

**UNIVERSIDAD TÉCNICA DE AMBATO**

**FACULTAD DE INGENIERÍA EN SISTEMAS**

**ANÁLISIS DE LA REPLICACIÓN DE BASES DE  
DATOS RELACIONALES CASO PRACTICO**

**AUTOR: JORGE EDUARDO CHICAIZA RUGEL**

**TUTOR: ING. EDISON ALVAREZ M.**

**Proyecto Informático de Grado, previo a la  
obtención del Título de Ingeniero en sistemas**

**Ambato – Ecuador**

**Abril/2005**

## **AGRADECIMIENTO**

Manifiesto mi más sincero agradecimiento a los directivos de la facultad de Ingeniería en sistemas de la Universidad Técnica de Ambato, a mis maestros que han contribuido en mi preparación profesional y a DIOS mi creador por la vida, valor y fuerza que me ha proporcionado en el transcurso de este tiempo.

A los: Ing. Edison Álvarez M. E Ing. Franklin Mayorga M. Tutor y Asesor respectivamente del proyecto de grado, quienes me brindaron su apoyo incansablemente en el crecimiento de la semilla fértil de la ciencia y el saber, aportando con animo positivo su participación eficaz como sus experiencias válidas para la realización del presente trabajo.

El Autor.

## DEDICATORIA

Este trabajo esta dedicado con mucho cariño para todos y cada una de las personas que más quiero.

A mi adoración, mi reina, mi amor, Mi Hija ANGIE DAYANA, porque tu presencia será el lucero que ilumine el camino de nuestro porvenir.

A mis PADRES queridos, que con su sacrificio y amor supieron enrumbarme en el camino del bien.

A mis HERMANOS que de una u otra manera me brindaron su apoyo incondicional.

A ti EARO.

Y a todas aquellas personas que confiaron en mi, va dedicado este proyecto.

Jorge Eduardo.

## **Introducción**

En los últimos tiempos la tecnología de almacenamiento y administración de datos ha tenido grandes cambios; partiendo desde el manejo de archivos en forma escrita y manual, hasta el uso de almacenamientos en sistemas magnéticos conocidos como bases de datos, cuyo acceso y manipulación se realiza en forma automática.

El manejo de una base de datos no sólo es requerido por una parte de la humanidad, si no también de todas y cada una de las personas, empresas u organizaciones que utilizan información en las tareas diarias que estas realizan tales como: consultas, procesos de recuperaciones, actualizaciones, etc. Debido a que esta información es muy importante, surge, la necesidad de mantener su integridad.

Un sistema de gestión de bases de datos (DBMS) consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a esos datos. La colección de datos, normalmente denominada base de datos, contiene información acerca de una empresa determinada. El objetivo primordial de un DBMS es proporcionar un entorno que sea, a la vez, conveniente y eficiente, para ser utilizado para crear, extraer y almacenar información de la base de datos.

Los sistemas de bases de datos están diseñados para gestionar grandes bloques de información, esta gestión implica tanto la definición de estructuras para el almacenamiento de la información como la provisión de mecanismos para la administración de ésta información. Además los sistemas de bases de datos deben mantener la seguridad de los datos almacenados, cuando se den caídas del sistema o a intentos de accesos no autorizados. Y si los datos van a ser compartidos por varios usuarios debe evitar posibles resultados anómalos.

La importancia de la información en la mayoría de las organizaciones, y por tanto el valor de la base de datos, ha llevado al desarrollo de una gran cantidad de conceptos y técnicas para la gestión eficiente de los datos.

Existen en la actualidad un sin número de RDBMS (Relational Data Base Management System), que permiten definir, manejar y administrar la información, tales como: SQL Server, Oracle, Informix, Sybase, en sus múltiples versiones; de los cuales éste estudio, se centrará en los administradores de bases de datos relacionales SQL Server 7.0 y Oracle8 Server Enterprise Edition, en lo concerniente a la administración de replications o conocido también como las duplicaciones.

Los sistemas Administradores de Bases de Datos SQL Server 7.0 y Oracle8 Server Enterprise Edition, al ser en si administradores permiten mantener, manipular, estructurar, visualizar, modificar, etc., toda la información en distintas formas siendo estas:

- a) La definición y almacenamiento de datos, usando para esto el LDD (Lenguaje de Definición de Datos), que permite la creación de las estructuras de datos: tablas, vistas, índices, etc., en donde se va a guardar la información. y
- b) Todo lo relacionado con la manipulación de la información utilizando para esto el LMD (Lenguaje de Manipulación de Datos), que permitirá realizar todos los procesos relacionados a: consultas, inserciones, actualizaciones, eliminaciones, modificaciones, etc., de los datos almacenados.

**Finalidad del trabajo.**

Este trabajo tiene como finalidad adentrarse en el proceso de replicación, que se realiza en los Administradores de Bases de Datos Relacionales, orientándose explícitamente a visualizar esto en los RDBMS SQL Server 7.0, con un caso práctico y Oracle8 Server Enterprise Edition, para así poder vislumbrar con mayor entendimiento este proceso, también observar: configuraciones, estimaciones de necesidades, ventajas y desventajas en cada uno y, el por qué y para qué del uso de la replicación.

**Antecedentes.**

Como se ha visto en la actualidad todas y cada una de las empresas e instituciones, sean estas grandes o pequeñas necesitan de una forma automática de almacenamiento de información, por el elevado cúmulo en que ésta se presenta, así como por su importancia, y tomando en cuenta estos aspectos se hace evidente la necesidad de utilizar una base de datos y por ende de un sistema que lo administre.

Partiendo de este punto y del auge sustancial de la Computación e Informática se han definido algunos tipos de manipuladores de información entre los que están:

- a) En primera instancia los archivos sean planos, textuales, binarios entre otros, en donde únicamente se almacenaba la información en una estructura simple y aislada sin la presencia directa de relaciones.
- b) Con el pasar del tiempo y el crecimiento continuo de la información se ha hecho necesario el estudio y apareamiento de mejoradas formas de almacenamiento, como lo son las Bases de Datos.
- c) Al manejar bases de datos grandes se presenta entonces la necesidad de implementar mecanismos o procedimientos para administrar éstas, orientados ya al manejo de la información

relacionada es decir, definidos a partir de los modelos de datos relacionales y el uso del álgebra relacional.

Al surgir estos nuevos mecanismos y procedimientos de administración (RDBMs), aparecen también otras características y aspectos que mejoran la manipulación de la información entre estos tenemos los siguientes: que el diseño de un RDBMS permite generar un conjunto de esquemas de relaciones; lo cual almacenará los datos sin la connotada redundancia que se presentaba en los anteriores almacenamientos, y que, a la vez permita recuperarlos rápida y fácilmente. Una técnica utilizada para esto es la que consiste en diseñar esquemas de datos que tengan una "Forma Normal" adecuada.

Dentro de los muchos Administradores de Bases de Datos Relacionales se encuentran:

- a) Microsoft SQL Server 7.0 que funciona en el sistema operativo Windows NT, el cual se constituye en un producto de bases de datos en donde se pueden construir aplicaciones para negocios. Las necesidades y requerimientos de los clientes han llevado a la creación de innovaciones significativas del producto para facilitar, la utilización, escalabilidad, confiabilidad y el almacenamiento de datos.
- b) Oracle8 es el Administrador de Bases de Datos Relacional que hace uso de los recursos informáticos en todas las arquitecturas de hardware, para garantizar su aprovechamiento al máximo en ambientes cargados de información. Es el conjunto de datos que proporciona la capacidad de almacenar y acudir a esto de forma consecuente con un modelo definido, como el relacional. Además es una suite de productos que ofrece una gran variedad de herramientas. Incluyendo cuatro generaciones de desarrollo de aplicaciones, herramientas de reportes así como varios utilitarios. Oracle se ejecuta en computadoras personales (PC), mainframes,

microcomputadoras y computadoras con procesamiento paralelo masivo. Soporta unos 17 idiomas, corre automáticamente en más de 80 arquitecturas de hardware y software distintos, sin tener la necesidad de cambiar una sola línea de código. Esto es porque más del 80% de los códigos internos de Oracle son iguales a los establecidos en todas las plataformas de sistemas operativos.

Estos RDBMS's tienen incorporado la utilidad de replicación usando un ambiente gráfico de manipulación (Enterprise Manager en el SQL Server 7.0 u Oracle8 DBA Studio en Oracle8), esta parte permitirá a la empresa u organización, incorporar una nueva manera de mantener segura su más preciada riqueza (la información), la tarea de replicación se efectúa en distintos sistemas de conectividad como: Distribuidos en sus diferentes capas, remotos, locales, etc.

La replicación, consiste en llevar la información del servidor principal hacia otro servidor (local o remoto), permitiendo esto, obtener respaldos o copias de la información con la finalidad de evitar su pérdida.



## **OBJETIVOS.**

### **General.**

- Analizar la Replicación (duplicación) de datos en los sistemas Administradores de Bases de Datos Relacionales (RDBMS), SQL Server 7.0, y Oracle8 Enterprise Edition, para determinar tanto las ventajas como desventajas que se presentan, mostrando un caso práctico en el SQL Server 7.0. en el que se muestre el manejo y uso del proceso de replicación.

### **Específicos.**

- Conceptualizar el proceso de la replicación de datos en los Administradores de Bases de Datos Relacionales.
- Definir los tipos de replicación de datos que más se utilizan en los Administradores de Bases de Datos Relacionales.
- Profundizar en el conocimiento de los procesos de replicación de datos en los Administradores de Bases de Datos Relacionales: SQL Server 7.0 y Oracle8 Enterprise Edition.
- Presentar un caso práctico en el cual se muestre el proceso de la replicación de datos en el Administrador de Bases de Datos Relacional SQL Server 7.0.

**Documentación de la recopilación de la información.**

Las fuentes de datos o la documentación de donde se extrae la información para este trabajo de estudio, están dadas en:

- Fuentes primarias que consisten en documentos definidos que muestran el uso de herramientas. Y
  
- Fuentes secundarias consistentes en libros, textos, revistas, folletos, periódicos, artículos extraídos del Internet, ayuda de paquetes y otras publicaciones.

Utilizando estas fuentes como base para la realización de la Investigación Bibliográfica – Documental, que se constituye en una búsqueda de información con el propósito explícito de: ampliar, profundizar y analizar el conocimiento producido en las fuentes.

## INDICE GENERAL

Portada	i
Agradecimiento	ii
Dedicatoria	iii
Introducción	iv
Finalidad del trabajo	vi
Antecedentes	vi
Objetivo general	ix
Objetivos específicos	ix
Fuente de datos	x
Índice General	xi
Índice de Gráficos	xv
Índice de Figuras	xv
Índice de Tablas	xvii
CAPÍTULO I	
TEORÍA DE BASES DE DATOS	1
1.1 Introducción	1
1.2 Modelos de datos	1
1.2.1 Modelos Lógicos Basados en Objetos	1
1.2.2 Modelos Lógicos Basados en Registros	2
1.2.3 Modelos Físicos de Datos	3
1.3 Bases de Datos	3
1.4 Sistemas de Gestión de Bases de Datos	4
1.4.1 Evolución histórica	4
1.4.2 Definición	4
1.5 Sistemas Administradores de Bases de Datos Relacionales (RDBMS/Relational Data Base Management System)	5
1.6 Sistemas Administradores de Bases de Datos Orientados a Objetos (OODBMS/Oriented Object Data Base Management System)	11

1.7 Replicación de datos	121
1.7.1 Por qué se quiere replicar	12
1.7.2 Qué se necesita replicar	13
1.7.3 Los sistemas fuentes y destinos	14
1.7.4 Grado de sincronización en la transferencia de datos	14
1.7.5 La necesidad de la replicación múltiple	15
1.7.6 Actualización de los datos replicados	15
1.7.7 Tipos de selecciones, consolidaciones y transformaciones necesarias	16
1.7.8 Requerimientos de confiabilidad y recuperación	16
CAPÍTULO II	
REPLICACIÓN EN EL RDBMS SQL SERVER	18
2.1 Introducción	18
2.2 Instalación	19
2.3 Replicación (Duplicación)	25
2.3.1 Replicación de mezcla	26
2.3.2 Replicación Transaccional (Transactional Replication)	31
2.3.2.1 Agente de Instantáneas (Snapshot Agent)	33
2.3.2.2 Agente Lector del Log (Log Reader Agent)	35
2.3.2.3 Agente de Distribución (Distribution Agent)	36
2.3.2.4 Replicación Transaccional con Partición	37
2.3.3 Replicación de Instantáneas (Snapshot Replication)	38
2.3.3.1 Agente de Instantáneas (Snapshot Agent)	39
2.3.3.2 Agente de Distribución (Distribution Agent)	41
CAPÍTULO III	
REPLICACIÓN EN EL RDBMS ORACLE	43
3.1 Introducción	43
3.2 Instalación	45
3.3 Replicación con ORACLE	49

3.3.1 Replicación de objeto, grupo y sitios	49
3.3.1.1 Sitios de Replicación	51
3.3.2 Replicación Multimaestro	52
3.3.2.1 Usos para la Replicación Multimaestro	53
3.3.2.2 Sitios de fallos	53
3.3.2.3 Distribución de Cargas de la Aplicación	54
3.3.3 Replicación Instantánea	55
3.3.3.1 Instantáneas de solo Lectura	55
3.3.3.2 Instantáneas Actualizables	57
3.3.3.3 Usos de la replicación de Instantáneas	58
3.3.3.4 Distribución de la información	59
3.3.3.5 transporte de la información	60
3.3.3.6 Ambientes desconectados	60
3.3.4 Configuraciones Híbridas multimaestro e Instantáneas	60
3.3.5 Administración de un ambiente de replicación	62
3.3.5.1 Catalogo de Replicación	62
3.3.5.2 Administración de la Replicación API y Requerimientos de la Administración	63
3.3.5.3 Administración de Replicación ORACLE	63
3.3.6 Conflictos de Replicación	64
3.3.7 Opciones Especializadas de Replicación	64
3.3.7.1 Procedimientos de Replicación	64
3.3.7.1.1 Descubrimiento del conflicto en los Procedimientos de Replicación	65
3.3.7.2 Propagación sincrónica (En tiempo-real) de los datos	66
3.3.7.2.1 Conflictos de Replicación en la propagación Sincronizada de Datos	66
3.3.8 Proceso de Replicación	67
Catalogo de replicación	67
Sitios de Replicación	67
Grupo de replicación	67

3.3.8.1 Replicación Básica	67
3.3.8.2 Replicación simétrica (Avanzada)	70

#### CAPÍTULO IV

DESARROLLO DEL CASO PRÁCTICO	72
Paso 1: Creación de la base de datos	72
Paso 2: Creación de tablas	74
Paso 3: Añadir registros (datos) a las tablas	76
Paso 4: Creación de la base de datos de distribución	78
Paso 5: Creación de la publicación	85
Paso 6: Creación de la suscripción de extracción	90

#### CAPÍTULO V

5 CONCLUSIONES Y RECOMENDACIONES	95
5.1 Conclusiones	95
5.2 Recomendaciones	97

#### BIBLIOGRAFÍA

## ÍNDICE DE FIGURAS

Figura 2.1: Proceso de la Replicación Transaccional	33
Figura 2.2: Replicación de Instantáneas	39
Figura 3.1 Grupo Maestro SCOTT_MG conteniendo los mismos objetos de replicación en todos los sitios.	50
Figura 3.2: Grupo de Instantáneas Correspondiendo con un Grupo Maestro	51
Figura 3.3: Tres Sitios Maestros y Un Sitio de instantáneas.	52
Figura 3.4: Sistema de Replicación Multimaestro.	54
Figura 3.5: Replicación Multimaestro Soportada por Múltiples Puntos de Acceso a las actualizaciones.	55
Figura 3.6: Replicación de Instantánea Solo Lectura	56
Figura 3-7: Ilustra un ambiente de replicación usando Instantáneas Actualizables.	57
Figura 3.8: Información de Descarga.	59
Figura 3.9: Ejemplo de una base de datos replicada a múltiples sitios.	59
Figura 3.10: Configuración Híbrida	61

## ÍNDICE DE GRÁFICOS

Gráfico 2.1. Pantalla inicial de la instalación SQL Server 7	20
Gráfico 2.2. Pasos para la configuración de duplicaciones.	27
Gráfico 2.3. Configuración de la base de datos de distribución	28
Gráfico 2.4. Pantalla que permitirá la creación de la publicación	29
Gráfico 2.5. Pantalla que permitirá escoger el tipo de publicación	29
Gráfico 2.6. Pantalla que permitirá la creación de la suscripción de mezcla	30
Gráfico 2.7. Pantalla que determina la planificación del agente de mezcla	30
Gráfico 5.1. Ventana para la creación de una nueva Base de Datos.	72
Gráfico 5.2. Ventana para la configuración de la Base de Datos.	73
Gráfico 5.3. Ubicación de la nueva Base de Datos replicación.	73

Gráfico 5.4. Ubicación del asistente para la creación De tablas.	74
Gráfico 5.5. Muestra las ventanas que permiten configurar tablas.	74
Gráfico 5.6. Muestra las nuevas tablas en la base de datos replicación.	75
Gráfico 5.7. Opción para la creación de un nuevo diagrama.	75
Gráfico 5.8. Ventana para añadir tablas al diagrama.	76
Gráfico 5.9. Opción para la creación de las relaciones entre las tablas.	76
Gráfico 5.10. Menú contextual para la manipulación de los stored procedures.	77
Gráfico 5.11. Ubicación del asistente para la creación de la base de datos de distribución de la replicación.	79
Gráfico 5.12. Ventana inicial del asistente para la configuración de la base de datos de distribución.	79
Gráfico 5.13. Escogitamiento del servidor en donde se configurara la base de datos de distribución.	80
Gráfico 5.14. Ventana de seteos de la publicación y la distribución.	80
Gráfico 5.15. Ventana para proveer información a la configuración personalizada de la base de datos de distribución.	81
Gráfico 5.16. Ventana que permite habilitar al servidor como un publicador.	81
Gráfico 5.17. Ventana que presenta las bases de datos disponibles para ser replicadas se incluye la base de datos “replicación”.	82
Gráfico 5.18. Ventana que permite habilitar al servidor como un suscriptor.	82
Gráfico 5.19. Ventana de finalización de configuración personalizada del suscriptor y la base de datos de distribución	83
Gráfico 5.20 Ventana de finalización con valores por defecto.	83
Gráfico 5.21. Puntos que el asistente configurará.	84
Gráfico 5.22. Ventana de aviso de finalización de configuración.	84
Gráfico 5.23. Ubicación de las nuevos componentes de replicación.	85
Gráfico 5.24.Creación de la publicación.	85
Gráfico 5.25.Ventana de inicio del uso del asistente para crear la publicación.	86



Gráfico 5.26. Tipos de replicación existentes en SQL Server.	86
Gráfico 5.27. Tipos de actualización del suscriptor.	87
Gráfico 5.28. Ubicación de realización de la suscripción.	87
Gráfico 5.29. Ventana que presenta los artículos a replicarse.	88
Gráfico 5.30. Ventana para identificar la publicación.	88
Gráfico 5.31. Ventana para especificar filtro de la replicación.	89
Gráfico 5.32. Ventana de finalización de la publicación.	89
Gráfico 5.33. Ventana que muestra la creación de la publicación.	89
Gráfico 5.34. Ubicación de la publicación en la consola.	90
Gráfico 5.35. Ventana que presenta la ubicación del asistente para crear la publicación de extracción.	90
Gráfico 5.36. Ubicación del publicador para la suscripción.	91
Gráfico 5.37. Escogimiento del servidor donde se manipulará el suscriptor.	91
Gráfico 5.38. Ubicación de la base de datos de suscripción.	92
Gráfico 5.39. Frecuencias de actualización de la suscripción .	92
Gráfico 5.40. Ventana especificar la inicialización de la suscripción.	93
Gráfico 5.41. Servicios que se levantarán automáticamente con la suscripción.	93
Gráfico 5.42. Ventana de finalización del asistente de configuración de suscripción.	94
Gráfico 5.43. Pasos que seguirá el asistente.	94
Gráfico 5.44. Ventana de finalización.	94

## ÍNDICE DE TABLAS

Tabla 2.1. Componentes hardware requeridos	24
Tabla 2.2. Componentes software requeridos	25

## **DECLARACIÓN DE AUTENTICIDAD Y RESPONSABILIDAD**

Yo, Jorge Eduardo Chicaiza Rugel.

Número de cédula de identidad: 160026669-4

Declaro que la investigación enmarcada en el diseño del proyecto de grado es absolutamente original, autentica y personal. En tal virtud, declaro que el contenido, efectos legales y académicos que se desprenden del trabajo en el proyecto de grado son y serán de mi sola y exclusiva responsabilidad legal y académica.

---

Jorge E. Chicaiza R.

## **CAPÍTULO I**

### **TEORÍA DE BASES DE DATOS Y REPLICACIÓN**

#### **1.1 Introducción**

Para adentrarse en el objeto de este proyecto, primeramente se necesitará tener una idea global de lo que es una base de datos, los sistemas de bases de datos, los administradores de bases de datos y una visión general de todo lo que se incluye en el proceso de replicación.

#### **1.2. Modelos de Datos**

Como punto de partida para manejar el conocimiento de las bases de datos es necesario definir el concepto de modelos de datos, que no es más que una colección de herramientas conceptuales para describir datos, relaciones entre ellos, semántica asociada a los datos y restricciones de consistencias. Los diversos modelos de datos que se han propuesto se dividen en tres grupos: modelos lógicos basados en objetos, modelos lógicos basados en registros y modelos físicos de datos.

##### **1.2.1 Modelos Lógicos Basados En Objetos**

Los modelos lógicos basados en objetos se usan para describir datos en los niveles conceptual y de visión. Se caracterizan por el hecho de que proporcionan capacidad de estructuración bastante flexible y permiten especificar restricciones de datos explícitamente. Hay muchos modelos diferentes, y es probable que aparezcan más, entre los que se conocen se tienen:

- El modelo entidad-Relación.
- El modelo orientado a objetos.
- El modelo binario.
- El modelo semántico de datos.

- El modelo infológico.
- El modelo funcional de datos.

### **1.2.2 Modelos Lógicos Basados En Registros**

Los modelos lógicos basados en registros se utilizan para describir datos en los niveles conceptual y físico. A diferencia de los modelos de datos basados en objetos, se usan para especificar la estructura lógica global de la base de datos y para proporcionar una descripción a nivel más alto de la implementación.

Se llaman así porque la base de datos está estructurada en registros de formato fijo de varios tipos. Cada tipo de registro define un número fijo de campos, o atributos, y cada campo normalmente es de longitud fija. El uso de registros de longitud fija simplifica la implementación del nivel físico de la base de datos.

No incluyen un mecanismo para la presentación directa de código en la base de datos. En cambio hay lenguajes separados que se asocian con el modelo para expresar consultas y actualizaciones de la base de datos. Algunos modelos incluyen código ejecutable como parte integrante del mismo modelo de datos. Los tres modelos de datos más ampliamente aceptados en estos están:

- Modelo relacional.- representa los datos y las relaciones entre los datos mediante una colección de tablas, cada una de las cuales tiene un número de columnas con nombres únicos.
- Modelo de red.- los datos se representan mediante colecciones de registros y las relaciones entre los datos se representan mediante enlaces, los cuales pueden verse como punteros.
- Modelo jerárquico.- es similar al modelo de red en el sentido de que los datos y las relaciones entre los datos se representan mediante registros y enlaces, respectivamente, se diferencian en que los

registros están organizados como colecciones de árboles en vez de grafos arbitrarios.

### **1.2.3 Modelos físicos de datos**

Se usan para describir datos de nivel más bajo. A diferencia de los modelos lógicos de datos, hay muy pocos modelos físicos de datos en uso. Los más ampliamente conocidos son:

- Modelo unificador.
- Memoria de elementos.

### **1.3. Bases de Datos.**

Es un conjunto de datos no redundantes, almacenados en un soporte informático organizado de forma independiente de su utilización y accesibles simultáneamente por distintas aplicaciones. Es decir, la diferencia de una base de datos respecto a un sistema de almacenamiento de datos, es que éstos se almacenan de forma que cumplan tres requisitos básicos:

1. No redundancia.- Los datos se almacenan una sola vez, si varias aplicaciones necesitan los mismos datos, no crearán cada una su propia copia, sino que todas accederán a la misma.
2. Independencia.- Los datos se almacenan teniendo en cuenta la estructura inherente a los propios datos y no a la de la aplicación que los crea. Esta forma de trabajar es la que permite que varias aplicaciones puedan utilizar los mismos datos.
3. Concurrencia.- varios usuarios, ejecutando la misma o diferente aplicación, podrán acceder simultáneamente a los datos.

## **1.4. Sistemas de Gestión de Base de Datos**

### **1.4.1 Evolución histórica**

Actualmente los Sistemas de Gestión de Base de Datos constituyen el núcleo fundamental del soporte lógico a los sistemas de información. Desde la aparición de los primeros Sistemas de Gestión de Base de Datos comerciales en la década de los 60 hasta la actualidad, se han sucedido tres generaciones distintas de los Sistemas de Gestión de Base de Datos basados en tres modelos de datos diferentes.

Los tres modelos en los que se ha basado el desarrollo de las bases de datos son el jerárquico, red y relacional. El modelo jerárquico dominó el mercado de los Sistemas de Gestión de Base de Datos hasta mediados de los 80. Durante este mismo período, surgió el modelo red con el que se pretendía sustituir los Sistemas de Gestión de Base de Datos Jerárquicos, lo que no se consiguió. A principio de los 80 el modelo jerárquico comenzó a ser sustituido por una nueva generación de Sistemas de Gestión de Base de Datos basados en el modelo relacional que, en la actualidad dominan totalmente el mercado mundial de almacenamiento de la información.

### **1.4.2 Definición.**

Es el conjunto de programas que permiten definir, manipular y utilizar la información que contienen las bases de datos, realizar todas las tareas de administración necesarias para mantenerlas operativas, mantener su integridad, confidencialidad y seguridad. Una base de datos nunca se accede o manipula directamente sino a través del SGBD considerando a este como la interfase entre el usuario y la base de datos.

El funcionamiento de los Sistemas de Gestión de Base de Datos está muy interrelacionado con el del sistema operativo, especialmente con el sistema de comunicaciones. El Sistema de Gestión de Base de Datos

utilizará las facilidades del sistema de comunicaciones para recibir las peticiones del usuario que puede estar utilizando un terminal remoto y devolver los resultados.

Un Sistema de Gestión de Bases de Datos debe proporcionar un amplio surtido de funcionalidades para cumplir adecuadamente su misión. Normalmente se clasifican en:

- Función de definición.- Permite describir los elementos de datos, sus estructuras, sus interrelaciones y sus validaciones a nivel externo, lógico e interno. Esta función es realizada por una parte del SGBD denominada lenguaje de definición de datos (LDD o DDL- Data Definition Language).
- Función de manipulación.- Permite buscar, añadir, suprimir y modificar los datos de una base de datos. Esta función es realizada por una parte del SGBD denominada lenguaje de manipulación de datos (LMD o DML-Data Manipulation Language).
- Función de utilización.- Incluye otras funcionalidades tales como: modificar la capacidad de los registros, cargar archivos, realizar copias de seguridad, re arranque, protección frente a accesos no autorizados, gestión de la concurrencia, estadísticas de utilización, etc.

### **1.5. Sistemas Administradores de Base de Datos Relacionales (RDBMS / Relational Data Base Management System)**

Basados en el modelo relacional, los datos se describen como relaciones que se suelen representar como tablas bidimensionales, consistentes en filas y columnas. Cada fila (tupla, en terminología relacional) representa una ocurrencia, las columnas (atributos) representan propiedades de las filas. cada tupla se identifica por una clave primaria o identificador.

Esta organización de la información, permite recuperar de forma flexible los datos de una o varias tablas, así como combinar registros de diferentes tablas para formar otras nuevas. No todas las definiciones posibles de tablas son válidas según el modelo relacional. En él deben emplearse diseños normalizados que garantizan que no se producirán anomalías en la actualización de la base de datos. En un diseño normalizado para cada tabla:

- No pueden existir tuplas duplicadas.
- El orden de las tuplas es irrelevante.
- La tabla es plana, es decir, en el cruce de un atributo y una tupla sólo puede haber un valor (el orden de los atributos no es significativo).

De todas formas otras consideraciones, principalmente de rendimiento, llevan en ocasiones a que los diseños que se implanta no estén totalmente normalizados. Hallar el punto de equilibrio entre normalización y rendimiento, es con frecuencia un punto clave para obtener un buen diseño de la base de datos cuando se utilizan los Sistemas Administradores de Base de Datos Relacionales. Se han impuesto llegar a dominar casi totalmente el mercado actual, debiéndose esto principalmente a la flexibilidad y sencillez de manejo, convenientemente se debe destacar la amplia implementación del lenguaje SQL, que se ha convertido en un estándar para el manejo de datos en el modelo relacional, lo que ha supuesto una ventaja adicional para su desarrollo.

No se debe confundir el Sistema Administrador de Base de Datos Relacional con la arquitectura que se elige para su implantación, algunos sólo se pueden implantar en una de las arquitecturas y otros en todas ellas, entre éstas están:

**La arquitectura centralizada.-** Es la más clásica. En ella, el



Sistema Administrador de Base de Datos Relacional está implantado en una sola plataforma u ordenador desde donde se gestiona directamente, de modo centralizado, la totalidad de los recursos. Es la arquitectura de los centros de proceso de datos tradicionales. Se basa en tecnologías sencillas, muy experimentadas y de gran robustez.

**La arquitectura distribuida.-** Aquí el Sistema Administrador de Base de Datos Relacional y la base de datos no están asociados a un determinado ordenador sino a una red, cuyos nodos se reparten las funciones. Una base de datos distribuida es vista por las aplicaciones igual si fuera una centralizada, es el Sistema Administrador de Base de Datos Relacional distribuido el que se encarga de preservar la integridad y coherencia de la base de datos. Sin embargo existe otra definición mucho menos estricta de base de datos distribuida si permite lecturas y modificaciones remotas, independientemente de que éstas sean transparentes o no, para las aplicaciones. Esta definición no es adecuada cuando se desea seleccionar una base de datos realmente distribuida.

