Anexo A:

Anteproyecto de Tesis Aprobado.
Por: José Luis Cisneros y Javier Jirón.

TEMA

SOLUCIÓN A PROBLEMAS DE BASES DE DATOS DISTRIBUIDAS EN SISTEMAS DE PEQUEÑA Y MEDIANA ESCALA

JUSTIFICACIÓN

En los últimos años se ha experimentado un notable desarrollo y avance en las investigaciones sobre sistemas distribuidos, especialmente en lo referente a flujo de información y bases de datos, esto ha dado un considerable impacto en aspectos de desarrollos tecnológicos y sociales.

«El crecimiento demográfico es de manera geométrica, pero el crecimiento tecnológico es lineal»[1], en conclusión, ha permitido pensar seriamente que la población en continua expansión crea cada vez un mayor número de necesidades y precisan urgentemente ser satisfechas. Esto, por ejemplo se ve reflejado en los servicios que un banco ofrece, mismos que se reflejan en la calidad del servicio, apoyados por una moderna tecnología, en particular informática, en lo cual se aplican los modelos y teorías sobre bases de datos distribuidas y se erradican los sistemas centralizados, ya que se debe lograr un óptimo rendimiento.

Las nuevas reglas de los negocios presionan a las empresas a migrar hacia las aplicaciones distribuidas, a pesar de que por experiencia todas coinciden en que estos sistemas son difíciles de implementar y es muy costoso, por ejemplo, el caso de una Universidad, que expande sus fronteras, tienen extensiones en otras provincias de la cual es la sede, e incluso tienen la modalidad de educación a distancia, significa que los datos se encuentran dispersos en todo el país, y se presenta la necesidad de acceder a la información en donde se la requiera, problema que nos sugiere utilizar una base de datos distribuida. Algunas de las razones para la utilización de estas bases de datos distribuidas son:

- La mayor parte de las empresas modernas tienen filiales y sucursales en todo el país y necesitan sistemas de información distribuida que pueden utilizar bases de datos distribuidas.
- Los nuevos paradigmas de atención al cliente y las exigencias para enfrentar a la competencia, sugieren un enfoque que otorga importancia total a los clientes, tanto internos como externos, y por lo tanto un excelente servicio.
- Las compañías esperan explotar las tecnologías de computación más novedosas para mejorar sus procesos de negocios con el fin de seguir siendo competitivas, o de volverse más eficientes y rentables.

Las bases de datos comerciales más populares del mercado[2], incluyen en sus motores, servicios de distribución de recursos bastante sofisticados (como por ejemplo espejos); pero con el inconveniente que necesitan equipo complejo y costoso para su funcionamiento, además de personal capacitado.

Pequeñas y medianas instituciones de nuestro medio, como universidades, empresas de transporte, cooperativas de ahorro y crédito, necesitan agilitar sus operaciones basados en el flujo de la información, pudiendo ser una solución a sus problemas la aplicación de una base de datos distribuida pero en muchos casos no es justificable la adquisición de un sofisticado y costoso motor de una base de datos comercial que dispone de este tipo de servicio, que implica también nuevas adquisiciones de

[1] Esto depende de sí es un país subdesarrollado o en vías de desarrollo, es decir en países desarrollados se puede decir que el crecimiento demográfico va de acuerdo con el crecimiento tecnológico. Este concepto se deduce de los últimos estudios realizados en nuestro país, acerca de la poca importancia que se le da a la investigación, es tal, que el presupuesto que la IBM ha dispuesto para esto, es mayor que el de toda Latinoamérica en conjunto, y que el presupuesto que el Ecuador dispone para investigación es del 0,01% del ingreso bruto del país.
[2] Tales como: Microsoft SQL Server, ORACLE, Informix y Sybase; esto es tomado de estadísticas realizadas por revistas especializadas en informática como PC Magazine.

equipo, ya que en estas instituciones la ganancia obtenida por agilitar sus operaciones, en cierta forma es mínima, y al realizar los estudios de costo-beneficio, prefieren no implantar un nuevo sistema o lo prefieren llevar de otra manera o inclusive como la realizaban tradicionalmente, trayendo como consecuencias atraso, acumulación y pérdida de la información.

Para definir un sistema de pequeña y mediana escala, se utilizará un método denominado «punto de fusión»[3], el cual evalúa un sistema de acuerdo a su funcionalidad y efectividad.

