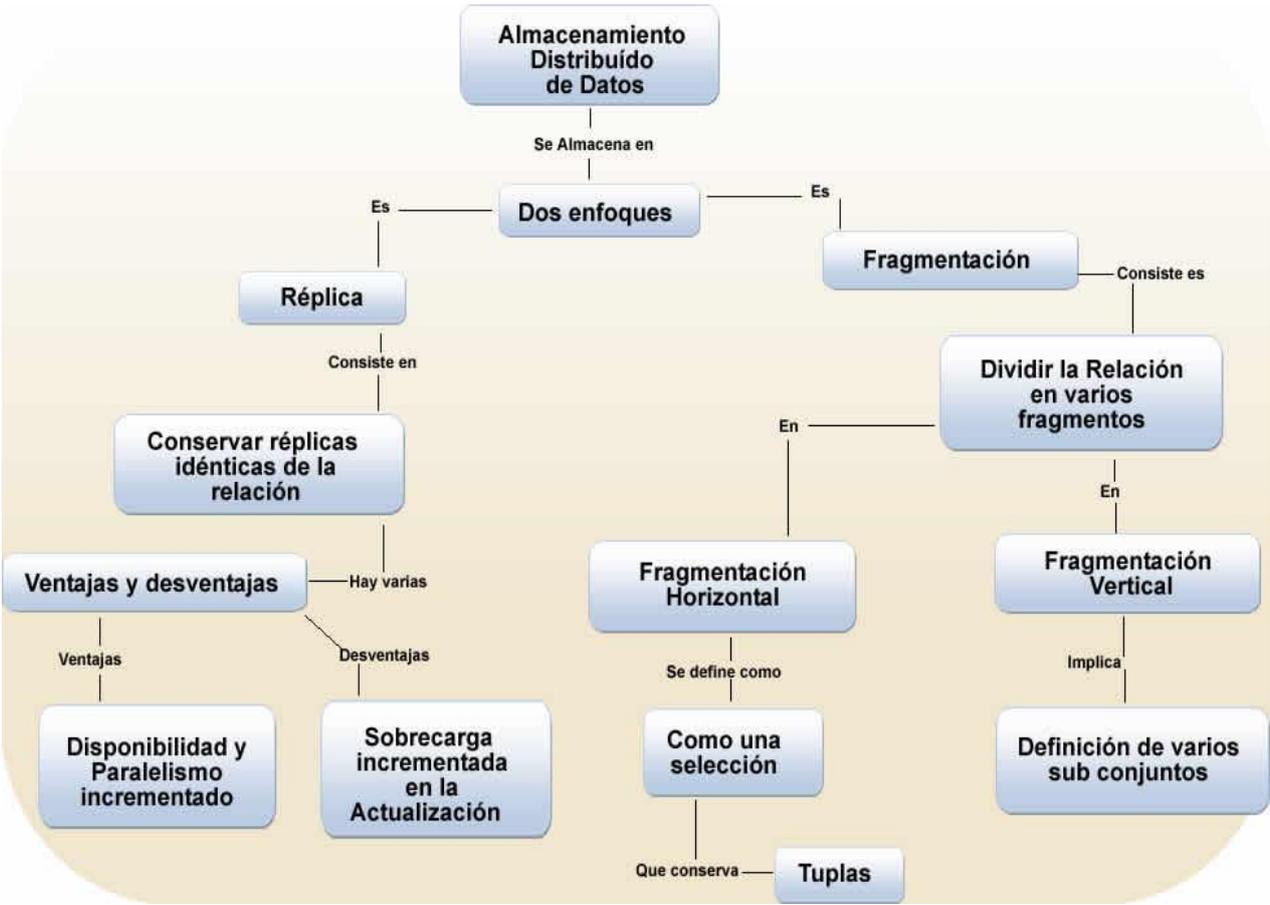


Figura 3.1. Almacenamiento Distribuidos de Datos

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

3.2.1 Tipología de la red.

En este caso, ambas soluciones coinciden totalmente, pudiendo usar ambas cualquier tipología de red, ya sea totalmente conectada, prácticamente conectada, estructura de árbol, de estrella, o de anillo.

3.2.1.1. Velocidad de la Red

Si la tipología de red no es relevante en la comparación entre ambientes de almacenamiento distribuido de datos, la velocidad de la red o ancho de banda que presenten las mismas si pueden ser un punto en que aparezcan divergencias.

En los SBDD es imprescindible el uso de vías de comunicación que den cobertura a las necesidades constantes de tráfico de información. Téngase en cuenta que cada vez que se desee realizar una consulta a un dato que esté fragmentado hay que consultar al menos dos localidades para obtener el dato íntegro. Esto enfoca la alta dependencia que tiene este tipo de sistema de la velocidad de la red.

En los SBDR, se hace un uso menos intensivo de la red comparado con el SBDD, considerando que en el caso de la sincrónica se usa después de cada transacción, y en la asincrónica, cada intervalos definidos por el administrador.

3.2.2. Ventajas y desventajas

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

En este aspecto, el SBDR tiene mayores méritos, ya que además de poseer todas las ventajas del SBDD, es mucho más fácil su implementación, partiendo del punto en que los protocolos que usa el SBDD son más fáciles de usar cuando se hacen dentro de una estructura jerárquica, como permite hacer el SBDR.

En cuanto a las desventajas, los SBDD son mucho más complejos que los SBDR, y aunque los primeros no tienen la sobrecarga excesiva en el proceso de réplica que presentan los SBDR, si tienen una complejidad muy alta en la realización de la transacción, específicamente cuando incluye consultas globales, ya que aunque en este caso, por trabajar con PostgreSQL, se evita el tema del manejo de bloqueos, quedan otros como selección de controlador de réplicas, de back ups, etc., que hacen que la realización de la transacción sea compleja, mientras que en los SBDR, la complejidad es solo al replicar esa información, que el Slony I disminuye considerablemente.

3.2.3. Aspectos importantes

En el caso de SBDD se hablaba del diseño, el procesamiento de consultas, control de concurrencia y confiabilidad, con todos los aspectos que hay que considerar dentro de cada elemento de los mencionados, mientras que en el caso de SBDR hay que pensar en los mismos elementos, pero con otros matices que se detallarán próximamente.

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

3.2.4. Estado del arte

Aquí se puede presenciar una de las causas por las que los desarrolladores se deciden por los SBDR, y es que hay muchas más experiencias en este tema, que el de los SBDD, lo que unido al hecho de que es más probable contactar con esos desarrolladores exitosos y aprovechar sus experiencias, que en el otro tema.

3.2.5. Diseño de la distribución y la replicación

En el SBDD, diseñar la distribución incluye información sobre los datos más accedidos, por quien son consultados, modificados, etc. Todo esto definirá la mejor manera en que se planteará la fragmentación de los datos y la distribución en la red de los mismo, mientras que en el SBDR, también se piensa en eso, aunque también es posible solo analizar el negocio para definir la manera en que se diseñará el modelo replicación, y que, la información de una localidad sea copia fiel de las otras.

3.2.6. Procesamiento de Consultas

Los SBDD, casi siempre, como parte de sus funcionalidades, necesitan hacer transacciones en las que es necesario consultar datos que están en otras locaciones, y es entonces cuando se debe analizar el costo de las operaciones, para hacer el mejor uso posible de las mismas. Es importante el proceso de

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

descomposición de consultas, la localización de los datos dentro del sistema, y la optimización de las consultas, tanto a nivel global, como local, para lo que se piensa en tipos, granularidad y tiempo de optimización.

En el SBDR no hay que pensar en esto, solamente en el proceso de replicar los datos que se modificaron como resultado de la interacción con el sistema

3.2.7. Control de Concurrencia y Consistencia

Este es un tema en que cuenta con aproximadamente los mismos algoritmos en los dos ambientes, en ambos aparecen algoritmos optimistas y pesimistas.