Se suele distinguir entre sistemas homogéneos y heterogéneos. Un sistema es homogéneo si el Sistema Administrador de Base de Datos Relacional usado en todas las máquinas es el mismo. Si existe más de un Sistema Administrador de Base de Datos Relacional distinto el sistema se denomina heterogéneo.

La distribución física, espacial o geográfica de la información, puede

aconsejar la utilización de esta arquitectura. Cada vez existen más productos disponible en el mercado aunque no existe estándares.

Un Sistema Administrador de Base de Datos Relacional que soporte una arquitectura de base de datos distribuida, es mucho más complejo que uno para base de datos centralizada y el número de Sistemas Administradores de Base de Datos Relacionales distribuidos disponibles en el mercado es mucho menor. Existen algunos Sistemas Administradores de Base de Datos Relacionales que ofrecen la disponibilidad de implementar una base de datos distribuida, sólo para sistemas homogéneos.

**La arquitectura cliente/servidor** separa las funciones de una aplicación en componentes que establecen diálogos entre sí para intercambiar información, servicios y recursos con el objeto de realizar una tarea común, cada componente puede estar en un ordenador diferente, el proceso que inicie el diálogo o solicita recursos se denomina cliente y suele ser la aplicación que el usuario está ejecutando, y el proceso que responde a las solicitudes se denomina servidor.

Esta arquitectura se basa, al igual que la anterior en varias plataformas interconectadas, una de las cuales actúa como servidor de la BD en la que los datos están físicamente localizados y centraliza las funciones de administración, las plataformas denominados clientes realizan funciones del manejo de las interfaces de usuario, lógica de aplicación, etc.

La arquitectura cliente/servidor no exige requisitos especialmente

complejos a los Sistemas Administradores de Base de Datos Relacionales ya que, aunque estén involucrados varios ordenadores, la base de datos está normalmente centralizada en un ordenador y su mantenimiento es igual de sencillo que una arquitectura centralizada clásica. Para esta arquitectura es importante que el Sistema Administrador de Base de Datos Relacional soporte sistemas de comunicación normalizados, ya que tendrá que recibir peticiones de diversos cliente, operando con máquinas y protocolos distintos.

La instalación de un Sistema Administrador de Base de Datos Relacional en un sistema que está funcionando sin él, normalmente proporciona una amplia serie de ventajas, entre las que se pueden citar las más importantes:

- Eliminan las inconsistencias en los datos. Algo especialmente difícil sin un Sistema Administrador de Base de Datos Relacional cuando los mismos datos se utilizan y actualizan en diferentes procesos.
- Permiten compartir los mismos datos entre diferentes aplicaciones con distintas necesidades. Por ejemplo: aplicaciones transaccionales junto con aplicaciones de soporte a la dirección.
- Se adaptan mejor a la existencia de aplicaciones rápidamente cambiantes. En estos casos con los enfoques tradicionales se puede requerir la conversión de los datos cada vez. Un Sistema Administrador de Base de Datos Relacional proporcionará independencia de los datos respecto a las aplicaciones.

- Ahorran espacio de almacenamiento al no existir redundancia o ser ésta muy baja. También porque muchos Sistema Administrador de Base de Datos Relacional utilizan mecanismos de compresión para almacenar los datos.
- Mejorar la seguridad de los datos, pues normalmente incorporan mecanismos de seguridad en el propio Sistema Administrador de Base de Datos Relacional.
- Permiten la creación de entornos de alta disponibilidad. Los Sistemas Administradores de Base de Datos Relacionales modernos suelen permitir realizar gran parte (a veces todo) el mantenimiento del sistema sin necesidad de parar las aplicaciones. Por tanto, con algunos Sistemas Administradores de Base de Datos Relacionales es posible llegar a disponer de las aplicaciones funcionando ininterrumpidamente.

Por otra parte, si se escoge adecuadamente el Sistema Administrador de Base de Datos Relacional, no suelen presentarse problemas de tipo técnico que se presentan con los sistemas anteriores de almacenamiento de datos, sino que los problemas suelen ser los típicos de cualquier equipo lógico completo:

- La puesta en funcionamiento puede ser larga. Pues antes de obtener los primeros resultados se necesita un período de formación y adaptación variable, según la complejidad.
- Se necesita personal especializado para su mantenimiento. En principio un diseñador de la base de datos y luego un administrador permanente de esta.

## **1.6 Sistemas Administradores de Base de Datos Orientados a Objetos (OODBMS / Oriented Object Data Base Management System).**

Una de las novedades más prometedoras y más desarrolladas comercialmente de los nuevos Sistemas de Gestión de Bases de Datos, son los basados en un nuevo modelo de datos, conocido como modelos orientados a objetos. La orientación a objetos es un paradigma que no se aplica sólo al desarrollo de Sistemas de Gestión de Bases de Datos sino, en general, al desarrollo de sistemas de información.

Aunque no existe una definición universalmente aceptada de lo que es un objeto, la idea fundamental es la integración de dos conceptos que tradicionalmente se han venido tratando de forma separada: datos y procesos. Cada objeto encapsula tanto datos (también llamados atributos) como procesos (o métodos). Los objetos se comunican entre sí mediante mensajes. Cada objeto se percibe por los demás como el encapsulamiento de una serie de servicios que se pueden invocar externamente. De esta forma, el encapsulamiento es una abstracción que permite ocultar a los usuarios la instrumentación del objeto, ofreciéndoles una interfase externa mediante la cuál interactúa con el.

Idealmente, los objetos son modulares independientes entre sí, lo que facilita su comprensión, reutilización y mantenimiento.

Los Sistemas de Gestión de Bases de Datos orientados a objetos ofrecen varias ventajas sobre los sistemas relacionales:

- Manejan más efectivamente tipos de datos complejos como imágenes.
- Son más sencillos de mantener gracias al encapsulamiento.
- Proveen un acceso más sencillo a los datos.

En cuanto a las desventajas:

- El modelo orientado a objetos no está totalmente desarrollado, ni académicamente en cuanto a investigación y desarrollo comercial.
- Aun no dispone de un lenguaje normalizado como SQL, ni otro tipo de estándar.
- No existen apenas instalaciones en funcionamiento y la estabilidad de los proveedores de Sistemas de Gestión de Bases de Datos orientados a objetos es cuestionable.

Se puede apreciar que los problemas de los Sistemas de Gestión de Bases de Datos orientados a objetos, no provienen de debilidades del propio modelo sino de su falta de desarrollo, por lo que es de esperar que se vayan resolviendo paulatinamente. De hecho, ya existen consorcios de suministradores que comienzan a intentar crear estándares. Por otra parte, entre los temas en los que se está trabajando para la próxima versión de SQL (SQL3) se encuentra el soporte de bases de datos orientados a objetos.

También existen varios fabricantes de Sistemas de Gestión de Bases de Datos relacionales que están incorporando, lentamente, capacidades de orientación a los objetos en sus Sistemas de Gestión de Bases de Datos, abriendo así otra vía al desarrollo de los Sistemas de Gestión de Bases de Datos orientados a objetos que parecen muy prometedores en un futuro muy próximo.

## **1.7 Replicación de Datos.**

### **1.7.1 Por qué se quiere replicar.?**

Esta interrogante permite visualizar los puntos por los cuales se utilizará el proceso de replicación así se tiene que sirve para:

- Utilizar sistemas adicionales para consultar datos puede reducir la carga en el sistema fuente y mejorar la respuesta a consultas en los sistemas destinos, también permite el acceso a la información desde dispositivos móviles cuando no están conectados al sistema central.
- Permitir el proceso de transacciones en sistemas distribuidos mientras se mantiene una copia consolidada de datos. El proceso de transacciones y sistemas múltiples puede mejorar el rendimiento y permitir procesamiento cuando uno o más de los sistemas en la red no estén disponibles.
- Mantener un sistema secundario para propósitos de recuperación en caso de que el sistema principal falle.
- Producir una copia seleccionada, consolidada o transformada de los datos para facilitar el análisis de datos o la migración hacia una nueva estructura de bases de datos.
- Mantener una base de datos adicional para propósitos de pruebas de aplicaciones.

### **1.7.2 Qué se necesita replicar?.**

Para cada uno de los puntos listados anteriormente se debe identificar las tablas de la base de datos que se necesita replicar ya sea parcial (algunas columnas o renglones) o completamente; tomando en cuenta la identificación del volumen de datos para cada tabla, considerando incluso otras clases de objetos, procedimientos almacenados, archivos de mensajes los que puedan requerir de la replicación.

Se debe considerar si es posible, el particionamiento de datos en forma lógica para reducir con esto el monto de datos que necesite réplica. Por ejemplo se podría particionar información de clientes en grupos regionales, y entonces replicar hacia cada sistema regional solamente a los

cliente de la región, reduciendo este enfoque en tráfico de comunicación para simplificar el manejo de situaciones donde los datos replicados pueden ser actualizados en el sistema de destino, tal y como están en el sistema fuente.

### **1.7.3 Los sistemas fuentes y destinos.**

Para estructurar de mejor manera el proceso de replicación se puede utilizar un diagrama que muestre cada sistema que sirve como fuente de los datos aplicados con flechas que se dirigen hacia cada sistema que será el destino de los datos aplicados, en este esquema se puede usar, flechas con una sola punta para aplicaciones que se establece en una sola dirección y flechas con dos puntas para replicación múltiple. De este diagrama, se debe crear una lista que muestre los sistemas operativo fuentes y destinos, los sistemas administradores de base de datos, así como lo que se deba replicar entre cada pareja de sistemas.

### **1.7.4 Grado de sincronización en la transferencia de datos.**

Hay que identificar aquellos casos en donde se necesite datos totalmente sincronizados. Esto puede ser requerido por ejemplo si se tiene transacciones que puedan originarse desde múltiples sistemas que tienen acceso a datos distribuidos. Los datos totalmente sincronizados requieren que el sistema de administración de la base de datos soporte transacciones distribuidas y control de compromiso en dos fases, considerando los servicios del DBMS, independientes de los productos y herramientas de replicación de datos.

Para otros casos se puede utilizar la replicación asíncrona, en la cual los cambios sobre el sistema fuente están comprometidos antes de que sean propagados al sistema destino. Con esta replicación se necesitará determinar si se preserva o no cada transacción en el sistema fuente como una transacción del sistema destino. Generalmente si se puede definir un



período de actualizaciones en el sistema destino durante el cual todos los cambios del sistema fuente al destino son realizados, entonces no se necesita preservar transacciones independientes. Un ejemplo de este tipo de período es la replicación completa realizada por una “ replicación instantánea” de un conjunto de tablas del sistema fuente; mientras la instantánea esta siendo tomada y no hay transacciones ejecutándose en el sistema fuente, las tablas replicadas al sistema destino deben ser sincronizadas después de que complete la replicación. En otros casos, se puede necesitar únicamente replicar sólo algunos datos de la fuente para propósito de análisis, a estar con la preocupación de la consistencia entre los datos en el sistema destino.

#### **1.7.5 La necesidad de la replicación múltiple.**

Cuando esta permitido las actualizaciones de datos particulares sólo en un sistema y los replica en copias de solo-lectura, se puede utilizar la replicación unidireccional, también conocida como replicación amo-esclavo. Este trabajo se complica un poco más si la copia de los datos debe ser modificada en más de un sistema. Obviamente entonces, esto requiere de replicación múltiple. Definir esto, constituye la parte fácil, el reto es como resolver o evitar conflictos que pueden ocurrir cuando dos copias son cambiadas a distintos valores entre los intervalos de replicación, es decir saber que valor es el nuevo valor correcto, aquí no hay respuesta simple por lo que es aconsejable evitar la replicación múltiple.

#### **1.7.6 Actualización de los datos replicados**

Con la replicación asíncrona, los destinos están con frecuencia ligeramente desactualizados. Se debe determinar que tan desactualizadas pueden estar las copias: segundos, minutos, días; entre más amplio sea el margen de desactualización permitido, se deberá ser menos demandante

con las soluciones de replicación. Por ejemplo en el caso donde se tiene una pequeña tabla replicada que necesite estar actualizada el mismo día que la tabla fuente. Una copia de la tabla completa, programada para ser enviada durante un período de baja actividad, puede satisfacer adecuadamente el requerimiento. En contraste, una tabla grande que es replicada en un sistema de “espera en caliente”, necesita ser actualizada en unos pocos segundos o minutos y posiblemente requiera que cada transacción en el sistema fuente sea replicada tan pronto como sea posible después de que se ejecute.

#### **1.7.7 Tipos de selecciones, consolidaciones y transformaciones necesarias.**

Muchos usos de la replicación, especialmente para análisis de datos y pruebas, requieren que sólo algunos renglones o columnas específicos sean copiados es decir *seleccionar*, y *consolidar* datos de una variedad de sistemas fuentes en un solo sistema destino de manera que todos los datos puedan ser analizados en conjunto, después de la operación derivada, de los valores agregados o valores duplicados en tablas “desnormalizadas” del sistema destino. La replicación entre diferentes bases de datos o la *conversión* de datos en una forma utilizada por una aplicación antigua hacia una versión más reciente puede requerir del mapeo de datos objetivos en la nueva estructura y la comprensión hacia nuevos tipos de datos o esquemas de codificación. Una respuesta completa a los requerimientos de la formulación de un conjunto de esquemas para el sistema destino así como reglas de papeleo entre los sistemas fuente y destino.

#### **1.7.8 Requerimientos de confiabilidad y recuperación.**

Son preocupaciones reales si: los datos replicados se pierden, o si algunos de los datos son replicados más de una vez en la base de datos destino, es aquí que se deberá decidir que tan esencial puede ser garantizar la entrega “una y sólo una vez”, de los procesos de replicación que están

planeados llevarse a cabo.

Suponiendo que una parte de la replica falla., cuál seria el tiempo que se deba esperar hasta que el sistema destino sea llevado al nivel de actualización?. Como parte de la respuesta a esto se debería establecer algunos lineamientos para actividades de bitácora y para la existencia de pantallas de ayuda para el usuario, creando así ciertos aspectos con los cuales lograr la máxima confiabilidad en el proceso de replicación y por ende el manejo de recuperación de información.

## **CAPITULO II**

### **REPLICACIÓN EN EL SISTEMA ADMINISTRADOR DE BASE DE DATOS RELACIONAL SQL SERVER.**

#### **2.1. Introducción**

Microsoft SQL Server es una base de datos relacional que funciona en el sistema operativo NT. El lenguaje estructurado de consulta SQL (Structured Query Language), es un estándar informático corrientemente utilizado para definir, modificar, gestionar datos y controlar cómo se realizan cambios en la base de datos usando tablas, índice, claves, filas y columnas para almacenar la información.

De la iniciativa del modelo relacional surgió Microsoft SQL Server, originalmente Microsoft compro la licencia de los bloques básicos de construcción de SQL Server a Sybase e hizo el producto disponible para plataformas PC que trabajaban con el sistema operativo OS/2 y, más recientemente, Windows NT. Un esfuerzo conjunto entre Sybase, Ashton-Tate y Microsoft en 1988, dio como resultado una de las primeras bases de datos relacionales que fuera más accesibles a los usuarios finales, justamente como el Dr. Codd había imaginado. Microsoft encabezó el proyecto SQL Server al retiro de Ashton-Tate, cuando SQL Server fue adaptado al sistema operativo NT desde OS/2. Microsoft y Sybase vendieron conjuntamente la base de datos para la plataforma hardware PC hasta la versión 4.21. pero la asociación Microsoft/Sybase se disolvió en 1993. Desde este punto Sybase se concentró en el campo de las microcomputadoras y Microsoft en las computadoras personales.

Microsoft, la computadora personal e Internet están transformando la visión del Dr. Codd sobre la real accesibilidad de datos para los usuarios

finales, algo en lo que otros han fracasado.

Hoy en día, la base de datos relacional es aceptada como método preferido de almacenamiento de información, y muchas empresas cada una con una base de datos relacional distinta, están compitiendo para ser líderes con el uso de SQL. Lo cierto es que el futuro de las bases de datos, y el SQL, está dirigido por las exigencias tanto de los clientes como del mercado, y siendo éstas exigencias las mismas que se tenían hace 30 años: bases de datos más grandes, más rápidas, más fáciles de usar y mucho más accesibles. Dando esto la generación de una presión continua a largo plazo en los encargados de desarrollar las bases de datos relacionales.

Las computadoras personales son las más sencillas de usar que hay en el mercado. La ampliación que Microsoft ha hecho de la interfaz gráfica de usuario en SQL Server 7 para PC, muestra claramente que Microsoft está escuchando al cliente y está centrado en canalizar las exigencias de facilidad de uso y administración.

## **2.2. Instalación**

Para manejar el proceso de instalación se deberá analizar el punto donde se encuentra la manipulación de información y que es lo que se desea hacer, ya que puede darse el caso que únicamente se necesite de una actualización o se necesite una instalación nueva la que permitirá el almacenamiento de datos .

Para dar comienzo a la instalación se insertará el CD—ROM de SQL Server y accediendo al CD—ROM se ejecutará de forma automática el programa setup.exe apareciendo la siguiente pantalla de interfaz gráfica de Usuario de Windows y comenzando así la instalación de SQL Server 7.



**Gráfico 2.1.** Pantalla inicial de la instalación SQL Server 7

SQL Server ofrece tres tipos de instalación:

1. Típica
2. Mínima
3. Personalizada

Estas instalaciones se deben realizar dependiendo de algunas características y circunstancia que adapte la base de datos, pudiéndose seguir los siguientes consejos para escoger el tipo de instalación más adecuado.

Se debe realizar una instalación típica si se dan las siguientes condiciones:

- No hay datos preexistentes que haya que convertir.
- Los protocolos de red son Name Pipes (canales identificados), TCP/IP Sockets (conectores TCP/IP) y multiprotocolo.
- Se utiliza el juego de caracteres predeterminado, el ISO1252.

- Se utiliza el orden de clasificación predeterminado: el orden alfabético, sin diferenciación de mayúsculas y minúsculas.
- Se utiliza la ordenación Unicode predeterminada: general sin diferenciación de mayúsculas y minúsculas.
- Se desea instalar todas las herramientas de gestión de clientes (Client Management, tools).
- Se desean instalar los libros interactivos.
- Desea conservar la ubicación predeterminada del archivo de programa en /MSSQL7.
- Se dispone de 148 MB libres en disco.

Se debe realizar una instalación mínima si las circunstancias son las siguientes:

- Se dispone sólo de un espacio de disco utilizable mínimo de 112 MB.
- No se precisa de los libros interactivos.

Es aconsejable la instalación personalizada cuando se necesita convertir bases de datos SQL Server 6.x a SQL Server 7, o cuando se quiera cambiar alguna de las siguientes opciones de instalación:

- Protocolos de red.
- Juego de caracteres.
- Orden de clasificación.
- Ordenación Unicode.
- Los componentes de servidor que hay que instalar.
- La instalación de la documentación interactiva.
- La ubicación de archivos de programa.
- La ubicación del archivo de datos.
- La cuenta de acceso para SQL Server
- La cuenta de acceso para SQL Server Agent.
- El auto arranque de SQL Server.

Una vez escogido el tipo de instalación que más se adecue a los requerimientos, hay que realizar la preinstalación, ya que siempre es buena idea planificar tanto como sea posible la instalación de cualquier aplicación así como el SQL Server; en primer lugar se verifica que el controlador de la caché de escritura al disco se haya desactivado en el equipo que va a ejecutar SQL Server 7. En caso contrario, se podrá corromper la base de datos en el momento más inesperado. Existen ciertas unidades cuya caché de escritura es compatible y por lo tanto, segura, con SQL Server. Se deberá consultar al suministrador del hardware, explicándole que SQL Server es un sistema de gestión de bases de datos, y que dependen del mecanismo de escritura anticipada para la recuperación, y no puede haber pérdida de páginas. Cada vez hay más cachés de escritura que llevan consigo una propia batería de alimentación, así como otros mecanismos que evitan dicha pérdida. A continuación se expresan los pasos físicos que han de darse antes de comenzar el proceso de instalación:

- Verificar si se dispone de suficiente espacio libre en disco para instalar SQL Server 7.
- Hacer una copia de seguridad de todas las bases de datos de SQL Server 6.x que se tengan.
- Cerrar cualquier otra aplicación o applet que se halle en ejecución en el servidor antes de arrancar el programa de instalación de SQL Server.

Así mismo se deberá reunir cierta información y tomar determinadas decisiones antes de iniciar el programa de instalación. Revisando las opciones de instalación utilizando la siguiente lista:

- Verificar que se dispone del hardware requerido.
- Verificar que se dispone del software preciso.
- Decidir el tipo de instalación que se utilizará.
- Averiguar el nombre del dominio.



- Averiguar el nombre del servidor SQL (tomando el nombre de la computadora Windows NT).
- Crear una cuenta de usuario de Windows NT, o utilizar una ya existente que inicie los servicios de SQL Server (SQL Server, SQL Server Agent, MSDTC, pudiendo utilizar cuentas comunes o separadas).
- Tomar una cuenta de usuario de Windows NT con privilegios de administrador para ejecutar la instalación de SQL Server.
- Determinar el conjunto de caracteres.
- Determinar el orden de clasificación.
- Elegir la ordenación Unicode predeterminada o averiguar cual se debe utilizar.
- Elegir un modo de autenticación para SQL Server (autenticación pura NT o mixta).
- Determinar el nombre primario de usuario SQL Server.
- Averiguar el nombre de la compañía.
- Averiguar el número de serie del paquete.
- Determinar que protocolos de red va ha utilizar.
- Decidir si se van a instalar los libros interactivos.
- Determinar la ubicación en la que residirán los archivos de datos de SQL Server.
- Determinar la ubicación en que se almacenarán los archivos de programa.
- Decidir si se utilizará el auto arranque para SQL Server.

Para una correcta instalación de SQL Server se necesita que tanto el hardware como el software funcionen correctamente tomando en cuenta los requisitos determinados en la siguiente tabla:

Componentes Hardware	Requisitos
Computadora	DEC Alpha AXP y sistemas compatibles. Intel y sistemas compatibles 486/33 Mhz. o superiores. Procesador Pentium o Pentium PRO
Memoria	Mínimo 32 MB de RAM, no se debe escatimar la memoria. Instalar la mayor cantidad de memoria que se pueda mejorará enormemente la velocidad de ejecución y permitirá a la aplicación hacer mejor uso de SQL Server.
Unidad de Disco	Estimando la cantidad de información que se va ha almacenar y las aplicaciones que se va a soportar, pudiendo ser en MB, GB, o TB.
CD-ROM	Para la instalación y manejo de un sin número de nuevas aplicaciones.
Espacio disco instalación mínima	80 MB
Espacio disco instalación típica	185 MB
Espacio disco instalación personalizada	185 MB y más si se está actualizando desde SQL Server 6.x, proveyendo un espacio adicional igual a 1.5 veces el espacio requerido por las bases de datos SQL Server 6.x que no sean del sistema.
Adaptador de red soportado por NT	Sólo se requiere si se pretende utilizar SQL Server en una red más no para una individual.

**Tabla 2.1.** Componentes hardware requeridos

<b>Componente</b> <b>Software</b>	<b>Requisitos</b>
Sistema Operativo	Windows NT Server 4.0 o Workstation. Service Pack 4 o cualquier versión posterior. Windows 95/98, Windows NT Workstation,
Clientes	Small Business Server.
Internet	Internet Explorer 4.01.
Software de Red	Si se está utilizando Banyan VINES o Apple Talk ADSP, necesitará software de red adicional.

**Tabla 2.2.** Componentes software requeridos

Si se planea utilizar una red, SQL Server necesita el nombre del dominio Windows que corresponda a los usuarios de red con conexiones confiables. Cuando SQL Server está estableciendo una cuenta de seguridad, utiliza el nombre del dominio predeterminado como prefijo para el nombre de la cuenta.

### **2.3. Replicación. (Duplicación)**

La replicación es una importante y poderosa tecnología para distribución de datos y procedimientos almacenados, a través de un Enterprise<sup>1</sup>. La tecnología de duplicación en Microsoft® SQL Server™, permite hacer copias de los datos almacenados, mover estas copias a diferentes localizaciones, y sincronizar automáticamente los datos para que todas las copias tengan el mismo valor de datos. La replicación puede ser implementada entre dos bases de datos sobre el mismo servidor o diferentes servidores conectados por LANs, WANs o el Internet.

<sup>1</sup> Enterprise.- (Traducción Empresa) Una aplicación que funcionará de manera que permita realizar el manejo de ciertas utilidades enmarcadas mediante una interfase gráfica.

Esto se consigue estableciendo una copia original de los datos y configurando, a continuación, el servidor SQL para sincronizar en lo sucesivo los datos mediante el desplazamiento de estos de un servidor a otro. Siendo uno el publicador (Publisher) y otro el suscriptor (Subscriber). Solo hay un publicador, pero puede haber varios suscriptores. En SQL Server 7.0 los suscriptores pueden actualizar datos del publicador, en oposición al modelo tradicional, en el que solo el publicador podía modificar los datos. Los suscriptores pueden ser también publicadores.

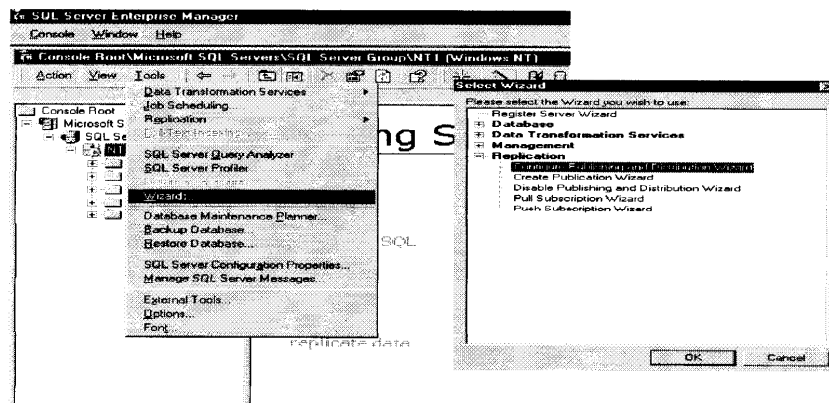
Se han mejorado las capacidades de duplicación de SQL Server 7 al añadirse dos nuevos agentes. Ahora, además de los agentes del lector del registro y de distribución se tienen los agentes de instantáneas y de mezclas, y existen tres opciones importantes a la hora de duplicar datos:

- Replicación de mezcla.
- Replicación de instantáneas.
- Replicación transaccional.

### **2.3.1 Replicación de mezcla**

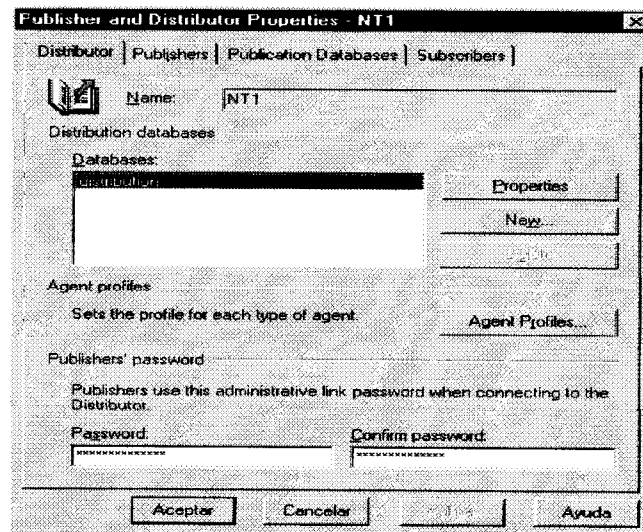
Es una aplicación de SQL Server 7, que es usada cuando lo más importante del sistema es su autonomía. En este proceso, los datos se desplazan solo durante la reconciliación par a par, cuando se realiza una combinación planificada. El proceso de mezcla de datos conecta a los dos servidores y el reconciliador (Reconciler) realiza dicha combinación mientras resuelve cualquier conflicto que se presente. Los valores entrantes se mezclan con los existentes y se resuelven los conflictos de acuerdo con las reglas de reconciliación que pueden ampliarse y configurarse según las preferencias del usuario. De forma alternativa pueden resolver los conflictos basándose en prioridades asignadas previamente. Los datos se trasladan desde el servidor SQL modificado, hasta aquel que necesita ser sincronizado.

En el caso de esta duplicación no se usan, ni el lector del registro ni las transacciones para la reconciliación. En lugar de esto, se usan los datos reales. Se pueden cambiar la columna varias veces, pero solo se usará en la reconciliación la última imagen de fila. La configuración de la duplicación de mezclas se la puede realizar usando el asistente para configurar la duplicación y distribución. (Configure Publishing and Distribution Wizard) representado en el siguiente gráfico



**Gráfico 2.2.** Pasos para la configuración de duplicaciones.

En primer lugar se debe elegir un distribuidor. Este proceso creará una base de datos de distribución, pudiéndose crear esta en el servidor de publicación o usar un servidor diferente para alojar dicha base de datos, después que se haya configurado el servidor como distribuidor

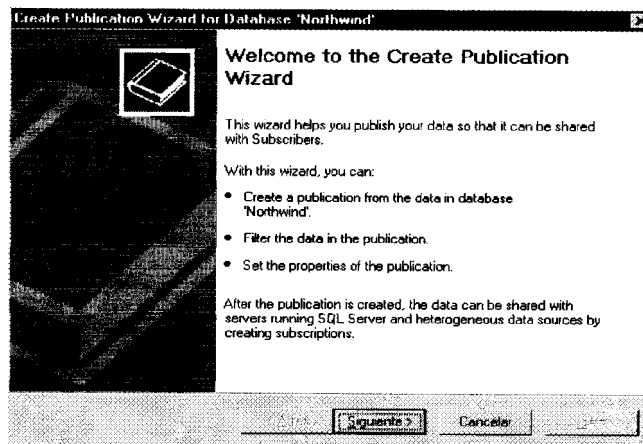


**Gráfico 2.3.** Configuración de la base de datos de distribución

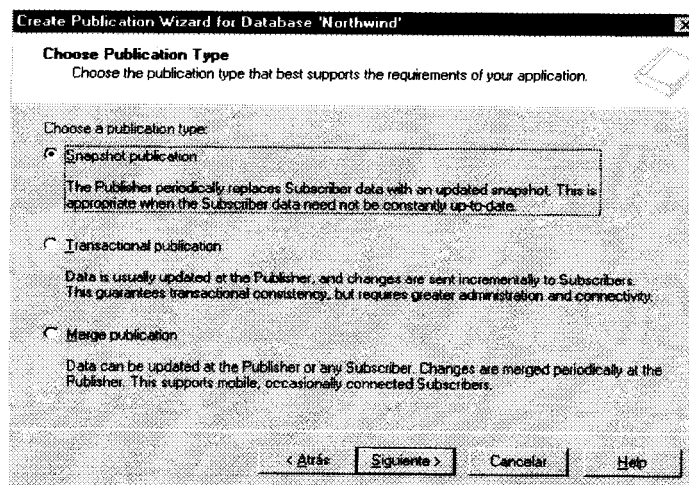
La siguiente pantalla muestra el proceso para activar el suscriptor, con opciones, incluyendo una descripción de cómo los agentes de duplicación acceden al suscriptor.