En vista de lo analizado anteriormente, se está en la capacidad de proponer y realizar nuestra propia tecnología en base de pequeñas herramientas no muy costosas[4], el objetivo es dar todas las soluciones posibles. Incluso el mantenimiento de este tipo de sistemas distribuidos basados en herramientas no tan complejas es mucho más barato, ya que no se necesita contratar un especialista en la materia, sino técnicos que conozcan dichas herramientas que son ampliamente estudiadas en institutos de educación superior que dictan carreras de informática y afines.

Cabe mencionar que la aplicación de estos modelos propuestos se puede basar en hardware no tan sofisticado como lo son PCs de escritorio y estaciones de trabajo.

El propósito de este trabajo es dar soluciones nuevas y efectivas, para problemas modernos, que en forma específica es bajar los costos de implantación, desarrollo y mantenimiento de bases de datos distribuidas en sistemas de pequeña y mediana escala, que se plantean especialmente en nuestro medio.

OBJETIVOS

OBJETIVO GENERAL

Solucionar problemas de bases de datos distribuidas en sistemas de pequeña y mediana escala

La propuesta central es demostrar que si se pueden resolver problemas propuestos mediante la generación de nuestra propia tecnología, misma que posibilite abaratar costos, ser amigable al usuario, permita una eficiente administración y fácil mantenimiento.

OBJETIVOS ESPECÍFICOS

- Localizar, puntualizar y definir los problemas de implementación de una base de datos distribuida.
- Elaborar propuestas y algoritmos para la solución de estos problemas.
- Comprobar y verificar los resultados obtenidos por estas propuestas y algoritmos.
- Realización de una aplicación práctica.

MARCO TEÓRICO

Los criterios que utiliza el método de fusión para la medición del software son:

1) número de entradas de usuarios,
2) número de salidas de usuarios,
3) número de peticiones al usuario,
4) número de archivos utilizados (aquí se incluye la base de datos), y,
5) número de interfaces externas.

[3] Un completo enfoque del método del punto de fusión, se detalla en: Ingeniería del Software, un enfoque práctico, de Roger S. Pressman, McGraw Hill, España 1993.
[4] Mucha de la infraestructura informática de pequeñas empresas, se basa especialmente en programas de Base de Datos como FoxPro, o aplicaciones desarrolladas en Clipper, y podemos aprovechar esto como punto de partida.

De acuerdo a este método se define que un sistema de pequeña escala es aquel en el que el número de usuarios que interactúan es menor a diez, y su capacidad para almacenar información es menor a un millón de registros por tabla en la base de datos; en cambio un sistema de mediana escala es aquel que puede agrupar hasta cien usuarios, y las tablas de la base de datos pueden almacenar hasta cinco millones de registros, e inclusive puede superar este límite. Por lo tanto se podría decir que este tipo de sistemas, son los que usan las organizaciones pequeñas y medianas.

Las aplicaciones de procesamiento de información incluyen cinco capas principales funcionales:

- Un administrador de presentación como Windows o X Windows para UNIX o LINUX, OSF/Motif;
- Una capa lógica de presentación, que maneja la interfaz de usuario;
- Una capa lógica de aplicación, que a menudo es llamado control lógico o capa lógica de flujo, la cual determina el comportamiento correcto de una aplicación;
- Una capa lógica de datos, que almacena y recupera la información mediante el mecanismo de base de datos;
- Un mecanismo que maneje los datos como un DBMS, o que pueda operar como tal.

El administrador de presentación y el mecanismo de base de datos son componentes estándar. Las tres capas lógicas deben ser desarrolladas. El reto en el diseño es asignar el contenido de las capas lógicas a los componentes físicos.

Las soluciones conceptuales simples se encuentran con facilidad. Por desgracia, su excesiva saturación en las comunicaciones ocasiona que tengan un desempeño muy deficiente.

Se encontrará con un término que lo utilizaremos de aquí en adelante: "***middleware***", que es un término, que etimológicamente nos da una definición vaga de su propio significado; la definición más enriquecida dice que hace referencia a las capas de software que permiten interactuar los componentes de aplicaciones distribuidas, es decir es el software que hace posible la comunicación entre las capas lógicas para que nos dé cómo resultado una aplicación distribuida. Aunque el middleware adopta varias formas, su función básica es permitir la comunicación entre los procesos.

