UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

CENTRO INTERNACIONAL DE POSGRADO

MAESTRÍA EN INFORMÁTICA APLICADA

MODELO PARA LA IMPLEMENTACIÓN DE BASES DE DATOS DISTRIBUIDAS EN LA UNIVERSIDAD AGOSTINHO NETO

Tesis presentada en opción al grado científico de Máster en Ciencias Técnicas

Lic. José Bernardo Dumbo

Tutores: Dra. Vivian Estrada Sentí

MSc. Yamilís Fernández Pérez

La Habana, Cuba

2015

AGRADECIMIENTOS

A Dios

A mi familia por su constante apoyo incondicional y por confiar en mis conocimientos.

A mis profesores que me enseñaron y formaron en todo lo largo de mis estudios universitarios.

A todos mis amigos/as que desde lejos y también a los que se encuentran presentes, por el apoyo y fuerzas brindadas para la culminación de este trabajo investigativo.

Al DATEC por liberar su laboratorio para poder realizar la implementación de la base de datos distribuida.

Quisiera también agradecer a todas las personas que han contribuido directa e indirectamente con su ayuda para la realización de esta investigación.

DEDICATORIA

A mis padres por todo el amor y apoyo que me han dado.

DECLARACIÓN JURADA DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste, se firma la presente los_____días del mes de_____ del año_____.

_____

José Bernardo Dumbo

_____

Vivian Estrada Sentí

_____

Yamilís Fernández Pérez

RESUMEN

El desarrollo de las tecnologías de información y comunicación (TIC), ha impactado en la sociedad actual en las diferentes áreas de la vida diaria; en el plano académico, investigación, industria y los servicios. Entre los aspectos positivos que se podría destacar está la posibilidad de acceso a la información desde cualquier parte sin importar ni dónde se crea ni las diferentes fuentes de origen. Lo anterior ha favorecido el desarrollo de entornos distribuidos. Esto ha provocado que las organizaciones que adaptan su estructura para un funcionamiento descentralizado es cada vez mayor. Una de estas es la Universidad Agostinho Neto (UAN) que hoy día mantiene sus instituciones de manera descentralizada, sus servicios han presentado dificultades en la disponibilidad y fiabilidad de los datos que manejan, cuya causa se constata en que sus sistemas informáticos se mantiene trabajando de forma centralizada. La utilización de bases de datos distribuidas es imprescindible para las organizaciones que se encuentran geográficamente distintas, así como su integración con herramientas que complementan su implementación, gestión y su adaptación a las características de la organización. En esta investigación se propone un modelo para la implementación de bases de datos distribuidas en la Universidad Agostinho Neto que contribuya a un incremento de la fiabilidad y disponibilidad en el tratamiento de los datos institucionales. El modelo es compuesto por cuatro componentes: diseño de los datos, diseño de la distribución, diseño físico y prueba exploratoria. El modelo fue validado con un cuasiexperimento, entrevista a profundidad, la técnica de Iadov y triangulación metodológica; también se realizó la implementación de base de datos distribuida donde se utilizó técnicas de fragmentación y replicación de datos, evidenciándose que existe una correspondencia satisfactoria entre el objetivo y los resultados.

Palabras-claves: base de datos distribuidas, replicación, fragmentación, asignación, modelo.

ÍNDICE

ÍNDICE DE FIGURAS

INDICE DE TABLA

INTRUDUCCIÓN

Con los recientes avances en las tecnologías de la información (TIC) y las redes de computadoras las organizaciones están cada vez más dispersas, por lo que surge la necesidad de mantener la información disponible desde varias localizaciones geográficas, con el objetivo de integrar los datos operacionales y proporcionar un acceso controlado a estos datos. La gestión de la información requiere un gran esfuerzo y se debe corresponder con los objetivos globales y estratégicos de las organizaciones. El uso generalizado de las TICs ha llevado una mayor integración de los recursos de información disponible a cualquier nivel para el ciudadano.

Esta integración implica una disponibilidad ubicua (en cualquier momento y cualquier lugar) y multidimensional (en todas las facetas de la actividad) y no solo un acceso pasivo a la información, sino la facilidad para ofrecer esta información al resto de la comunidad. Con el surgimiento de estos paradigmas surge el concepto de descentralización e integración en las bases de datos.

La descentralización [Belloch 2012]: es la transferencia o distribución de recursos de una localidad a otras, con el objetivo de decidir sobre estos con operaciones preestablecidas. La descentralización de los datos es una técnica de dispersar los mismos en diferentes ubicaciones, donde estos pueden ser manejados con mayor frecuencia en estas ubicaciones.

La integración de los datos [Heuser 2009]: es el proceso de combinar datos que residen en diferentes fuentes y permitirle al usuario final tener una vista unificada de todos estos datos.

Las bases de datos (todavía de un punto de vista centralizado) tuvieron como antecedentes los sistemas de gestión de ficheros. Estos fueron el primer acercamiento a la resolución del problema de la existencia de enormes cantidades de datos en una organización. Tenían por objetivo automatizar algunas tareas realizadas manualmente hasta cierto punto. Mientras los procesos eran ejecutados de la misma forma [Özsu y Valduriez et al. 2011].

Contrario a lo que sucede en un sistema de ficheros, los datos de una base de datos son organizados en conjunto. Además, el acceso físico a la base de datos es hecho por un Sistema Gestor de Bases de Datos (SGBD), y sus aplicaciones tienen apenas una interface lógica con este SGBD. Un SGBD es un conjunto de

aplicaciones destinadas a gestionar todo el almacenamiento y manipulación de los datos del sistema, haciendo la interface entre el nivel de aplicación y la propia base de datos. El SGBD es una herramienta poderosa para crear y gestionar grandes cantidades de datos de forma eficiente y permitir que estos datos persistan durante largos espacios de tiempo y con seguridad [Garcia, Ullman y Widom 2008].

Las bases de datos almacenan los datos y su descripción (metadatos) en un diccionario de datos que ayuda a interpretar la estructura de los datos almacenados, permitiendo disponibilidad al nivel de la aplicación de una interface lógica. Y estos conceptos son aplicados en la práctica a través de un modelo lógico de datos.

Un modelo lógico de datos [Heuser 2009]: es un conjunto de conceptos que pueden ser utilizados para describir la estructura de la base de datos, en que la estructura de la base de datos son los tipos de datos, las relaciones y las restricciones que soportan los datos. Además, muchos modelos poseen una serie de operaciones básicas aplicadas en las tareas de utilización y recuperación en la base de datos [Elmasri y Navathe 2008]. En la actualidad, el modelo lógico de datos más utilizado es el modelo relacional.