Dentro de los optimistas están los basados en candados y los de estampas de tiempo.

Los basados en candados engloban a los centralizados, los de copia primaria y los distribuidos.

Los de estampas de tiempo reúnen los básicos, los multi versión y los conservadores.

Los pesimistas son los basados en candados, estampas de tiempo y los híbridos

En el caso de los SBDR, se presentan estos mismos protocolos, además de que se usan las dos variantes de réplica para controlar la consistencia de los datos, la réplica sincrónica (propagación ávida de las actualizaciones) y la réplica asíncrona (propagación perezosa de las actualizaciones)

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

3.2.8. Confiabilidad

Este es un tema muy importante en los dos sistemas, por ser el que garantizará que los sistemas de manejo sean confiables, pudiendo continuar la ejecución de solicitudes del usuario a pesar de que el sistema sobre el que se opere no sea confiable.

En los sistemas manejadores se pueden presentar fallas, que son de distinta naturaleza, en dependencia del ambiente del que se trate. En SBDD hay fallas de transacciones, fallas del sistema, fallas del medio de almacenamiento y fallas de comunicación, mientras que en los SBDR solo aparecen las 3 primeras, lo que aporta otra buena razón para que la propuesta del trabajo esté orientada a este tipo de solución.

Los SBDD usan protocolos de recuperación, a nivel local y global, siendo estos últimos muy complicados, debiendo garantizar atomicidad y durabilidad de las transacciones, además de garantizar que los nodos sigan trabajando normalmente después de un fallo de transacción, y no forzar a otros a que la aborten, conocido como no-bloqueante, igualmente hay que valorar como terminar la transacción que se ejecutaba al momento de la falla.

En los protocolos locales, están los de recuperación in-place, y la out-of-place.

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

Dentro de los protocolos globales, están los de Compromiso en dos fases, y dentro de las fallas en los nodos, hay protocolos de terminación para el coordinador y para los participantes, conocidos como timeouts. También hay protocolos de recuperación a nivel de coordinador y de participantes.

En el caso de los SBDR, solo existe la recuperación a nivel local, y se usan protocolos de recuperación explícita, basados en logs, en versiones, entre otros.

3.2.9. Selección del Coordinador

Este es un tema referido a los SBDD, y es el coordinador que se usa para manejar las transacciones y los bloqueos, además de los back ups.

En los SBDR, son los gestores de bases de datos los encargados de supervisar el estado de las operaciones que se realizan con los datos de la BD. Estos sistemas hacen buen uso de estas propiedades. Cada vez que se tramita una transacción esta lo hace de forma local, cada acción a realizar no implica directamente la búsqueda de información en otro lugar. De este modo cada gestor puede controlar la ejecución de las transacciones que tienen efecto en su localidad. Del mismo modo en caso de presentarse bloqueos los gestores locales de bases de datos son los encargados de solucionar el problema.

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

En el caso de los SBDD, el procesamiento de las transacciones se realizaría a través de controladores globales. Estos son los responsables de tramitar las acciones, coordinar el proceso de consultas a los datos que ejecutan en la transacción.

Al tratar el caso de los bloqueos, los controladores adquieren gran importancia puesto que ellos son los encargados de detectar posibles casos de bloqueos. La importancia radica en que si permanecen los bloqueos sin ser detectados, debido a las características de las transacciones globales puede efectuarse una sucesión de bloqueos que provoque el colapso o la inutilidad del sistema, debido a que la información no puede ser accedida.

Los controladores globales presentan un inconveniente: si falla el controlador global el sistema debe impedir que los resultados parciales de las transacciones sean desechadas. Para ello debe contar con un respaldo para casos de emergencia, como la salida de servicio de un coordinador.

3.3. Características del MAC

Uno de los elementos a tener en cuenta es que el MAC posee centenares de trabajadores a nivel central, con necesidad de acceso a la información relacionada con los diferentes procesos de auditoría y control. En cada nivel, ya sea provincial o ministerial, los trabajadores no solo leen información sino que lo hacen de forma constante.

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

Por las investigaciones realizadas, se estima que con la cantidad de trabajadores existentes solo a nivel central del MAC y tomando en cuenta las direcciones que posee, deben conectarse al servidor nacional centenares de usuarios concurrentemente, por lo que es imprescindible dotar al mismo de una capacidad de procesamiento que responda a la demanda. Esto, sin contar que recibe información desde otros servidores de BD, dispersos por el país.

La velocidad actual de conexión es de 56 Kb/s, y tratándose de un ministerio que requiere de tráfico de información de forma constante, dicha velocidad se puede catalogar de insuficiente. Pensar en almacenamiento distribuido de datos con estas condiciones o en dotar al ministerio de sitios web es casi imposible.

3.4. Software a emplear en el entorno de replicación

Para contribuir a la concepción de un diseño para gestionar la información del MAC, son necesarios ciertos recursos, entre los que se encuentran los SW a usar para gestionar la información.

3.4.1. Sistema operativo Linux

La política que existe en el país es migrar los sistemas informáticos hacia SW libre. La utilización de GNU/Linux resulta ser una opción muy favorable. El software libre ofrece amplias posibilidades para

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

países en vías de desarrollo, como el nuestro. También es importante el uso de SW que no se adhieran a la política de bloqueo que es impuesto al pueblo cubano, para tener estabilidad tecnológica.

De las distintas versiones UNIX para PC y servidores, son altamente estimadas a nivel internacional, la velocidad y confiabilidad con que opera el sistema operativo Linux, producto derivado del proyecto GNU. Un aspecto a tener en cuenta es la existencia de un gran número de aplicaciones sobre el sistema operativo, lo que le pone por delante de muchos sistemas UNIX, como el caso de Solaris.

Pudiera cuestionarse por muchas razones el hecho de no usar Windows. Este potente sistema operativo creado por Microsoft ofrece muchas facilidades, también para el trabajo de oficina, especialmente debido a que tiene todo un cúmulo de programas que son creados por esa misma compañía para complementar su sistema operativo. Para el trabajo de oficina posee el paquete de Office, que es altamente aceptado por sus prestaciones y las comodidades que incluye. Sin embargo Linux en sus diferentes versiones posee un paquete para el trabajo de oficina similar al que tiene Windows, el OpenOffice. Además Windows en sus versiones actuales requiere cada vez de más rendimiento de hardware, en contraposición a los bajos requerimientos de Linux, que permiten hacer un sistema potente y útil para aquellas naciones que no pueden darse el lujo de adquirir la última tecnología en hardware. En cambio, si son adquiridos dispositivos con una alta velocidad de procesamiento, el uso de Linux será mucho más eficiente.

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

Haciendo un análisis respecto al sistema multitarea y de administración de memoria, resulta fácil entender la superioridad de Linux así como también, las capacidades superiores de manejo de conectividad a redes y sistema multiusuario. Otro elemento de consideración, es que este sistema operativo es poco atacado por virus informáticos, lo que junto al poco requerimiento de hardware que necesita para operar, provoca que normalmente los servidores en casi cualquier parte del mundo estén basados en esta tecnología, elección también amparada en que optimiza muy bien el uso de los recursos que el sistema le proporcione.

3.4.1.1. Características

- Multiprocesador: Puede usarse también en computadoras con más de un procesador.
- Multitarea: Referido a la posibilidad de realizar la ejecución de procesamiento de varios peticiones de disimiles programas al mismo tiempo. La multitarea preventiva es una de las habilidades de este sistema operativo, a través de ella se asegura que todos los programas que se están utilizando en un momento dado serán ejecutados, es responsabilidad del sistema operativo otorgar tiempo de procesamiento del microprocesador a cada programa que lo requiera.
- Multiusuario: Como tantos otros, provee la posibilidad de tener varios usuarios, aun de distinto tipo, realizando actividades en un mismo instante de tiempo sobre la misma máquina.