Uno de los beneficios teóricos de la computación distribuida es la oportunidad que proporciona para dividir el trabajo entre una red de computadoras independientes, en la que cada una trabaja a su propio ritmo.

Pero cualquier servidor tiene una capacidad limitada para procesar las solicitudes concurrentes; cada una de las cuales consume una cantidad de memoria y los recursos del CPU del servidor. Si tratamos de configurar un servidor en un nivel de concurrencia más alto del que permite su capacidad de memoria y de CPU, en realidad hacemos un daño, debido a la saturación de intercambio de memoria y a la conmutación de tareas o contextos que requieren más del procesador.

Si una gran cantidad de estaciones de trabajo cliente continúan enviando solicitudes sincrónicas a un alto nivel más de lo que el servidor puede manejar, el sistema se satura, las solicitudes se regresan y la espiral del desempeño cae a un ritmo acelerado hasta que los usuarios se desesperan y dejan de lado su trabajo. Cuando algunos usuarios dejan de utilizar el sistema, el desempeño mejora para los que continúan.

Cuando se unen los componentes de software con conexiones sincrónicas, pueden ocurrir dos cosas negativas:

Una gran cantidad de demandas de los clientes, pueden crear condiciones de saturación en el servidor.

Cuando cualquier componente de una aplicación distribuida tiene que esperar, el retraso tiende a propagarse a través del sistema. Sin importar cual sea el poder de su procesador o la cantidad de procesadores que tenga, usted no ganara nada si estos quedan ociosos, esperando al componente. Entre más diversa sea la empresa y su ambiente computacional, es más probable que enfrente dichos problemas. Existen cuatro áreas comunes de diversidad en las redes empresariales:

Capacidad Variable. Los componentes pueden trabajar a distintas velocidades.

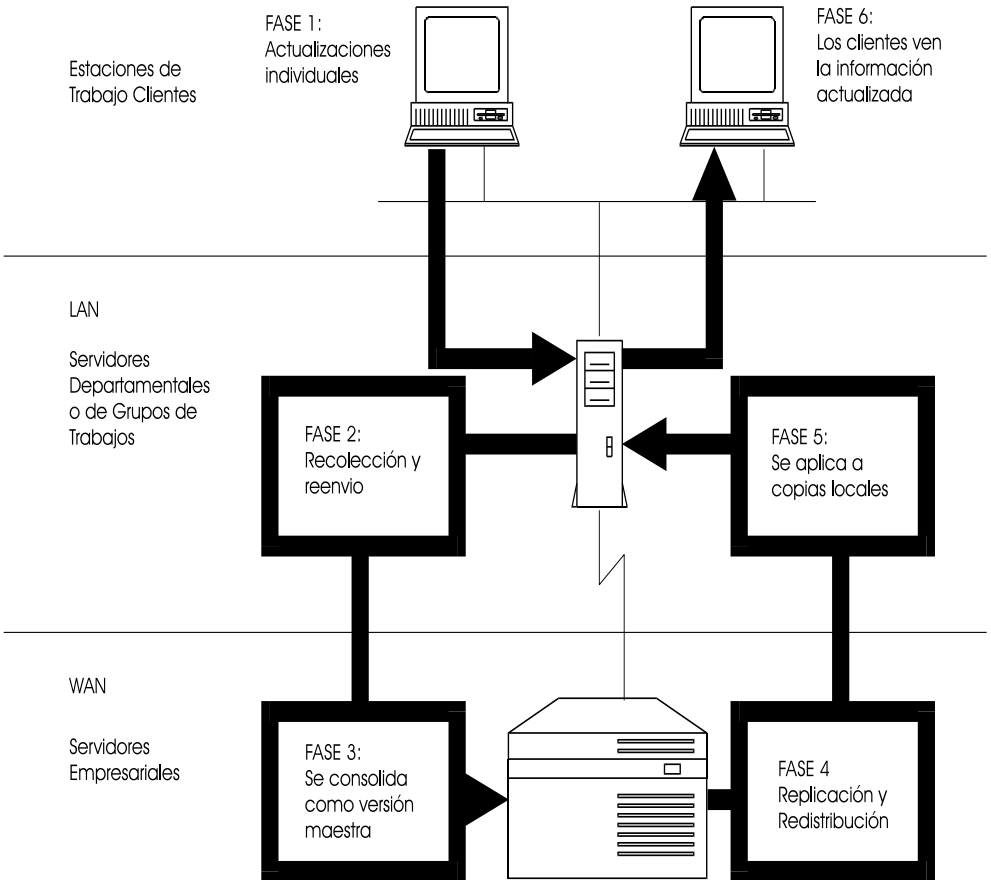
Contención Variable. Los componentes trabajan en diferentes ambientes de procesamiento, de tal forma que algunos pueden sufrir retrasos por compartir recursos con otros trabajos.

