

UNIVERSIDAD “TÉCNICA DEL NORTE”



**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
ESCUELA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**TESIS PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERA
EN SISTEMAS COMPUTACIONALES.**

TEMA:

**“FRAGMENTACIÓN DE TABLAS EN BASES DE DATOS
DISTRIBUIDAS”**

APLICATIVO:

**“IMPLEMENTACIÓN DE UNA BASE DE DATOS DISTRIBUIDA EN
EL SISTEMA ACADÉMICO DE LA FICA”**

AUTORA:

LÓPEZ TULCANAZA VERÓNICA ALEXANDRA

DIRECTOR:

Ing. MIGUEL ORQUERA.

**Ibarra – Ecuador
2010**



ÍNDICE

CAPITULO I	3
BASES DE DATOS DISTRIBUIDAS	3
OBJETIVOS DE LAS BASES DE DATOS DISTRIBUIDAS	3
SISTEMA DE GESTIÓN DE BASE DE DATOS DISTRIBUIDAS (SGBDD)	3
DOCE REGLAS DE LAS BASES DE DATOS DISTRIBUIDAS	4
BASES DE DATOS DISTRIBUIDAS VRS. CENTRALIZADA	4
CAPITULO II	5
FRAGMENTACIÓN DE TABLAS	5
TIPOS DE FRAGMENTACIÓN	5
FRAGMENTACIÓN HORIZONTAL	5
FRAGMENTACIÓN VERTICAL	6
FRAGMENTACIÓN MIXTA	6
GRADOS DE FRAGMENTACIÓN	6
CAPITULO III	7
PARTICIONAMIENTO EN ORACLE	7
PARTICIONAMIENTO POR RANGO	7
PARTICIONAMIENTO POR LISTA	7
PARTICIONAMIENTO POR HASH	7
PARTICIÓN POR COMPOSICIÓN	8
EXTENSIONES DE PARTICIONAMIENTO	8
CAPITULO IV	9
ARQUITECTURA DE UN SISTEMA MANEJADOR DE BASES DE DATOS DISTRIBUIDAS	9
ARQUITECTURA DE UN SISTEMA DE MANEJO DE BASES DE DATOS DISTRIBUIDOS HOMOGÉNEOS	9
TRANSPARENCIA	10
NIVELES DE TRANSPARENCIA	10
TRANSPARENCIA DE LOCALIZACIÓN	10
TRANSPARENCIA DE FRAGMENTACIÓN	10
TRANSPARENCIA DE REPLICA	11
FORMAS DE RÉPLICAS	11
CAPITULO V	12
ADMINISTRADOR DE LA BASE DE DATOS	12



CAPITULO I

BASES DE DATOS DISTRIBUIDAS.

Es una colección de datos que pertenecen lógicamente a un sólo sistema, pero se encuentra físicamente esparcido en varios "sitios" de la red.

OBJETIVOS DE LAS BASES DE DATOS DISTRIBUIDAS

Es una especie de objeto virtual, cuyos componentes se almacenan físicamente en varias bases de datos “reales” distintas, ubicadas en diferentes sitios.

Cada sitio tiene sus propias bases de datos “reales” locales, sus propios usuarios locales, sus propios DBMS y programas para la administración de transacciones y su propio administrador local de comunicación de datos.

SISTEMA DE GESTIÓN DE BASE DE DATOS DISTRIBUIDAS (SGBDD)

Es necesaria una aplicación que actúe de interfaz entre el usuario, los modelos y el sistema físico. Esta es la función que desempeñan los SGBD. Una base de datos permite al usuario acceder a datos almacenados sin que él deba saber cómo están organizados esos datos en el sistema, garantiza que ningún usuario sin autorización pueda ver o manipular los datos.

OBJETIVOS DE LOS SGBDD

- Permitir la definición de todos los datos.
- Permitir la manipulación de los datos:
- Permitir los accesos concurrentes.



DOCE REGLAS DE LAS BASES DE DATOS DISTRIBUIDAS

1. Autonomía Local.
2. No dependencia de un sitio central:
3. Operación Continua
4. Independencia con respecto a la localización
5. Independencia con respecto a la fragmentación.
6. Independencia de réplica.
7. Procesamiento Distribuido de Consultas.
8. Manejo Distribuido de Transacciones.
9. Independencia con respecto al equipo.
10. Independencia con respecto al Sistema Operativo.
11. Independencia con respecto a la red.
12. Independencia con respecto al DBMS.