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

- Tiene la mayor comunidad de desarrollo de software libre.
- No requiere de procesadores tan potentes y sofisticados para funcionar a plenitud.
- Provee de una herramienta parecida al Administrador de Tareas de Windows pero con la confiabilidad de que cuando se ordene eliminar un proceso esta acción es realizada de manera instantánea. Con esto se evita que la máquina quede “congelada”.
- Linux es capaz de soportar distintos sistemas de ficheros, como pueden ser: FAT 16, FAT 32, ISO, HPFS, etc., y cuenta con uno propio: Ext2.

3.4.2. PostgreSQL.

PostgreSQL es un poderoso gestor de bases de datos (GBD) libre. Es un sistema relacional de bases de datos que cuenta ya con más de 15 años de desarrollo. PostgreSQL es considerado uno de los mejores y más populares gestores de BD de código abierto que compite al nivel de los más avanzados del mundo. Posee características que le proporcionan gran credibilidad y prestigio. (Desarrolladores, 2008)

3.4.2.1. Características

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

PostgreSQL ofrece facilidades y tiene características de alta estimación para quienes lo usan:

- Interfaces naturales para C, C++, PHP, Perl, ECPG, Python, entre otros.
- Réplica que permite la duplicación de la base de datos principal a las máquinas auxiliares múltiples.
- Compilador ACID Completo
- PG_Dump para realizar salvadas.
- Multiplataforma.
- Diseñado para los ambientes de alto volúmenes de almacenamiento.
- No usa sistema de bloqueo de forma predeterminada.
- Trabajo en servidores: es efectivo posee reglas, triggers.
- Trabaja a una misma velocidad tanto para bajos volúmenes de datos como para altos volúmenes.
- Es libre

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

3.4.3. Slony I.²⁶

Esta herramienta ha sido diseñada específicamente para el trabajo con muchas versiones de PostgreSQL. Existen en el mundo múltiples soluciones que permiten llevar a cabo la réplica de datos enfocadas al gestor de base de datos utilizado en cuestión. Pueden citarse entre ellos Postgres-R, PgReplicator, Usogres, ErServer. Estas aplicaciones se dividen de acuerdo a los entornos de réplica donde pueden ser aplicadas, para el entorno “maestro-esclavo”.

Slony I es un replicador que utiliza en modo maestro a múltiples esclavos. Implementa un sistema de replicación en cascada y promociona (en caso de ser necesario) los esclavos a maestros. La principal característica de Slony I es que es un sistema maestro - esclavo que incluye la clase de capacidad necesaria para replicar grandes bases de datos hacia un número razonables de esclavos. Si el número de esclavos es muy extenso el costo de las comunicaciones incrementa prohibitivamente, por lo que se pierden los beneficios de tener múltiples servidores a los que se le envía información.

Se trata de un sistema pensado para realizar centros de datos y la salva mediante la distribución de los mismos. Normalmente el modo de operación del mismo es concibiendo que los nodos implicados en el sistema de réplica estén disponibles en el momento de la replicación y todos puedan ser asegurados.

²⁶ Informacion obtenida del Sitio Web de Slony

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

Está pensado para sistemas donde los servidores estén disponibles la mayor parte del tiempo. También permite indicar qué cambios replicar de un servidor a otro.

3.4.3.1. Características

- Sincrónico de un origen a múltiples esclavos.
- Sincrónico de múltiples orígenes a múltiples esclavos.
- Asíncrono entre múltiples maestros.
- Multiplataforma. Cualquier plataforma en la que pueda funcionar PostgreSQL debe servir para ejecutar Slony I.

3.5. Elección del tipo de Almacenamiento Distribuido para el MAC

Luego de la comparación realizada a los Ambientes de Almacenamiento Distribuido de Datos, el equipo llega a la conclusión que la opción a utilizar es la de SBDR. Por las características que presenta resulta destacable el hecho de que es menos dependiente de los altos niveles de rendimiento en la red lo cual es muy conveniente debido a los problemas de velocidad presentados en las redes del MAC. Resulta importante el hecho de que el objetivo la distribución del almacenamiento de las BD del MAC

CAPÍTULO 3. COMPARACIONES Y ELECCIÓN DE ESTRATEGIA DE ALMACENAMIENTO.

no busca dividirlos o fragmentarlos, sino que se trata de centralizar los datos de las BD. Pesa mucho en la decisión de decantarse por la opción de los SBDR el hecho de que ya existen experiencias positivas en la aplicación de los mismos, en cambio para el caso de los SBDD no se reconoce como un sistema ampliamente utilizado salvo en sistemas experimentales, y no es recomendable usar sistemas experimentales cuando se trata de un proyecto de informatización de un ministerio, al menos no con las condiciones con que cuenta la infraestructura informática actual, y la que se prevé que usará a corto y mediano plazo.

3.6. Conclusiones

En este capítulo se han realizado comparaciones entre las características de los SBDD y los SBDR. Se han visto las ventajas y desventajas presentada por cada uno, con la finalidad de elegir uno para su posterior aplicación al modelo de almacenamiento distribuido de bases de datos del MAC. Además se han analizado, sin detallar demasiado, las características del ministerio, en consecuencia fueron descritas las herramientas que, por las facilidades que brindan, pudieran utilizarse para llevar a cabo el sistema de almacenamiento.

CAPITULO 4: PROPUESTA

4.1. Introducción

Hasta el momento se han visto las características los sistemas SBDD y SBDR para la distribución de la información del MAC. Aunque la opción de utilizar un SBDD resulta interesante y puede parecer muy sofisticada y tentativa; como ya se ha visto, su aplicación presenta claras desventajas tales como: complejidad, tiempo extra de procesamiento, alta dependencia de los sistemas de redes, entre otros aspectos.

Tras haber analizado las potencialidades de ambos sistemas y las necesidades del ministerio a informatizar, teniendo en cuenta los resultados de las investigaciones realizadas acerca de cómo debe fluir la información dentro de las redes del MAC y llegado a la conclusión de que un sistema de réplica es apropiado para las características de flujo de la datos, se procede a fundamentar la propuesta.

4.2. Diseño

Resulta necesario apelar a la distribución de la información, tratando siempre de mantener la confidencialidad de los mismos, o sea, que solo aquellos usuarios implicados directamente con determinada información, tengan acceso a ella. También hay que eliminar la posibilidad de que la información se pierda. No menos importante resulta ofrecer la posibilidad de obtener y guardar datos en todo momento.

4.2.1. Jerarquía

Como se ha enfatizado con anterioridad, este es un sistema donde cada servidor posee su propia base de datos. Cada central trabaja de forma independiente, de modo que a los efectos del diseño, son servidores, que aunque pertenecen a un mismo sistema, gestionan la información de manera que cada uno es responsable únicamente, de la información que se encuentran en su localidad.

Los servidores se comunican, en la concepción lógica, de forma escalonada con el objetivo de enviar los datos hacia la BD Central. La jerarquía de los servidores estará condicionada al objetivo que le sea asignado al mismo, es decir, unos recibirán información proveniente de la interacción de los trabajadores del ministerio con el sistema, luego replicarán dicha información, mientras que otros reciben los datos mediante el proceso de réplica.

En la parte más baja de la concepción lógica, se encuentran los servidores provinciales. En estas centrales se concentrará la labor de recibir y almacenar directamente desde las estaciones de trabajo, los datos correspondientes a la gestión auditora y controladora de la provincia. También son recogidos los resultados de los procesos de auditoría así como toda la información referida al centro y sus trabajadores.