Una base de datos es una colección de datos persistentes que son utilizados por el sistema de aplicaciones de una determinada organización [Date 2004]. Cualquier base de datos presenta un conjunto de propiedades, y la misma representa aspectos del mundo real, según [Elmasri y Navathe 2007], una base de datos es una colección lógica y coherente de datos con algún sentido intrínseco, es diseñada, construida y cargada con datos para un fin específico.

Cuando los datos organizacionales son interrelacionados y almacenados juntos para servir a una o más aplicaciones, esta colección de datos es generalmente llamada base de datos [Wetherbe 2001]. Una base de datos debe ser capaz de mantener los datos a lo largo del tiempo. De estos conceptos dado por diferentes autores se deduce que una base de datos es una colección coherente y lógica de datos persistentes que sirven a una o más aplicaciones de una o más organizaciones.

Una de las mayores motivaciones detrás del uso de los sistemas de bases de datos es el deseo de integrar los datos operacionales de una organización y de

proporcionarlos centralizados, así como un acceso controlado a los datos. La tecnología de redes de computadoras, por otro lado, promueve una forma de trabajar que va en contra de los esfuerzos centralizados.

A simple vista parece difícil comprender cómo integrar dos tecnologías basadas en metodologías opuestas para producir una tecnología más poderosa y con mayores expectativas que cualquiera de las dos por sí mismas. Y la clave está en observar que el objetivo principal de la tecnología de bases de datos es la integración y no la centralización; ninguno de estos términos implica el otro. Es posible lograr la integración sin centralización, lo cual es el objetivo de la tecnología de bases de datos distribuidas [Özsu y Valduriez 2011].

Una base de datos distribuida es un conjunto de múltiples bases de datos lógicamente relacionadas las cuales se encuentran distribuidas entre diferentes sitios o nodos interconectados por una red de comunicaciones, las cuales tienen la capacidad de procesamiento autónomo, y pueden realizar operaciones locales o distribuidas [Carranza y Athó 2006]. Algunos aspectos importantes que están relacionados con este concepto son la distribución y la correlación lógica. El primer aspecto tiene como característica de que los datos no residen en el mismo local, sino en diferentes localidades, lo que la hace diferente a una base de datos centralizada. El segundo es caracterizado por el hecho de que los datos tienen algunas propiedades o características que los relacionan, y de este modo se puede distinguir una base de datos distribuida de un conjunto de base de datos locales o ficheros residentes en diferentes localidades de una red de computadoras [Elmasri y Navathe 2007].

Con la amplia prevalencia de los sistemas de bases de datos centralizados, los usuarios y los programas de aplicaciones acceden a la base de datos desde diversos sitios. En contraste una base de datos distribuida no se almacena completamente en una localidad central, sino que se distribuye en una red de comunicaciones, donde cada localidad tendrá su propia base de datos y está capacitada para acceder a los datos de otras localidades.

Según [Özsu y Valduriez 2011], una base de datos distribuida es caracterizada por tener sus datos almacenados en nodos y sus procesadores a través de una red de computadoras. En una base de datos distribuida (BDD), los nodos pueden hacer consultas locales o consultas que hacen en los datos que se encuentran en otros

nodos de la red. Como el propio nombre dice, una base de datos distribuida, es compuesta por datos distribuidos en diferentes localizaciones, las cuales son transparentes ante los usuarios.

La distribución de la base de datos contempla determinados aspectos, como: la fragmentación, la replicación y la localización y distribución de los datos que deben ser clarificados y entendidos para que se pueda establecer la estrategia para su diseño. En el ámbito de distribución de datos, existe el Sistema Gestor de Base de Datos Distribuida o SGBDD (también denominado *DDBMS* de sus siglas en inglés *Distributed Database Management System*).

Un SGBDD es el software que gestiona la base de datos y provee el mecanismo de acceso que hace la distribución transparente ante el usuario [Özsu y Valduriez 2011]. Los usuarios que tienen acceso a los datos pueden compartir e inclusive actualizar, de forma integral a las bases de datos ubicadas en diferentes sitios geográficos sin necesidad de conocer los detalles de implementación y donde las mismas están ubicadas, o sea, para los usuarios parece como si se tratase de un sistema único o centralizado.

La fragmentación consiste en dividir en fragmentos menores la relación que corresponde a una tabla, donde cada fragmento se guarda en sitios diferentes. El proceso de fragmentación en las bases de datos distribuidas se hace mediante tres alternativas lógicas que son la horizontal, vertical y mixta. La fragmentación horizontal se realiza por tuplas individuales, la fragmentación vertical por atributos individuales, y la combinación de las dos anteriores (vertical y horizontal) resulta en una fragmentación mixta o híbrida. La replicación es un mecanismo utilizado para propagar y diseminar datos en un ambiente distribuido, con el objetivo de tener mejor performance y confiabilidad, mediante la reducción de dependencia de un sistema de base de datos centralizado.

En los últimos años las bases de datos distribuidas han tenido gran auge [Ramakrishnan, et al. 2003]. Varias organizaciones que se dedican al desarrollo de software comercial, el software de código abierto y experimental se han involucrado en esta área [Burbano 2006]. Otros factores han favorecido el incremento de la tecnología distribuida. Factores como la reducción del costo del hardware, el incremento de la complejidad en los datos manejados, como en las aplicaciones orientadas a objetos, deductivas y multimedia. Lo anterior lleva a que en la mayoría

de las ocasiones se procese una gran cantidad de datos, creando un ambiente propicio para que las aplicaciones sean desarrolladas en una plataforma de bases de datos distribuidas [Ríos 2011].

Existen diferentes herramientas para la implementación de bases de datos distribuidas. Entre ellas se destacan *SyBase Data Integration Suite* de IBM y *BDLinks* de Oracle. SyBase es una herramienta que ofrece servicios para integración y replicación de los datos, así como para el análisis, extracción, transformación y carga de datos (ETL), (de sus siglas en inglés, *Extraction-Transform and Load*). Pero esta herramienta es propietaria y de código cerrado, posee precio elevado, y no utiliza patrones de interoperabilidad y de mensajes abiertos, lo que torna difícil la interoperabilidad entre los sistemas de los datos [Brito, 2009].