BASES DE DATOS DISTRIBUIDAS VRS. CENTRALIZADA

CENTRALIZADO	DISTRIBUIDO
Control centralizado: Un solo DBA	Control jerárquico: DBA global y DBAs locales
Independencia de Datos: Organización de los datos es transparente para el programador	Transparencia en la Distribución: Localización de los datos es un aspecto adicional de independencia de datos
Reducción de redundancia: Una sola copia de datos que se comparta	Replicación de Datos: Copias múltiples de datos que incrementa la localidad y la disponibilidad de datos
Estructuras físicas complejas para accesos eficientes	No hay estructuras intersitios. Uso de optimización global para reducir transferencia de datos
Seguridad	Problemas de seguridad Intrínsecos



CAPITULO II

FRAGMENTACIÓN DE TABLAS

FRAGMENTACIÓN

Consiste en dividirla en fragmentos menores, estos se guarda en sitio diferente.

VENTAJAS	DESVENTAJAS
Mejorar el rendimiento de las aplicaciones	Disminuye la eficiencia en las aplicaciones que trabajan con varios fragmentos
Da una respuesta eficiente	La comprobación de las restricciones de integridad puede ser más costosa.
Los fragmentos permiten aumentar el número de ejecuciones concurrentes	

CONDICIONES PARA DEFINIR FRAGMENTOS

Se deben cumplir tres reglas durante el proceso de fragmentación.

- ✓ Completitud
- ✓ Reconstrucción
- ✓ Disyunción

TIPOS DE FRAGMENTACIÓN.

FRAGMENTACIÓN HORIZONTAL.

Se realiza sobre las tuplas de la relación, es decir que cada fragmento será un subconjunto de las tuplas de la relación.

Una tabla T se divide en subconjuntos, T1, T2, ...Tn. Los fragmentos se definen mediante una operación de selección. Su reconstrucción se realizará mediante la unión de los fragmentos componentes.



FRAGMENTACIÓN VERTICAL

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

La partición vertical resulta más complicada que la horizontal. Esto se debe al aumento del número total de alternativas que tenemos disponibles.

Existen dos enfoques heurísticos para la fragmentación vertical de relaciones: Agrupación y Escisión

FRAGMENTACIÓN MIXTA

La fragmentación mixta puede llevarse a cabo de tres formas diferentes: desarrollando primero la fragmentación vertical y, posteriormente, aplicando la fragmentación horizontal sobre los fragmentos verticales (denominada partición VH), o aplicando primero una división horizontal para luego, sobre los fragmentos generados, desarrollar una fragmentación vertical (llamada partición HV), o bien, de forma directa considerando la semántica de las transacciones.

GRADOS DE FRAGMENTACIÓN

El grado de fragmentación puede considerarse en el que cada tupla o atributo forme un fragmento, o una ausencia de la división, considerando a las relaciones como unidades de fragmentación.



CAPITULO III

PARTICIONAMIENTO EN ORACLE

PARTICIONAMIENTO POR RANGO

Los datos se distribuyen de acuerdo con el rango de valores de la clave de particionamiento. La distribución de datos es continua.

Se requiere que los registros estén identificado por un “partition key” relacionado por un predefinido rango de valores, el valor de las columnas “partition key” determina la partición a la cual pertenecerá el registro.

PARTICIONAMIENTO POR LISTA

El particionamiento por lista no soporta claves de particionamiento formada por varios atributos.

La clave de particionado es una lista de valores, que determina cada una de las particiones, la distribución de datos se define por el listado de valores de la clave de partición. El valor DEFAULT sirve para definir la partición donde iran el resto de registros que no cumplen ninguna condición de las diferentes particiones.

PARTICIONAMIENTO POR HASH

La correspondencia entre las filas y las particiones se realiza a través de una función de hash. Es una opción útil cuando:

- Se desconoce la correspondencia en función de los rangos o no hay unos criterios de particionado claros.
- El rango de las particiones difiere sustancialmente o es difícil balancearla manualmente.



La clave de particionado es una función hash, aplicada sobre una columna, que tiene como objetivo realizar una distribución equitativa de los registros sobre las diferentes particiones.

PARTICIÓN POR COMPOSICIÓN

Se conjuga el uso de dos particionados a la vez. Primero la tabla se particiona con un primer método de distribución de datos y luego cada partición se vuelve a dividir en subparticiones utilizando un segundo método de distribución de datos.