Por último se muestra una pantalla que indica que la configuración de la base de datos de distribución fue realizada satisfactoriamente.

Al igual que se puede configurar un publicador para la publicación de mezcla, o borrarse o desactivarse un distribuidor y un publicador, se puede configurar la publicación para la duplicación del mismo, utilizando el asistente para crear y administrar una publicación de Enterprise Manager, permitiendo escoger el tipo de publicación: por instantáneo, transaccional o de mezcla.



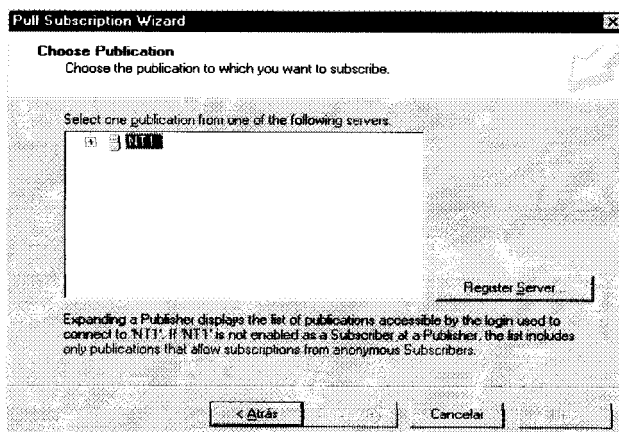
**Gráfico 2.4.** Pantalla que permitirá la creación de la publicación



**Gráfico 2.5.** Pantalla que permitirá escoger el tipo de publicación

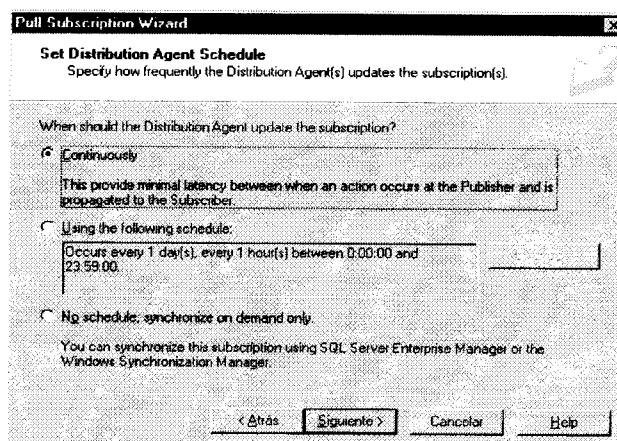
El asistente pedirá que se especifique el nombre y la descripción de la publicación. Si es que todo fue realizado con normalidad, al cerrar las pantallas de este asistente, aparecerá la nueva publicación en el árbol jerárquico de Enterprise Manager, en el elemento Supervisor de duplicación.

La suscripción tiene que crearse ahora como una suscripción de mezcla utilizando también el asistente pero de creación de suscripción de extracción desde el publicador. El primer paso es elegir una publicación para el suscriptor.



**Gráfico 2.6.** Pantalla que permitirá la creación de la suscripción de mezcla

A continuación se deberá escoger el momento de inicialización: al momento de suscripción o espera, para hacer la combinación, hasta que los agentes se ejecuten de forma planificada. Pasando de este punto a la creación de la planificación para el agente de mezcla.



**Gráfico 2.7.** Pantalla que determina la planificación del agente de mezcla



Si se desea cambiar la planificación del agente de mezcla se selecciona el segundo apartado y el botón cambiar en el cual se presenta una planificación definida en donde se puede realizar modificaciones en la programación periódica del trabajo.

La siguiente pantalla que aparece permite establecer la prioridad de suscripción, logrando de esta manera resolver los conflictos en las actualizaciones de múltiples sitios. Si dos suscripciones intentan modificar la misma columna en una fila de datos, la columna mostrará la modificación del servidor que tuviera mayor prioridad, a continuación la última pantalla completa el proceso de creación de una suscripción de extracción que utilizará duplicación de mezclas.

### **2.3.2 Replicación Transaccional (Transactional Replication)**

Se puede usar esta aplicación de Microsoft® SQL Server™ para replicar dos distintos tipos de objetos, tablas y procedimientos almacenados. Seleccionando todo o parte de una tabla para ser replicada como un artículo dentro de una publicación. Similarmente, se puede seleccionar uno o más procedimientos almacenados para ser replicados como un artículo dentro de la misma o una diferente publicación.

La replicación transaccional usa el log de transacciones<sup>2</sup> para capturar cambios que se hicieron a los datos en un artículo. El SQL Server monitoriza sentencias INSERT, UPDATE, DELETE, u otras modificaciones hechas en los datos, y almacena estos cambios en la base de datos de distribución, como nodos o elementos en una cola fiable. Los cambios entonces son enviados al suscriptor y aplicados en el mismo orden.

---

<sup>2</sup> Log de transacciones. Bloque de memoria en donde se guardaran un conjunto de tareas o transacciones realizadas en un procesamiento de datos

Con la replicación transaccional, cualquier elemento de datos dado a un publicador, cambios hechos en el flujo del publicador en forma continua o en el esquema de intervalos de salida para los sitios suscritos. Son usualmente propagados en un muy cercano tiempo real, típicamente con una latencia de segundos. Debiéndose esto a que los cambios de los datos pueden ser hechos en un sitio de publicación (o ambos el suscriptor y el sitio de publicación si hay actualizaciones inmediatas del suscriptor), evitándose los conflictos de actualización. Y logrando garantizar la consistencia transaccional. Por último, todos los sitios de suscripción alcanzan los mismos valores que tiene el publicador, que es donde se hicieron las actualizaciones.

Si el suscriptor necesita recibir cambios de datos en un muy cercano tiempo real, este necesita una conexión permanente a una red de trabajo con el publicador. La replicación transaccional en un ambiente bien instalado puede proveer una muy baja latencia del suscriptor.

El publicador de extracción usualmente recibe los cambios que vienen del publicador dentro de un minuto o más pronto, necesitando de una red con enlace y recursos de procesamiento adecuados.

De cualquier modo, el suscriptor puede también poner estos cambios según como se necesiten. Por ejemplo un representante de ventas de viajes puede ser un suscriptor y poner solo en la noche cambios de incremento para la lista de precio, que es modificado solamente en la oficina de la corporación. El uso de la replicación transaccional para usuarios desconectados pueden ser muy efectivos únicamente para la lectura de datos.

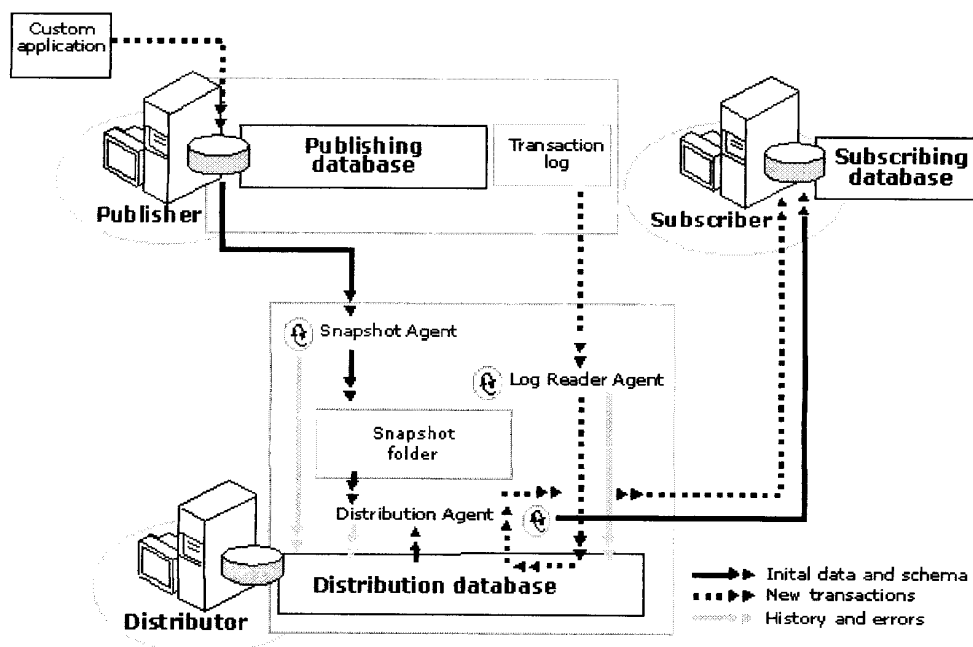


Figura 2.1 Proceso de la replicación transaccional.

Como se ilustra en la figura, la replicación transaccional es llevada a cabo por el Snapshot Agent, Log Reader Agent, y Distribution Agent. El Snapshot Agent prepara los archivos de instantánea conteniendo esquemas y datos, de las tablas de publicación, y almacena los archivos en la carpeta de instantánea sobre el distribuidor. El Log Reader Agent monitoriza el log de transacciones de cada base de datos seteada para la replicación y copia las marcas de las transacciones para la replicación desde el log de transacciones en la base de datos de distribución. El Distribution Agent mueve las transacciones e inicializa los trabajos de instantáneos que tiene en las tablas de la base de datos de distribución para el suscriptor.

### 2.3.2.1 Agente de Instantáneas. (Snapshot Agent)

Antes de recibir cambios incrementales desde un publicador, el nuevo suscriptor necesita un punto de inicio. El suscriptor puede contener tablas

con el mismo esquema y datos similares a los de la tabla en el publicador. Copiando la publicación actual completa desde el publicador hacia el suscriptor haciendo el llamado y aplicando la sincronización inicial. A menos que el suscriptor haya específicamente optado por enviar la sincronización inicial, la replicación transaccional ocurre solamente después de que SQL Server asegure que el suscriptor tiene una instantánea actual del esquema de la tabla y de los datos.

Cuando las instantáneas son distribuidas y aplicadas al suscriptor, solamente esos suscriptores esperan por instantáneas iniciales para ser afectadas. Otros suscriptores a esa publicación (esos que ya están recibiendo inserciones, actualizaciones, borrados, u otras modificaciones para los datos publicados) no son afectados.

Mientras la instantánea es generada se tienen cerraduras de porción (bloqueos), así un llenado, lógico, y un consistente juego de datos es producido. Esto significa, que mientras los datos son requeridos, no pueden ser actualizados durante el tiempo que toma generar la instantánea. Para minimizar cualquier inconveniente en la operación, siempre se debe planear la generación de una instantánea cuando las actualizaciones sean mínimas.

Los procedimientos por los cuales el Agente de la instantánea implementa la instantánea inicial en la replicación transaccional son los mismos procedimientos utilizados en la replicación de instantáneas.

Cuando se instala una Suscripción se puede cargar la instantánea inicial en el suscriptor manualmente en lugar de enviarlo sobre la red de trabajo, refiriéndose esto al manejo de la "sincronización manual". Si la publicación es muy grande, podría ser más eficiente leer la instantánea desde una cinta u otro dispositivo de almacenamiento. Por ejemplo, si se tiene una base de datos de 20 GB, podría ser más fácil y rápido volcar la

base de datos a una cinta, o agrupación de cintas en la localización del Suscriptor, y recargar la base de datos en lugar de enviar los archivos sobre una lenta red de trabajo. Si se opta por leer la instantánea, el procedimiento anterior puede evitar correr el Agente de Instantáneas y SQL Server debería no sincronizar los artículos de publicación con las tablas de destino. SQL Server entonces asume que el Publicador y el Suscriptor ya están sincronizados, e inmediatamente inicia el envío de inserciones, actualizaciones, borrados, u otras modificaciones a los datos publicados.

#### **2.3.2.2 Agente Lector del Log (Log Reader Agent)**

El Agente Lector del Log corre, o continuamente o en concordancia a un plan establecido en el momento en que la publicación es creada. Cuando se ejecuta el Agente Lector del Log primero lee el log de transacciones de la publicación (el mismo log de la base de datos usado para rastrear las transacciones y recobrar durante la operación normal del SQL Server ) e identifica cualquier sentencia de inserción, actualización y borrado, u otra modificación hecha a los datos de transacciones que tienen que ser marcados para la replicación. Luego, el agente copia lotes de esas transacciones a la base de datos de distribución en el Distribuidor. El Agente Lector del Log usa el procedimiento almacenado del sistema `sp_replcmds` para obtener el siguiente juego de comandos marcados para la replicación desde el log. La base de datos de distribución entonces llega a ser la cola de almacenamiento y reenvío desde el cual los cambios son enviados al Suscriptor. Solamente las transacciones entregadas son enviadas a la base de datos de distribución.

Hay una correspondencia uno a uno entre las transacciones en el publicador y la replicación transaccional en la base de datos de distribución.

Una simple transacción almacenada en Msrepl\_transactions<sup>3</sup> puede consistir de muchos comandos y cada comando puede ser creado por unos 500 caracteres Unicode limitados en la tabla Msrepl\_commands. Después los lotes entrantes de transacciones tienen que ser escritos satisfactoriamente en la base de datos de distribución, esto es realizando el proceso de entrega (committed). A continuación de la entrega de cada lote de comandos a el Distribuidor, el Agente Lector del Log llama a sp\_repldone para marcar donde la replicación fue últimamente completada. Finalmente el agente marca que fila en el log de transacciones fue leída para ser truncado. Las filas aún esperando a ser replicadas no son truncadas. El log de transacciones en el Publicador puede ser descargado sin interferir con la replicación, por que solamente las transacciones no marcadas para replicación son purgados.

El Agente Lector del Log corriendo bajo el Agente SQL Server, se puede administrar directamente al usar SQL Server Enterprise Manager. El Agente Lector del Log se ejecuta en el Distribuidor.

### **2.3.2.3 Agente de Distribución (Distribution Agent)**

Los comandos de transacciones son almacenados en la base de datos de distribución hasta que el Agente de Distribución los pone en todos los Suscriptores (o el Agente de Distribución saca los cambios del Suscriptor). La base de datos de distribución es usada solamente por la replicación y no contiene tablas de usuario. Bajo ninguna circunstancia se debe crear otros objetos en la base de datos de distribución. Las acciones que cambian datos en el Publicador fluyen al Suscriptor donde ellos cambian los datos exactamente de la misma manera. Esto asegura que el Suscriptor recibe las transacciones en el mismo orden en que ellos fueron aplicados al

---

<sup>3</sup> Msrepl\_transactions. Procedimiento almacenado que permite manipular las transacciones de la replicación en base a instrucciones codificadas.

Publicador.

Los procedimientos por los que el Agente de Distribución mueve los comandos al Suscriptor son los mismos procedimientos usados en la replicación de instantáneas.

#### **2.3.2.4 Replicación Transaccional con Partición.**

Es mejor utilizar esta duplicación cuando la aplicación necesite que haya coherencia entre los publicadores y los suscriptores unos pocos segundos después de realizada la transacción. El uso de particiones es una gran herramienta para diseñar un escenario de duplicación, siendo mucho mejor particionar la aplicación para evitar conflictos de actualización, que tratar de resolverlos cuando se producen. La resolución de conflictos tiene siempre un perdedor y un ganador, y cada vez que hay un perdedor, hay una transacción que no puede llevarse a cabo. Se puede evitar esta situación haciendo particiones de los datos y usando la duplicación transaccional.

Ésta utiliza un publicador y uno o más suscriptores, el primero envía los datos al segundo en forma de artículos. El registro de transacciones se usa para anotar los cambios en una base de datos de distribución, desde donde se envía al suscriptor en el mismo orden que están en el publicador. Este proceso requiere una conexión de red entre el publicador y el suscriptor, y la duplicación de los datos se hace con una latencia de unos pocos segundos.

El agente de instantáneas, el agente lector del registro y el agente de distribuciones participan en la duplicación transaccional, pudiéndose configurar esta duplicación con suscriptores actualizables y suscriptores de extracción normales o anónimas.

### **2.3.3 Replicación de Instantáneas (Snapshot Replication)**

Como su nombre lo indica, replicación de instantáneas toma una imagen o instantánea, de los esquemas de la base de datos y de los datos en un instante de tiempo específico, siendo la base para la primera sincronización; en el traslado del esquema y de los datos se usa la utilidad BCP. La replicación de instantáneas a diferencia de la replicación transaccional necesita de muy poco uso constante del procesador que ya no requiere el monitoreo continuo de los cambios en los datos sobre el origen del servidor. En lugar de copiar sentencias de Inserción, Actualización Y Borrado (característica de la replicación transaccional), o Modificación de Datos (característico de la replicación de cambios). Los suscriptores son actualizados por un refrescamiento total del juego de datos. Desde la replicación de instantáneas envían todos los datos para el suscriptor en lugar de enviar solo los cambios. Si el artículo es muy grande, puede requerir un sustancial recurso de la red de trabajo para la transmisión. Al decidir si la replicación de instantáneas es apropiada, se debe balancear el tamaño del juego de datos enteros contra la volatilidad de los cambios en los datos.

La replicación de instantáneas es el más simple tipo de replicación, y garantiza la latente consistencia transaccional entre el publicador y el suscriptor. La replicación de instantáneas es a menudo usada por grupos que requieren buscar datos tal como una lista de precios, o que requieren datos para soportar una decisión, donde los datos más actuales no son esenciales. Este tipo de replicación es una buena solución para el suscriptor de solo lectura que no requiere de los datos más actuales. Tal suscriptor puede estar desconectado totalmente si los datos no están actualizados. La replicación de instantáneas es también una buena solución para actualizaciones inmediatas.

Como se ilustra en el figura 2.2, la replicación de instantáneas es



llevada a cabo por el Agente de Instantáneas y el Agente de Distribuciones. El agente de Instantáneas prepara los archivos de instantáneas conteniendo esquemas y datos de las tablas a publicar, almacena los archivos en la carpeta de instantáneas en el distribuidor, y archiva la sincronización de trabajos en la base de datos de distribución en el Distribuidor. El agente de Distribución mueve el trabajo de instantáneas que tuvo en las tablas de la base de datos del distribuidor hacia las tablas de destino al suscriptor. La base de datos de distribución es usada solamente para la replicación y no contiene cualquier tabla de usuario.

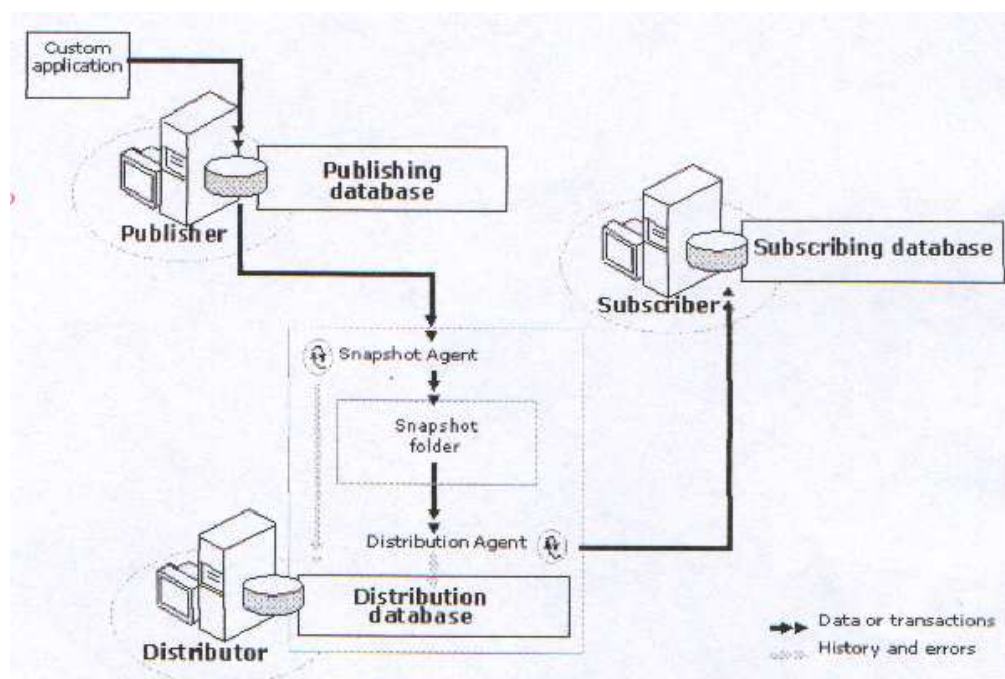


Figura 2.2: Replicación de Instantáneas

### 2.3.3.1 Agente de Instantáneas (Snapshot Agent)

Cada momento en que el agente de instantáneas corre, crea esquemas y archivos de datos para ser enviados al suscriptor. El agente hace esto siguiendo varios pasos tales como:

1. El agente establece una conexión desde el distribuidor hacia el publicador y fija un bloqueo en todas las tablas (artículos) incluidos en la publicación. Este bloqueo asegura una total consistencia de datos en la instantánea. Por que la cerradura prevendrá que el resto de usuarios actualicen las tablas, el Agente de Instantáneas debe ser puesto en un horario de ejecución durante la no actividad de la base de datos.
2. El agente establece una conexión desde la espalda del Publicador al Distribuidor y escribe una copia de los esquemas de la tabla por cada artículo a un archivo *.sch* en el distribuidor. Este archivo es almacenado en un subdirectorio del directorio de trabajo de la base de datos de distribución. Si se requiere incluir índices y declaraciones de integridad referencial, el agente crea un script de salida seleccionando índices para un archivo *.idx* en el distribuidor.
3. El agente toma una “instantánea” de los datos de la tabla de publicación en el Publicador y escribe los datos a un archivo en el Distribuidor. Los archivos son almacenados en un subdirectorio del directorio de trabajo de la base de datos de distribución. Si todos los Suscriptores fueron determinados como parte de Microsoft SQL Server™ las instantáneas son almacenadas como un archivo nativo bull copy program (*.bcp*). si uno o más Suscriptores están en un origen heterogéneo de datos la instantánea es almacenada como un archivo modo caracter (*.txt*). los archivos *.sch* y *.bcp* son los juegos de sincronización que representan la tabla como un simple punto en el tiempo. Hay un juego de sincronización por cada artículo dentro de una publicación.
4. El agente añade filas a las tablas *Msrepl\_commands* y *Msrepl\_transactions* en la base de datos de distribución. Las entradas en la tabla *Msrepl\_commands* son comandos indicando la localización de los juegos de sincronización (archivos *.sch* y *.bcp*) y ordenes de referencia para cualquier script específico de pre creación. Las

entradas en la tabla Msrepl\_transactions son comandos referenciando la tarea de sincronización del Suscriptor.

Finalmente el agente descarga los bloqueos en cada tabla de publicación y finaliza escribiendo en el archivo log history.

### **2.3.3.2 Agente De Distribución.(Distribution Agent)**

Cada momento en que el Agente de Distribución se ejecuta, mueve los esquemas y datos al suscriptor. El agente de distribución realiza esto siguiendo los siguientes pasos:

1. El agente establece una conexión desde el servidor donde el agente localiza al distribuidor. Para poner suscripciones, el Agente de Distribución es localizado en el Distribuidor, pero para sacar suscripciones el Agente de distribución esta localizado en el Suscriptor.
2. El agente examina las tablas Msrepl\_commands y Msrepl\_transactions en la base de datos de distribución en el Distribuidor. Lee la localización de los juegos de sincronización desde la primera tabla y los comandos de sincronización del Suscriptor desde esas tablas.
3. El agente aplica los esquemas y comandos para la base de datos de suscripción. Sí el suscriptor no es una base de datos SQL Server, el agente convierte los tipos de datos como sean necesarios. Todos los artículos de una publicación son sincronizados, preservando la transaccionalidad y la integridad referencial entre las tablas subyacentes.

Quando se manipula un gran número de Suscriptores, corriendo el Agente de Distribución en la computadora Suscriptora (esto es como sacar suscripciones) se puede salvar recursos valiosos de procesamiento en el Distribuidor.

Se pueden aplicar las instantáneas, o cuando la suscripción es creada o según un horario definido en el momento que se creó la publicación. Cuando se llega al horario solamente esos Suscriptores quienes no se sincronizaron después del último evento de sincronización del horario ocurrido (por eso, no todos los suscriptores a esa publicación) obtienen sincronización.

Esto minimiza el efecto en el Publicador por que la sincronización automática de las bases de datos o tablas iguales individuales requieren incrementar el sistema de movimiento de las cabezas, un beneficio del horario de sincronización automática para los pequeños intervalos frecuentes es que permiten la sincronización de la instantánea para ser planificada en un período de baja actividad en el Publicador.

## CAPÍTULO III

### REPLICACIÓN EN EL SISTEMA ADMINISTRADOR DE BASE DE DATOS RELACIONAL ORACLE

#### 3.1 Introducción

Es un administrador de base de datos objeto relacional que hace uso de los recursos del sistema informático en todas las arquitecturas de hardware, para garantizar su aprovechamiento al máximo en ambientes cargados de información.

Es el conjunto de datos que proporciona la capacidad de almacenar y acudir a estos de forma consecuente con un modelo definido como relacional. Además es una suite de productos que ofrece una gran variedad de herramientas, es el ,mayor y más usado Sistema Manejador de Base de Datos Relacional (RDBMS) en el mundo. La corporación Oracle ofrece este RDBMS como un producto incorporado a la línea de producción, incluyendo también cuatro generaciones de desarrollo de aplicación, herramientas de reportes y utilitarios.

Oracle corre en computadoras personales (PC), microcomputadoras, mainframes y computadoras con procesamiento paralelo masivo. Soporta unos 17 idiomas, corre automáticamente en más de 80 arquitecturas de hardware y software distinto sin tener la necesidad de cambiar una sola línea de código. Siendo esto porque más del 80% de los códigos internos de Oracle son iguales a los establecidos en todas la plataformas de sistemas operativos.

El administrador de base de datos ORACLE, surgió a final de los años 70 y principios de los años 80, George Koch y su equipo fue el primero en llegar al terreno de Oracle en 1980, durante un proceso de evaluación del sistema de gestión de base de datos para una importante aplicación comercial que George estaba diseñando y construyendo. Cuando terminó, la evaluación fue descrita en Computer World como el estudio más severo de los Sistemas Administradores de Base de Datos que se había hecho nunca. La empresa ORACLE conocido en aquel entonces como Relational Software, tenía poco más de 25 empleados y unos pocos clientes importantes, con el termino del estudio el Sistema Administrador de Base de Datos Oracle era técnicamente el mejor producto del mercado, siendo estas declaraciones hechas en una época en la que muy poca gente conocía el significado del término "Relacional", y los que lo conocían no tenían muchas cosas que decir de él.

La compañía Oracle Corporation estaba trabajando entonces para perfeccionar su joven producto, para comprender los tipos de características y funcionalidades que podría hacerlo útil y productivo en el mundo de los negocios. El esfuerzo contribuyo a su refinamiento, y la presentación de algunas características tales como las salidas de SQL\*FORMS.

Oracle ha presentado cuatro generaciones para el desarrollo de aplicaciones:

- Oracle5 y Oracle6: fueron las dos primeras versiones , quedando aun rezagadas por las versiones sucesoras.
- Oracle7: La base de datos relacional componentes de Oracle Universal Server. Posee además las versiones 7.1; 7.1.2; 7.1.3.
- Oracle7 Parallel: Ofrece a los usuarios un método seguro y administrable para incrementar la performance de sus bases de datos existentes introduciendo operaciones en paralelo y sincrónicas dentro de sus ambientes informáticos.

- Oracle8: Incluye mejoras de rendimiento y de utilización de recursos. Independiente de que se necesite dar soporte a decenas de miles de usuarios y cientos de Terabytes de datos, o se disponga de un sistema mucho más pequeño, pero igualmente crítico, todos se benefician del rendimiento de Oracle8, soporta aplicaciones de procesamiento de transacciones on line (OLTP) y de datawarehouse<sup>4</sup> mayores y más exigentes.

### 3.2. Instalación.

Para comenzar con el proceso de instalación de Oracle en Windows NT, se debe realizar las tareas de preinstalación con las cuales se ayudará a completar la instalación satisfactoriamente en el primer intento, siendo estas:

a) Elección de un Oracle Server apropiado .- Oracle 8 Server esta disponible para Windows NT en tres paquetes diferentes. Esto es para cubrir el rango entero, desde los autónomas a los de nivel empresarial.

El paquete de nivel de entrada *Oracle 8 Personal Edition*, es útil para las máquinas de Windows NT autónomas, Como Windows NT rara vez se utiliza en un entorno que no sea de red, este paquete es útil principalmente para el desarrollo de aplicaciones, prueba y entrenamiento. Los desarrolladores pueden mantener sus propias imágenes del software de Oracle8 y, mantener sus propias bases de datos. Las herramientas de entrenamiento basadas en computadoras (CBT) se pueden utilizar también convenientemente en este entorno.

El paquete *Oracle8* ( conocido también como edición Estándar), proporciona la funcionalidad completa del Oracle8 Server, menos las

---

<sup>4</sup> Datawarehouse.- Palabra usada para describir a un gran almacén de bases de datos, es decir un repositorio de información con una muy alta capacidad.

opciones añadidas y los cartuchos. Este paquete se puede utilizar para los sitios que desean ejecutar aplicaciones Cliente/servidor ya que se incluye el software Net8 Server y el software cliente. Sin embargo, algunas herramientas de administración no están disponibles con este paquete.