El *Database Links* o *DBLinks* de Oracle, es utilizada para definir un camino unidireccional desde una base de datos Oracle hasta la otra. El usuario local puede acceder a través de un *Link* a objetos de esquemas de otros usuarios en bases de datos remotas (siempre que tenga permiso suficiente para hacerlo) como si se tratara de una única base de datos. Este sistema es cerrado y uno de los más caros del mercado.

La Universidad Agostinho Neto, es una institución de enseñanza de Educación Superior, compuesta por ocho facultades y un instituto, con diversos cursos que son administrados en estos mismos centros. Durante los cinco últimos años, enfrentó el reto de desarrollar aplicaciones de gestión para el soporte de los servicios que frecuentemente son efectuados dentro de la institución. Entre estas aplicaciones se encuentran el SIGEA (Sistema Integrado de Gestión de Exámenes de Acceso), y el SIGA (Sistema Integrado de Gestión Académica).

El primero es una solución informática integrada para la gestión del proceso de candidaturas y exámenes de acceso a la Universidad Agostinho Neto. Este sistema automatiza el flujo de trabajo requerido para las fases de inscripciones, exámenes, resultados y otras funcionalidades que son capaces de corresponder con las necesidades de los candidatos de la Universidad y de sus unidades orgánicas. El (SIGA) es una herramienta de gestión universitaria de apoyo a todos los servicios académicos asociados a la Universidad y sus respectivas instituciones (el mismo se encuentra en fase de desarrollo). Estos sistemas, para realizar el procesamiento de

los datos, utilizan los SGBDs (MySQL y Oracle). Los datos se encuentran centralizados en una base de datos única y ubicada en un solo servidor.

Basado en el estudio realizado sobre las aplicaciones informáticas de la Universidad, con el proceso de acceso y el tratamiento de los datos en la Universidad Agostinho Neto, es posible destacar las siguientes dificultades y/o limitaciones:

- Dificultad en tener los datos siempre disponibles, por el hecho de la base de datos ser única
- Cuando el sistema de la base de datos falla, no se logra la disponibilidad de procesamiento y sobre todo de información confiable al sistema
- Con frecuencia se presentan dificultades en el acceso a la información, por parte de los departamentos institucionales
- El repositorio de los datos es muy limitado, aunque hay necesidad de tener bastante información
- Las cargas de trabajo no se pueden difundir entre varias computadoras, ya que los trabajos siempre se ejecutan en la misma máquina

Esta situación fundamentó el siguiente **problema científico:**

¿Cómo lograr un incremento de la fiabilidad y la disponibilidad en el tratamiento de los datos institucionales en la Universidad Agostinho Neto?

Se plantea como **objeto de estudio** el proceso de implementación de las bases de datos distribuidas y el **campo de acción** los modelos de distribución, fragmentación y replicación de datos para la implementación de bases de datos distribuidas.

Como **objetivo general** de la investigación se plantea:

Diseñar un modelo basado en técnicas de replicación y fragmentación para la implementación de bases de datos distribuidas, que contribuya a un incremento de la fiabilidad y la disponibilidad en el tratamiento de los datos institucionales en la Universidad Agostinho Neto.

Como **Hipótesis de investigación** se plantea la siguiente:

Un modelo basado en la integración de técnicas de fragmentación y replicación de datos para la implementación de bases de datos distribuidas contribuirá a un incremento de la fiabilidad y la disponibilidad en el tratamiento de los datos institucionales en la Universidad Agostinho Neto.

Se plantean los siguientes **objetivos específicos:**

- Elaborar el marco teórico referencial de la investigación, relacionado con las bases de datos distribuidas y sus aplicaciones, así como los modelos de distribución, fragmentación y replicación de datos para la implementación de bases de datos distribuidas
- Definir un modelo para la construcción de bases de datos distribuidas, basadas en técnicas de fragmentación y replicación
- Seleccionar herramientas computacionales para la distribución y la replicación de datos para sostener el modelo planteado
- Validar la propuesta

Aportes prácticos

- Un modelo de bases de datos distribuidas, basados en arquitecturas distribuidas y esquemas de procesamiento distribuido
- Un esquema de arquitectura para las bases de datos distribuidas
- El diseño y la implementación de una base de datos distribuida, basada en el modelo elaborado

Importancia y contribución metodológica

La presente investigación resulta importante para el logro de un incremento de la fiabilidad y la disponibilidad en el tratamiento de los datos institucionales en la Universidad Agostinho Neto. Desde el punto de vista metodológico tiene un destacado valor ya que incidirá en el mejoramiento de la carrera de ciencias de la computación en particular en la asignatura de bases de datos.

Durante el proceso de investigación se utilizaron los siguientes métodos científicos:

Métodos Teóricos:

Análisis Histórico-Lógico: se inicia la investigación teniendo en cuenta las descripciones históricas de las soluciones sobre implantación de bases de datos en general y de las bases de datos distribuidas, los procesos de fragmentación y replicación de datos. Se pondrán en función los resultados del estudio para obtener una base que ayude a definir las fases de dicho modelo.

Analítico-Sintético: permitirá profundizar y desglosar toda la información encontrada sobre las soluciones de bases de datos distribuidas y los servicios de su implantación. A partir de ello se podrá determinar la solución más apropiada para el problema planteado y definir la estructura del modelo de acuerdo con la situación de la Universidad.

Modelación: es un método de investigación que permite la creación de modelos, descubrir y estudiar nuevas relaciones y cualidades del objeto de estudio. Fue utilizado en el momento de crear el modelo para implantar las soluciones de bases de datos distribuidas, pues no se encontró ninguna propuesta para el establecimiento de las mismas.

Métodos Empíricos:

Observación: contribuirá al incremento de la visión sobre las condiciones en las que se encuentran los servicios de implantación de la base de datos distribuida en la Universidad.

Encuesta: servirá de apoyo para el diagnóstico de la situación actual y la validación de la propuesta.

La tesis está estructurada de introducción, tres capítulos, conclusiones, recomendaciones, bibliografía y anexos. En el **capítulo 1** se analiza y se exponen los conceptos teóricos relacionados con el objeto de estudio y el campo de acción en que se trabajan. En el **capítulo 2** se fundamenta la concepción del modelo para la implementación de bases de datos distribuidas en la Universidad Agostinho Neto, y son definidos sus principios, cualidades y componentes. En el **capítulo 3** se presenta el análisis y valoración de los resultados a partir de las técnicas y métodos empleados (cualitativos y cuantitativos) y se incluye la triangulación realizada de los métodos empleadas.