EXTENSIONES DE PARTICIONAMIENTO

Oracle Database 11g brinda extensiones de particionamiento, su principal objetivo es:

- Mejorar significativamente la capacidad de administración de una tabla particionada.
- Extender la flexibilidad para definir una clave de particionamiento.

Las extensiones son:

- Particionamiento por Intervalos
- Particionado System
- Particionamiento basado en Columnas Virtuales



CAPITULO IV

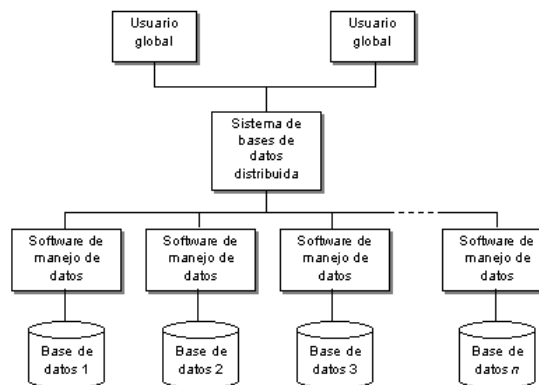
ARQUITECTURA DE UN SISTEMA MANEJADOR DE BASES DE DATOS DISTRIBUIDAS

Los sistemas de datos distribuidos están divididos en dos clases:

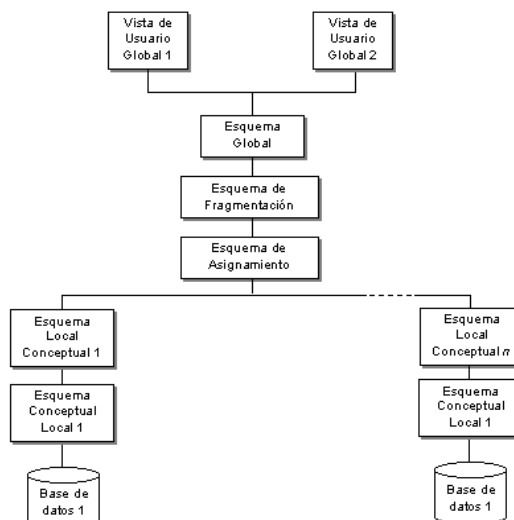
1. Sistemas de manejo de bases de datos distribuidos homogéneos
2. Sistemas de manejo de bases de datos distribuidos heterogéneos

ARQUITECTURA DE UN SISTEMA DE MANEJO DE BASES DE DATOS

DISTRIBUIDOS HOMOGÉNEOS



Se parece a un sistema centralizado, a diferencia que estos sus datos se distribuyen en varios sitios comunicados por la red. No existen usuarios locales y todos ellos accesan la base de datos a través de una interfaz global. Para manejar los aspectos de la distribución, se deben agregar dos niveles a la arquitectura estándar ANSI-SPARC como se muestra





El esquema de fragmentación describe la forma en que las relaciones globales se dividen entre las bases de datos locales.

El esquema de asignamiento especifica el lugar en el cual cada fragmento es almacenado. De aquí, los fragmentos pueden migrar de un sitio a otro en respuesta a cambios en los patrones de acceso.

TRANSPARENCIA

El objetivo es ocultar al usuario los detalles de diseño, es decir, el usuario no tiene que saber que se encuentra trabajando sobre un sistema distribuido, por ejemplo la independencia de los datos es una forma de transparencia.

NIVELES DE TRANSPARENCIA.

Primer Nivel. Se soporta la transparencia de red.

Segundo Nivel. Se permite la transparencia de replicación de datos.

Tercer Nivel. Se permite la transparencia de la fragmentación.

Cuarto Nivel. Se permite transparencia de acceso (por medio de un lenguaje de manipulación de datos)

TRANSPARENCIA DE LOCALIZACIÓN.

La transparencia de localización le permite acceder al usuario a los datos sin tener en cuenta la ubicación de estos, es decir debe ser transparente al usuario, ya q este no necesita saber dónde está el dato para utilizarlo.

TRANSPARENCIA DE FRAGMENTACIÓN

El acceso a una base de datos distribuida debe hacerse en forma transparente. Los usuarios deben comportarse, como si los datos en realidad no estuvieran fragmentados, lo cual es necesario por razones de rendimiento.



TRANSPARENCIA DE REPLICA

La transparencia sobre replicación de datos se refiere a que si existen réplicas de objetos de la base de datos, esta debe ser controlada por el SMBDD, más no por el usuario.