*Oracle8 Enterprise Edition* es un conjunto de productos exhaustivos que incluyen los añadidos y cartuchos. Las herramientas de administración también están disponibles con esta edición.

b) Asegurar la disponibilidad de recursos.-Se debe determinar las necesidades de hardware y software de los productos que se desea instalar desde la guía de instalación. Como recomendaciones generales para la instalación de Oracle8 Server tenemos:

- Un PC basado en un procesador Pentium o superior.
- Microsoft Windows NT 4.0 o superior con Tecnología NT.
- 48 Mb de RAM o más.
- 500 Mb de espacio libre en el disco duro.
- Dispositivo lector de CD-ROM.

c) Reunir una lista de productos para la instalación.- Oracle proporciona distintos paquetes de productos para Windows NT. Se debe asegurar que se reúna una lista de productos que se desea instalar y que se tiene el dispositivo necesario. Se sugiere que primero se instale el Oracle8 Server y a continuación el Web Aplicación Server, después de esto se puede instalar las herramientas de desarrollo que se desee en cualquier orden. El Oracle Installer llevará a cabo automáticamente una comprobación de las condiciones esenciales para el producto que intenta instalar y copiará todos los archivos necesarios.

Elección entre autónoma, Cliente/Servidor y NCA.- se debería tener una estrategia para desarrollar las aplicaciones en los sitios, las aplicaciones



autónomas son muy raras hoy en el mundo de las computadoras en red, sin embargo puedes ser muy útil para pruebas y desarrollo. Si se utilizan aplicaciones Cliente /Servidor se necesitará instalar el software Net8 Server en el servidor y el Software Net8 Client en todos los clientes. Si se piensa utilizar la arquitectura de tres niveles, se debe pensar en instalar un servidor Web.

d) Listar el software necesario para la conectividad.- Si se piensa desarrollar las aplicaciones de ORACLE en una arquitectura Cliente/servidor, se debe instalar el Software Net8 en el servidor y los clientes para disponer de conectividad. Si se piensa utilizar las presentaciones que no son de Oracle como Power Builder y Visual Basic con un ORACLE Server, se debe utilizar la configuración del ODBC (Conectividad Abierta a bases de datos). Si se desea utilizar una aplicación ORACLE con una base de datos que no sea ORACLE se deberá instalar el ORACLE Open Client Adapter. Se puede configurar ODBC utilizando el applet configuración de ODBC que se encuentra en el panel de control. También se puede acceder al administrador ODBC haciendo clic en Inicio/Programa/Oracle para Windows NT/Microsoft ODBC Administrador.

e) Verificar la integridad del sistema de archivos.- Se debería tomar los minutos necesarios para verificar la integridad del sistema de archivos. Se puede hacer esto utilizando el explorador de Windows NT o una herramienta de terceras partes como: Norton Disk Doctor. Se debe Iniciar el explorador de Windows NT y seleccionar el disco lógico que se desee comprobar. Se debe seleccionar Archivo/Propiedades y hacer clic en la pestaña de herramientas, luego hacer clic en el botón comprobar ahora y seguir las instrucciones del asistente hasta que termine la comprobación. Si se tiene una utilidad de compresión de disco instalada, se sugiere que sea desactivada en el disco donde esta instalada la base de datos de ORACLE éste calcula el espacio de disco con algoritmos internos. Si no se desactiva

la utilidad de compresión de disco se puede corromper la base de datos, puesto que estos algoritmos pueden fallar con la compresión de disco.

Si se instala software de Oracle en otros discos distintos de donde se creara la base de datos se puede tener activada la compresión de disco, sin embargo, no se debería comprimir el disco en el que se creará la base de datos.

### **Proceso de instalación de Oracle.**

Oracle Installer se utiliza para instalar todos los productos. El instalador lleva a cabo las tres tareas siguientes:

- Comprueba dependencias.
- Copia los archivos necesarios.
- Configura el sistema para su uso.

La primera tarea del instalador es asegurarse de que dispone de suficiente espacio de disco duro para instalar los productos deseados. Después de esto se realiza comprobaciones de dependencia. Una gran parte de instrucciones esta escrito en código compatible para los productos de Oracle y éste se pone a disposición en forma de componentes como: *Required Support Files* y *GUI Common Files*.

El instalador actualiza automáticamente cualquier componente que necesita actualizarse. Después de estas comprobaciones se copian los archivos necesarios en directorios adecuados en la carpeta destino seleccionada. Todos los ejecutables de Oracle y las bibliotecas dinámicas de enlace (DLLs) se colocan en la carpeta **\Orant\bin**, mientras que otros archivos del producto se colocan en sus correspondientes carpetas de productos. Por ultimo, el instalador crea los grupos de programas y accesos directos necesarios. También se realiza en este momento las modificaciones necesarias al registro. Se mantiene una lista de componentes instalados de

Oracle en la maquina en un archivo llamado **nt.rgs** en la carpeta **\orant\orainst.**

### **3.3. Replicación Con ORACLE**

En el Sistema Administrador de Base de Datos ORACLE la replicación es el proceso de copiar y mantener bancos de objetos de base de datos en múltiples bancos de base de datos realizados en un sistema de base de datos distribuidos. Los cambios aplicados en un sitio son capturados y almacenados localmente antes de iniciar el proceso y puestos en cada una de las aplicaciones remotas. La replicación provee a los usuarios un rápido acceso local a datos compartidos, y protege la disponibilidad de aplicaciones pero permitiendo manipular opciones de acceso de datos alternativos. Siempre si un sitio no esta disponible, los usuarios pueden continuar consultando o actualizar la actual localización.

#### **3.3.1 Replicación de objeto, grupos y sitios.**

Una replicación de objetos, es un objeto de base de datos existiendo sobre múltiples servidores en un sistema de base de datos distribuida. Las facilidades de ORACLE le habilita para replicar tablas y soportar objetos tal como vistas, disparadores a base de datos, paquetes, índices, y sinónimos. Como se visualiza en la figura 3.1.

En un medio ambiente de replicación, los administradores ORACLE de replicación de objetos usan los grupos de replicación. Para organizar objetos de base datos relacionados con un grupo de replicación, esto facilita administrar muchos objetos juntos. Típicamente se crea y usa un grupo de replicación para organizar esquemas de objetos necesarios para soportar una aplicación particular de base de datos. Esto no quiere decir que los grupos de replicación y esquemas pueden corresponder el uno con el otro. Los objetos en grupo de replicación puede originarse desde diversos esquema de base de datos y un esquema puede contener objetos que son

miembros de diferentes grupos de replicación. La restricción es que un objeto de replicación puede ser un miembro de solo un grupo.

En un medio ambiente de replicación multimaestro (multimaster), los grupos de replicación son llamados grupos maestros (MASTER GROUPS), correspondiendo un grupo maestro a diferentes sitios el que debe contener los mismos servicios de objetos de replicación.

Representado en la siguiente figura

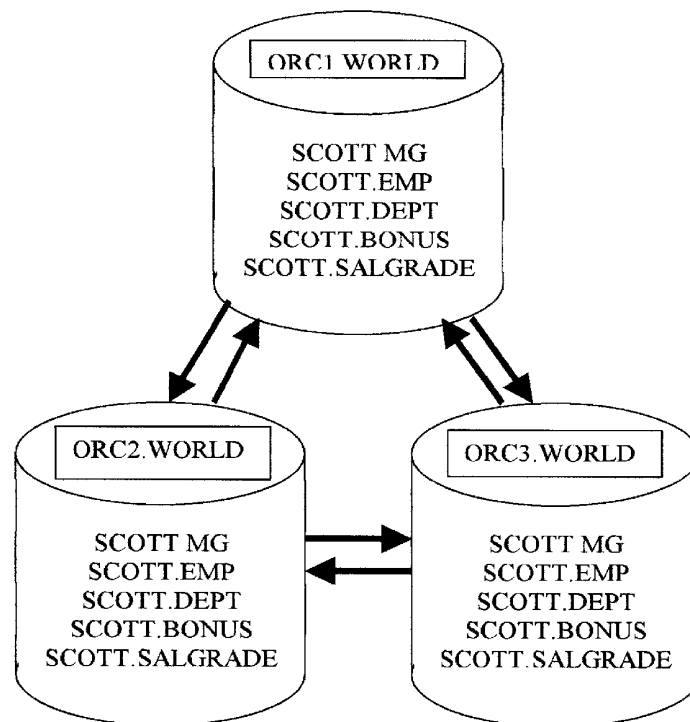


Figura 3.1 Grupo Maestro SCOTT\_MG conteniendo los mismos objetos de replicación en todos los sitios.

En un sitio de instantáneas (snapshot), la organización es mantenida usando un grupo de instantáneas, éste mantiene una copia parcial o completa de los objetos en el grupo maestro. Esto se presenta en la siguiente figura:

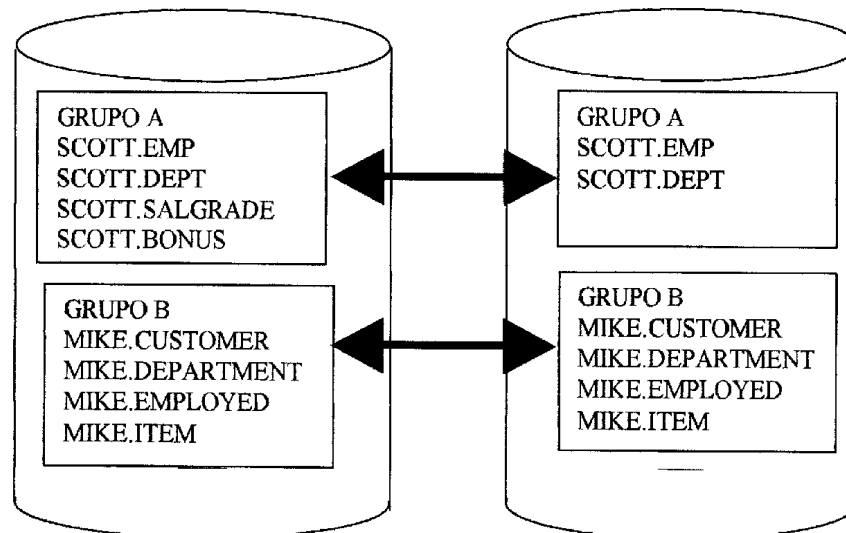


Figura 3.2: Grupo de Instantáneas Correspondiendo con un Grupo Maestro

### 3.3.1.1 Sitios de Replicación

Un grupo de replicación puede existir en múltiples sitios de replicación. Los ambientes de Replicación soportan dos tipos básicos de sitios: sitios maestros y sitios instantáneos.

Un sitio maestro mantiene una copia completa de todos los objetos en un grupo de replicación. Todo sitio maestro en un ambiente de replicación multimaestro se comunica directamente el uno con el otro para propagar datos y esquemas cambiando en el grupo de replicación. Un grupo de replicación en un sitio maestro es más específicamente referido como un grupo maestro. Adicionalmente, cada grupo maestro tiene uno y solo un sitio maestro de definición (por ejemplo, ORC1.WORLD en la Figura 3-1 puede ser el sitio maestro de la definición). Un grupo de replicación del sitio maestro de definición, es un sitio maestro que sirve como el punto de control

para administrar el grupo de replicación y los objetos en el grupo.

Un sitio instantáneo soporta solo leer y actualizar instantáneas de los datos de la tabla a un sitio maestro asociado. Las instantáneas de la tabla de un sitio instantáneo pueden contener todo o un subconjunto de los datos de la tabla dentro de un grupo de replicación. De cualquier modo, éstos deben ser instantáneas simples con una correspondencia uno-a-uno para tablas del sitio maestro. Por ejemplo un sitio de instantáneas contendría instantáneas para solo seleccionar tablas en un grupo de replicación. Y una instantánea particular puede hacer solo selecciones de porción de una cierta tabla reproducida. Un grupo de replicación en un sitio de instantáneas es más específicamente referida como un grupo de instantáneas. Un grupo de instantáneas puede contener también otros objetos de replicación.

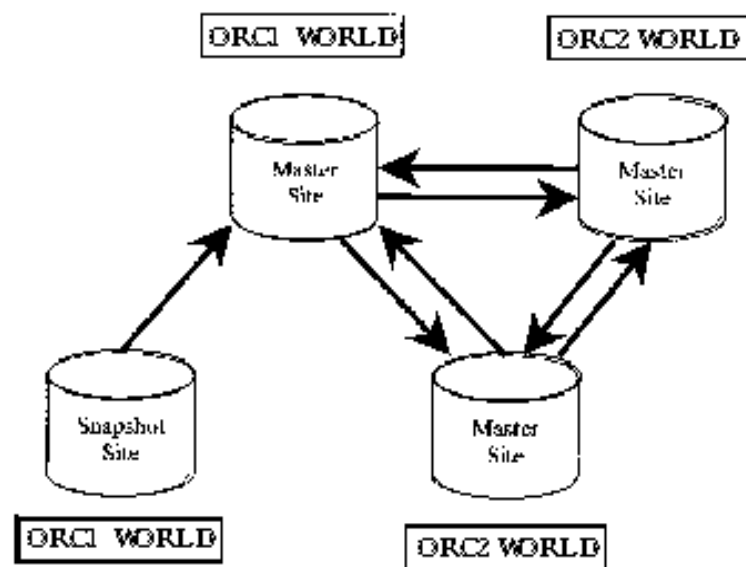


Figura 3.3: Tres Sitios Maestros y un Sitio de Instantáneas

### 3.3.2 Replicación Multimaestro

La replicación multimaestro de ORACLE permite manejar sitios múltiples, acciones como: pares iguales, manejar grupos de reproducción de

objetos del banco de datos. Son aplicaciones que pueden poner al día cualquier tabla reproducida en cualquier sitio de una configuración multimaestro. En la figura 3-4 se ilustra un sistema de replicación multimaestro.

Los servidores de bases de datos ORACLE operan como sitio maestros en un ambiente multimaestro, automáticamente trabaja para converger los datos de todas las réplicas de las tablas, y asegura consistencia de la transacción global e integridad de los datos.

### **3.3.2.1 Usos de la Replicación Multimaestro**

La replicación multimaestro es útil para muchos tipos de sistemas de aplicación con requisitos especiales. A continuación se describen algunos de los usos para la replicación multimaestro:

#### **3.3.2.1.1 Sitios de fallos**

La replicación multimaestro puede ser útil para proteger la disponibilidad de una misión crítica de la base de datos. Por ejemplo un ambiente de replicación multimaestro puede reproducir todos los datos en su base de datos para establecer un sitio de fallos debiendo el sitio primario llegar a no estar disponible, debido al sistema o caídas de la red. En contraste con los rasgos de espera de la base de datos ORACLE, tal como un sitio de fallos puede también servir como una base de datos totalmente funcional para soportar aplicaciones de acceso cuando el sitio primario este operando concurrentemente.

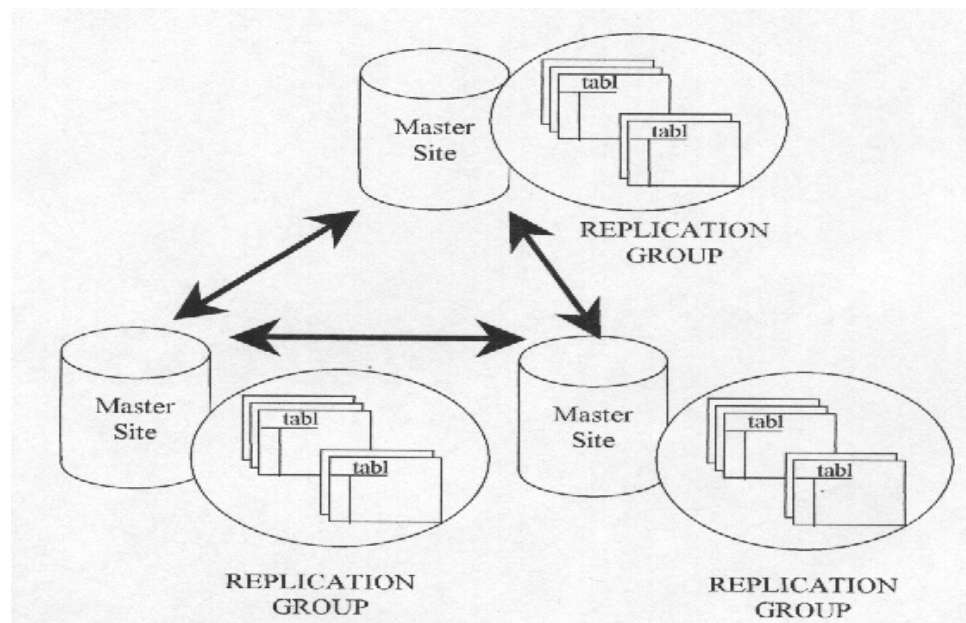


Figura 3.4: Sistema de Replicación Multimaestro.

### 3.3.2.1.2 Distribución de Cargas de la Aplicación

La replicación multimaestro es útil para aplicaciones de procesamiento de transacciones que requieren múltiples puntos de acceso a la información de base de datos para el propósito de distribuir una aplicación de carga pesada, asegurando disponibilidad continua, o proveyendo más localizadores de acceso a datos.

Transacciones que tienen requerimientos de cargas de aplicación de distribución, normalmente incluyen clientes de servicios orientado a aplicaciones. La distribución de cargas de Aplicación puede también ser usado para actualizaciones de instantáneas



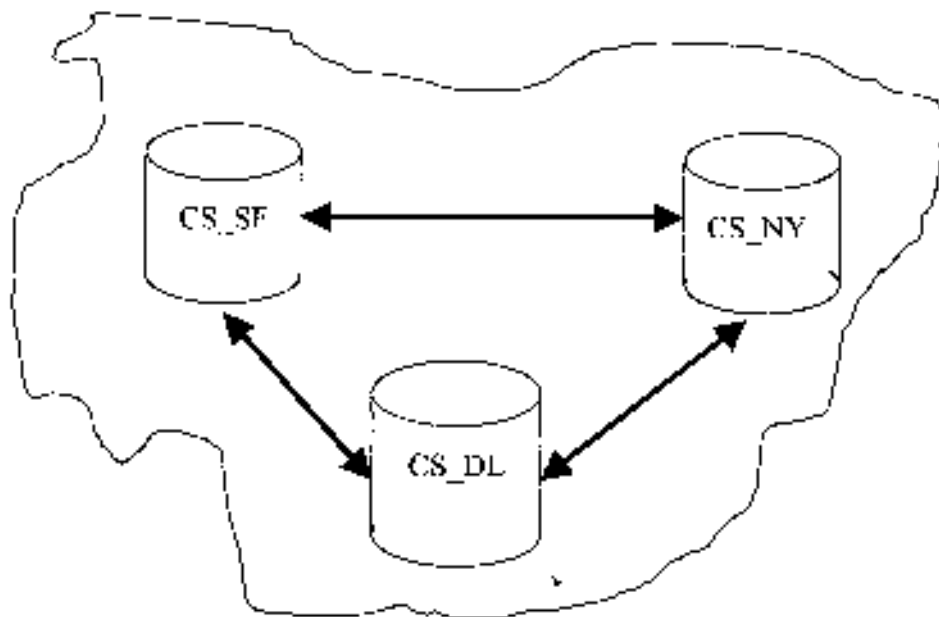


Figura 3.5: Replicación Multimaestro Soportada por Múltiples Puntos de Acceso a las actualizaciones.

### 3.3.3 Replicación de Instantáneas

Una instantánea contiene una réplica completa o parcial de una tabla maestra objetiva desde un solo punto en tiempo real. Una instantánea podría ser de solo lectura o actualizable.

#### 3.3.3.1 Instantáneas de solo Lectura

En una configuración básica, las instantáneas proveerían acceso de solo lectura a los datos de la tabla, originada desde una primaria o sitio “maestro”. Las aplicaciones pueden consultar datos desde réplicas locales de datos para evitar acceso de red a diferentes redes disponibles. De cualquier modo, las aplicaciones de todo el sistema deben acceder a los datos del sitio primario cuando son necesarias las actualizaciones. La figura 3.6 ilustra básicamente, la replicación solo lectura.

La siguiente es una lista de beneficios de las instantáneas de solo lectura:

- Tablas maestras no necesitan pertenecer a un grupo maestro.
- Puede soportar complejas instantáneas (las instantáneas pueden ser basadas en una o más tablas y contener agregados, uniones, operaciones fijas, o una cláusula CONNECT BY).
- Provee acceso local para entregar mejores tiempos de respuesta y disponibilidad.
- Descargar consultas del sitio del maestro.

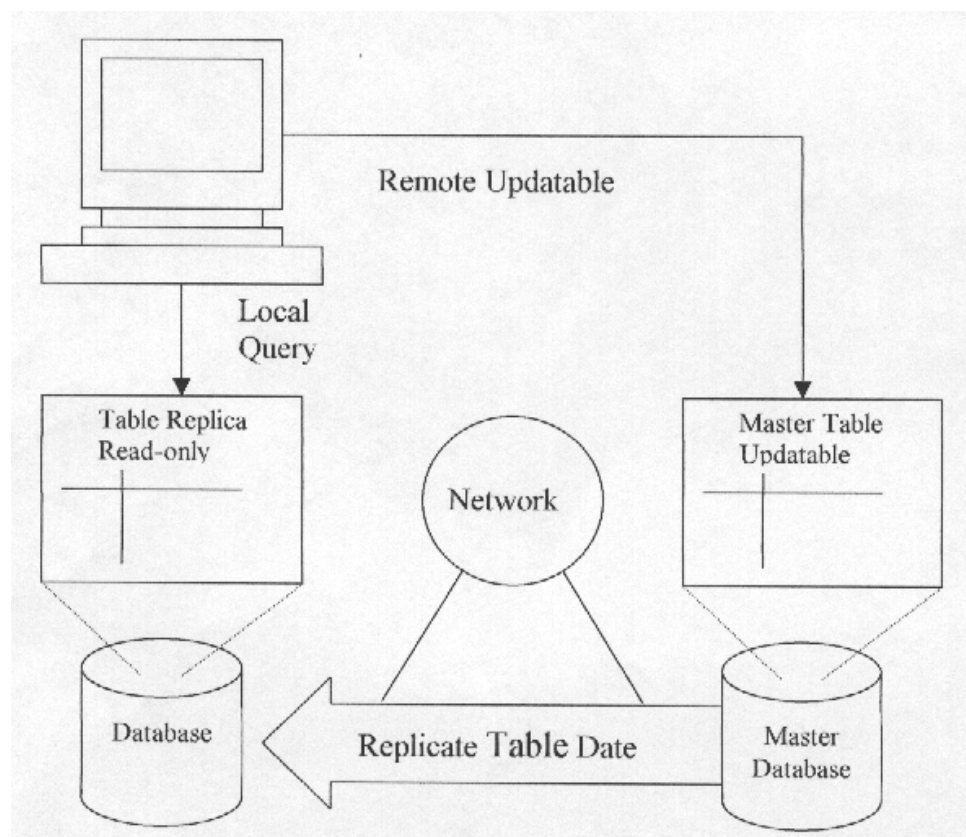


Figura 3.6: Replicación de Instantánea Solo Lectura

### 3.3.3.2 Instantáneas Actualizables.

En una configuración más avanzada, se puede crear una instantánea actualizable que permitiría a los usuarios insertar, actualizar, y borrar filas de una tabla maestra objetiva. Una instantánea actualizable contendría también sólo un subconjunto de la tabla maestra objetiva del conjunto de datos que posee.

Las instantáneas actualizables son basadas en tablas de un sitio maestro que han sido seteadas para soportar replicación multimaestro. De hecho, las instantáneas actualizables deben ser parte de un grupo de instantáneas que son basadas en un grupo maestro, de un sitio maestro. Esto se representa en la siguiente figura.

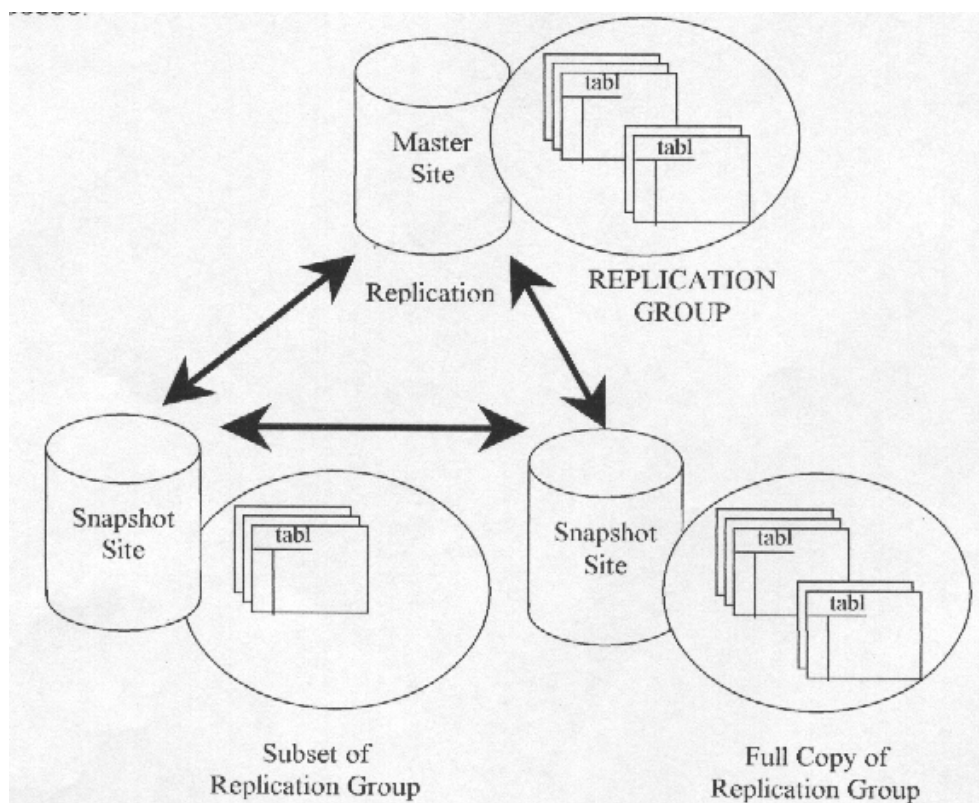


Figura 3.7: Ilustra un ambiente de replicación usando Instantáneas Actualizables.

Las Instantáneas actualizables tienen las propiedades siguientes.

- Se basan siempre en una tabla simple y puede ser incrementalmente (o “rápidamente”) refrescadas.
- ORACLE propaga los cambios hechos por una instantánea actualizable para la tabla maestra remota. Si necesita las actualizaciones en cascada a todos los sitios maestros.
- ORACLE refresca una instantánea actualizable como parte de un refrescamiento idéntico para instantáneas de solo lectura. (Un grupo de refrescamiento es un mecanismos organizacional que mantienen consistencia transaccional.

Las instantáneas actualizables tienen los beneficios siguientes:

- Deja que los usuarios consulten y actualicen una replica de datos local, cuando el sitio del maestro este desconectado.
- Incrementa la seguridad de los datos permitiendo replicar sólo un subconjunto seleccionado del conjunto de las tablas maestras.
- Una señal mas pequeña que la replicación multimaestro.

### **3.3.3.3 Usos de la Replicación de Instantáneas**

La replicación de instantánea es usada por varios tipos de aplicaciones. Las secciones siguientes describen algunos de los usos típicos para la replicación de instantáneas:

- Información descargada
- La replicación de instantáneas es usada como una manera de reproducir bases de datos enteras o información descargada. Por ejemplo cuando la ejecución de sistemas de procesamiento de transacciones de alto volumen es crítica, puede ser ventajoso mantener un base de datos duplicada para aislar las demandas de consultas de decidir soportar aplicaciones.

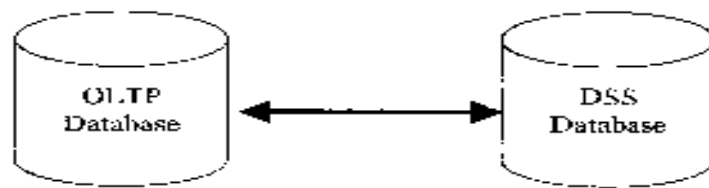


Figura 3.8: Información de Descarga.

### 3.3.3.4 Distribución de la información

Las replications de instantáneas de solo lectura son útiles para distribución de la información. Por ejemplo considere el funcionamiento de un gran cadena de bodegas de ventas. En este caso, esto es crítico para asegurar que la información del precio de los productos siempre este disponible y relativamente presente para las salidas. Para alcanzar estas metas, cada tienda puede tener una copia propia de los datos de los precios del producto que sea refrescada en la noche desde una tabla de precios primaria.

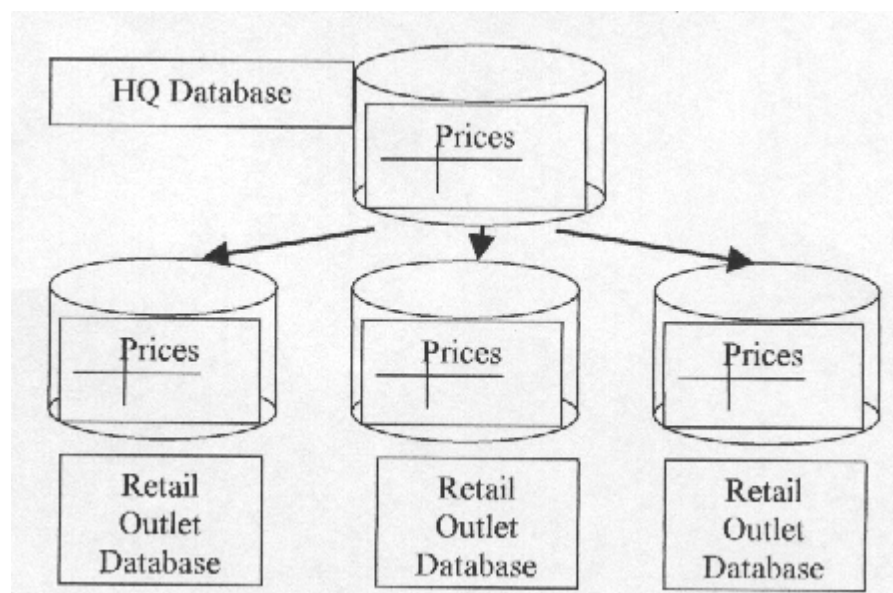


Figura 3.9: Ejemplo de una base de datos replicada a múltiples sitios.