CAPÍTULO 1: MARCO TEÓRICO

Introducción

En el presente capítulo se examinan las bases de datos en general y sus aplicaciones, entre ellos los sistemas gestores de bases de datos, además las ventajas y desventajas que las mismas presentan y sus principales características. Se analizan los conceptos de las bases de datos distribuidas, las principales ventajas que presentan aplicadas a través de la necesidad de manipular y gestionar grandes cantidades de datos de manera confiable, y las técnicas de fragmentación y modelos de replicación junto con el interese en descentralizar la información, y se fundamenta la opinión del autor siempre que sea necesario.

1.1 Bases de datos

Según [Connoly y Begg 2009], una base de datos (BD), es una colección de datos relacionados lógicamente, almacenados con carácter más o menos permanente, para satisfacer las necesidades de una organización.

Una base de datos es una colección de datos persistentes que son utilizados por el sistema de aplicaciones de una determinada organización [Date 2004]. Una base de datos es una colección lógica y coherente de datos con algún sentido intrínseco, es diseñada, construida y cargada con datos para un fin específico. De estos conceptos dado por diferentes autores se deduce que una base de datos es una colección coherente y lógica de datos persistentes que sirven a una o más aplicaciones de una o más organizaciones.

Por persistentes se entiende intuitivamente, que los datos de la base de datos difieren de otros tipos de datos, tales como datos de entrada, datos de salida, consultas SQL, resultados intermedios y otros datos más generales. Pero precisamente se dice que los datos en una base de datos "persisten" porque, una vez que han sido aceptados por el sistema de base de datos para entrar en la base de datos, estos pueden subsecuentemente ser borrados de la misma, solamente por una petición explícita del sistema de base de datos [Date 2004]. Donde los componentes principales del sistema de base de datos son: los datos, el hardware, el software y los usuarios [Garcia-Molina, Ullman, Widom 2008]. La Figura 1.1 muestra la relación entre los usuarios, la base de datos, y los datos.

Figura 1.1: Sistemas de bases de datos. (Ullman 2008).

Para que los datos sean tratados y manipulados es necesario el uso de un sistema de gestión, que es conocido como Sistema Gestor de Bases de Datos (o SGBD). Un Sistema Gestor de Base de Datos, también llamado de DBMS (de sus siglas en inglés *Database Management System*) es donde se almacena una colección de datos estructurados y organizados. Los datos almacenados se pueden actualizar, manipular y recuperar, y diferentes usuarios y aplicaciones pueden acceder y gestionar los datos [Ramos y Montero 2011]. En la Figura 1.2 se presenta una estructura de sistema de base de datos y su interacción con los usuarios.

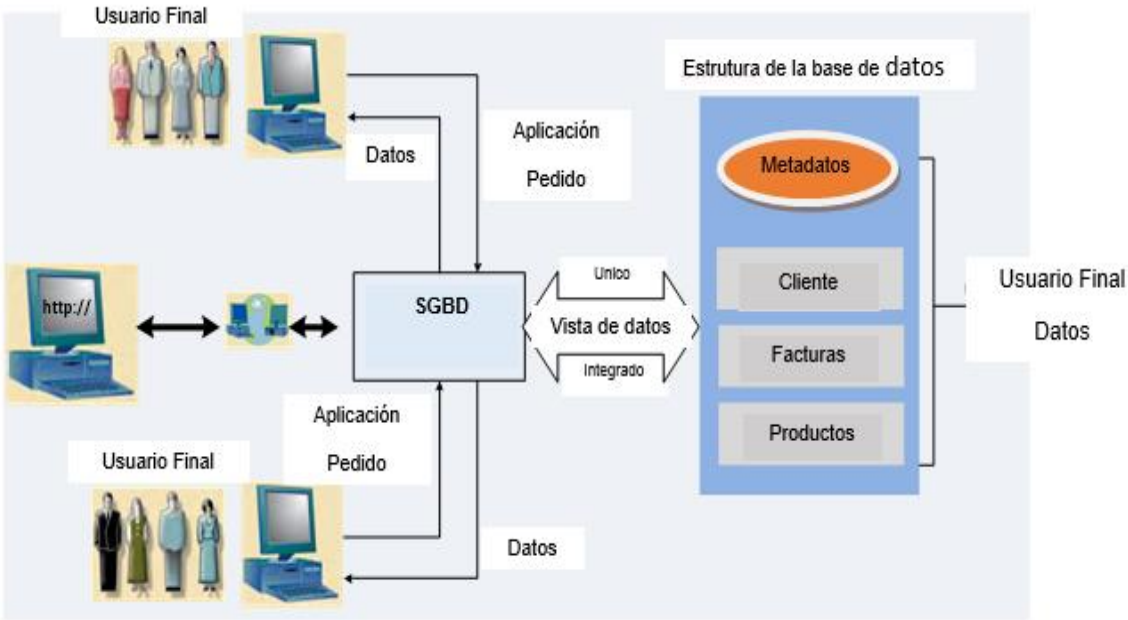

Figura 1.2: Gestión de la interacción entre los usuarios y la base de datos. (Coronel, Morris y Rob 2011).

El SGBD es una aplicación que permite al usuario definir, crear y mantener la base de datos y proporciona un acceso controlado de la misma [Martín y Ramos 2010]. Debe prestar los siguientes servicios:

- **Creación y definición de la BD:** especificación de la estructura, el tipo de los datos, las restricciones y relaciones entre ellos mediante lenguajes de definición de datos. Toda esta información se almacena en el diccionario de datos, el SGBD proporciona mecanismos para la gestión del diccionario de datos
- **Manipulación de los datos**, realizando consultas, inserciones y actualizaciones de los mismos utilizando lenguajes de manipulación de datos
- **Acceso controlado a los datos de la BD**, mediante mecanismos de seguridad de acceso a los usuarios
- **Mantener la integridad y la consistencia** de los datos utilizando mecanismos para evitar que los datos sean perjudicados por cambios no autorizados
- **Acceso compartido a la BD**, controlando la interacción entre usuarios concurrentes
- **Mecanismos de respaldo y recuperación**, para restablecer la información en caso de fallos en el sistema

Las bases de datos en la actualidad deben cumplir con:

1. Minimizar redundância
2. Flexibilidad e independência
3. Concurrencia
4. Servir eficientemente los *Data Warehouse*
5. Adaptarse al desarrollo orientado a objetos
6. Incorporar el tiempo como un elemento de caracterización de la información
7. Adaptarse al mundo de internet

Uno de los propósitos principal de un sistema de base de datos es proporcionar a los usuarios una visión abstracta de los datos. Es decir, el sistema oculta ciertos detalles de cómo los datos se almacenan y mantienen.