El usuario no necesita saber sobre la replicación de los datos, la función principal de la transparencia de replicación es la de mantener la consistencia entre las copias, esta funciona en forma transparente a las aplicaciones.

FORMAS DE RÉPLICAS

Las réplicas pueden asumir los siguientes formatos:

- ✓ Periódica
- ✓ Continua
- ✓ Check-in / Check-out

CREACIÓN DE RÉPLICAS

Es posible crear réplicas completas o parciales:

Réplica completa: Contiene todos los documentos y elementos de diseño de la base de datos como son: formularios y agentes.

Réplica parcial: Contiene documentos abreviados, resúmenes de documentos o documentos cuyos campos contienen información especificada, documentos preseleccionados, documentos de carpetas o vistas especificadas.

Este tipo de réplica le resultará útil cuando sólo necesite una parte de la base de datos y quiere ahorrar espacio en el disco.



CAPITULO V

ADMINISTRADOR DE LA BASE DE DATOS

Sus responsabilidades son: planear y crear bases de datos, administrar el acceso, los recursos y estructuras (tanto físicas como lógicas) de las mismas y administrar usuarios y sus privilegios.

COMPARACIÓN

CENTRALIZADO	DISTRIBUIDO
Existe un control total por parte del administrador de la base de datos global. Por razones de seguridad, hay gran énfasis en el control y este debe centralizarse.	Hay un alto grado de autonomía por parte de los administradores locales. En bases de datos distribuidas es posible identificar una estructura de control jerárquica basada en el administrador de la bases de datos global y en los administradores de las bases de datos locales. Estos pueden llegar a tener mucha autonomía.
Deben preservar la atomicidad de las transacciones en presencia de fallas. La recuperación debe hacerse hacia atrás o hacia delante, dependiendo de si la transacción se ha comprometido o no.	Deben preservar la atomicidad de las transacciones aun en casos de fallas en algunos sitios involucrados. La recuperación debe hacerse basado en protocolos complejos de compromiso en dos o en tres fases.
El DBA tiene un control centralizado a través del acceso autorizado a los datos. Pero es más vulnerable a las violaciones de seguridad y privacidad que en archivos separados, desde el punto de vista de la globalidad de la información	Los administradores locales tienen el mismo problema de vulnerabilidad y privacidad de los administradores globales. Por la autonomía de cada sitio, los propietarios de los datos tienen sus propias protecciones, a veces buenas, a veces no tanto. Las bases de datos distribuidas presentan los problemas intrínsecos a los sistemas distribuidos ya que las redes representan un punto débil en protección.



CHAPTER I

DISTRIBUTED DATABASES

It is a collection of information that belong logically to one only system, but it is physically spread in several "sites" of the network

AIMS OF THE DISTRIBUTED DATABASES

It is a species of virtual object, which components are stored physically in several "royal" different databases located in different sites.

Every site has his own "royal" local databases, his own local users, his own DBMS and programs for the administration of transactions and his own local administrator of communication of information.

SYSTEM OF BASE MANAGEMENT OF INFORMATION DISTRIBUTED

(SGBDD)

There is necessary an application that acts of interface between the user, the models and the physical system. This one is the function that the SGBD recover. A database allows to the user to accede to stored information without he should know how this information is organized in the system, guarantees that no user without authorization could see or manipulate the information.

AIMS OF THE SGBDD

- To allow the definition of all the information.
- To allow the manipulation of the information.
- To allow the competing accesses.

TWELVE RULES OF THE DISTRIBUTED DATABASES

1. Local Autonomy.
2. Not dependence of a central site:
3. Operation Continues
4. Independence with regard to the location



5. Independence with regard to the fragmentation.
6. Independence of reply.
7. Processing Distributed of Consultations.
8. Managing Distributed of Transactions.
9. Independence with regard to the equipment.
10. Independence with regard to the Operating system.
11. Independence with regard to the network.
12. Independence with regard to the DBMS.

DISTRIBUTED DATABASES VRS. CENTRALIZED

CENTRALIZED	DISTRIBUTED
Centralized control: An alone DBA	Hierarchic control: Global DBA and local DBAs
Independence of Information: Organization of the information It is transparent for the programmer	Transparency in the Distribution: Location of the information is one Additional aspect of independence of information
Reduction of redundancy: An alone copy of information That is shared	Replicación of Information: Multiple copies of information That increases the locality and the availability of information
Physical complex structures For efficient accesses	There are no structures intersites. Use of global optimization to reduce transfer of information
Security	Safety problems Intrinsic



CHAPTER II

FRAGMENTATION OF TABLES

FRAGMENTATION

It consists of dividing her in minor, these fragments guards in different site.