En el nivel intermedio se encuentran los servidores regionales. La principal labor de estos es permanecer como punto intermedio de almacenamiento de los datos que se generan en las provincias que atiende. También puede considerarse como un respaldo de los datos provenientes de las delegaciones provinciales.

En el nivel superior se encuentra el servidor residente en la sede central del ministerio de auditoría. En este nivel, a igual que el primero, se recibe información que es proporcionada de forma directa por las estaciones de trabajo adheridas al mismo. Estos datos pueden ser relacionados con cualquier entidad perteneciente a cualquier provincia. Además de recoger información de este tipo, también recibe de sus estaciones de trabajo toda la información generada por las diferentes direcciones que se derivan del trabajo ministerial. También almacena todos los datos pertenecientes a las bases de datos de las provincias. Por estos motivos es justamente considerado como servidor principal.

Se debe aclarar que el enfoque jerárquico que se menciona es solo a nivel lógico. No importa si es un servidor provincial, regional o nacional, tienen igual posibilidad de conexión en el nivel físico debido a que cada nodo puede acceder a la red en anillo sin necesidad de un intermediario. Con este método se garantiza la posibilidad de cualquier servidor de establecer conexión con otro, sin restricciones de conexión física de algún tipo, excepto en caso de fallas del servicio.

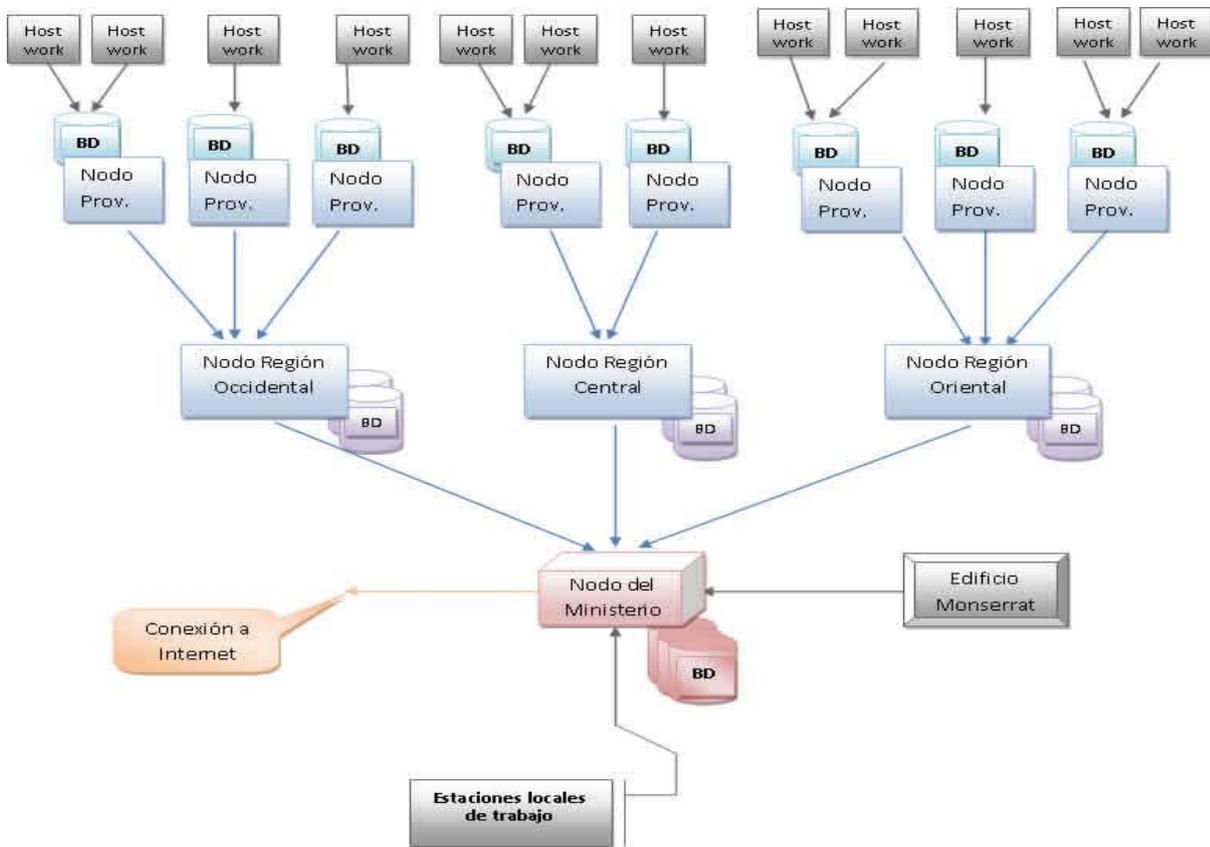


Figura 4.1. Concepción Lógica

4.2.2. Nodos Implicados

Es importante destacar que el proceso de distribución de los datos debe realizarse de forma que todas las delegaciones provinciales puedan enviar sus datos hacia la sede principal. Para lograr dicho objetivo la mejor alternativa es tener, en cada provincia, un servidor con una base de datos y su correspondiente gestor así como los diferentes SW asociados al sistema de replicación.

Además, se concibió de manera lógica, la división del país en tres zonas: Oriental, Central y Occidental, destinando para cada una un servidor de BD.

Un tercer elemento es el servidor en la sede del ministerio. Este servidor será el principal del conjunto de servidores distribuidos a nivel nacional.

4.2.3. Multiusuario

El sistema debe ser multiusuario. Cada nodo contará con determinada cantidad de usuarios que tienen diferentes permisos para acceder a los datos almacenados. Los trabajadores del ministerio interactuarán con la información de la BD a través de una aplicación web que manipule los datos.

Las bases de datos a nivel de provincia tendrán estos tipos de usuarios:

- Usuarios con permiso de lectura/escritura: Serán aquellos usuarios que inserten, modifiquen, eliminen o realicen búsquedas sobre la base de datos local. Son los trabajadores del ministerio que por sus funciones, estén vinculados a la información.
- Usuarios con permiso de solo lectura: Será aquellos que solo puedan leer la información correspondiente pero no tienen permiso de modificación. Son los trabajadores que necesiten manejar y no crear, eliminar o modificar información.
- Administradores locales: Serán aquellos usuarios encargados de administrar la BD, proveyendo seguridad a los datos guardados. Estos usuarios deben ser informáticos experimentados en el uso de las herramientas de manejo de las bases de datos.

Por su parte, los nodos regionales no tendrán usuarios que aporten información aunque si debe existir un personal encargado de darles mantenimiento, así como al menos un administrador de la base de datos. Recordar que estos solo actúan como punto de almacenamiento provisional para que los datos puedan ser enviados más tarde hacia la BD del Ministerio.

El nodo central tendrá los mismos tipos de usuarios que los mencionados en el nivel provincial, y en este caso, los administradores podrán realizar operaciones de administración a las BD localizadas en los servidores de las delegaciones provinciales y los servidores regionales debido a que poseerán privilegios de administración en dichos nodos.

Como se ha visto, atendiendo a las necesidades de los usuarios de consultar los datos, no hay necesidad de buscarlos en otra localidad. En todo caso, se realizarían chequeos de disponibilidad de los nodos a los que se desea replicar algún tipo de información. Toda la información que un usuario requiera estará en su propio servidor de bases de datos, lo cual hace mucho más rápida y efectiva la consulta.

4.2.4. Conexiones de red

En los momentos actuales se debe sostener la idea de mantener una red privada Para resolver las necesidades informativas del ministerio, para lo que es imperante contar con una velocidad de conexión que garantice el envío de datos por la red sin que esta llegue a colapsar.