1.1.1 Arquitectura de los sistemas de bases de datos

En el año 1975, el comité ANSI-SPARC (de sus siglas en inglés, *American National Standard Institute - Standards Planning and Requirements Committee*) propuso una arquitectura de tres niveles para los SGBD cuyo objetivo principal era el de separar los programas de aplicación de la BD física. En esta arquitectura el esquema de una BD se define en tres niveles de abstracción distintos:

Nivel interno o físico: el más cercano al almacenamiento físico, es decir, tal y como están almacenados en el ordenador. Describe la estructura física de la BD mediante un esquema interno. Este esquema se especifica con un modelo físico que describe los detalles de cómo se almacenan físicamente los datos: los archivos que contienen la información, su organización, los métodos de acceso a los registros, los tipos de registros, la longitud, los campos que los componen, etcétera.

Nivel externo o de visión: es el más cercano a los usuarios, es decir, es donde se describen varios esquemas externos o vistas de usuarios. Cada esquema describe la parte de la BD que interesa a un grupo de usuarios, en este nivel se representa la visión individual de un usuario o de un grupo de usuarios.

Nivel conceptual: describe la estructura de toda la BD para un grupo de usuarios mediante un esquema conceptual. Este esquema describe las entidades, atributos, relaciones, operaciones de los usuarios y restricciones, ocultando los detalles de las estructuras físicas de almacenamiento, que representa la información contenida en la BD. En la Figura 1.3 se representan los niveles de abstracción de la arquitectura ANSI.

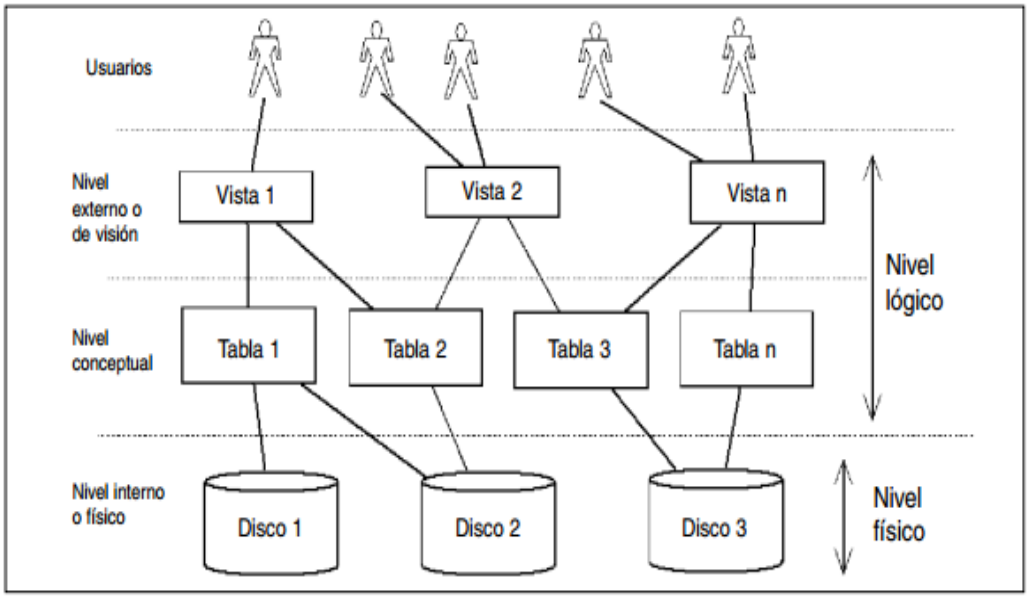

Figura 1.3: Niveles de abstracción de la arquitectura ANSI. (Martin y Ramos 2010).

Esta arquitectura describe los datos a tres niveles de abstracción. En realidad los únicos datos que existen están a nivel físico almacenados en discos u otros dispositivos. Los SGBD basados en esta arquitectura permiten que cada grupo de usuarios haga referencia a su propio esquema externo. El SGBD debe de transformar cualquier petición de usuario (esquema externo) a una petición expresada en términos de esquema conceptual, para finalmente ser una petición expresada en el esquema interno que se procesará sobre la BD almacenada. El proceso de transformar peticiones y resultados de un nivel a otro se denomina correspondencia o transformación, en que el SGBD es capaz de interpretar una solicitud de datos y realizar los siguientes pasos:

- El usuario solicita unos datos y crea una consulta
- El SGBD verifica y acepta el esquema externo para este usuario
- Transforma la solicitud al esquema conceptual
- Transforma la solicitud al esquema físico o interno
- Selecciona la(s) tablas implicadas en la consulta y ejecuta la consulta
- Transforma del esquema interno al conceptual, y del conceptual al externo
- Finalmente el usuario ve los datos solicitados

Para una BD específica solo hay un esquema interno y uno conceptual, pero puede haber varios esquemas externos definidos para uno o para varios usuarios.

1.1.2 Ventajas y desventajas de los SGBD

Los sistemas de bases de datos presentan numerosas ventajas, porque ayudan a que la gestión de los datos sea más eficiente y eficaz. En particular un SGBD presenta las siguientes ventajas [Coronel, Morris y Rob 2011]:

- Evita redundancia, inconsistencias y problemas de integridad
- Provee mejor (y estandarizado) acceso a los datos
- Datos compartidos, y acceso concurrente (aislamiento)
- Mantiene la independencia de los datos y su tratamiento
- Proporciona eficiencia en términos de operaciones y almacenamiento
- Hace con que el desarrollo de las aplicaciones sea más rápido, y facilita la satisfacción de demandas cambiantes
- Administración en la seguridad de los datos
- Economía de escala
- Mejores procedimientos de respaldo y recuperación

Aunque son muchas las ventajas del uso del SGBD, también existen las desventajas acerca de los mismos [Coronel, Morris y Rob 2011], abajo se listan algunas de ellas:

- Mayores costos: licencias (BD comercial), hardware, necesidad de administración y control
- Costos de conversión
- Procesamiento más lento en algunas aplicaciones
- Mayor vulnerabilidad (recursos centralizados)
- Recuperación más difícil

1.2 Bases de datos distribuidas

Una base de datos distribuida consiste en un conjunto de localidades, cada una de las cuales mantiene un sistema de bases de datos local. Cada localidad puede procesar transacciones locales, o bien transacciones globales entre varias localidades, requiriendo para eso comunicación entre ellas, donde en cada localidad tiene su propio sistema gestor de base de datos distribuidos o SGBDD (de sus siglas en inglés *DDBMS, Distributed Database Management System*) que permite la gestión de la base de datos, y hace que la distribución sea transparente

al usuario [Özsu y Valduriez 2011]. La Figura 1.4 muestra la estructura de un sistema de base de datos distribuidas.