ADVANTAGES	DISADVANTAGES
To improve the performance of the applications	It diminishes the efficiency in the applications that work with several fragments
It gives an efficient response	The checking of the restrictions of integrity can be costlier.
The fragments allow to increase the number of competing executions	

CONDITIONS TO DEFINE FRAGMENTS

Three rules must be fulfilled during the process of fragmentation.

- ✓ Completitud
- ✓ Reconstruction
- ✓ Disjunction

TYPES OF FRAGMENTATION

HORIZONTAL FRAGMENTATION

There is realized on the tuplas of the relation, that is to say that every fragment will be a subset of the tuplas of the relation.

A table T divides in subsets, T1, T2... Tn. The fragments are defined by means of an operation of selection. His reconstruction will realize by means of the union of the fragments components.



VERTICAL FRAGMENTATION

The aim of the vertical fragmentation consists of dividing the relation in a such set of smaller than relations any of the user's applications only use a fragment. On this frame, an ideal fragmentation is that one that produces a scheme of division that it minimizes the time of execution of the applications that use these fragments.

The vertical partition turns out to be more complicated than the horizontal one. This owes to the increase of the total number of alternatives that we have available.

Two heuristic approaches exist for the vertical fragmentation of relations: Group and Split

MIXED FRAGMENTATION

The mixed fragmentation can be carried out of three different forms: developing first the vertical fragmentation and, later, applying the horizontal fragmentation on the vertical fragments (partition called VH), or applying first a horizontal division for then, on the generated fragments, to develop a vertical fragmentation (so called partition HV), or, of direct form considering the semantics of the transactions.

DEGREES OF FRAGMENTATION

It can be considered to be the degree of fragmentation in that every tupla or attribute should form a fragment, or an absence of the division, considering to the relations as units of fragmentation.



CHAPTER III

PARTICIONAMIENTO IN ORACLE

PARTICIONAMIENTO FOR RANGE

The information is distributed in agreement by the range of values of the key of particionamiento. The distribution of information is constant.

There is needed that the records are identified by one "partition key " related by a predefined range of values, the value of the columns " partition key " determines the partition to which the record will belong.

PARTICIONAMIENTO FOR LIST

The particionamiento for list does not support keys of particionamiento formed by several attributes.

The key of partition is a list of values, which each of the partitions determines, the distribution of information is defined by the list of values of the key of partition. The value DEFAULT serves to define the partition where continue the rest of records that do not fulfill any condition of the different partitions.

PARTICIONAMIENTO FOR HASH

The correspondence between the rows and the partitions is realized across a function of hash. It is a useful option when:

- The correspondence is not known depending on the ranges or there are no a few criteria of partition clear.
- The range of the partitions differs substantially or it is difficult to balance it manually.



The key of partition is a function hash, applied on a column, which has as aim realize an equitable distribution of the records on the different partitions.

PARTITION FOR COMPOSITION

There brings together the use of two partitions the time. First the table partition with the first method of distribution of information and then every partition returns to divide in subpartitions using the second method of distribution of information.

PARTICIONAMIENTO's EXTENSIONS

Oracle Database 11g offers extensions of partition, his principal aim is:

- Improves significantly the capacity of administration of a table partition.
- To extend the flexibility to define a key of partition

The extensions are:

- Partition for Intervals
- Partition System
- Partition based on Virtual Columns