### **3.3.3.5 Transporte de la información**

Las replicaciones de instantáneas de solo lectura y actualizables pueden ser usadas como un mecanismo de transporte de la información. Por ejemplo: La replicación de instantáneas de solo lectura pueden periódicamente mover datos desde una transacción de producción procesando una base de datos, a un almacén de datos(DATA WAREHOUSE).

### **3.3.3.6 Ambientes desconectados.**

La replicación de instantáneas actualizables son usadas por el despliegue de transacciones procesando aplicaciones que operan usando componentes desconectados. Por ejemplo considere las típicas ventas obligadas de un sistema de automatización para el seguro de vida de una compañía. Cada vendedor debe visitar clientes regularmente con una computadora portátil (LAPTOP) y registrar las ordenes en una base de datos personal, mientras esta desconectado de la red de computadoras corporativa y del sistema centralizado de base de datos. Al volver a la oficina, cada vendedor debe reenviar todas las ordenes a una base de datos centralizada.

Ayuda a desplegar un ambiente de instantáneas, por ejemplo, una venta forzada. Las plantillas de despliegue permite al administrador de la base de datos pre-crear un ambiente de instantáneas al sitio maestro para facilitar, la costumbre, la distribución segura e instalación de un ambiente de instantáneas. Las plantillas de despliegue permiten que el DBA<sup>5</sup> cree y despliegue un ambiente de instantáneas una vez y tan a menudo como requiera en el sitio de instantáneas.

---

<sup>5</sup> DBA.- Persona encargada de administrar la base de datos.

### 3.3.4 Configuraciones Híbridas Multimaestro e Instantáneas

La replicación Multimaestro e instantáneas se pueden combinar en configuraciones híbrido o “mixto” para encontrar diferentes requisitos de la aplicación. Las configuraciones mixtas pueden tener cualquiera número de sitios maestros y múltiples sitios de instantáneas por cada maestro.

Por ejemplo como se muestra en la Figura 3.10, n-caminos ( o multimaestro), la replicación entre dos maestros pueden soportar replicación de tablas-llenas entre las bases de datos, para soportar dos regiones geográficamente alejadas. Las instantáneas se pueden definir sobre el maestro para reproducir tablas llenas o subconjuntos de tablas para sitios dentro de cada región.

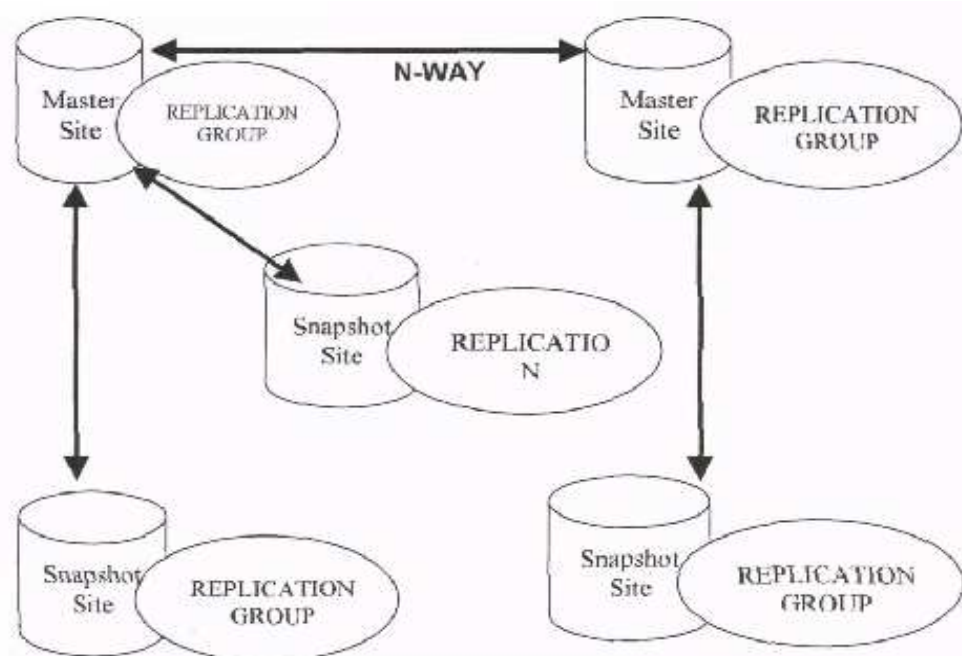


Figura 3.10: Configuración Híbrida

Diferencias importantes entre instantáneos y replicaciones maestros incluyen lo siguiente:

- Replicaciones maestro debe contener datos para hacer replicaciones de tablas llenas, considerando que las instantáneas pueden reproducir subconjuntos de datos de la tabla maestro.
- La replicación Multimaestro permite que la reproducción cambie por cada transacción cuando el cambio ocurre. La instantánea se refresca con una orientación fija, propaga los cambios de transacciones múltiples en un muy eficaz lote orientado al funcionamiento, pero con menos intervalos frecuentes.
- Si ocurren conflictos desde los cambios hechos para múltiples copias del mismo dato, los sitios maestros detectan y resuelven los conflictos.

### **3.3.5 Administración de un ambiente de Replicación**

Hay varias herramientas que están disponibles para ayudar a administrar y monitorear los ambientes de replicación. La administración ORACLE provee una poderosa guía para ayudar a administrar el ambiente, mientras la Administración de Replicación API provee una familiar interfase de programación de aplicaciones (API), para construir SCRIPT<sup>6</sup> personalizados para la administración de replicaciones. Adicionalmente, el catálogo de replicaciones guarda información acerca del ambiente de reproducción.

#### **3.3.5.1 Catalogo de Replicación**

Cada maestro y sitio de instantáneas en un ambiente de replicación tiene un catálogo de replicación. El catalogo de replicación de un sitio es un conjunto distinto de tablas y vistas del diccionario de datos, este mantiene

---

<sup>6</sup> SCRIPT.- Conjunto de comandos agrupados como un archivo, que permiten realizar alguna tarea específica.



información administrativa acerca de los objetos de replicación y grupos de replicación de los sitios. Cada servidor participando en un ambiente de replicación puede automatizar la replicación de objetos en grupos de replicación usando la información en un catalogo de replicación.

### **3.3.5.2 Administración de la Replicación API y Requerimientos de la Administración**

Para administrar y configurar un ambiente de replicación, cada servidor participante usa la interfase de programación de aplicación ORACLE (API). Un servidor de administración de replications API es un conjunto empaquetado de PL/SQL<sup>7</sup> encapsulando procedimientos y funciones de administración, pueden usarse para configurar rasgos de la replicación ORACLE, también utiliza los procedimientos y funciones de cada sitio de administración de replicación API para ejecutar trabajos.

Un requerimiento a la administración es una llamada a un procedimiento o función en el administrador de replicación ORACLE API. Por ejemplo cuando se usa la Administración de Replicación para crear un grupo maestro nuevo, la Administración de Replicación completa la tarea para elaborar una llamada al procedimiento DBMS\_REPCAT.CREATE\_MASTER\_REPGROUP. Algunos requerimientos de administración generan administraciones de replicación adicional API llamados para completar los requerimientos.

### **3.3.5.3 Administración de Replicación ORACLE**

ORACLE provee una herramienta de administración, denominada como el administrador de replicación ORACLE, que no es más que una ayuda en la configuración y administración de la replicación en ambientes tanto multimaestros como de instantáneas.

---

<sup>7</sup> PL/SQL.- Lenguaje estructurado de consultas que es utilizado para la manipulación de la base de datos en ORACLE

### **3.3.6 Conflictos de Replicación.**

Los ambientes multimaestros asíncronos y ambientes de replicación de instantáneas actualizables deben permitir la posibilidad de direccionarse a los conflictos de replicación cuando estos ocurran, por ejemplo, dos transacciones originadas desde sitios diferentes actualizan la misma fila casi al mismo tiempo.

Cuando los datos hacen ocurrir un conflicto, se requiere un mecanismo para asegurar que el conflicto pueda ser resuelto de acuerdo con las reglas de negocio, y que los datos converjan correctamente a todos los sitios.

Además se anotan cualesquiera de los conflictos que ocurrieran en el ambiente de replicación, La replicación de ORACLE ofrece una variedad de métodos para la resolución de conflictos que pueden permitirle definir un sistema de resolución de conflictos para crear una base de datos que permitiría resolver conflictos de acuerdo con las reglas del negocio. Si se tiene una situación única que ORACLE pre-construye métodos de resolución de conflictos que no los pueden resolver, se tiene la opción de construir y usar rutinas propias de conflicto.

### **3.3.7 Opciones Especializadas de Replicación**

Algunas aplicaciones tienen requisitos especiales de un sistema de replicación. A continuación se enuncian y explican las opciones únicas de replicación ORACLE, incluyendo:

- Procedimientos de replicación.
- Propagación sincrónica (En tiempo real) de los datos.

#### **3.3.7.1 Procedimientos de Replicación.**

Aplicaciones de procesamiento por lotes, pueden cambiar cantidades grandes de datos dentro de una sola transacción. En tales casos,

típicamente la replicación a nivel-fila podría cargar una red de trabajo con una cantidad grande de datos cambiados. Para evitar tales problemas, una aplicación de procesamiento por lotes, operando en un ambiente de replicación puede usar procedimientos de replicación ORACLE para llamar procedimientos almacenados para converger datos de replicación. Los procedimientos de replicación, reproducen solamente la llamada a un procedimiento almacenado que una aplicación lo que se usa para actualizar una tabla. Los procedimientos de replicación no reproducen modificaciones de los datos.

Para usar procedimientos de replicación, se debe reproducir los paquetes que modifican datos en el sistema para todos los sitios. Después de reproducir un paquete, se debe generar una envoltura para este paquete en cada sitio. Cuando una aplicación llama un paquete de procedimientos en el sitio local para modificar datos, la envoltura asegura que la llamada finalmente sea hecha para el mismo paquete de procedimiento en todos los otros sitios en el ambiente de replicación. Los procedimientos de replicación pueden ocurrir asincrónicamente o sincrónicamente.

#### **3.3.7.1.1 Descubrimiento del conflicto en los Procedimientos de Replicación**

Cuando un sistema de replicación reproduce datos usando procedimientos de replicación, los procedimientos que reproducen datos son responsables de asegurar la integridad de los datos reproducidos. Esto es, se debe diseñar tales procedimientos tanto para evitar o descubrir conflictos de replicación y resolver éstos apropiadamente. Por consiguiente, un procedimiento de replicación es típicamente usado cuando las base de datos están disponibles sólo para un procesamiento de grandes operaciones por lote. En tales situaciones los conflictos de replicación son inciertos porque numerosas transacciones no contienen los mismos datos.

### **3.3.7.2 Propagación sincrónica (En tiempo-real) de los Datos.**

La propagación sincrónica de los datos es la normal configuración para ambientes de replicación. Sin embargo, ORACLE también soporta propagación sincrónica de los datos para aplicaciones con requisitos especiales. Esta propagación sincrónica de datos ocurre cuando una aplicación actualiza una réplica local de una tabla, y dentro de la misma transacción, también lo hace con todas las otras réplicas de la misma tabla. Por consiguiente, la replicación de datos se llama también replicación de datos en tiempo-real. Se usa la replicación sincrónica sólo cuando se requieren aplicaciones que reproduzcan sitios, en los cuales deban continuamente sincronizarse.

Puede crearse un ambiente de replicación con algunos sitios propagando cambios sincronizadamente, mientras otros usan propagación sincronizada (transacciones diferidas). Un sistema de replicación que usa propagación en tiempo-real de la replicación, es muy dependiente del sistema y requiere disponibilidad de red, debido a que pueden funcionar solo cuando todos los sistema de sitios estén concurrentemente disponibles.

#### **3.3.7.2.1 Conflictos de Replicación en la propagación Sincronizada de Datos**

Cuando un sistema propietario compartido reproduce todos los cambios sincronizadamente (replicación en tiempo-real), los conflictos de replicación no pueden ocurrir. Con la replicación en tiempo-real, las aplicaciones usan transacciones distribuidas para actualizar todas las réplicas de una tabla al mismo tiempo. Como es el caso en ambientes de base de datos no distribuidas, ORACLE automáticamente bloquea las filas sobre el nombre de cada transacción distribuida para prevenir todos los tipos de interferencia destructiva entre transacciones. Los sistemas de replicación en Tiempo-Real pueden prevenir los conflictos de replicación.

### 3.3.8 Proceso de Replicación

Hay una terminología amplia detrás de la replicación en Oracle8 Server que fue analizada anteriormente pero que cabe recalcar en este instante, debido a que se deberá tomar muy en cuenta antes de la implementación sobre NT, entre estos tenemos:

- **Catalogo de replicación.-** Consiste en un conjunto de tablas de diccionario de datos y visualizaciones necesarias para mantener, administrar y coordinar la replicación. Donde todo sitio maestro e instantáneo debe tener uno.
- **Sitios de replicación.-** Existen una amplia variedad de sitios de replicación definidos en un sistema. Un sitio maestro contiene una copia de todos los objetos que se desea replicar, y se mantiene una copia completa de los objetos maestros. Un sitio instantáneo soporta instantáneas de datos de sólo lectura o actualizables desde el sitio maestro.
- **Grupo de replicación.-** Un conjunto de objetos de base de datos que se debe replicar pueden ser colocados en un grupo de replicación. Normalmente esta constituido de objetos que tienen una entidad lógica y pertenecen a una aplicación.

En el Oracle8 hay disponibles dos tipos de replicación:

- 1) Replicación básica
- 2) Replicación simétrica.

#### 3.3.8.1 Replicación Básica

Esta replicación proporciona una visualización de sólo lectura de datos. Las replicas de datos se crean en el nodo local generando instantáneas de datos desde un nodo maestro. Como éstas son copias de sólo lectura, las aplicaciones deben acceder al nodo maestro si necesitan modificar datos. Oracle8 Server soporta replicación básica utilizando

instantáneas, enlaces de bases de datos y distintos objetos de diccionario de datos.

Como un ejemplo que permita ilustrar la replicación de este tipo se podría citar la configuración de las tablas EMP y DEPT para la replicación con los siguientes pasos:

Paso 1: Se debe crear el enlace de la base de datos en ORCO, que se puede utilizar para conectarse como el usuario scott a la base de datos ORCL.

```
SVRMGR> connect system/manager@child  
SVRMGR> create public database link oracle connect to scott identified  
by tiger using 'mast';
```

El nombre del enlace de la base de datos debe ser el mismo que el nombre de la base de datos si el parámetro siguiente está ajustado en el INIT.ORA.  
Global\_names = TRUE

Paso2: Crear un registro instantáneo en el nodo maestro.

Oracle8 Server soporta dos tipos de actualizaciones: 1) actualización completa y 2) actualización rápida. La primera reemplaza todos los datos del nodo Snapshot con los datos desde el nodo Master ejecutando una consulta actualizada. La segunda modifica el nodo Snapshot aplicando los cambios en el nodo Master desde la última actualización (cambios incrementales). Es necesario además llevar un registro de instantáneos para llevar un catálogo de los cambios y el que debe ser obligatorio para las actualizaciones rápidas:

```
SVRMGR> connect system/manager@mast  
SVRMGR> create snapshot log on scott.emp;
```

```
SVRMGR> create snapshot log on scott.dept;
```

Paso 3: Crear las instantáneas del nodo snapshot

Una instancia proporciona una visualización de sólo lectura de la tabla en el snapshot. Puede crearse una instantánea utilizando la orden CREATE SNAPSHOT:

```
SVRMGR> connect system/manager@child
```

```
SVRMGR> create snapshot log on scott.emp as select * from  
scott.emp@oracle;
```

```
SVRMGR> create snapshot log on scott.dept as select * from  
scott.dept@oracle;
```

Paso 4: Crear un grupo de refresco

Un grupo de refresco permite crear un conjunto de objetos para replicación, generalmente contiene los objetos necesarios para una aplicación.

```
SVRMGR> connect system/manager@child
```

```
SVRMGR> execute dbms_refresh.make (\  
2> name=> 'scott.refgrd', \  
3> list => 'scott.dept, scott.emp', \  
4> next_date => sysdate, \  
5> interval => 'sysdate + 1/96');
```

```
SVRMGR> commit;
```

Paso 5: Replicación de prueba

Esta se llevará a cabo con la ejecución de consultas a la tabla EMP del Master como Snapshot

```
SVRMGR> connect system/manager@mast
```

```
SVRMGR> select count(*) from emp;
```

```
SVRMGR> connect system/manager@child
```

```
SVRMGR> select count(*) from emp;
```

```
SVRMGR> connect system/manager@mast
```

```
SVRMGR> delete from emp where empno in(7902,7934);
```

```
SVRMGR> commit;
```

```
SVRMGR> select count(*) from emp;
```

```
SVRMGR> connect system/manager@child
```

```
SVRMGR> select count(*) from emp;
```

### 3.3.8.2 Replicación simétrica (Avanzada)

También llamada avanzada, permite que las replicas de datos se actualicen automáticamente a través del sistema. Las aplicaciones pueden modificar los datos en cualquier nodo participante y estos cambios se reflejan en forma automática en todos los nodos. El Oracle Server asegura que haya una consistencia de transacción global, manteniendo la integridad de datos. Para llevar a cabo este tipo de replicación se debe estimar necesario seguir los siguientes pasos:

#### Paso1. Creación de alias de Net8.

Se debe crear alias de Net8 para los sitios maestros, se recomienda hacer con el mismo nombre que la base de datos.

#### Paso2. Configurar sitios maestros.

Para esto se debe invocar el Setup Wizard en REPMGR y seleccionar la opción para instalar los sitios maestros, dar clic en New y añadir los sitios maestros.

El usuario system de Oracle es el que se utiliza para crear: al administrador de replicación, el propagador, el receptor de cuentas, y el intervalo de



actualización, de manera similar se debe configurar los parámetros para planificar purgas.

Paso3. Creación de los grupos maestros.

Un grupo de replicaciones en un sitio maestro es llamado grupo maestro, se puede crear un grupo maestro utilizando REPMGR. Seleccionando la conexión de la base de datos para un sitio maestro con la selección de File | New | New Master Group, desde el menú. Proporcionando un nombre para el grupo maestro y añadiendo objetos para la replicación, así como la dirección de destino.

Paso4. Configuración de los parámetros de resolución de conflictos.

La resolución de conflictos asegura que los datos converjan a todos los sitios maestros y además evitan errores en cascada.

Paso5. Concesión de acceso a los objetos replicados. Para esto se deberá utilizar el Security Manager o la orden GRANT de SQL para conceder acceso apropiado a los objetos replicados en los distintos sitios.

## CAPÍTULO IV

### DESARROLLO DEL CASO PRÁCTICO

Todo el proceso que se va a realizar está basado en la utilización de las nuevas herramientas de administración, como lo son los Wizard (asistentes), que proporcionan una manera fácil de crear y manipular ciertos procedimientos en forma automática, únicamente con la inclusión de ciertos parámetros, en este caso práctico se utilizará la replicación de mezcla ya que aquí se trabaja con los datos reales. Con esta pequeña introducción se procede a continuación a especificar los pasos que permitirán determinar como se realizará la presentación de datos en el Administrador de Bases de Datos Relacional SQL Server 7.0.

#### Paso 1: Creación de la Base de Datos Ejemplo (REPLICACION).

Para iniciar esta práctica se hace necesario crear una base de datos de la cuál se replicará la información:

En el Enterprise Manager de SQL Server se deberá expandir lo necesario hasta llegar a la pestaña Databases, y dar clic derecho y escoger la opción New database como se muestra a continuación.

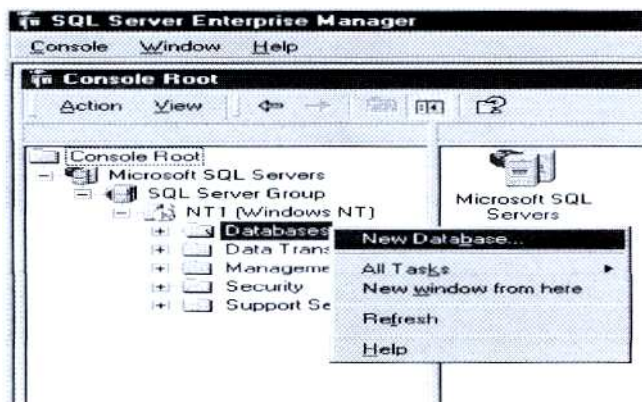


Gráfico 5.1. Ventana para la creación de una nueva Base de Datos.

Al realizar esto se presenta la primera venta del asistente donde se especificará cierta información que permitirá la correcta configuración en la creación de la base de datos.

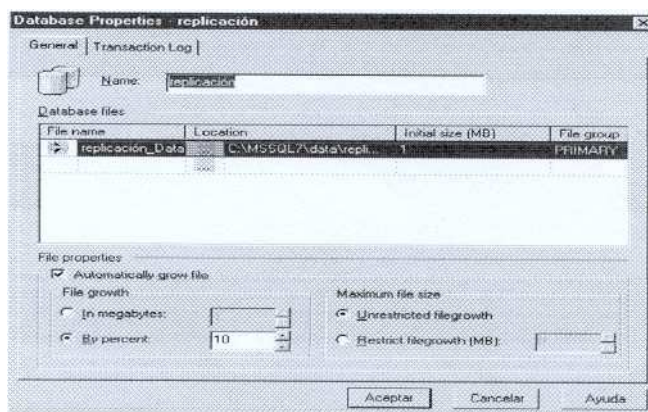


Gráfico 5.2. Ventana para la configuración de la Base de Datos.

Luego de realizada la creación, está nueva base de datos llamada “replicación”, aparecerá formando parte del grupo de bases de datos en la pestaña databases, incluyendo en esta todos los componentes necesarios como son diagramas, tablas, vistas, procedimientos almacenados, etc.

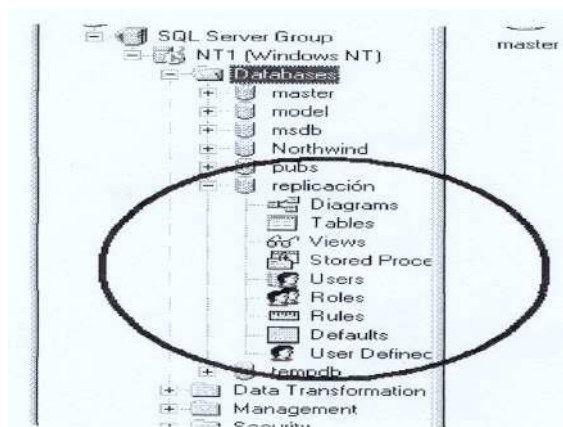


Gráfico 5.3. Ubicación de la nueva Base de Datos replicación.

## Paso2: Creación de Tablas

Para este caso práctico se hará necesario crear unas tablas que contengan información (datos), estas son: Productos, factura, detalle de factura con sus respectivos campos, incluyendo estos el tipo y tamaño, para esta creación se debe expandir las pestañas databases y replicación en tablas, dar clic derecho y escoger la opción New table

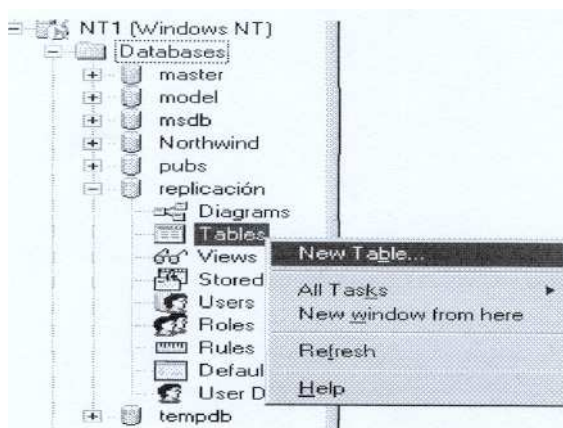


Gráfico 5.4. Ubicación del asistente para la creación De tablas.

Casi de la misma forma que en la creación de la base de datos aquí se presenta un asistente en el cual se procederá al llenado de la información correspondiente, con la cuál se realizará la configuración y creación de las tablas una por una, como en el siguiente ejemplo.

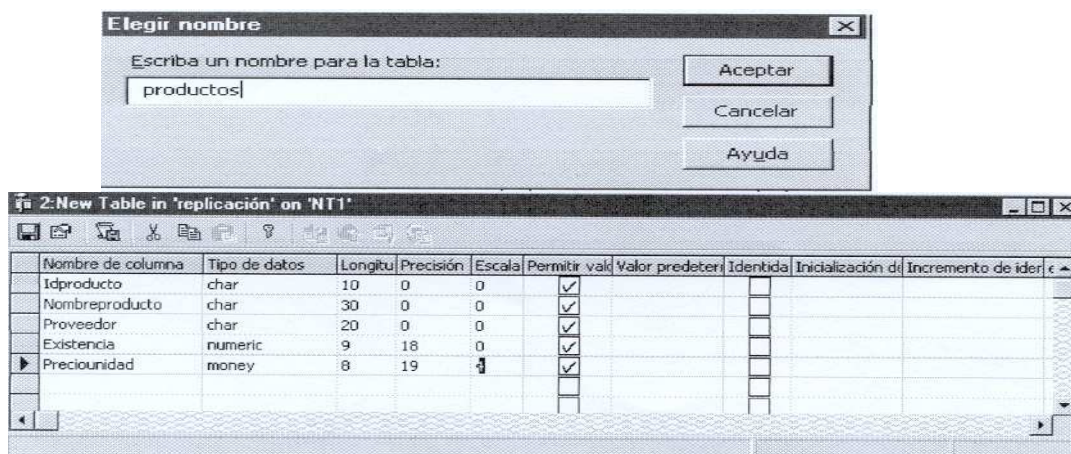
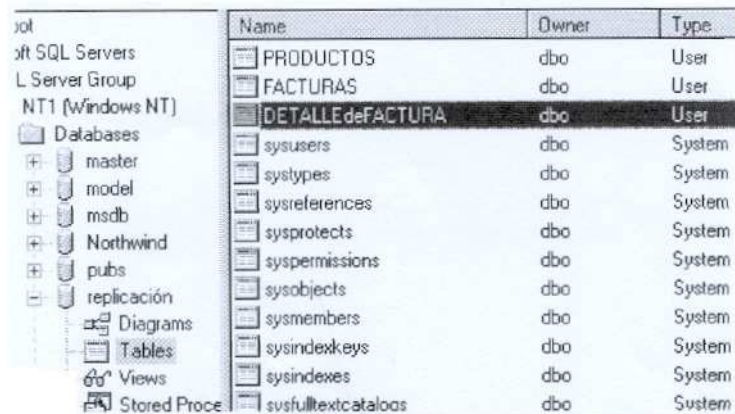


Gráfico 5.5. Muestra las ventanas que permiten configurar tablas.

Al utilizar este asistente se crean las tablas con las que se presentará el caso práctico de la replicación, estas se muestran internamente en la pestaña tables de la nueva base de datos denominada “replicación”.



Name	Owner	Type
PRODUCTOS	dbo	User
FACTURAS	dbo	User
DETALLE deFACTURA	dbo	User
sysusers	dbo	System
systypes	dbo	System
sysreferences	dbo	System
sysprotects	dbo	System
syspermissions	dbo	System
sysobjects	dbo	System
sysmembers	dbo	System
sysindexkeys	dbo	System
sysindexes	dbo	System
sysfulltextcatalogs	dbo	System

Gráfico 5.6. Muestra las nuevas tablas en la base de datos replicación.

Para una mejor administración de los datos que se almacenan en la base de datos “replicación”, se debe crear un diagrama con el cuál configurar las relaciones que existirán entre las tablas que fueron creadas anteriormente. En la pestaña de Diagrams en el menu contextual escogiendo la opción New Database Diagram de la siguiente manera:

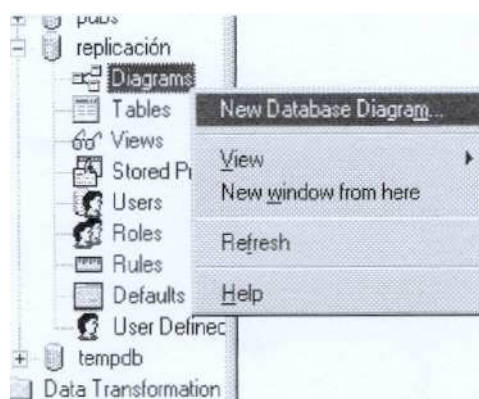


Gráfico 5.7. Opción para la creación de un nuevo diagrama.

Al escoger esta opción se abrirá una ventana de edición en donde se realizarán algunas tareas adicionales para conseguir el objetivo siendo estas: Tarea 1: Agregar las tablas en la ventana de edición

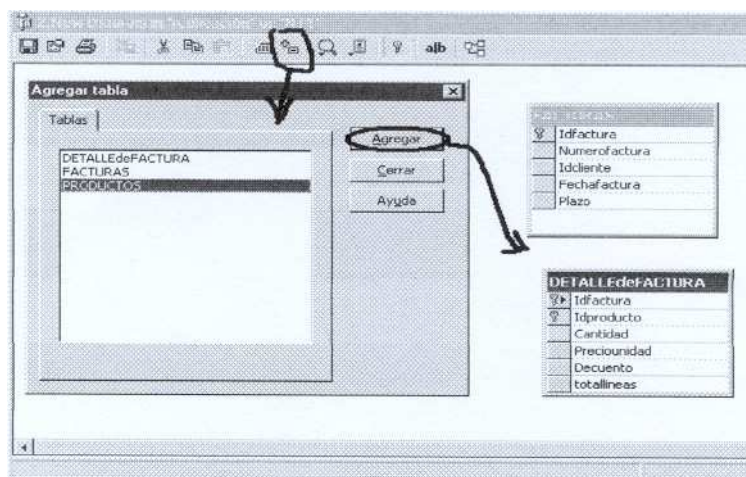


Gráfico 5.8. Ventana para añadir tablas al diagrama.