Disponibilidad Variable. Es difícil asegurar que los componentes controlados en forma independiente estarán disponibles al mismo tiempo.

Demanda Variable. Una sola Transacción de negocios podría generar demandas desiguales en los componentes. Mientras que una parte de trabajos podría finalizar en segundos, otra tal vez necesite algunos minutos o incluso horas.

Para minimizar el impacto de estas variables en el desempeño de la aplicación, los diseños distribuidos deben estar basados en la comunicación asincrónica empleando filas de mensajes. El uso de las filas para separar a los solicitantes de los servidores elimina saturaciones y permite que cada servidor procese las solicitudes a su propio ritmo con el fin de enfrentar las demandas variables. Al eliminar la saturación de las interrupciones, las filas maximizan el desempeño del servidor y minimizan el desempeño del servidor y minimizan el tiempo que debe esperar un solicitante.

PATRONES DE DISEÑO DISTRIBUIDO



Basándose en los principios de diseño que se analizó, se podrá llegar a algunas conclusiones acerca de los patrones de diseño que trabajarán mejor en las aplicaciones de las empresas.

Debido a que los usuarios distribuidos necesitan compartir información corporativa, para satisfacer las demandas de localidad debemos emplear técnicas de caché y de replicación con el fin de trasladar copias de datos esenciales mas cerca de los usuarios.

Para encontrar un desempeño óptimo se deberá evitar diseños que se requieran actualizaciones remotas o de múltiples sitios con una sola transacción, y sé optará por un enfoque asincrónico de múltiples fases como el que se muestra en la figura anterior.

FASE 1: **Actualización**. Las actualizaciones se graban primero en una copia de la base de datos del nodo local. La copia local de la base de datos actúa como un cache más actual que la versión maestra almacenada en forma centralizada. Si es posible, la versión actual será fragmento de la versión maestra aunque solo contendrá el subconjunto necesario para las operaciones locales típicas.

FASE 2: **Recolección.** Las copias de las actualizaciones locales son reenviadas en forma asincrónica hacia una copia maestra de la información que se encuentra en un sitio central. Existen muchas formas de hacer esto: mensajes de actualización individual, por lotes o si se emplea un software de replicación de DBMS.

FASE 3: **Consolidación.** Las actualizaciones son colocadas en versión maestra de la información posiblemente en lotes según el balance apropiado de la paridad de la información y de la eficiencia de la actualización.

FASE 4: **Redistribución.** Las actualizaciones destinadas a las copias dependientes de los nodos locales, se vuelven a distribuir en forma asincrónica.

FASE 5: **Aplicación.** Las actualizaciones son aplicadas al fragmento de los datos del nodo local.

FASE 6: **Visualización.** Los usuarios ven la información actualizada.

Este estilo de procesamiento algunas veces es llamado replicación bidireccional, aunque no requieran un producto de replicación; puede ser implementado empleando la lógica de aplicación distribuida y middleware de aplicación. Los productos middleware actuales evolucionaran hasta convertirse en agentes de mensajes diseñados para soportar este estilo de procesamiento. El software para procesamiento de transacciones fuera de línea también está apareciendo para soportar clientes móviles que demanden un estilo de procesamiento similar.

El desarrollo de estos algoritmos para la distribución de datos, tanto en redes LAN como WAN, nos dará una mayor visión de la tecnología que podemos alcanzar, y se pondrá en práctica, como se cito anteriormente, muchas de las técnicas de planificación informática y programación. Además se usará para la aplicabilidad y demostración de estos, herramientas que estén a precios bajos, e inclusive técnicas usadas en Internet, y nuevos paquetes de programación para el ya tan popular sistema operativo LINUX, que como se tiene conocimiento es freeware, es decir software de bajo costo, por no decirlo gratuito, y estaremos aplicando tendencias que van a ser utilizadas en el próximo milenio.