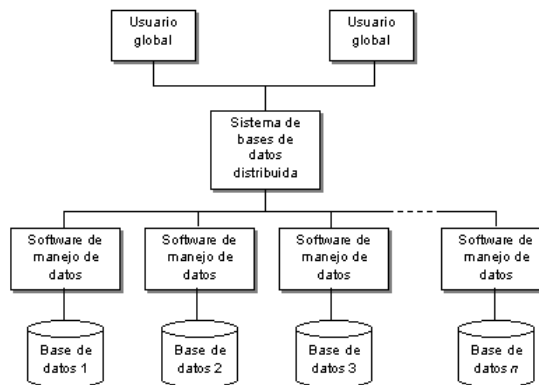
CHAPTER IV

ARCHITECTURE OF A SYSTEM OF MANAGING OF DISTRIBUTED DATABASES

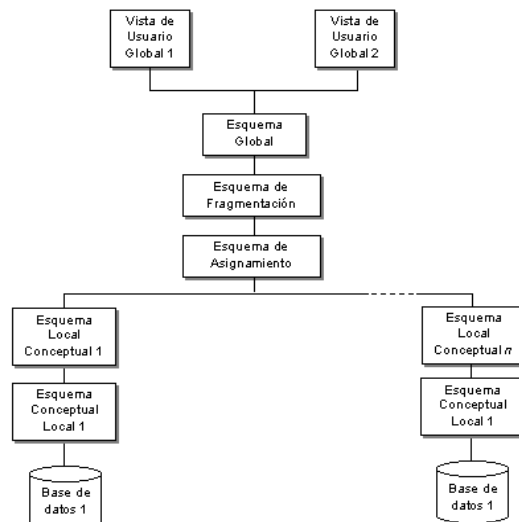
The systems of distributed information are divided in two classes:

1. Systems of managing database distributed homogeneous
2. Systems of managing database distributed heterogeneous

ARCHITECTURE OF A SYSTEM OF MANAGING DATABASE DISTRIBUTED HOMOGENEOUS



It looks like a centralized system, to difference that these his information distributes in several sites communicated by the network. Local users do not exist and all of them access the database across a global interface. To handle the aspects of the distribution, they must add two levels to the standard architecture ANSI-SPARC like it shows.





The scheme of fragmentation describes the form in which the global relations divide between the local databases.

The scheme of assignment I lie it specifies the place in which every fragment is stored. Of here, the fragments can migrate from a site to other one in response to changes in the bosses of access.

TRANSPARENCY

The aim is to conceal from the user the details of design, that is to say, the user does not have to know that it is working on a distributed system, for example the independence of the information is a form of transparency.

LEVELS OF TRANSPARENCY

The first Level. The transparency of network is supported.

The second Level. The transparency is allowed of replicación of information.

The third Level. The transparency of the fragmentation is allowed.

Fourth Level. There is allowed transparency of access (by means of a language of manipulation of information)

TRANSPARENCY OF LOCATION

The transparency of location allows him to accede to the user to the information without bearing the location in mind of these, that is to say it must be transparent to the user, already q this one does not need to know where the information is to use it.



TRANSPARENCY OF FRAGMENTATION

The access to a distributed database must be done in transparent form. The users must behave, as if the information actually was not fragmented, which is necessary for reasons of performance.

TRANSPARENCY OF REPLY

The transparency on reply of information refers that if there exist replies of objects of the database, this one must be controlled for the SMBDD, more not for the user.

FORMS OF REPLIES

The replies can assume the following formats:

- ✓ Periodic
- ✓ It continues
- ✓ Check-in / Check-out

CREATION OF REPLIES

It is possible to create complete or partial replies:

Reply completes: It contains all the documents and elements of design of the database since they are: forms and agents.

Partial reply: It contains brief documents, summaries of documents or documents which fields contain specified information, short-listed documents, documents of folders or conference specified.

This type of reply him will turn out to be useful when only he needs a part of the database and it wants to save space on the disc.



CHAPTER V

ADMINISTRATOR OF THE BASE OF INFORMATION

His responsibilities are: to glide and to create databases, to administer the access, the resources and structures (so much physical as logics) of the same ones and to administer users and his privileges

COMPARISON

CENTRALIZED	DISTRIBUTED
A total control exists on the part of the administrator of the global database. For safety reasons, there is great emphasis in the control and this one must be centralized.	There is a high degree of autonomy on the part of the local administrators. In distributed databases it is possible to identify a hierarchic structure of control based on the administrator of global bases de datos and in the administrators of the local databases. These can manage to have autonomy great.
They must preserve the atomicity of the transactions in presence of faults. The recovery must be done backward or towards ahead, depending on if the transaction has compromised itself or not.	They must preserve the atomicity of the transactions even in cases of faults in some involved sites. The recovery must become based on complex protocols of commitment on two or on three phases.
The DBA has a control centralized across the access authorized for the information. But it is more vulnerable to the safety violations and privacy that in separated files, from the point of view of the globalidad of the information	The local administrators have the same problem of vulnerability and privacy of the global administrators. For the autonomy of every site, the owners of the information have his own protections, sometimes good, sometimes not so much. The distributed databases present the intrinsic problems to the distributed systems since the networks represent a weak point in protection.