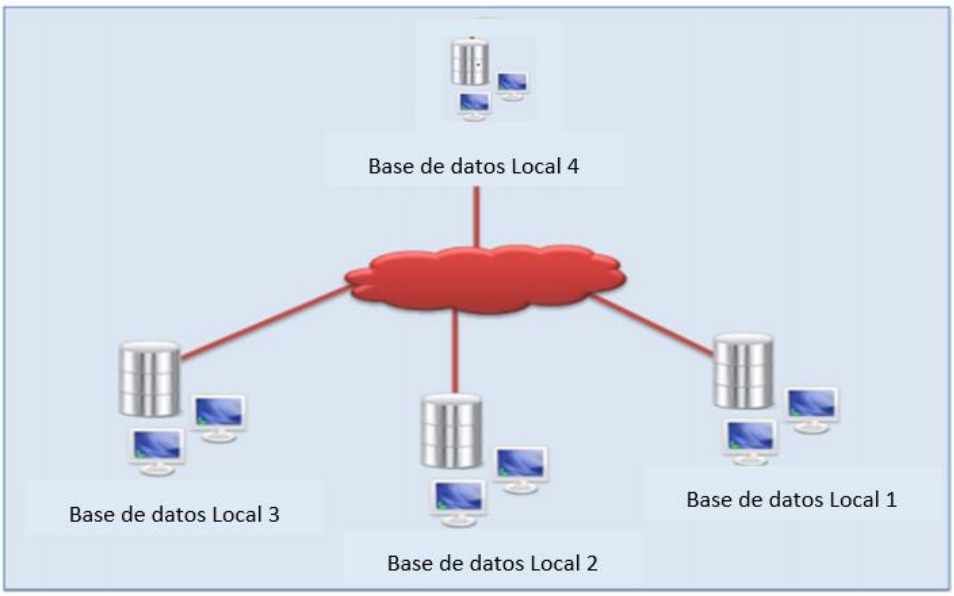

Figura 1.4: Sistema de base de datos distribuidas. (Özsu y Valduriez 2011)

En un modelo de bases de datos distribuidas, los datos se almacenan en varias computadoras. Las computadoras de un sistema distribuido se comunican entre sí a través de diversos medios de comunicación, tales como cables de alta velocidad o líneas telefónicas, donde cada sitio o nodo tiene un sistema completo de procesamiento de la información, con propia función de administración de datos, usuarios, hardware y software, inclusive una base de datos local, sistema de gestión de base de datos y software de comunicaciones.

1.2.1 Características de las bases de datos distribuidas

En la práctica las bases de datos distribuidas se pueden caracterizar desde varios aspectos [Cuadra 2007]:

- Los datos deben estar físicamente en más de un ordenador, es decir, los datos se encuentran almacenados en distintos sitios o nodos
- Los sitios deben estar interconectados mediante una red de computadoras (donde cada sitio será un nodo de la red). Para realizar el diseño no se tiene en cuenta la topología, tipo y rendimiento de la red, aunque estas propiedades tengan relevancia en el buen funcionamiento del sistema

- Los datos deben estar lógicamente integrados en una única estructura o esquema lógico común
- Los usuarios tienen acceso (recuperación y actualización) a los datos pertenecientes a la BDD, ya residen estos en el mismo sitio (acceso local) o en otro sitio (acceso remoto)
- Cada nodo o emplazamiento facilita un entorno para la ejecución de transacciones tanto local o global
- En una única operación, tanto de consulta como de actualización, se accede a datos que se encuentran en más de un sitio sin que el usuario sepa la distribución de los mismos en los distintos sitios. Es decir, que la distribución de la información es transparente para el usuario

1.2.2 Diseño de bases de datos distribuidas

El diseño de un sistema de base de datos distribuido implica la toma de decisiones sobre la ubicación de los programas que accederán a la base de datos y sobre los datos que la constituyen a lo largo de los diferentes puestos que configuren una red de computadoras. En el caso del SGBD distribuido, la distribución de las aplicaciones involucra dos cosas: la distribución del SGBD y la distribución de los programas de aplicaciones que corren en el mismo sistema [Ruz 2010].

Para el diseño de una base de datos distribuida, es necesario seguir una metodología, la cual consta de cuatro fases [Özsu, y Valduriez 2011]:

1. **Diseño del esquema conceptual**: describe la integración de la base de datos; es decir, los datos que se encuentran almacenados para ser utilizados por las aplicaciones.
2. **Diseño físico de la base de datos**: es el direccionamiento al esquema conceptual de áreas de almacenamiento y la determinación de métodos de acceso apropiados. En una base de datos distribuida, estos dos elementos son parte del diseño global del esquema y diseño físico del sistema de información en cada localidad.
3. **Diseño de la fragmentación**: determina como serán subdivididas las relaciones, ya sea de manera horizontal, vertical o mixta.
4. **Diseño de la distribución de fragmentos**: determina como serán direccionados los fragmentos en las imágenes, y también se determina la replicación de los datos.

El diseño de la fragmentación es conocido como una característica distinta de las bases de datos distribuidas. La distinción entre fragmentación y asignación de fragmentos es conceptualmente relevante; la primera se refiere a la fragmentación lógica de una relación global, mientras que la segunda trata de la localización física de los datos fragmentados en varias localidades.

1.2.3 Técnicas para el diseño de la distribución de los datos

En todo proceso de diseño de BDD existen dos técnicas para el diseño de la distribución de los datos, las cuales se describen a continuación [Özsu y Valduriez 2011]:

El diseño **ascendente** o ***Top-Down***, que se utiliza cuando se crea una BDD a partir de varias BDD locales existentes. Se parte de varios esquemas lógicos locales (ELL) con ubicaciones diferentes que se integran en un único esquema lógico global (ELG).