Para finalizar, se distinguirá entre sistemas operativos distribuidos, redes distribuidas y bases de datos distribuidas. Un sistema operativo es mucho más general, el que implica tanto hardware y software, distribuyendo los recursos de la máquina, en cambio en una red distribuida se distribuyen los recursos de la red y en una base de datos distribuida, se distribuyen los datos, y esta puede ser aplicada tanto en un sistema operativo distribuido como en una red distribuida.

HIPÓTESIS

Se pueden encontrar alternativas informáticas que permiten soluciones óptimas a problemas de bases de datos distribuidas, en sistemas de pequeña y mediana escala.

METODOLOGÍA

Para cumplir con los objetivos propuestos, se utilizará un método experimental, que posibilitará el análisis y ayudará a tomar la decisión de la mejor solución, en el transcurso del desarrollo del trabajo, ya que aplicaremos las teorías que plantean algunos autores que tratan acerca del tema, en la práctica, utilizando especialmente técnicas de programación y comprobando los resultados.

Soluciones a Problemas de Base de Datos Distribuidas en Sistemas de Pequeña y Mediana Escala

CONTENIDOS

Capítulo I: Introducción.

1.1. Introducción teórica a la distribución de datos.
1.2. Conceptos sobre Bases de Datos Distribuidas.
1.3. Problemas a resolver en el diseño de bases de datos distribuidas.
1.4. Aplicabilidad de las bases de datos distribuidas, él por que.

Capítulo II: Estudio, Análisis y Evaluación de DBMS que poseen la capacidad de distribuir datos.

2.1. Microsoft SQL Server.
2.2. Informix.
2.3. Oracle.
2.4. DB2.

Capítulo III: Desarrollo de Algoritmos para distribución de datos.

3.1. Solucionar el problema de tener varias tablas con la misma estructura en diferentes lugares físicos.
3.2. Solucionar el problema de la integridad de los datos en el momento de realizar una transacción.
3.3. Solucionar el problema de la integridad de la indexación.
3.3. COMMIT y ROLLBACK, COMMIT EN CASCADA.
3.4. Transparencia de los resultados finales al usuario.
3.5. Consultas a la base de datos.
3.5. Espejos.
3.6. Integración de lo antes expuesto.
3.7. Conclusiones y comentarios.

Capítulo IV: Aplicación práctica.

4.1. Sistema Informático para la demostración práctica (consta de por lo menos tres nodos de distribución).
4.2. Pruebas alfa y beta de la aplicación.
4.3. Documentación técnica y de usuario de la aplicación.

Evaluaciones, Conclusiones y Recomendaciones.

CRONOGRAMA DE ACTIVIDADES

Actividades / **Tiempo**	Oct.	Nov.	Dic.	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Agt.	Sept.
Recopilación de Información	X	X	X	X								
Observación de Bases de Datos		X	X	X								
Identificación de Problemas			X	X	X	X	X					
Elaboración de Alternativas					X	X	X					
Seleccionar Alternativas						X	X	X				
Desarrollo de Algoritmos						X	X	X	X			
Evaluación de Resultados									X	X		
Aplicación Práctica								X	X	X	X	
Pruebas de la Aplicación										X	X	
Elaboración de Conclusiones y Recomendaciones												X

Documentación	X	X	X	X	X	X	X	X	X	X	X	X

PRESUPUESTO

***** Precio en USD.

	Disponible	Por adquirir
• 1 PC Server	3000	
• 1 Computadoras	1000	
• 1 Tarjetas de Red		50
• 1 Tarjeta de Módem		120
• Útiles y Suministros		150
• Horas de Internet (6 horas al mes)		360
• Libros y Software		1200
• Gastos en Viajes y Viáticos		100
Imprevistos (15% total)		740
TOTAL		6720

BIBLIOGRAFIA

- **KROENKE, David M.**; Procesamiento de Bases de Datos; Prentice Hall; México 1996.
- **PRESSMAN, Roger S.**; Ingeniería del Software, un enfoque práctico; McGraw Hill; España 1993.
- El Comercio; reportajes tomados desde enero de 1998 hasta la fecha.
- PC Magazine; Editorial Televisa; revistas mensuales desde enero de 1998.
- Documentación en línea de Microsoft Windows NT Server versión 4.0.
- Documentación en línea de Slakware Linux kernel versión 2.0.0 y S.U.S.E. Linux kernel versión 2.0.33.
- Información tomada de Internet, en las siguientes direcciones: www.microsoft.com y www.linux.org.