Se puede observar que con el uso de una red privada se agregan ciertos beneficios:

- Uso exclusivo de los servicios de red: La entidad correspondiente es la única capaz de hacer uso de los recursos de red. No existen centros de trabajo o servidores foráneos consumiendo de los servicios que se brindan.
- Mayor confiabilidad: Los datos viajan por un ambiente poco hostil.
- Pocos ataques: La probabilidad de sufrir ataques informáticos provenientes del exterior (internet) se reduce considerablemente.
- Configuración: Se pueden hacer cambios en las comunicaciones lógicas sin tener que consultar u obtener permisos de alguna entidad foránea.

Se recomienda tener una red de tipo anillo para conectar todos los servidores que contienen las BD en cuestión. Esta tipología es de gran relevancia para la presente propuesta ya que por esta vía es que serán transferidos los datos mediante el proceso de réplica. Tiendo en cuenta que la velocidad actual es muy limitada y que, según la concepción del proyecto se piensa en la posibilidad de brindar servicios web, se hace necesario realizar inversiones para aumentar la velocidad de conexión.

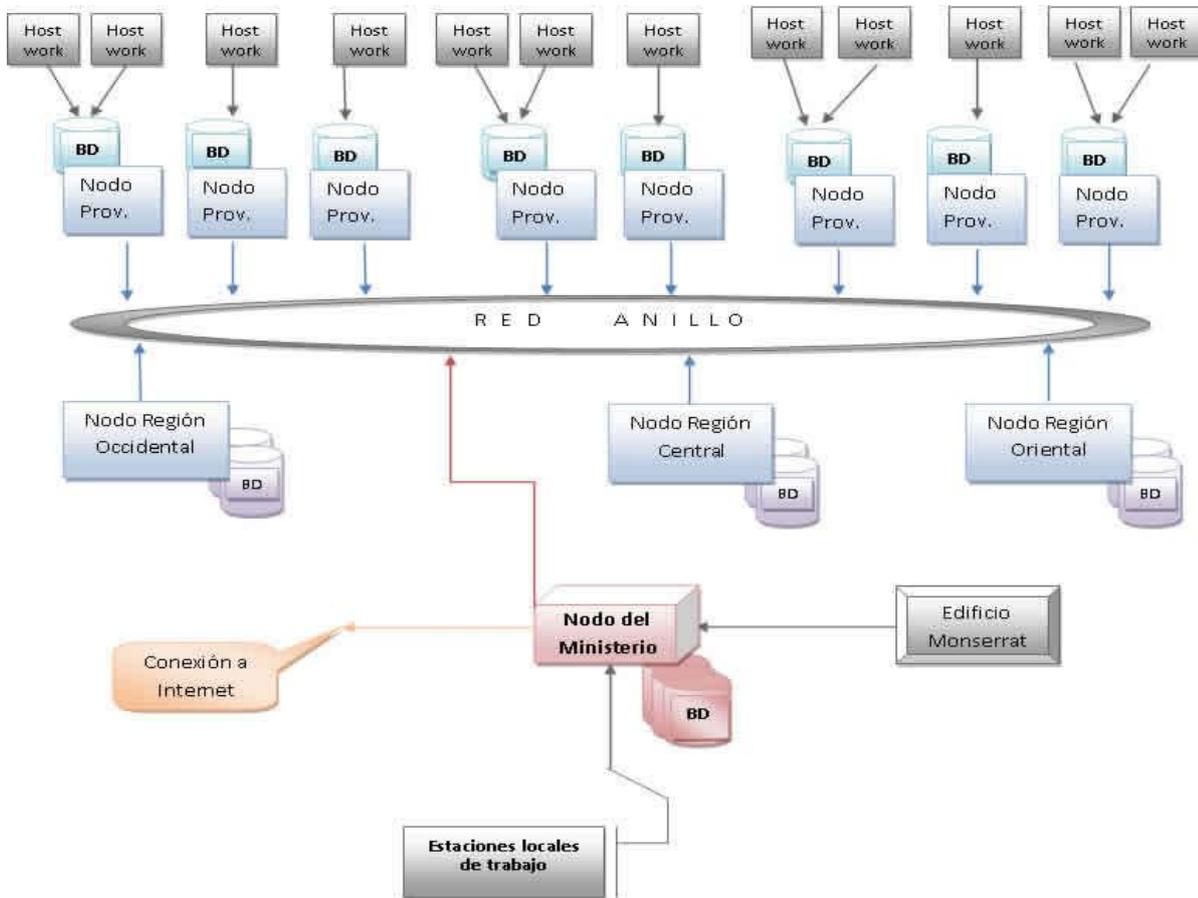


Figura 4.2. Concepción física

4.3. Bases de Datos

4.3.1. Estructura de la Información en los Servidores

La estructura de las bases de datos de las delegaciones provinciales será similar. Cada una poseerá una BD donde se almacene la información sobre las empresas que son de interés para el ministerio al nivel de la provincia en cuestión. Por ejemplo: dentro de la BD existirán tablas que representen las quejas sobre, otras que representen los resultados de auditorías realizadas en la provincia o simplemente que contengan los datos de los trabajadores de esa delegación.

En cambio, las bases de datos de los nodos regionales tendrán una BD por cada provincia que atiende. En dichas BD se almacena la información proveniente de cada BD provincial, por ejemplo: la

base de datos Oriental tiene una BD para Granma, otra para Santiago de Cuba, etc. Siempre se tratará de una copia fiel, de la información almacenada al nivel más bajo, hasta el superior. De esta forma se logra diferenciar la información proveniente de cada BD provincial en vez de mezclar toda la información en una sola BD que contenga todos los objetos que recibe, en vez de usar otra variante de las que plantea la teoría de BDD en el tema de asignación de nombres y autonomía local, como asignador central de nombres o asignarle a cada localidad un prefijo.

Por último la BD del nodo central, tendrá tantas BD como provincias tenga el país más la del Municipio Especial. Al igual que las bases de datos de los nodos regionales, en la BD de la sede ministerial los datos provenientes de los nodos regionales son ubicados en la BD correspondiente a la provincia a la que pertenece la información. En el caso de los datos que son aportados desde el propio ministerio, serán ubicados en una BD que es creada para guardar los datos de la sede.

4.3.2. Consideraciones generales

Si se observa el comportamiento de la arquitectura propuesta, la idea central radica en usar una base de datos principal (BD ministerial) y varios nodos (BD provinciales) que poseen parte de la información que tiene la principal. La información no fluye, en ningún momento, desde la BD principal hacia las BD asociadas sino que es el proceso contrario. La información que posee cada nodo es solo referente a su área de trabajo. De modo que se trata de una base de datos que se encuentra totalmente en una localidad principal y es el resultado de la unión de los datos de las BD provinciales, a través del proceso de réplica. Se toman estas BD como las primordiales porque es en ellas donde se realizan los cambios significativos a los datos.

No se trata como primordial el tema de los nodos regionales porque estos son solo puntos de almacenamiento y distribución, que proveen estabilidad en la carga de trabajo del nodo principal, ya que reúnen la información de los nodos provinciales bajo su responsabilidad y luego se encargan de enviar los datos al ministerio. Serán, por tanto, solo tres nodos los que envíen información al ministerio y no 15 los implicados directamente en el proceso de actualización de los datos sobre el servidor principal.

Todo este proceso, resultará invisible a los ojos de los usuarios del sistema, quienes al realizar los cambios no necesitan enviarlos a ninguna parte. Para el caso de los usuarios de las provincias, cuando

se realiza una modificación de datos y se guardan los cambios aparentemente quedan en su localidad pero en realidad una copia es enviada a la BD ministerial.

Este modelo resulta el adecuado para la distribución de información ya que se hace uso de la replicación según la lógica del sistema de trabajo del Ministerio. La idea no es enviar al resto de los nodos, datos o fragmentos de estos para hacer más rápidas las consultas, puesto que los datos que se manejan, solo son de importancia para la provincia que los genera y para la sede ministerial; sino que se hacen copias fieles de las bases de datos provinciales hacia la sede del MAC utilizando los servidores regionales, con la finalidad de mantener la disponibilidad en los lugares requeridos y manteniendo siempre la confidencialidad y autenticidad de los datos.