Tarea 2: Luego de tener las tablas se procede a la creación de las relaciones escogiendo los campos coincidentes en las tablas que se relacionan de la siguiente forma:

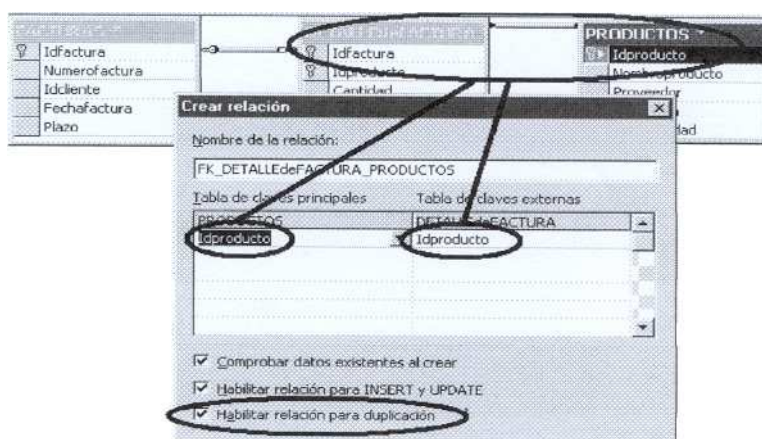


Gráfico 5.9. Opción para la creación de las relaciones entre las tablas.

### Paso 3: Añadir Registros (Datos) a las Tablas.

Para realizar este paso se procederá a crear un procedimiento almacenado, utilizando el asistente e incrementando las instrucciones

transact SQL necesarias para este efecto, en la base de datos “replicación” en la pestaña stored procedures, al hacer clic en el botón derecho del mouse para abrir el menú contextual y escogiendo la opción New Stored Procedure.

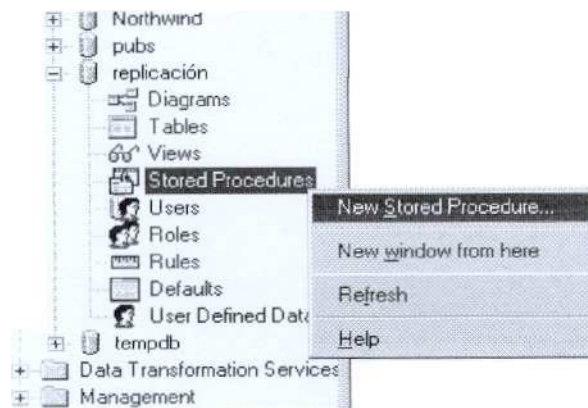


Gráfico 5.10. Menú contextual para la manipulación de los stored procedures.

Un ejemplo de esto son los siguientes procedimientos almacenados:

Para la tabla productos se usa sp\_productos

```
CREATE PROCEDURE sp_productos
```

```
AS
```

```
declare @contar int
```

```
set @contar = 1
```

```
while @contar<=100
```

```
begin
```

```
    insert productos (idproducto, nombreproducto, proveedor, existencia,  
                   preciounidad)
```

```
        values (@contar, "PERAS", "Juan", 10, 20);
```

```
    set @contar=@contar+1
```

```
end
```

Para la tabla FACTURA se usa sp\_factura

```
CREATE PROCEDURE sp_factura
AS
declare @contar int
set @contar = 1
while @contar<=100
begin
    insert facturas (idfactura, descripcion , fecha, cliente, forma_pago.total)
        values (@contar, "Venta de productos", '10/10/2002',
            "Juan", "Mensual" 200);
    set @contar=@contar+1
end
```

Para la tabla DETALLE de FACTURA se usa sp\_detalledefacturas

```
CREATE PROCEDURE sp_detalledefactura
AS
declare @contar int
set @contar = 1
while @contra<=100
begin
    insert detalle_factura (idproducto, idfactura, cantidad)
        values (@contar, @contar, 15);
    set @contar=@contar+1
end
```

Las tablas se llenaran con información, ejecutando estos procedimientos en el analizador de consultas que permite verificar y poner en marcha sentencias realizadas donde se puede incluso ver todos los resultados en las ventanas de respuesta de sentencias en forma de: texto, tabla y plan de ejecución.



#### Paso 4: Creación de la Base de Datos de Distribución.

Es aquí que inicia el proceso de replicación, con la configuración del distribuidor que no es más que la creación de la base de datos de distribución que contendrá la información relacionada con: los suscriptores, los publicadores y el distribuidor, esta configuración se logra utilizando el asistente para configurar publicaciones y distribución, el que se encuentra expandiendo el servidor SQL en la pestaña console root del explorador y escogiendo la opción herramientas (tools) del menú, en la otra ventana se escoge la pestaña Replication y luego el asistente para la configuración de publicación y distribución así:

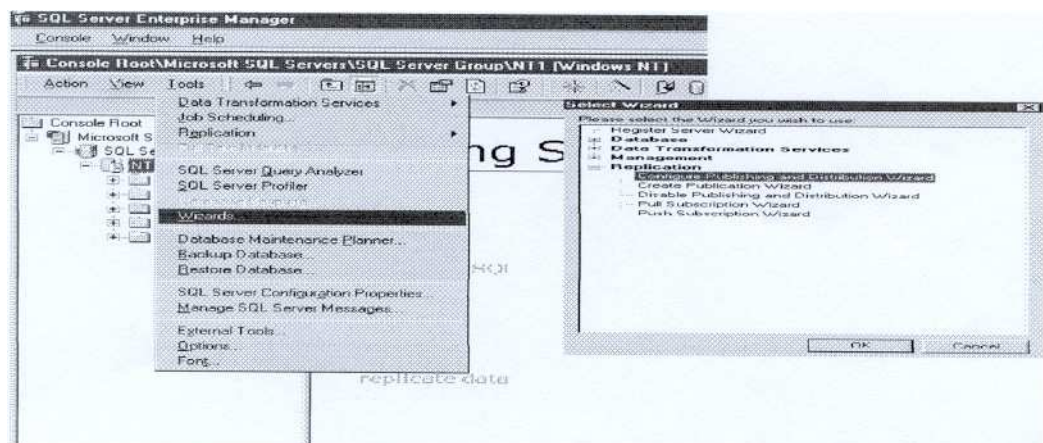


Gráfico 5.11. Ubicación del asistente para la creación de la base de datos de distribución de la replicación.

Al escoger la opción se presenta una ventana inicial del asistente que muestra el proceso que se llevará a cabo para la configuración

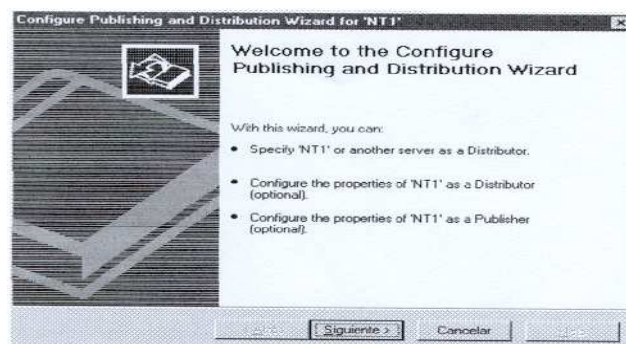


Gráfico 5.12. Ventana inicial del asistente para la configuración de la base de datos de distribución.

Al dar siguiente, se pasará a la ventana de selección en la que se escoge el servidor en el cual se configurará la base de datos de distribución

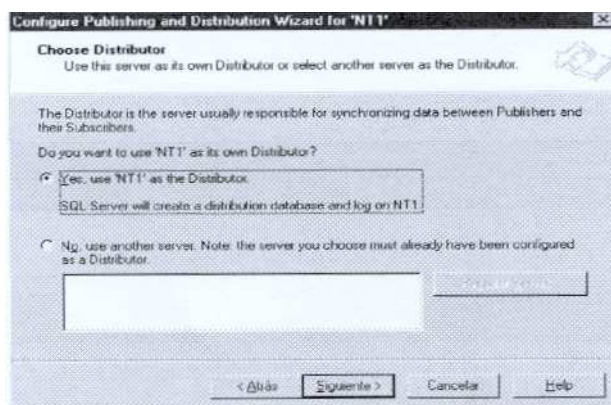


Gráfico 5.13. Escogitamiento del servidor en donde se configurara la base de datos de distribución.

Continuando con esta configuración luego de escoger el servidor se debe setear en forma personalizada o con los valores por defecto, la publicación y la distribución

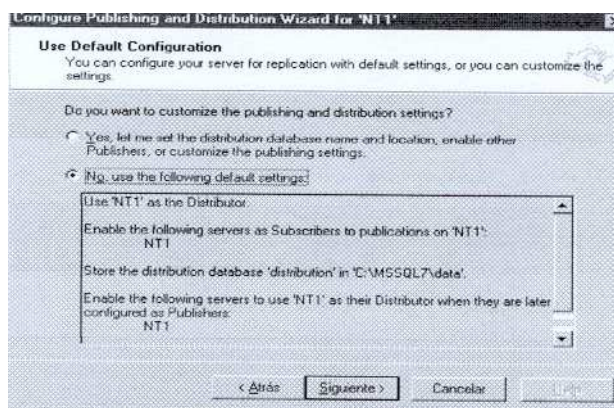


Gráfico 5.14. Ventana de seteos de la publicación y la distribución.

Si se escoge personalizada, se muestran otras pantallas para la configuración entre las cuales están: el proveer la información de la base de datos de distribución.

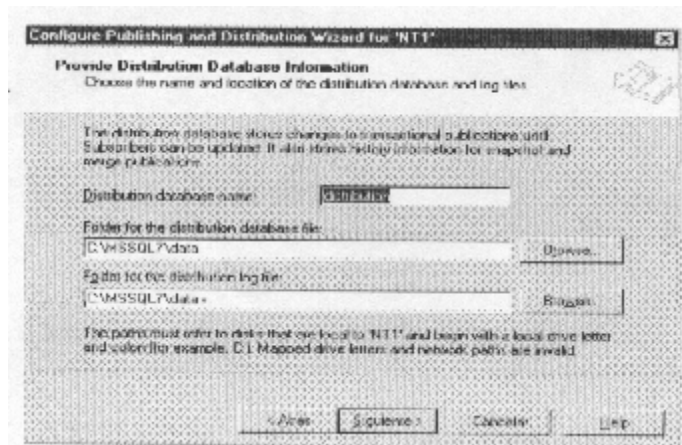


Gráfico 5.15. Ventana para proveer información a la configuración personalizada de la base de datos de distribución.

A continuación está la ventana que habilita al servidor para ser usado después de ser configurado como un publicador.

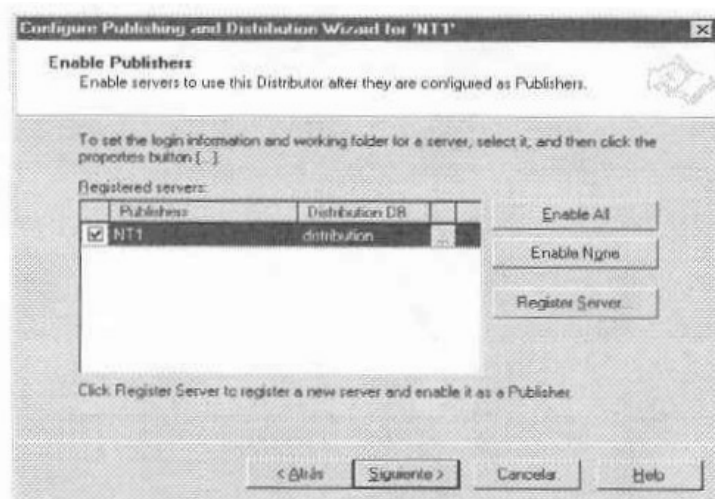


Gráfico 5.16. Ventana que permite habilitar al servidor como un publicador.

La siguiente ventana presenta las bases de datos que se pueden

configurar para ser replicadas, entre las que esta la base de datos "replicación" que es el ejemplo de este caso práctico.

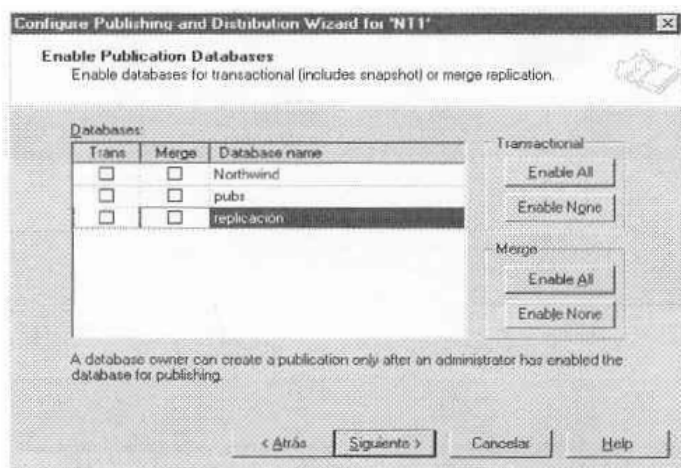


Gráfico 5.17. Ventana que presenta las bases de datos disponibles para ser replicadas se incluye la base de datos “replicación”.

De la misma manera que se debía habilitar el servidor para un publicador, hay que habilitar el servidor para un suscriptor desde el publicador.

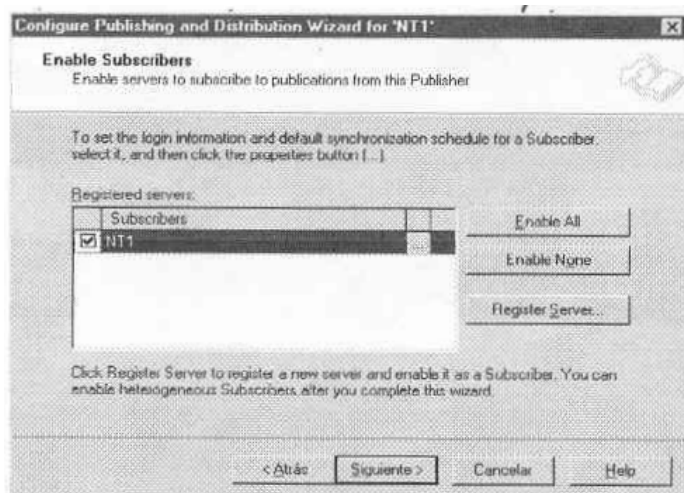


Gráfico 5.18. Ventana que permite habilitar al servidor como un suscriptor.

Finalmente dentro de esta definición de configuración personalizada esta la ventana de finalización que muestra como se va a crear la configuración de la base de datos de distribución con los respectivos publicadores y suscriptores.

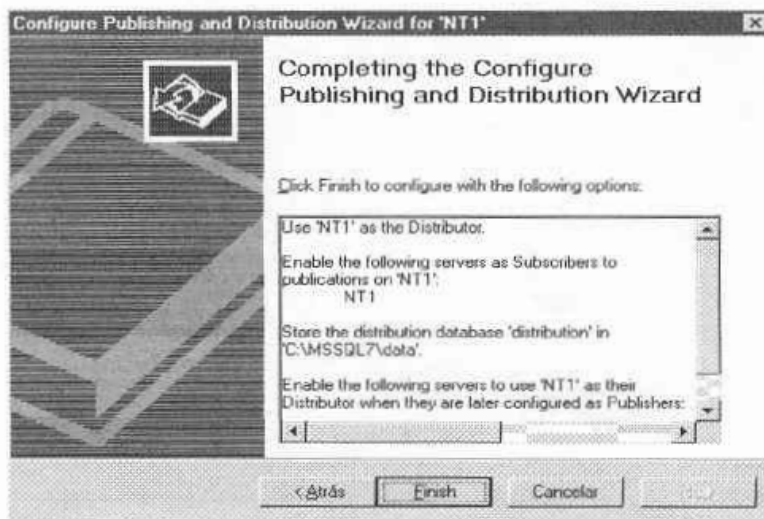


Gráfico 5.19. Ventana de finalización de configuración personalizada del suscriptor y la base de datos de distribución

Si se escogió la opción de configurar tomando los valores por defecto se pasa directo a la ventana de finalización de configuración solo faltaría dar por finalizado el asistente para que empiece el trabajo real de configuración.

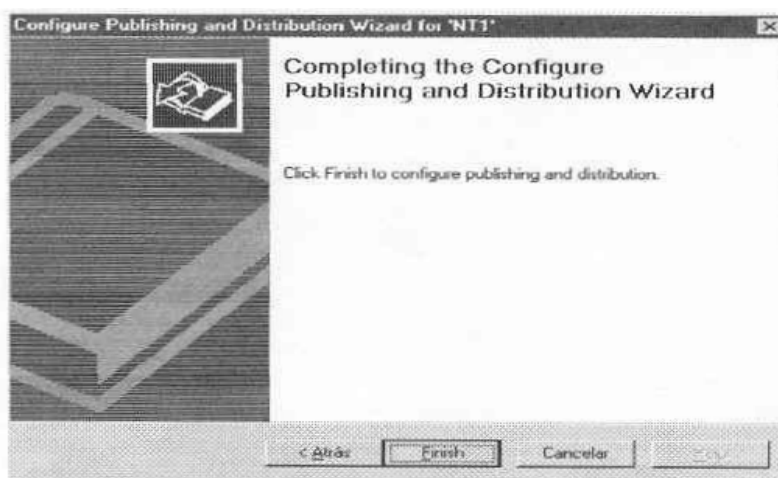


Gráfico 5.20 Ventana de finalización con valores por defecto.

Escogida la configuración por cualesquiera de estas formas la configuración se la realiza de los siguientes puntos.

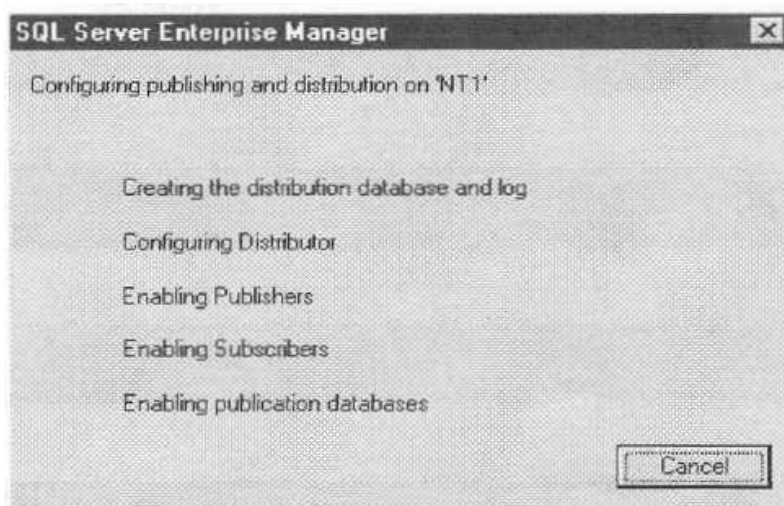


Gráfico 5.21. Puntos que el asistente configurará.

Cuando termina de realizar la configuración el asistente presenta una ventana de aviso en la que se especifica que se creará también una opción de monitoreo de la replicación.

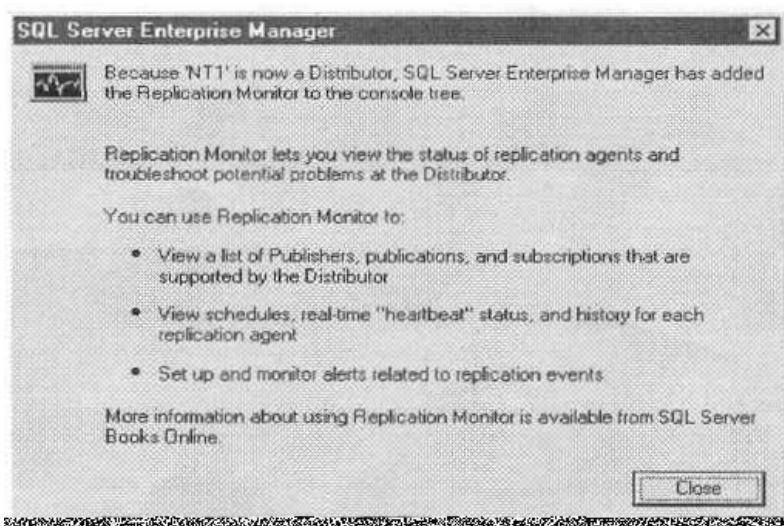


Gráfico 5.22. Ventana de aviso de finalización de configuración.

Concluido el proceso de configuración de la distribución así como el monitoreo de la replicación se presenta en la consola root la base de datos de distribución así como el monitoreo.

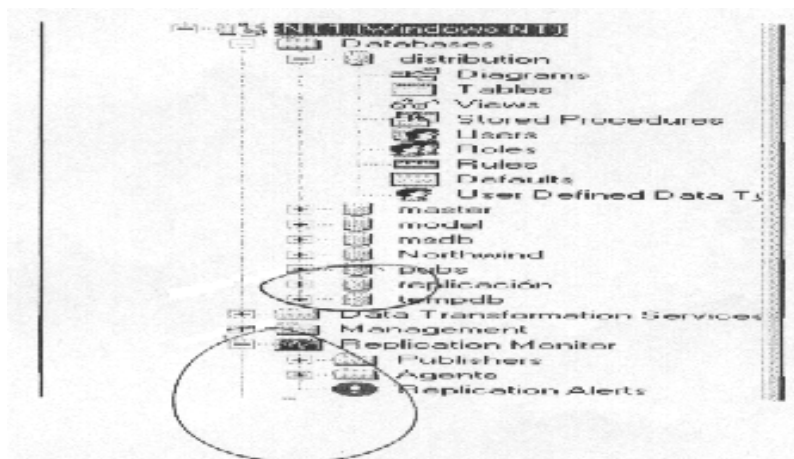


Gráfico 5.23. Ubicación de las nuevos componentes de replicación.

### Paso 5: Creación de la Publicación

En este caso se creará la publicación con la cual se procede a poner la información, así como la definición del tipo de publicación que se procesará. De la misma manera como se ha hecho hasta esta parte de este caso práctico se utilizará el asistente definido para aquello. En primera instancia se deberá escoger la base de datos a ser replicada, en este ejemplo “replicación”.

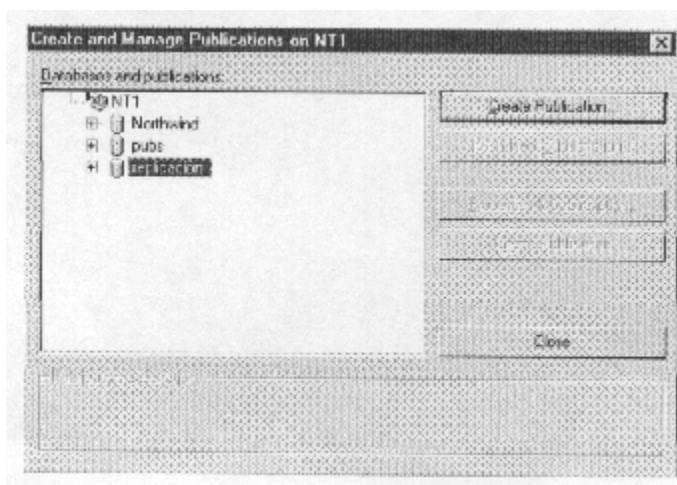


Gráfico 5.24. Creación de la publicación.

Seguidamente al dar clic en el botón Create Publication se pasará a la siguiente ventana de creación en donde da comienzo al manejo del asistente.

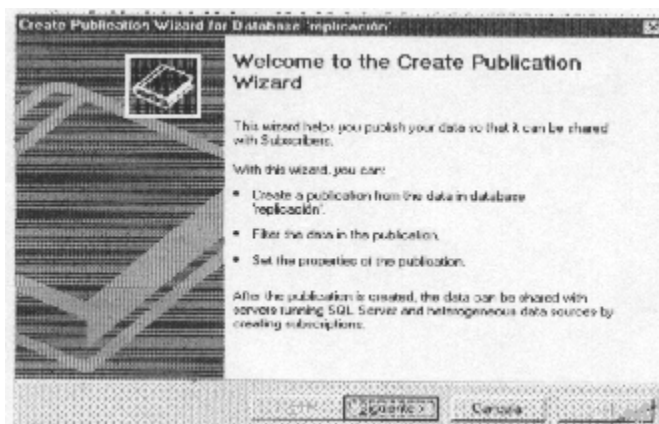


Gráfico 5.25 Ventana de inicio del uso del asistente para crear la publicación.

La siguiente tarea es definir el tipo de replicación que se ejecutará, existiendo tres posibilidades: Instantánea, Transaccional y de Mezcla, se deberá tener precaución en el momento de escoger una de las tres, en sentido que cuando se creó la base de datos del distribuidor se configuró para un tipo específico, si se desea escoger una que no este configurada, el asistente procederá a cambiar tal configuración que a la postre podría llevar a la modificación del plan de replicación.

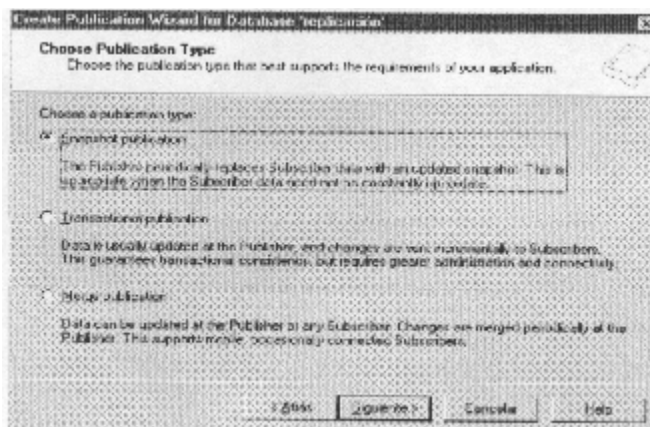


Gráfico 5.26. Tipos de replicación existentes en SQL Server.



Continuando con las ventanas del asistente se pasará, a aquella en la cual se pide que se escoja el tipo de actualización del suscriptor que está entre: inmediata y no inmediata, la diferencia es que en la primera en el momento que se actualiza en publicador, el suscriptor también se actualiza, y la otra espera una situación de commit.

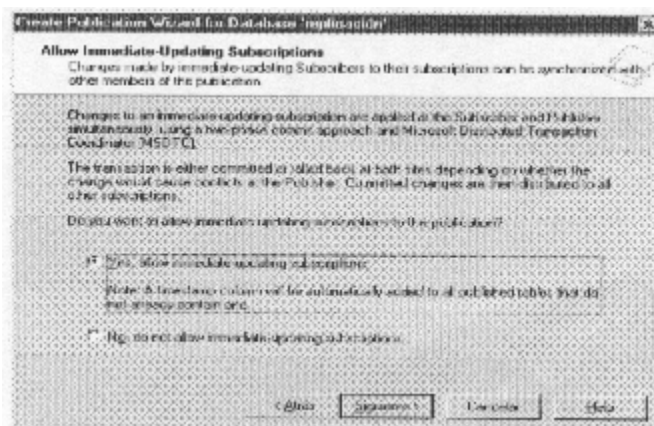


Gráfico 5.27. Tipos de actualización del suscriptor.

En la siguiente ventana se especifica cómo y dónde se realizará la suscripción, estando esto entre: la ejecución en el servidor SQL o en otro administrador como puede ser Oracle, Access, Infomix, etc. En los cuales obligadamente se debe tener configurado ya sea el ODBC o el OLEDB para acceder al origen de los datos.

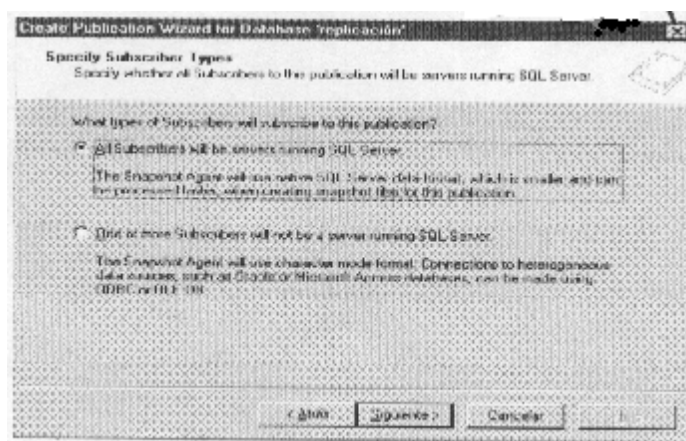


Gráfico 5.28. Ubicación de realización de la suscripción.

En este punto se presentan los artículos que se van a permitir replicarse, se debe seleccionar todos los necesarios, para nuestro ejemplo solo están las tablas que se crearon en el paso 2.

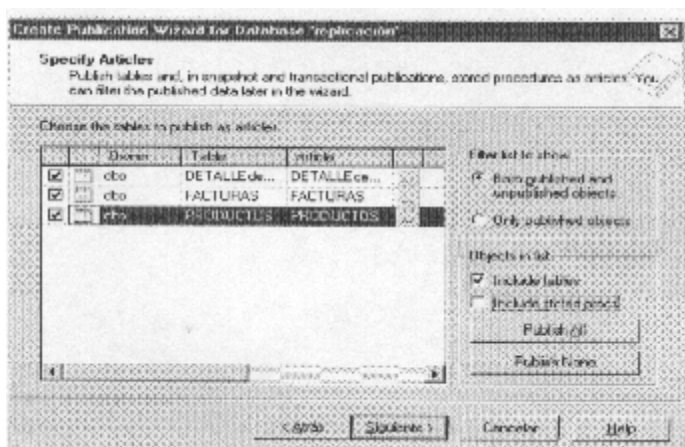


Gráfico 5.29 Ventana que presenta los artículos a replicarse.

A continuación se presenta la ventana donde se ingresa el nombre que tendrá la publicación con la respectiva descripción de lo que se esté configurando.