El diseño **descendente** o ***Bottom-Up***, parte de un ELG y construye varios ELL definidos a partir de esquemas de fragmentación, asignación y replicación que determinan la distribución de los datos en los distintos nodos de la red.

1.2.4 Objetivos del diseño de la distribución de los datos

El diseño de la distribución de datos tiene los siguientes objetivos [Coronel, Morris y Rob 2011]:

Localidad de procesamiento: el distribuir los datos para maximizar el procesamiento local de cada localidad, corresponde al sencillo principio de colocar los datos donde corresponde, para que estos se encuentren disponibles para la aplicación que los utilizará. La forma más fácil de caracterizar el procesamiento local de una localidad es considerar dos tipos de referencias de datos: locales y remotos, lo cual depende solamente de la distribución de los mismos.

Disponibilidad y confiabilidad en la distribución de los datos: un alto grado de disponibilidad para las aplicaciones de solo lectura se logra a través del almacenamiento de múltiples copias de información; el sistema debe ser capaz de cambiar a otra alternativa cuando la información que debe ser accesible en condiciones normales no se encuentra disponible. La confiabilidad también se logra

almacenando copias o réplicas de información para que el sistema sea capaz de recuperarse, ya sea de caídas del sistema o de destrucción física de una de las copias utilizando la otra copia disponible.

Distribución de la carga de trabajo: la distribución de la carga de trabajo sobre las diferentes localidades es una característica importante de las bases de datos distribuidas. La distribución de la carga de trabajo se realiza para aprovechar las ventajas de la computadora en cada localidad y maximizar el grado de paralelismo de la ejecución de aplicaciones.

Costo de almacenamiento y disponibilidad de los datos: la distribución de la base de datos debe reflejar el costo y disponibilidad de almacenamiento en las diferentes localidades: esto es posible al tener una localidad especial para almacenar datos. Generalmente, el costo de almacenamiento de datos no es relevante si se compara con el costo de un procesador, dispositivos de entrada/salida y los costos de transmisión de aplicaciones: es por ello que la limitante de disponibilidad de almacenamiento en cada localidad debe ser considerada.

1.2.5 Diseño de la fragmentación

La fragmentación de los datos es también uno de los aspectos a considerar durante el diseño de una base de datos distribuida, la cual consiste en dividir una relación en partes o fragmentos para propósitos de almacenamiento físico por medio de operaciones como **SELECT, INSERT, UPDATE, JOIN, PROYECT, DELETE** etcétera.

La fragmentación es un conjunto de técnicas para dividir la BD en unidades lógicas, llamadas fragmentos, cuyo almacenamiento puede asignarse a los diversos nodos. Esta técnica se utiliza durante el proceso de diseño de la BDD [Elmasri y Navathe 2011]. Un fragmento es cualquier subrelación derivada de una tabla global mediante operaciones de selección y proyección. Donde la reconstrucción de la relación fragmentada, se hace mediante operaciones de join y unión.

Esta facilidad de fragmentación de datos y reconstrucción, es una de las razones por la cual, la mayoría de las bases de datos distribuidas utilizan el modelo relacional. En la práctica existen distintas formas de fragmentar una relación en una

BDD; la fragmentación horizontal, la fragmentación vertical y la fragmentación mixta o híbrida [Özsu y Valduriez 2011], [Coronel, Morris y Rob 2011].

Según [Özsu y Valduriez 2011], [Coronel, Morris y Rob 2011], existen cuatro razones para que sea aplicada fragmentación en el diseño de bases de datos distribuidas, que son:

- Es útil ya que las aplicaciones de BD suelen funcionar con vistas, y por ello se pueden utilizar distintas relaciones en distintos nodos para formar la unidad distribuida
- Se consigue una mayor eficiencia, dado que los datos suelen estar almacenados cerca del nodo que más los utiliza
- Permite aumentar el grado de concurrencia; la fragmentación de las relaciones permite que una transacción pueda dividirse en subconsultas que operan sobre estos fragmentos
- Aporta una mayor seguridad, dado que los datos no utilizados por un nodo local no se almacenan en él y por lo tanto no están al alcance de usuarios sin autorización

Además, la fragmentación presenta algunos inconvenientes; si las aplicaciones tienen requisitos conflictivos que impiden la descomposición de la relación en fragmentos mutuamente exclusivos, estas aplicaciones cuyas vistas se definen en más de un fragmento pueden sufrir una degradación del rendimiento [Martín 2010].

El segundo problema está relacionado con el control semántico de los datos, especialmente para el control de integridad. Como resultado de la fragmentación, los atributos que participan en una dependencia pueden ser descompuestos en diferentes fragmentos que pueden ser asignados a diferentes sitios. En este caso, incluso la simple tarea de comprobar las dependencias daría lugar a la búsqueda de los datos en diferentes sitios [Martín 2010], [Elmasri y Navathe 2011], [Tiwari 2011].

Reglas de fragmentación

Durante la etapa de fragmentación se definen las siguientes normas que determinan la calidad de la fragmentación de una relación, que son: **completitud**,

reconstrucción y **disyunción**, que en conjunto, asegúrense de que la base de datos no sufrirá cambios semánticos [Özsu y Valduriez 2011].

Completitud: cuando se descompone una relación R en los fragmentos $F_R = (R_1, R_2,\ldots, R_n)$ se dice completa si y solamente si cada elemento de datos en R se encuentra en algún fragmento descompuesto. Hay que tener en cuenta que en el caso de la fragmentación horizontal, el "elemento" se refiere normalmente a una tupla, mientras que en el caso de la fragmentación vertical, se refiere a un atributo.

Reconstrucción: si una relación R se descompone en los fragmentos $F_R = (R_1, R_2,\ldots, R_n)$ entonces debe existir un operador Δ que nos permita reconstruir la relación original R, de tal forma que $R=\Delta R_i, R_i \epsilon F_R$.

El operador Δ será diferente para diferentes formas de fragmentación; es importante, sin embargo que, puede ser identificado. La reconstrucción de la relación de sus fragmentos asegura que las restricciones definidas en los datos de la forma de dependencias se conservan.