4.3.3. Manejo de datos de forma local

Como se ha descrito en el diseño, se propone y asume que la gestión de los datos requeridos se realice de manera local. Para realizar operaciones con la debida eficacia resulta necesario el uso de un potente gestor de bases de datos. No existe la posibilidad ni la necesidad de que un usuario no administrador global, o un gestor de una localidad determinada gestionen la información contenida en otra localidad.

4.3.4. Recuperación de las BD

4.3.4.1. Utilizando Archivos LOG

Salvar los datos puede resultar un proceso tedioso pero necesario. Sin embargo existe la posibilidad de que el SGBD de cualquier servidor, presente problemas con las BD y provoque que ciertas transacciones no sean realizadas. En ese caso la BD puede haber quedado en un estado inconsistente.

En caso de presentarse una caída de la BD, se puede restaurar a través del uso de los archivos log, conocidos como: *write ahead log* (WAL) contenido en el `pg_xlog/` y que es una de las características de PostgreSQL que ayudan al buen rendimiento en la gestión de los datos. Este archivo describe cada cambio hecho a la BD. De esta manera los posibles cambios realizados o en proceso de realizarse en las distintas transacciones, pueden ser retomados o desechados según sea conveniente.

4.3.4.2. Salvas (Backup)

Siempre es de gran utilidad tener un respaldo para casos de modificación indeseada o pérdida de los datos. Aunque es posible replicar desde niveles inferiores en caso de pérdida de información, es más conveniente usar salvas para lograr la recuperación de los datos, teniendo en cuenta que no necesita sobrecargar la red ni invierte demasiado tiempo en el proceso. Es deber de los administradores de la BD central realizar de manera periódica salvas o backup de las bases de datos contenidas en dicho servidor.

El gestor de BD PostgreSQL, provee opciones de salva que los administradores pueden usar. De la misma manera, es necesario señalar que la frecuencia con que se harán las salvas será determinada por los administradores según las necesidades del ministerio. Siempre que se sea necesario hay que realizar salvas de los datos, la cuestión es como realizar las mismas en un ambiente donde se están utilizando de forma casi constante los recursos de la BD. Pueden tomarse dos casos:

1. Detener los servicios del SGBD antes de realizarlo.
2. Realizar el proceso de con el SGBD en plena ejecución.

4.3.4.2.1. Salvas en Frío (Cold Backup)

En este caso se debe realizar la salva en horas de la noche o madrugada que es cuando los datos son menos requeridos, puesto que los usuarios están en horario de descanso. Pero tiene claros inconvenientes, al tener que poner fuera de servicio el servidor:

- Los usuarios del servidor ministerial, a esa hora, no dispondrán de ningún tipo de servicios datos.
- Los servicios de replicación deben estará detenidos (en el caso de que exista personal trabajando a esas horas en las BD provinciales, los cambios realizados no actualizarán inmediatamente la BD nacional)

4.3.4.2.2. Salvas en Caliente (Hot Backup)

El segundo caso resulta más cómodo aunque el tiempo del proceso demora un poco más. Son visibles las ventajas, entre las que se pueden citar:

- No requiere poner fuera de servicio el SGBD por lo que en todo momento los trabajadores del MAC que requieran de suministro de información la podrán adquirir sin interrupciones (incremento de la disponibilidad).
- Los servicios de replicación de datos no se verán afectados por la no accesibilidad a la BD.

Una desventaja que se presenta es que cuando las BD son muy grandes los servicios asociados a `pg_dump` (que es una aplicación cliente que realiza salvadas en caliente) suelen ser inestables. Esta es una de las razones por las que se recomienda crear una BD por provincia. (PostgreSQL, 2008)

4.4. Definiendo Maestros y Esclavos

4.4.1. Flujo de Replicación

El proceso comienza en las provincias, que es donde se insertan los datos. Cada tabla tendrá asociado un trigger que se accionará en caso de modificación de los datos de dicha tabla (INSERT, UPDATE y DELETE). Este trigger accionará una función que activa el slon (motor del Slony I). El slon realizará las tareas correspondientes de conexión al esclavo que resultará ser el nodo regional.

Una vez establecida la conexión se procede al traspaso de manera asíncrona de los datos desde el servidor maestro hacia el esclavo. Como Slony I trabaja con el envío de grupos de datos, cada grupo lleva la dirección de la tabla donde debe ser adicionado o insertado cada dato. Por su parte en el servidor esclavo, el slon es el encargado de recibir y almacenar los datos según la información que contenga.

Una vez almacenada la información en el nodo regional y finalizada la transmisión de todos los datos según sea la BD a la que corresponden (recordar que en los nodos regionales hay una BD para cada provincia que atiende), son accionados triggers que vigilan el comportamiento de las tablas locales. Es puesto en marcha el slon local para un nuevo proceso de replicación, esta vez hacia el servidor nacional con el objetivo de enviar los datos modificados.

En el servidor nacional el slon local recibe los datos que son enviados por el servidor maestro. A la hora de almacenarlos se tiene en cuenta la base de datos a la que pertenece el grupo de datos recibidos, luego son almacenados consecuentemente. Para todos los casos, una vez culminado el envío de los datos la conexión se pierde hasta que un nuevo proceso de réplica se ponga en marcha.

4.4.2. Alternativa

El Gestor de base de datos PostgreSQL permite trabajar con SCHEMA (esquema), dentro de los esquemas, se crean tablas, vistas, triggers, funciones, etc., es decir, todos los elementos que se puedan necesitar para la administración de los datos. Al poseer esquemas, prácticamente se puede trabajar como si fuera una base de datos diferente, es más, un esquema puede interpretarse como una pequeña base de datos dentro de una gran base de datos. (Digital, 2007)

Un variante a considerar es tener una BD en cada servidor regional, dicha BD poseerá un esquema para cada provincia, o sea que la información proveniente una determinada Delegación Provincial, será almacenada en una tabla de la provincia correspondiente a la misma. De igual modo en el servidor nacional puede tenerse una BD con tantas tablas como provincias tenga el país, mas otra tabla para la información proveniente del municipio especial Isla de la juventud. De la misma forma en que es almacenada la información en los servidores regionales, se puede hacer en el servidor nacional.

En este caso, la base de datos nacional deberá tener un esquema para gestionar la información que se tramita en la sede del ministerio.

La desventaja de esta propuesta está dada para el caso en que los volúmenes de información sean muy grandes. En este caso, a medida que pase el tiempo y sean realizadas operaciones, la capacidad de almacenamiento de dichos esquemas puede resultar insuficiente ante los altos volúmenes de información a guardar. Una posible solución a este problema sería la realización, de forma periódica, de salvadas de la información más antigua de la BD que pueda resultar prácticamente innecesaria para el trabajo del ministerio.

4.4.2.1. Pérdida del Servicio

En el análisis de este tema, como pérdida del servicio se asumirá en primero lugar, la pérdida de conexión entre la localidad y el resto del sistema, además de la caída del nodo, relacionadas con fallos no referidos a la conexión.

4.4.2.1.1. Pérdida del Servicio del Nodo Provincial

Este es el primero de tres casos a tener en cuenta:

Si el servidor de la provincia de Granma pierde la comunicación con el nodo Oriental (Servidor Regional); los usuarios de la sede ministerial pueden seguir trabajando con los datos que tienen almacenados en su BD local. Cuando se recupere la conexión se activa el sistema de réplica y los datos del ministerio son actualizados. En caso de destrucción de la información de la BD en el nodo provincial, se puede hacer el servidor de BD de la región maestro de la provincial y replicar los datos que posee sobre esa provincia. Para cuando la Delegación Provincial cuente con todos los datos perdidos, se pone en ejecución el orden de réplica que existía originalmente. De este modo la delegación afectada recupera los datos perdidos.