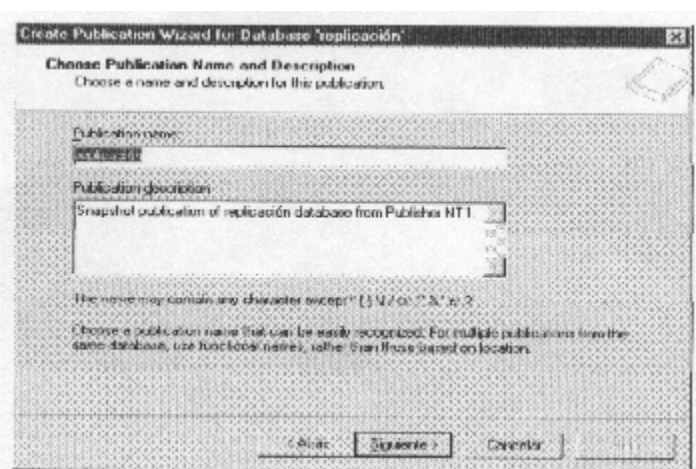


Gráfico 5.30. Ventana para identificar la publicación.

Siguiendo con la creación de la publicación, el asistente pedirá que se especifique si se coloca un filtro de datos o no.

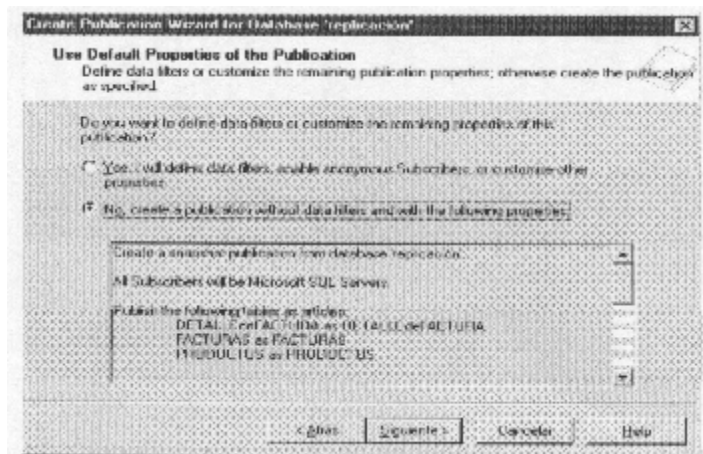


Gráfico 5.31 Ventana para especificar filtro de la replicación.

Por último el asistente completará la creación de la publicación con la finalización del asistente y realizando el proceso de creación.

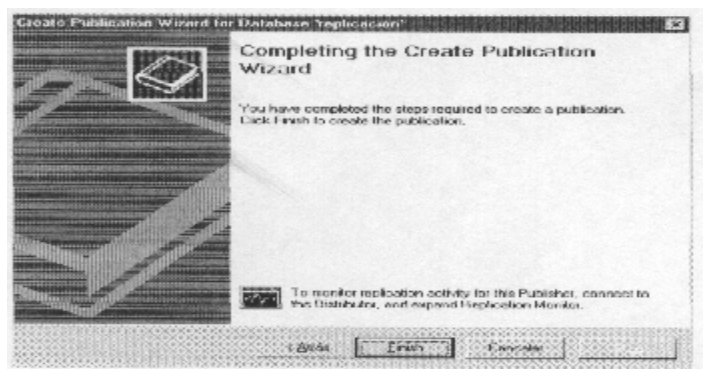


Gráfico 5.32. Ventana de finalización de la publicación.

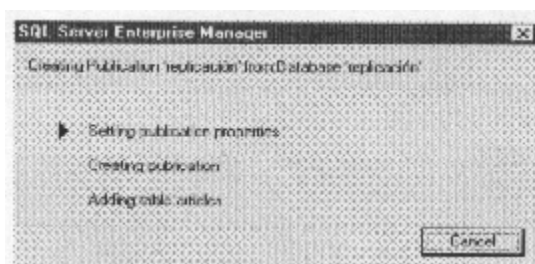


Gráfico 5.33. Ventana que muestra la creación de la publicación.

Al concluir con la creación se presentará esta nueva configuración en la consola root tanto en el monitor de replicación como en publicaciones de la base de datos replicación de la siguiente manera.

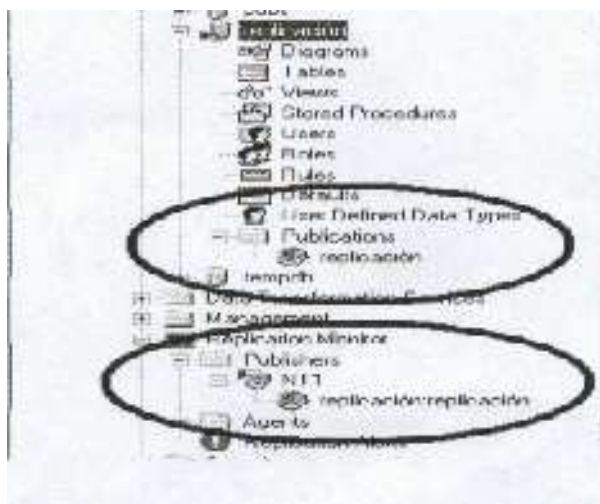


Gráfico 5.34. Ubicación de la publicación en la consola.

### Paso 6: Creación de la Suscripción de Extracción.

Como para todo lo que se ha realizado hasta este punto se han utilizado los asistentes, en la creación de suscriptores no será la excepción, entonces se abrirá el asistente para la creación de suscriptores de extracción.

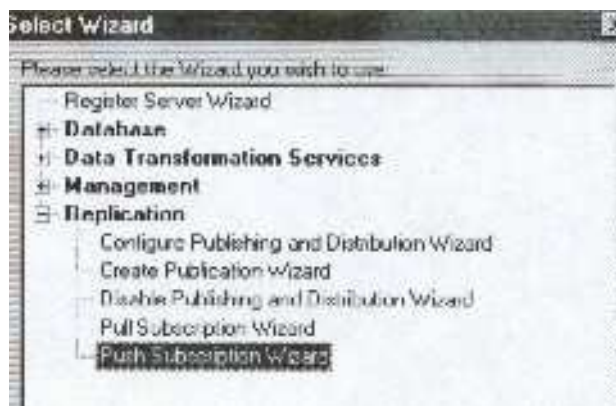


Gráfico 5.35. Ventana que presenta la ubicación del asistente para crear la publicación de extracción.

Al escoger el asistente que permitirá crear el suscriptor de extracción se debe seleccionar a que publicador pertenecerá, y dar clic al botón Push New Subscription.(Nueva Suscripción de Extracción)

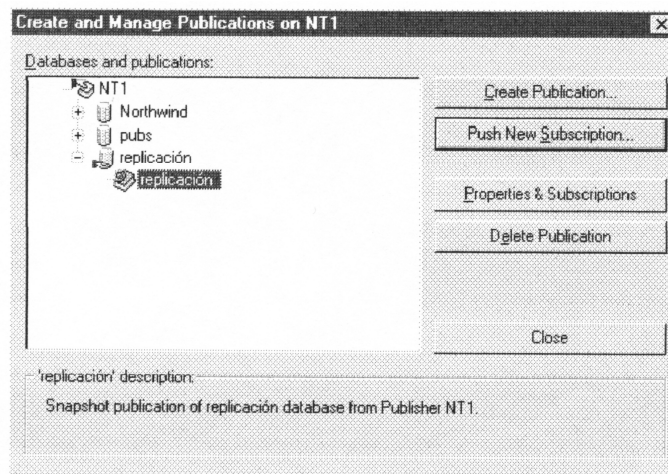


Gráfico 5.36. Ubicación del publicador para la suscripción.

De aquí se pasará a la ventana que muestra en si el trabajo del asistente, escogiendo primeramente el suscriptor que anteriormente se había configurado y con el cual se creará la suscripción.

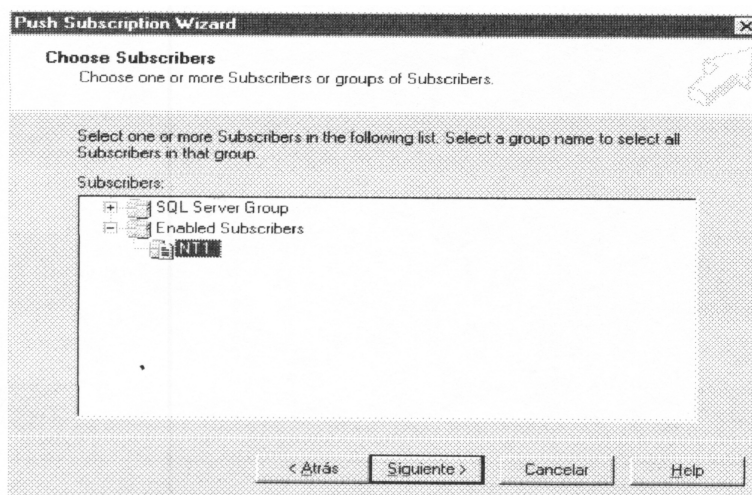


Gráfico 5.37. Escogitamiento del servidor donde se manipulará el suscriptor.

Se especifica a continuación la base de datos de suscripción para el suscriptor del servidor SQL, así como se muestra a continuación:

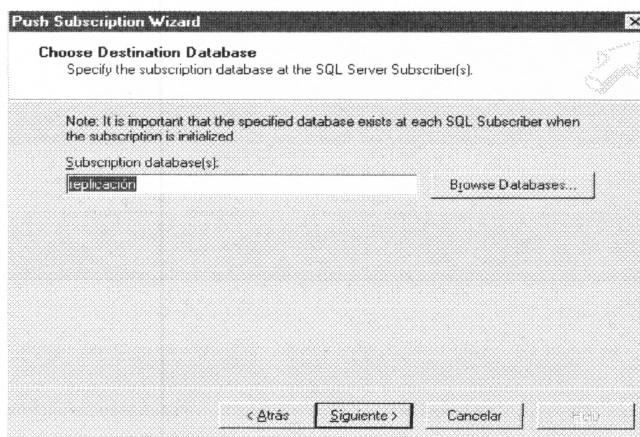


Gráfico 5.38. Ubicación de la base de datos de suscripción.

Continuando con la creación de la suscripción la siguiente tarea del asistente es pedir la frecuencia con la que el agente de distribución actualizará la suscripción. Teniendo dos posibilidades; la primera continuamente, es decir provee una mínima latencia siendo esto cuando ocurre una acción en el publicador se propaga al suscriptor; y la segunda siguiendo una planificación, especificando el tiempo en que realizará la replicación, para el ejemplo se seguirá un plan.

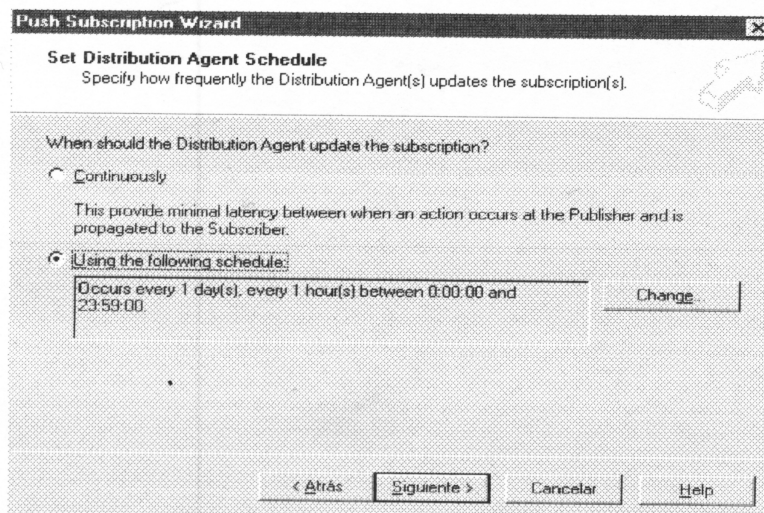


Gráfico 5.39. Frecuencias de actualización de la suscripción .

Para seguir con la creación de la suscripción, se especifica si las suscripciones necesitan ser inicializadas y si es el caso cuando comienza el proceso de inicialización.

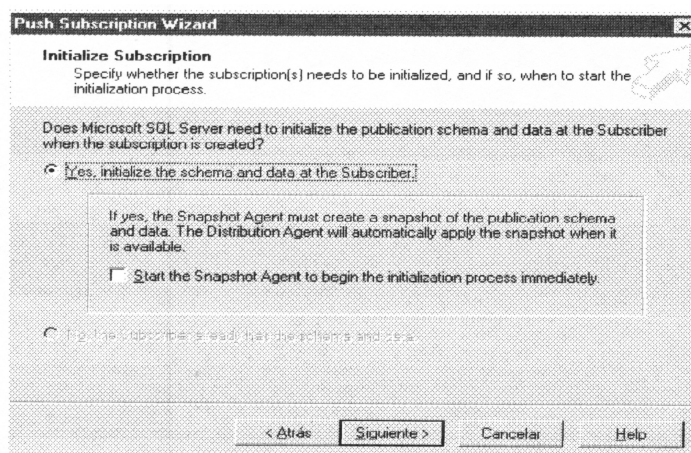


Gráfico 5.40. Ventana que especifica la inicialización de la suscripción.

Al haber especificado la inicialización del suscriptor, la siguiente tarea es escoger un servicio para que sea automáticamente levantado, después que la suscripción sea creada. Un servicio que no se ha seleccionado tendrá que ser levantado manualmente para que la suscripción trabaje.

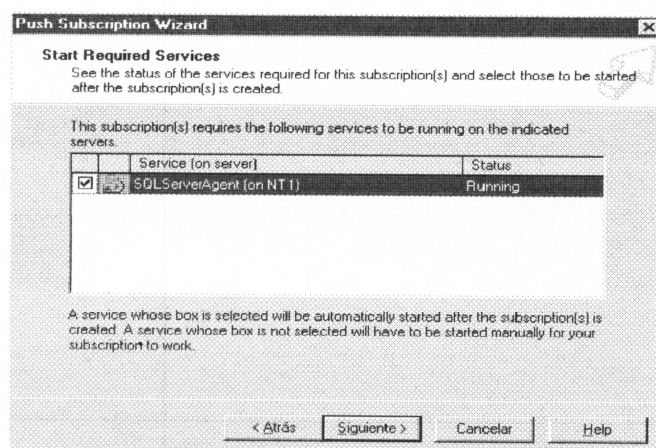


Gráfico 5.41. Servicios que se levantarán automáticamente con la suscripción.

La ventana de finalización del asistente indica como se configurará la suscripción con los parámetros anteriormente especificados.

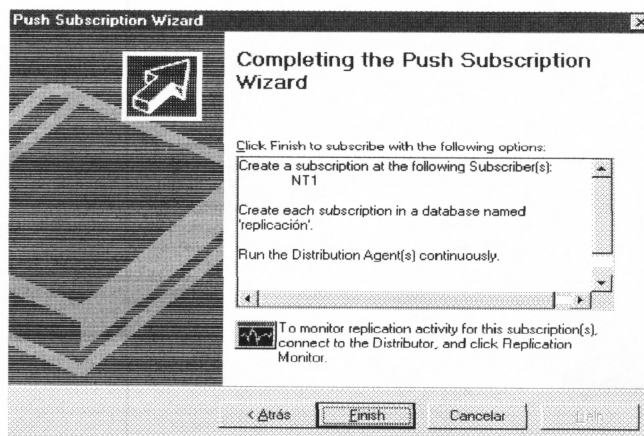


Gráfico 5.42. Ventana de finalización del asistente de configuración de suscripción.

Las siguientes ventanas indicarán, la del lado izquierdo el proceso de configuración del suscriptor y la del lado derecho la finalización en si del proceso.

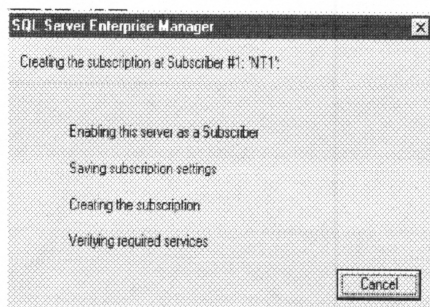


Gráfico 5.43. Pasos que seguirá el asistente.

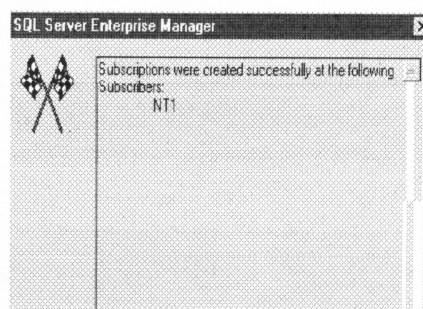


Gráfico 5.44. Ventana de finalización.

Luego de realizado esto con las configuraciones satisfactoriamente realizadas, el administrador de base de datos SQL Server 7.0 esta listo para realizar en forma planificada y automática la replicación de los artículos



seleccionados.

## CAPÍTULO V

### 5. CONCLUSIONES Y RECOMENDACIONES

#### 5.1. Conclusiones.

Luego de haber terminado el trabajo de estudio de este proyecto se puede concluir que:

- El manejo de la replicación en los Administradores de Bases de Datos Relacionales actuales, como lo son el SQL Server 7.0 y Oracle8 Enterprise Edition, con la ayuda de los conocidos asistentes en un ambiente visual se vuelve una tarea sumamente fácil de administrarla.
- Las replications se las puede hacer de diferentes maneras como son las replications de mezclas, las replications transaccionales y las replications de instantáneas. Todas estas muy fácilmente manejables.
- Para el manejo de replications se debe crear una base de datos de distribución con la cual se referenciará tanto el publicador como el suscriptor, y con estos realizar el proceso de traslado de datos de un servidor a otro, pero hablando de servidor, los que se hayan definido como tal, en una arquitectura centralizada como en una distribuida.
- Al tener el Administrador de Base de Datos en un sistema operativo de red se presenta una amplia seguridad en los datos tanto de la base de datos original como en la de replicación. Ya que para el ingreso a trabajar en el computador se hace necesario la contraseña de usuarios validos. Y como los Administradores de Bases de Datos Relacionales estudiados no son la excepción ya que se utilizan las autenticaciones, se logra tener mayor

confidencialidad en la información, hacia personas que no estén autorizadas para la manipulación de ésta.

- Al utilizar el proceso de replicación en los Administradores de Bases de Datos Relacionales, se logra mantener segura la información, en ocasiones en que la base de datos original se deteriore, o corrompa por múltiples causas, ya sean de software como de hardware. Teniendo la misma información en una base de datos espejo, lo único que se deberá hacer es refrescarla de forma inmediata o planificada según sea el tipo de replicación que se haya escogido o definido.

## 5.2 Recomendaciones.

Las recomendaciones que se pueden expresar de este estudio son:

- La persona encargada de la administración, deberá conocer también acerca de la teoría de bases de datos, tanto para la creación, como para el futuro mantenimiento de la base de datos.
- Se debe conocer el uso de todos los componentes y opciones manejadas por los asistentes que realizan los procesos de replicación en los Administradores de Bases de Datos Relacionales estudiados para así lograr sacar el máximo provecho de ellos.
- Según las necesidades, requerimientos y la importancia de la información de las empresas se debe seleccionar una de las diferentes maneras de realizar las replicas con la mayor especificación de uso.
- Según el tipo de replicación escogida se deberá especificar claramente en donde estará la base de datos de distribución, cual será el publicador (únicamente uno) y cual o cuales serán los suscriptores (uno, dos o más) y en donde estarán ubicadas.
- Si es que, se da el caso que se utilice una arquitectura distribuida se deberá especificar la completa configuración de la red así como las capas con las que trabajará el sistema (una, dos, tres capas, etc.).
- Se deberá tener muy en claro el conocimiento sobre el manejo, manipulación, administración, etc., del sistema operativo sobre el cual estará trabajando el RDBMS.
- Se necesitará conocer adecuadamente lo que se refiere al manejo y administración del RDBMS (incluyendo a la base de datos, tablas, usuarios, permisos, replications, etc.) en la que se va a mantener la información.

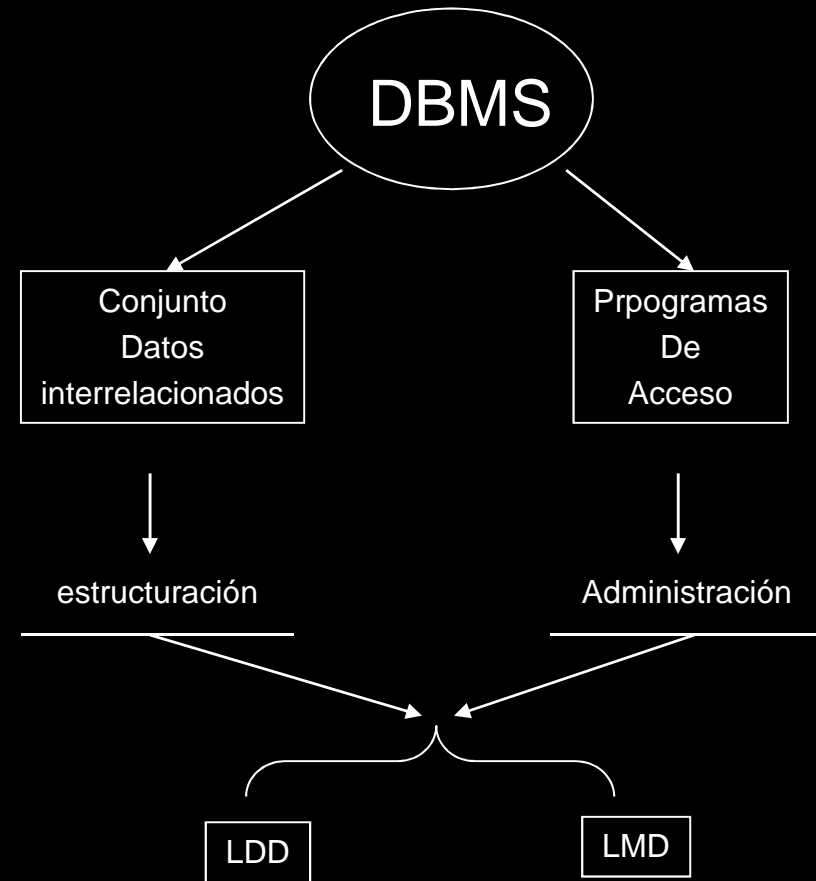
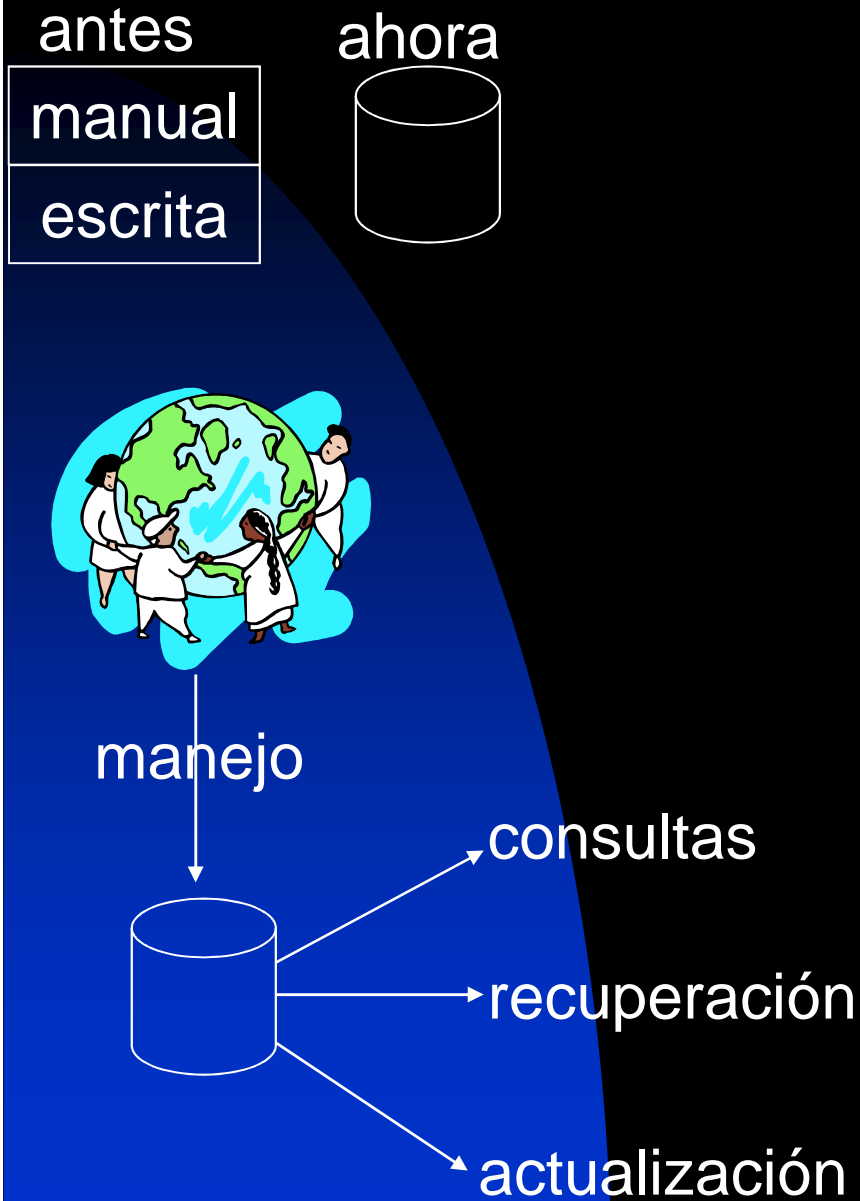
# REPLICACIÓN DE BASES DE DATOS

Disertante  
JORGE E. CHICAIZA R.  
Aspirante al título de  
Ingeniero en Sistemas

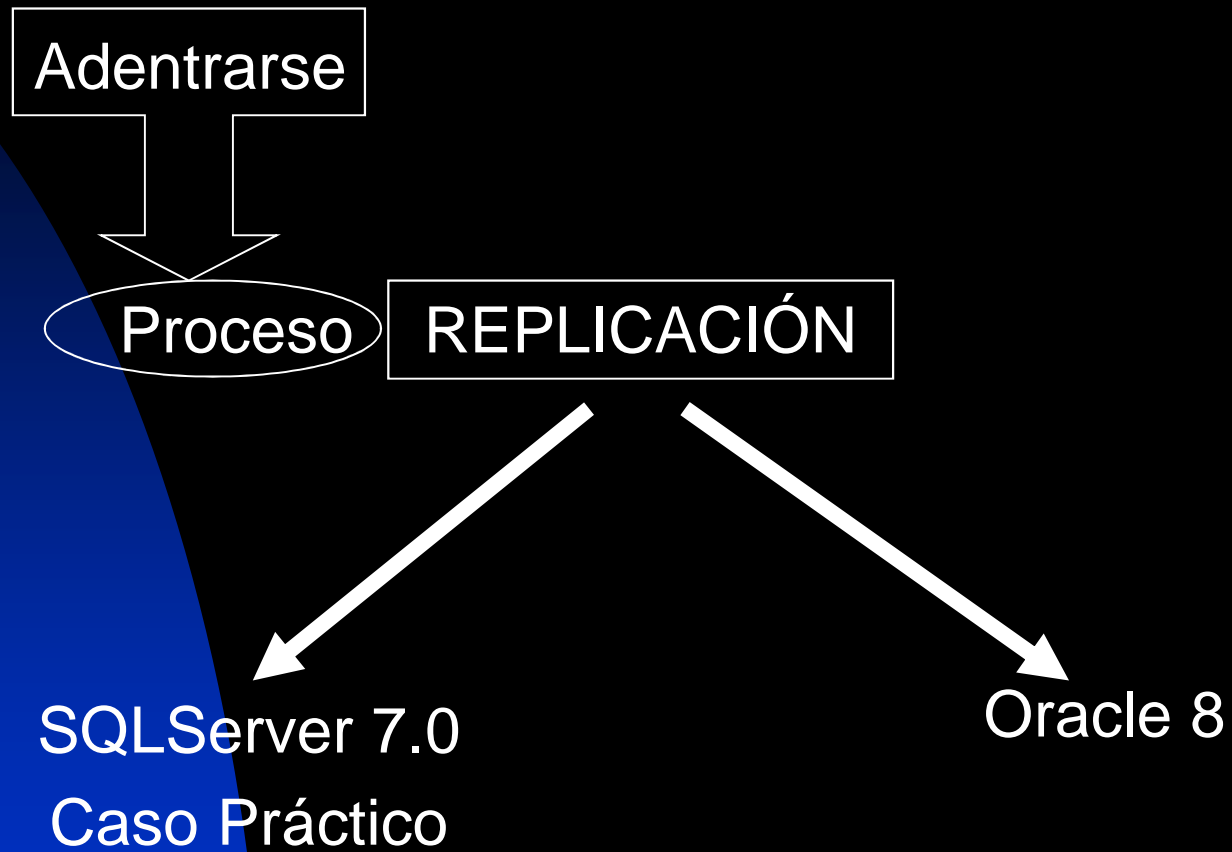
FACULTAD DE INGENIERIA  
EN SISTEMAS

UNIVERSIDAD TÉCNICA DE  
AMBATO

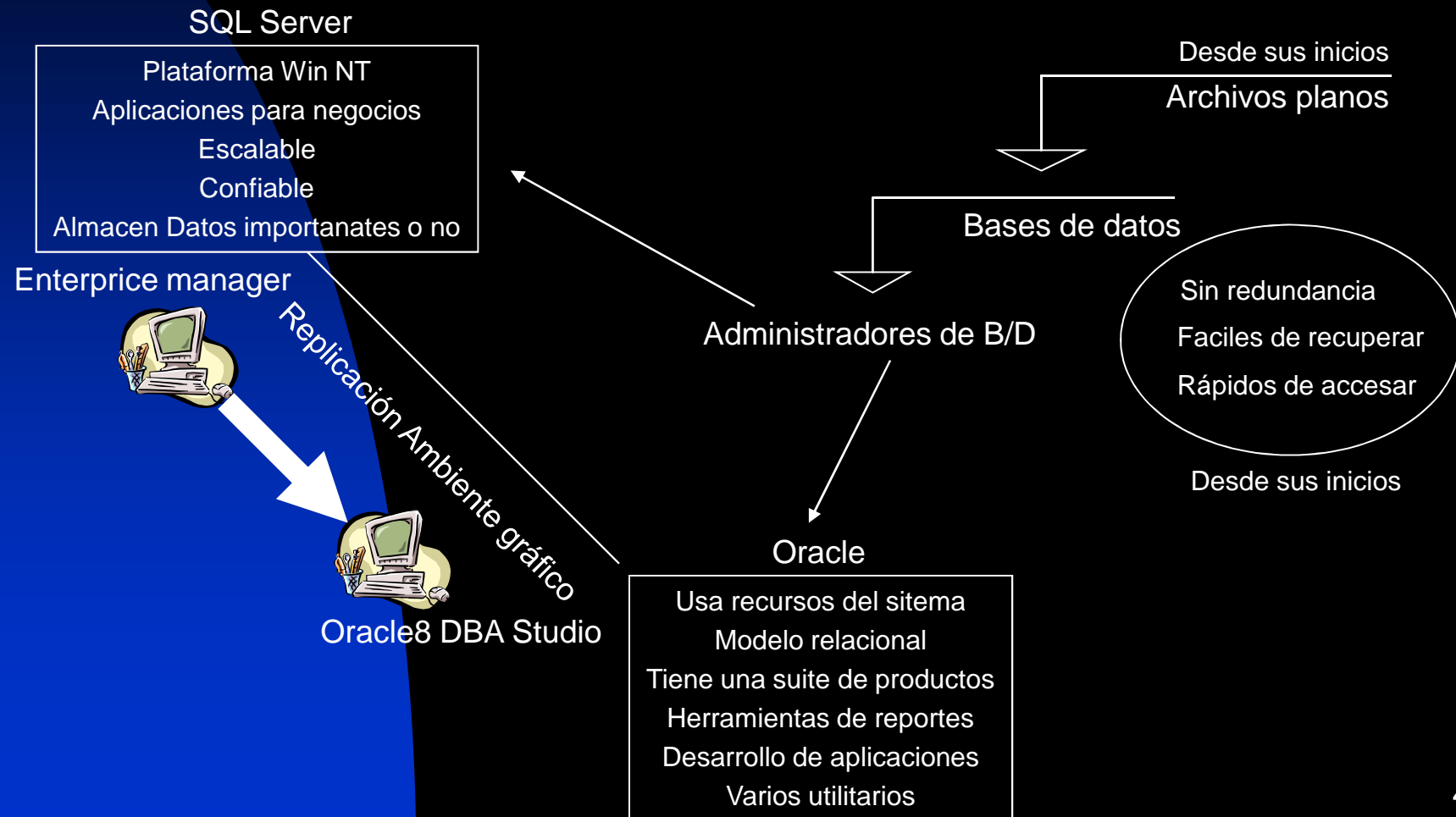
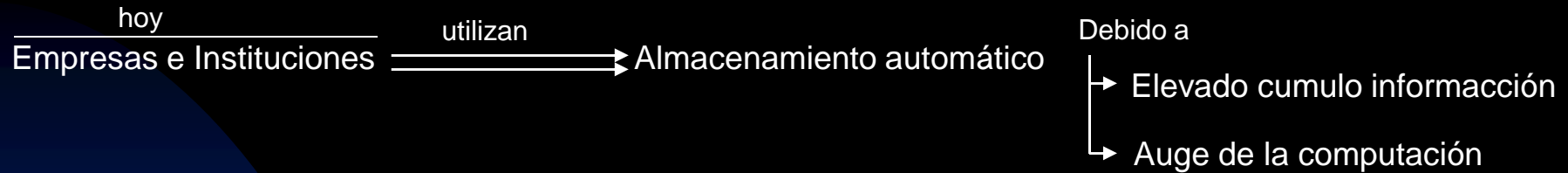
# Introducción