4.4.2.1.2. Pérdida del Servicio del Nodo Regional

El segundo caso que puede ocurrir está dado por la siguiente situación:

Si el servidor regional pierde conexión con el nodo ministerial y/o con los nodos provinciales a los cuales atiende, se puede realizar una nueva conexión, esta vez, redireccionando en las provincias, el IP hacia donde deben ser replicados los datos; este será el IP del nodo del ministerio, en consecuencia en este último, se reconfigura el script que le indica de donde deben ser aceptados los pedidos de réplica. Dicho de otro modo, ahora el servidor de BD provincial dejaría de ser maestro del servidor regional, el servidor de BD regional dejaría de ser maestro del servidor del ministerio. El nuevo sistema maestro esclavo se realiza entre la provincia y el ministerio de forma directa. Los servidores provinciales pertenecientes a la zona afectada, pasarían a ser los nuevos maestros (aunque de forma temporal) del servidor central. Cuando se restablezcan los servicios del nodo Regional, se toman las rutas originales de replicación de la información.

4.4.2.1.3. Pérdida del Servicio del Nodo del Ministerio

Un tercer caso puede ser el que se presenta a continuación:

Ante esta circunstancia hay que esperar a que la conexión del servidor nacional sea restablecida, para que pueda adquirir los datos que han sido modificados en las provincias, y que esperan a ser replicados. Los usuarios locales pueden continuar con sus labores contando con la información que contiene el servidor antes de la incomunicación.

4.4.2.1.4. Recomendación para el caso de caída de nodos

En todos los casos de caída de los nodos, es la mejor práctica usar siempre el backup de la BD hasta donde sea posible, de modo que hacer réplicas a gran escala, con la necesaria sobrecarga de la red, sea la última opción.

4.4.3. Alternativa de Comunicación usando la Red de Fibra Óptica Nacional

En la actualidad existe una red de fibra óptica a nivel nacional, con la cual se pretende dar cobertura a las necesidades de comunicación de las diferentes instituciones del país. Integrar el sistema de redes del Ministerio a la misma resultaría beneficioso para transmitir todo tipo de información a gran velocidad. Téngase en cuenta que servirá de apoyo en caso de sobrecarga del sistema de redes del ministerio. El hecho de conectar cada uno de los nodos de las delegaciones provinciales, los nodos regionales, y el nodo residente en la sede del MAC a la red de fibra óptica nacional resultaría beneficioso, si se tiene en cuenta que la red privada puede fallar. En ese caso, si cada nodo tuviese acceso, de alguna forma, a la red de fibra óptica, esta puede tomarse como vía alternativa para el envío de información.

Para lograr los objetivos de una distribución y recuperación de datos de manera confiable hay que considerar las redes de comunicaciones necesarias. De modo que, cada nodo debe tener al menos dos vías de comunicación: una por la red privada y otra por la red nacional. Esta última es obviamente una conexión de respaldo.

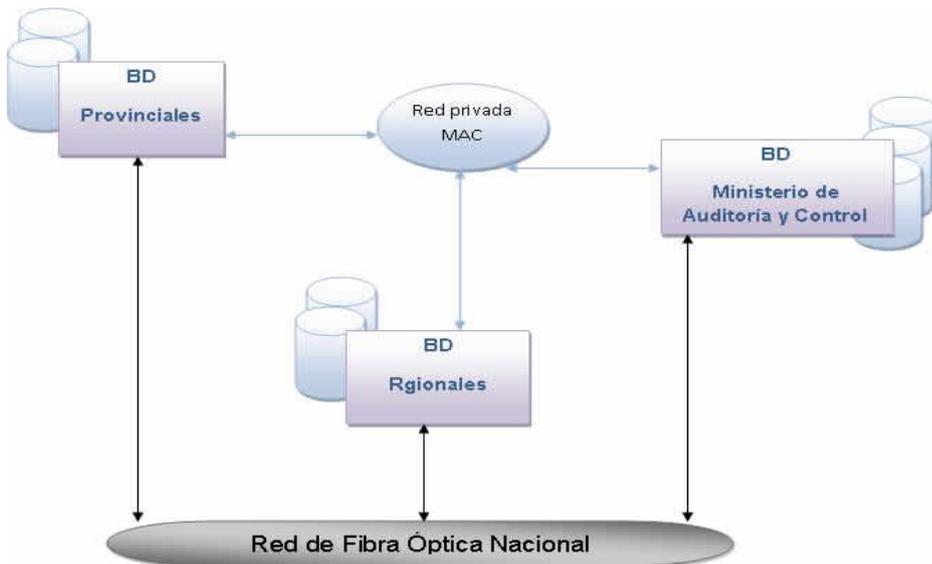


Figura 4.3. Conexión con la red nacional de fibra óptica

4.5. Aspectos Importantes

Una característica que se debe puntualizar es que en este sistema los datos replicados no pueden ser modificados por los usuarios de la BD ministerial debido a su condición de servidor esclavo. En consecuencia, si hay que actualizar datos sobre tablas que se corresponden con las replicadas desde las delegaciones provinciales, la estructura y diseño del modelo de datos debe tener en cuenta la posibilidad de tener en BD del ministerio tablas donde se pueda guardar dicha información.

La presente propuesta, que puede catalogarse de flexible, se presenta como una replicación en cascada (escalonada). Además está basada en pruebas realizadas en el proyecto SIGAC, al que pertenece el equipo de investigadores, donde fue simulado un ambiente de replicación según las características jerárquicas anteriormente expresadas, obteniéndose resultados satisfactorios.

Tómese como referencia la velocidad de transmisión que sostiene la UCI con sus facultades regionales de 56kb, característica que emula con la presentada por el MAC. En las entrevistas realizadas a los administradores de los servidores en la UCI, se ha llegado a la conclusión (en opinión de los administradores) de que resulta insuficiente a la hora de realizar la transmisión de datos. Para ilustrar el ejemplo que se muestra, entiéndase que el envío de 1 Mb de información requiere de aproximadamente 18 segundos, en caso de que sea un solo usuario el que requiera los servicios de la

red. Tratándose de un sistema donde se requiere replicar información de manera periódica, este ancho de banda resulta insuficiente.

4.6. Conclusiones

Hasta aquí se ha presentado la propuesta realizada por el equipo de investigadores, mostrando las ventajas de tener un SBDR para la distribución de los datos del MAC. Esta alternativa es muy eficaz teniendo en cuenta que los datos son ubicados, únicamente, en aquellos lugares donde son necesarios evitando su propagación innecesaria. Esta propuesta ha sido fundamentada a partir de los lineamientos de flujo de la información que presenta el ministerio. El previo estudio de las diferentes herramientas de trabajo como PostgreSQL y Slony I ha posibilitado la recomendación de las mismas para usarlas como las ejecutoras de dicho sistema. Ha sido un aporte importante el hecho de dar solución eficaz en caso de fallas en uno o varios puntos del sistema prevaleciendo siempre por encima de todo fenómeno la seguridad de la información.

CONCLUSIONES

El desarrollo de sistemas que posibiliten la eficiente disponibilidad de la información contenida en las bases de datos, resulta de extrema importancia para los ambientes en que los usuarios se encuentran en diferentes zonas geográficas. Debido a la necesidad de realizar consultas rápidas a los datos, es que surgen ideas y/o alternativas donde se resuelven estos problemas. Si bien las redes de comunicaciones pueden ayudar en este sentido, no resulta beneficioso su uso intensivo para enviar y recibir los datos requeridos.