# Finalidad



# Antecedentes





# Objetivos

## General

Analizar replicación RDBMS → SQL Server 7.0 Caso Práctico  
→ Oracle8

## Específicos

Conceptualizar REPLICACIÓN

Definir tipos de replicación

Profundizar en el conocimiento del proceso de replicación

Presentar un caso práctico en SQL Server 7.0

# Metodología de investigación

## Bibliográfica – documental

Búsqueda de información

en fuentes

Primarias

Documentos

Secundarias

Libros

Revistas

Folletos

Periódicos

Internet

Ayuda de paquetes

Para

Ampliar

Profundizar

Analizar

Conocimientos

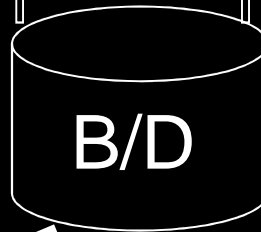
# Capítulo I

## Conjunto Datos

No redundantes  
Soporte Informático  
Independiente utilización  
Acceso Distintas Aplicaciones

## Requisitos

No redundancia  
Independencia  
Concurrencia



## Modelos de datos

Herramientas conceptuales describir datos



# Capítulo I

Recuperación Flexible de datos

Normalización

Modelo Relacional

Tablas bidimensionales

Filas

ocurrencia  
Clave P.

Columnas

Propiedades filas

Implantación

RDBMS

Desventajas

Puesta funcionamiento larga  
Personal especializado

En el uso presentan

Ventajas

Elimina inconsistencias  
Compartir datos aplicaciones  
Adaptan aplicaciones cambiantes  
Ahorro espacio (no redundancia)  
Mejores seguridades  
Craeción entornos alta disponibilidad

Centralizada

Agrupar todo en una sola plataforma

Distribuida

Basada Red

Administración  
Integridad  
Coherencia

B/D

Compleja

Homogenea Mismo RDBMS  
Heterogenea Otros RDBMS

Cliente/Servidor

Funciona en componentes comunicados para conseguir un fin común  
Cliente.- solicita recursos  
Servidor.- responde solicitud  
Sistemas comunicación normalizados

Por peticiones de diferentes:

Cientes  
Máquinas  
protocolos

# Capítulo I

Reducir carga sistema fuente  
Permitir procesos transacciones  
Mantener Sis. Secundarios recuperación  
Producir copias seleccionadas  
Mantener B/D adicional para pruebas

Para

Por qué? Necesidad

## REPLICACIÓN DATOS

Tablas. Columnas. Renglones  
Clases de objetos. Sp.  
Archivos de mensajes. Etc.

Qué?

Fuentes y destinos  
Esquemas:  
De donde  
Hacia donde

Grado sincronización

Sincronizados

Fuente y destino  
Múltiples sitios

No Sincronizados

Datos comprometidos  
Para propagarse

Recuperación y confiabilidad

Llevar Bitácora

Ver perdidas y repeticiones

Entregar "una y solo una vez"

Replicación múltiple

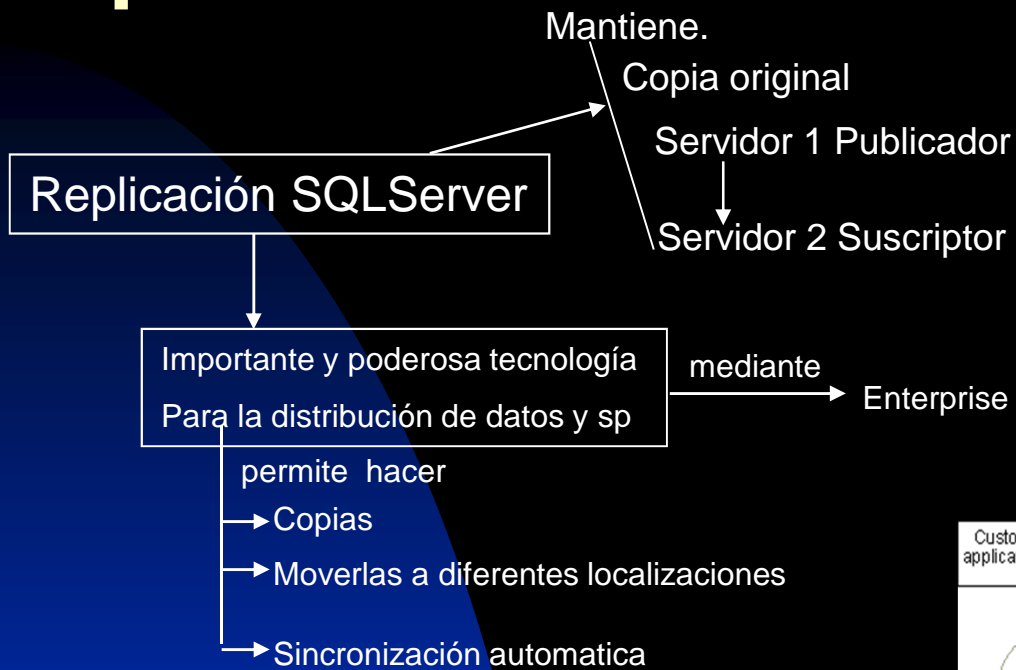
Copias en múltiples sitios

Actualización datos replicados

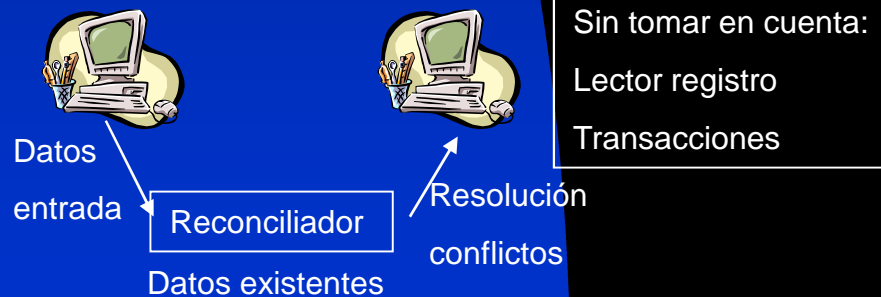
Establecer tiempos actualización:

Segundos. Minutos. Días

# Capítulo II



MEZCLA Usa los datos reales Tomando la ultima imagen de los datos

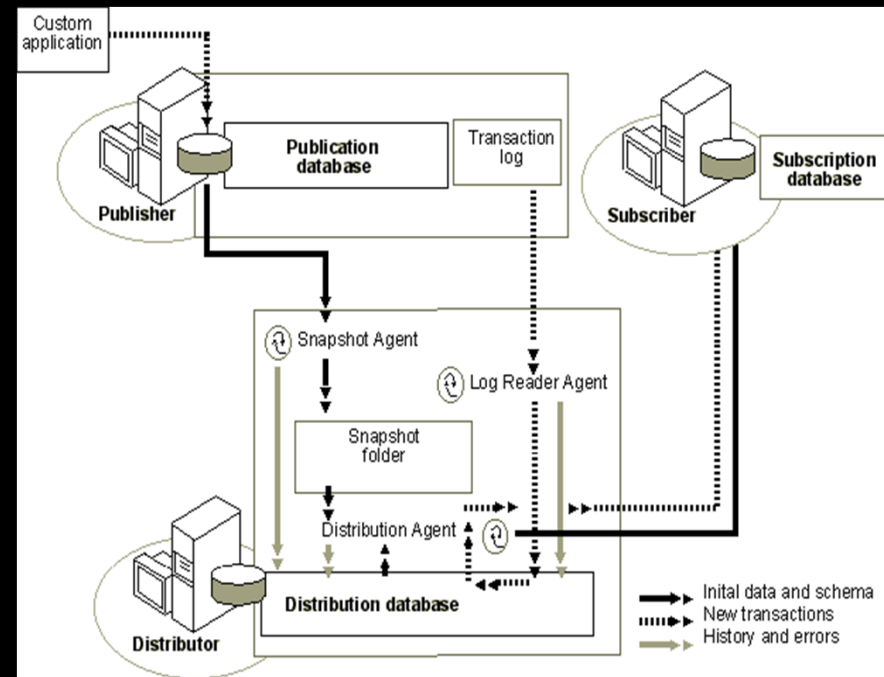


Tipos de replicación:

- Mezcla (merge)
- Transaccional (transactinal)
- Instantanea (snapshot)

## TRANSACCIONAL

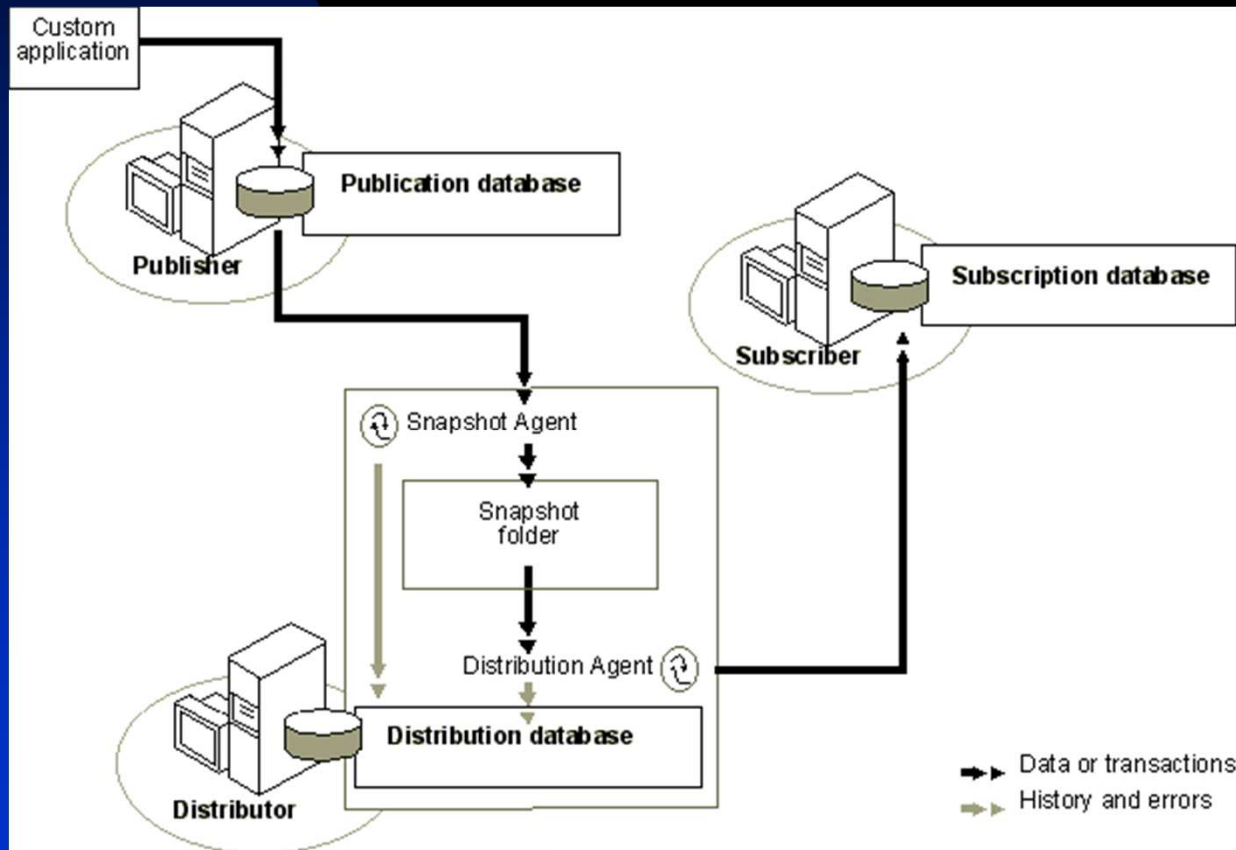
Trabaja utilizando una red y e log de transacciones



# Capítulo II

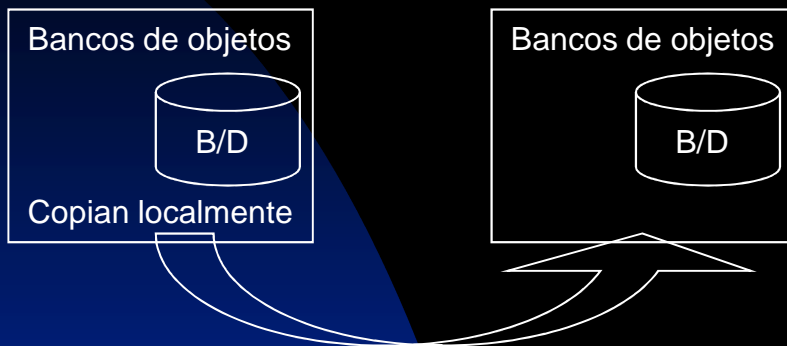
## REPLICACIÓN DE INSTANTANEA

Hace muy poco uso del procesador,  
No monitoriza los cambios en los datos

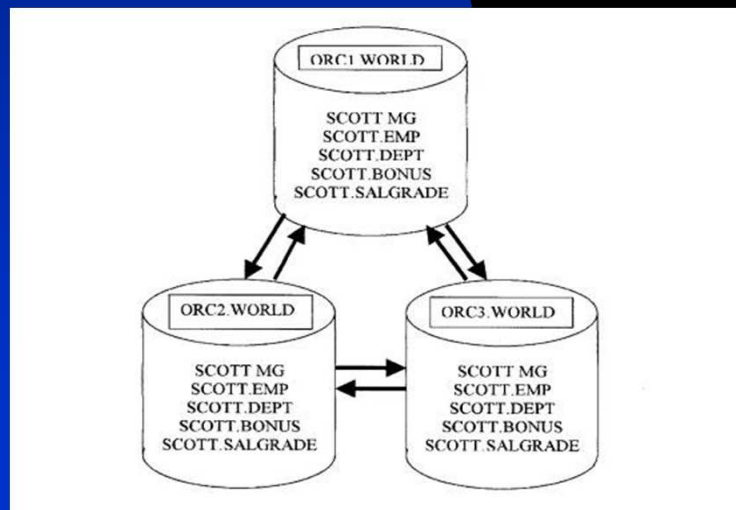


# Capítulo III

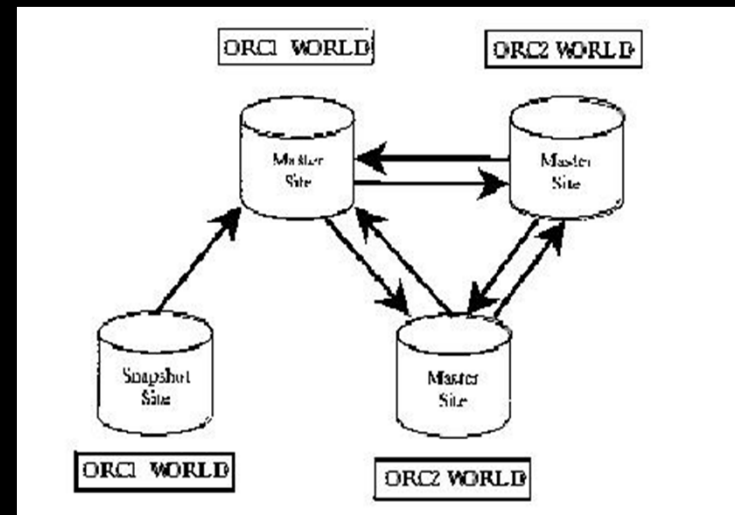
## Replicación



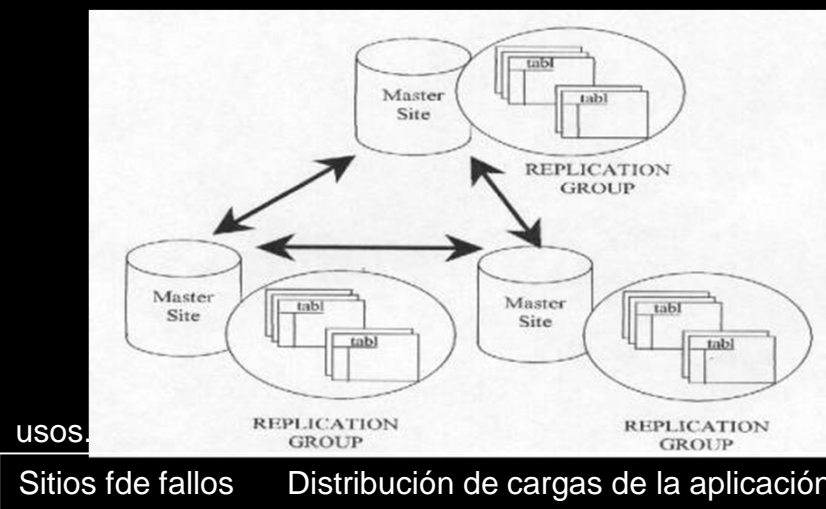
## Replicación Objetos, Grupos y Sitios



## Sitios de Replicación



## Replicación Multimaestro



USOS

Sitios fde fallos

Distribución de cargas de la aplicación



# Capítulo III

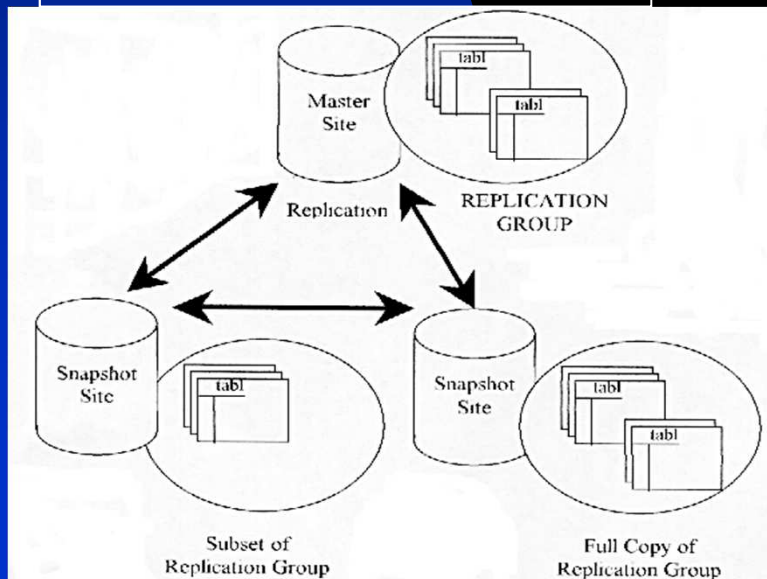
## Replicación de Instantaneas

Contiene una replica completa o parcial de  
Una tabla maestra objetivo desde un solo  
Punto en tiempo real

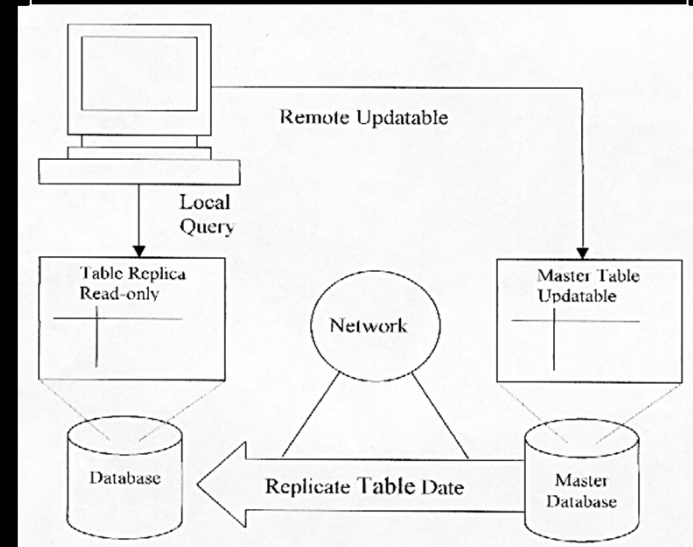
Uso:

- En manejo de información descargada
- Distribución de la información
- Transporte de la información
- En Ambientes desconectados

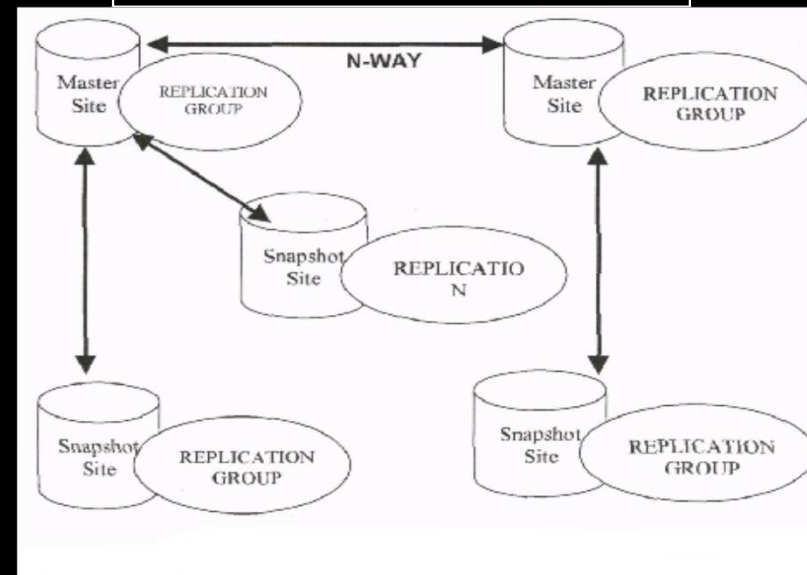
## Instantaneas Actualizables



## Instantaneas de solo Lectura



## Configuraciones Híbridas



# Capítulo V

## CONCLUSIONES

El manejo de la replicación en los Administradores de Bases de Datos Relacionales actuales, como lo son el SQL Server 7.0 y Oracle8 Enterprise Edition, con la ayuda de los conocidos asistentes en un ambiente visual se vuelve una tarea sumamente fácil de administrarla.

Las replicaciones se las puede hacer de diferentes maneras como son las replicaciones de mezclas, las replicaciones transaccionales y las replicaciones de instantáneas. Todas estas muy fácilmente manejables.

Para el manejo de replicaciones se debe crear una base de datos de distribución con la cual se referenciará tanto el publicador como el suscriptor, y con estos realizar el proceso de traslado de datos de un servidor a otro, pero hablando de servidor, los que se hayan definido como tal, en una arquitectura centralizada como en una distribuida.

Al tener el Administrador de Base de Datos en un sistema operativo de red se presenta una amplia seguridad en los datos tanto de la base de datos original como en la de replicación. Ya que para el ingreso a trabajar en el computador se hace necesario la contraseña de usuarios validos. Y como los Administradores de Bases de Datos Relacionales estudiados no son la excepción ya que se utilizan las autenticaciones, se logra tener mayor confidencialidad en la información, hacia personas que no estén autorizadas para la manipulación de ésta.

Al utilizar el proceso de replicación en los Administradores de Bases de Datos Relacionales, se logra mantener segura la información, en ocasiones en que la base de datos original se deteriore, o corrompa por múltiples causas, ya sean de software como de hardware. Teniendo la misma información en una base de datos espejo, lo único que se deberá hacer es refrescarla de forma inmediata o planificada según sea el tipo de replicación que se haya escogido o definido.

# Capítulo V

## RECOMENDACIONES

La persona encargada de la administración, deberá conocer también acerca de la teoría de bases de datos, tanto para la creación, como para el futuro mantenimiento de la base de datos.

Se debe conocer el uso de todos los componentes y opciones manejadas por los asistentes que realizan los procesos de replicación en los Administradores de Bases de Datos Relacionales estudiados para así lograr sacar el máximo provecho de ellos.

Según las necesidades, requerimientos y la importancia de la información de las empresas se debe seleccionar una de las diferentes maneras de realizar las replicas con la mayor especificación de uso.

Según el tipo de replicación escogida se deberá especificar claramente en donde estará la base de datos de distribución, cual será el publicador (únicamente uno) y cual o cuales serán los suscriptores (uno, dos o más) y en donde estarán ubicadas.

Si es que, se da el caso que se utilice una arquitectura distribuida se deberá especificar la completa configuración de la red así como las capas con las que trabajará el sistema (una, dos, tres capas, etc.).

Se deberá tener muy en claro el conocimiento sobre el manejo, manipulación, administración, etc., del sistema operativo sobre el cual estará trabajando el RDBMS.

Se necesitará conocer adecuadamente lo que se refiere al manejo y administración del RDBMS (incluyendo a la base de datos, tablas, usuarios, permisos, replications, etc.) en la que se va a mantener la información.

## **BIBLIOGRAFÍA.**

### **Textos**

ADKOLI Anand y VELPURI Rama, Manual de ORACLE8 para Windows NT; La guía práctica para administrar ORACLE8 en Windows NT, Sánchez García José Ignacio y Quijada Arteaga Ma. Pilar, Primera edición, España, Mc Graw – Hill, © 1999, Oracle NT Handbook.

GOMEZ, Ángel Lucas (et all), Diseño y gestión de sistemas de bases de datos, Primera edición, Colombia, Paraninfo, © 1993.

LONEY Kevin, ORACLE8 Manual del administrador, Todo lo que un administrador de sistemas debe saber para la gestión eficaz y eficiente de una base de datos, Vuelapluma S. L., Primera edición, España, Mc Graw – Hill, © 1999, ORACLE( DBA Handbook.)

SOUKUP Ron y DELANEY Kalen, A fondo Microsoft SQL Server 7.0, la guía del desarrollador sobre diseño, arquitectura e implementación, Vallejo Pinto José Ángel, Primera edición, España, Mc Graw – Hill, © 1999, Incide SQL Server 7.0.

### **Revistas:**

COMPU MAGAZINE, Número 50, Septiembre, 1992.

COMPU MAGAZINE, Número 51, Octubre, 1992.

### **Direcciones Internet:**

<http://www.sqlmax.com>

<http://www.microsoft.com/latam/sql>

<http://cybercursos.com>

### **Otros documentos**

Books Online de Microsoft SQL Server 7.0, Microsoft Corporation, © 1988 – 1998.

Librería de documentación de Oracle8 Personal Edition, Oracle Corporation, Redwood Shred, California, © 1998.

Librería de documentación de Oracle8i, Oracle Corporation, Redwood City, California, © 1998.

## **GLOSARIO DE TÉRMINOS**

Distribuidor.- Base de datos que maneja toda la información en el proceso de replicación.

Publicador.- Parte encargada de mantener los datos que van a ser replicados.

Suscriptor.- Parte encargada de poner los datos de replicación en la base de datos espejo.