Durante el estudio realizado se han buscado alternativas para brindar la posibilidad de que, haciendo uso de técnicas para el almacenamiento distribuido de datos, poder garantizar que los trabajadores del MAC accedan a estos, y que estén debidamente actualizados en la BD. Para ello se ha hecho un estudio profundo sobre los tipos de sistemas a utilizar y las vías para distribuir la información contenida en las BD. También se ha analizado la prioridad de salvaguardar la información contenida en las BD precisamente mediante la distribución de la misma así como la posibilidad de recuperar los mismos en caso de pérdida.

Lo anteriormente enunciado demuestra que los objetivos que se trazó el equipo al momento de comenzar la investigación, fueron cumplidos, apoyados en la metodología investigativa que se usó, que resultó definitiva a la hora de encauzar el trabajo.

RECOMENDACIONES

A lo largo del este trabajo se analizaron las características del funcionamiento del MAC y la forma de solucionar la cuestión de almacenamiento distribuido de su información. Sin embargo, teniendo en cuenta que la solución propuesta no es infalible, se recomienda tener presente la utilización de la fibra óptica nacional para la comunicación entre los distintos servidores de bases de datos del ministerio. Las restricciones de acceso a la misma estarán condicionadas por el nivel de acceso que le sea concedido al ministerio.

Se recomienda continuar con el estudio de los sistemas que brinden la posibilidad de distribuir de una u otra forma la información contenida en las BD, con el objetivo brindar un acceso los mismos de manera eficiente.

Asimismo, es recomendable que las personas implicadas en la puesta en marcha de esta solución, consulten la bibliografía referenciada, y no solo este documento.

REFERENCIAS BIBLIOGRÁFICAS

BIBLIOGRAFÍA

Ángel Roldán Carrasco, Beatriz Mora Rivas. Bases de Datos Distribuidas. [Online] [Cited: Enero 10, 2008.]

Arévalo, Sergio. 2004. *Consistencia de Datos en un sistema tolerante a fallos.* [Online] Febrero 2004. [Cited: Junio 15, 2008.] <http://gsyc.esct.urjc.es/moodle/file.php/33/Teoria/T6-ConsistenciaDatos.pdf>.

Autores, Colectivo de. 2006. Teleformacion. *Conferencia 2 Tema 1: Teoría científica como sistema.* [Online] 2006. [Cited: Diciembre 20, 2007.]

Briz, Luis Irún. Gestión de la Consistencia en Sistemas Replicados. [Online] [Cited: Junio 15, 2008.]

CyD. G.Coulouris, J. Dollimore, T. Kindberg. 2008. [Online] Junio 1, 2008. <http://www ldc.usb.ve/~yudith/docencia/Telematica/TemasSistDist/ProcesamientoTransII.pdf>.

Date. 2003. *Introducción a los Sistemas de Bases de Datos III.* La Habana : Félix Varela, 2003.

Diaz, A. *Sistemas de Bases de Datos Distribuidas.* [Online] [Cited: Enero 10, 2008.] http://www.cs.cinvestav.mx/SC/prof_personal/adiaz/Disdb/.

Elías, J. Jesús Acosta. 2003. Algoritmos de consistencia rápida orientados por la demanda en sistemas distribuidos de gran escala. [Online] 2003. [Cited: Junio 15, 2008.] <http://personals.ac.upc.edu/leandro/pubs/tesis-jacosta.pdf>.

Ensergi. 2005. Ensergi. [Online] 2005. [Cited: Febrero 15, 2008.] http://www.ensergi.com/downloads/assignatures/cinque/sdb/tema5_4x.pdf.

Escoí, Francesc Daniel Muñoz i. 2005. HIDRA: Invocaciones Fiables y Control de Concurrencia. *UNIVERSIDAD POLITECNICA DE VALENCIA.* [Online] 2005. [Cited: Febrero 15, 2008.] <http://web.iti.upv.es/~fmunyo/research/pdf/phd.pdf>.

Expresion Digital. 2007. Expresion Digital. *Expresion Digital.* [Online] 2007. [Cited: Marzo 7, 2008.] http://www.expresiondigital.cl/articulo.php?id_artic=9.

Gabriela Basconcel, Antonio Benitez, Diana E. I Fernández, Natalia E. Jensen, Mariela A. Maidana, Guido J. Pace, Silvina I. Podestá, Alina Rivero, Gisela L. Rodríguez Gómez, Santiago B. Sanchez de Bustamante. 2002. Universidad Nacional del Nordeste. *Monografía: Replicación.* [Online] 2002. [Cited: Febrero 15, 2008.] <http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/MonogSO/REPLIC02.html>.

Gustavo González, Paola Di Yorio, Luis Ignacio Alonzo, Cesar Barreto. 2003. Instituto de Computación Facultad de Ingeniería Universidad de la República Oriental del Uruguay. *Técnicas Avanzadas para Gestión de Sistemas de Información.* [Online] 2003. [Cited: Junio 15, 2008.] <http://www.fing.edu.uy/inco/cursos/tagsi/Trabajos/2003/TAGSI03-TareaTecnSI-grupo4.pdf>.

María de los Angeles Ayala Mosqueda, Silvia Elizabeth Ibarra, Natalia Avalos, Daniel Andrés García, Santiago Jacobacci, Jorge Lecca, Alejandro Aguirre, Leonardo Avalos, Mauricio Falabella, Walter Larroza, Mario Chaves, Daniel Marcelo Iramain, Julio Maiz., [Online] [Cited: Abril 22, 2008.] <http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/MonogSO/TRANS02.htm>.

Martinez, Janhil Aurora Trejo. Monografías. [Online] [Cited: Marzo 6, 2008.] <http://www.monografias.com/trabajos11/basda/basda.shtml#d>.

México, Universidad Autónoma Indígena de. 2007. Universidad Autónoma Indígena de México. [Online] 2007. [Cited: Abril 22, 2008.] <http://www.uaim.edu.mx>.

PostgreSQL. 2008. PostgreSQL. [Online] 2008. [Cited: Febrero 15 15, 2008.] <http://www.postgresql.org>.

Santos, Jorge Rodríguez. 2005. Rincon del Vago. [Online] 2005. [Cited: Enero 10, 2008.] http://html.rincondelvago.com/bases-de-datos-distribuidas_1.html.

Slony. 2008. Slony. [Online] 2008. [Cited: Febrero 15, 2008.] <http://www.slony.info/documentation/slonyintro.html#INTRODUCTION>.

Soledad Aravena, Alejandra Díaz, Colomba Machiavello, Andrea Nieto, Evelyn Ortiz. 2003. Estructura de Base de Datos Distribuidas. [Online] 2003. [Cited: Febrero 12, 2008.] http://asignaturas.inf.udec.cl/~basedato/trabajos/estructura_debasededatosdistribuidas.doc.

Sybase. Sybase. [Online] [Cited: Febrero 15, 2008.] http://www.sybase.com/content/1028711/6143_whitepaper_v2.pdf.

U.P.M., Departamento de O.E.I. -. 2007. Departamento de O.E.I. - U.P.M. *Departamento de O.E.I. - U.P.M.* [Online] 2007. [Cited: Marzo 7, 2008.] <http://www.oei.eui.upm.es/Asignaturas/BD/DYOBDDISTRIBUIDAS.pdf>.

Vega, Norge Fajardo. 2007. *Sistema de réplica para bases de datos distribuidas en PostgreSQL.* La Habana : s.n., 2007.

ANEXOS

GLOSARIO Y SIGLAS

MAC: Ministerio de Auditoría y Control

BD: Base(s) de Datos

SBD: Sistema(s) de Bases de Datos

SBDD: Sistema(s) de Bases de Datos Distribuidas

SBDR: Sistema(s) de Bases de Datos Replicadas

SMBD: Sistema(s) Manejador(es) de Bases de Datos

PYMES: Pequeñas y medianas empresas