Disyunción: si la relación R se descompone en los siguientes fragmentos $F_R = (R_1, R_2,\ldots, R_n)$, y el dato d_i está en R_j, entonces no debe encontrarse en ningún otro fragmento de la relación R_k ($k\neq j$). Este criterio asegura que los fragmentos horizontales son disjuntos. Si la relación R se descompone verticalmente, sus atributos de clave primaria son típicamente repetidos en todos sus fragmentos (de reconstrucción). Por lo tanto, en el caso de la fragmentación vertical, la disyunción se define solo en los atributos no clave primaria de una relación.

Fragmentación horizontal

La fragmentación horizontal consiste en la existencia de una relación R, la cual puede dividirse en varios subconjuntos $r_1, r_2, r_3,\ldots r_n$, donde cada tupla de la relación R deberá pertenecer a alguno de los fragmentos, de manera que si es preciso, pueda reconstruirse la relación global. La reconstrucción de la relación R puede obtenerse al calcular la unión de los fragmentos [Elmasri y Navathe 2011]. En la Tabla 1.1 se muestra un ejemplo de la fragmentación horizontal, donde R es la tabla de Matriculados (que muestra los estudiantes matriculados en una determinada escuela).

Tabla 1.1: Estudiantes matriculados.

Escuela	*#Num_Matriculados*	*Estudiante*	*Curso*
A	0011	Pedro	Derecho
A	0022	Rodríguez	Derecho
B	0044	José	Economía
A	0099	González	Mecánica
B	0012	Juan	Economía
B	2310	Marcos	Medicina

La relación puede dividirse en n fragmentos diferentes, cada uno de los cuales consiste en tuplas de *Num_estudiante* que pertenecen a una escuela determinada.

Tabla 1.2: Fragmentación horizontal de los estudiantes de la escuela A y B.

Escuela	*#Num_Matriculados*	*Estudiante*	*Curso*
A	0011	Pedro	Derecho
A	0022	Rodríguez	Derecho
A	0099	González	Mecánica

Escuela	*#Num_Matriculados*	*Estudiante*	*Curso*
B	0044	José	Economía
B	0012	Juan	Economía
B	2310	Marcos	Medicina

Fragmentación Vertical

La fragmentación vertical consiste en dividir los atributos (columnas) de una tabla global en subtablas, añadiendo un atributo especial llamado *Id_tupla* a la tabla global, donde *Id_tupla* es una dirección lógica de una tupla. Esta dirección permite recuperar de forma directa la tupla sin necesidad de un índice [Elmasri y Navathe 2011]. En la Tabla 1.3 se muestra un ejemplo de fragmentación vertical utilizando la relación de la Tabla 1.1.

Tabla 1.3: Fragmentación vertical.

Escuela	*Estudiante*	*Id_tupla*
A	Pedro	1
A	rodríguez	2
B	José	3
A	González	4
B	Juan	5
B	Marcos	6

#Num_Matriculados	*Curso*	*Id_tupla*
0011	Derecho	1
0022	Derecho	2
0044	Economía	3
0099	Mecánica	4
0012	Economía	5
2310	Medicina	6

Fragmentación Mixta

La fragmentación mixta consiste en aplicar la fragmentación vertical seguida de la fragmentación horizontal o viceversa, o sea, incluye una o varias secuencias de aplicación de las fragmentaciones anteriormente mencionadas ya que en muchos casos una fragmentación horizontal o vertical de un esquema de una base de datos, no será suficiente para satisfacer los requerimientos de aplicaciones de usuario.

La forma más sencilla de construir este tipo de fragmentación consiste en que, una fragmentación vertical puede ser seguida de una horizontal, o viceversa, produciendo un árbol de particionamiento estructurado ya que los dos tipos de particionamiento se aplican uno después del otro.

1.3 Replicación de datos

La replicación de datos es el proceso de compartir objetos y datos de una base de datos a múltiples bases de datos, en diferentes localizaciones [Buretta 2009]. La replicación de datos permite que ciertos datos de la base de datos sean almacenados en más de un sitio, aumentando la disponibilidad y mejora el funcionamiento de las consultas globales a la base de datos.

En la Figura 1.5 se muestra el proceso de replicación de datos entre varios nodos de una base de datos distribuida.

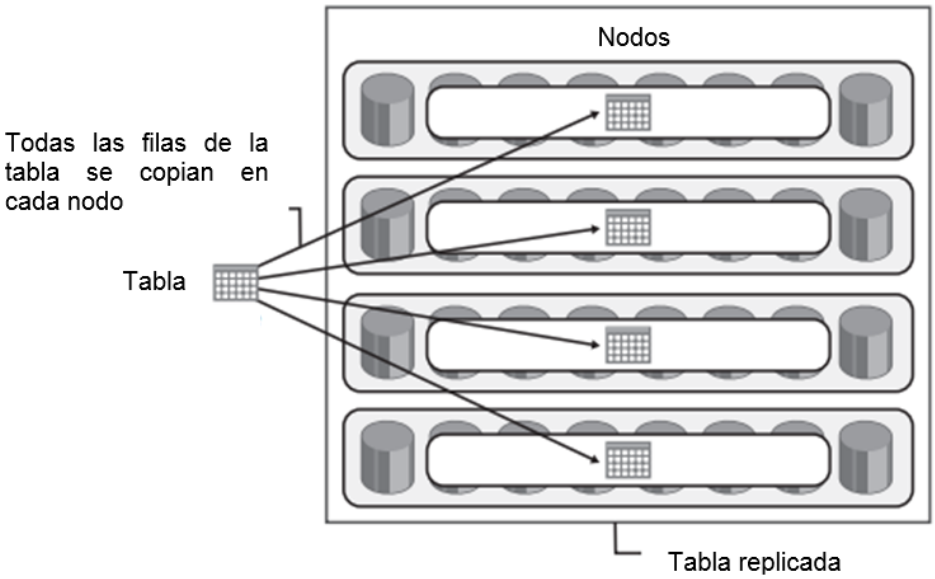

Figura 1.5: Replicación de datos. (Mistry y Misner 2010).

Para esto, existen tres escenarios de replicación: una base de datos puede ser totalmente replicada, parcialmente replicada o no replicada.

- Una base de datos totalmente replicada almacena varias copias de cada fragmento de la base de datos en múltiples sitios. En esto caso todos los fragmentos de la base de datos se replican. Este enfoque puede ser poco práctico debido la cantidad de sobrecarga que impone en el sistema
- Una base de datos parcialmente replicada almacena múltiples copias de algunos fragmentos de la base de datos en múltiples sitios. La mayoría de los SGBDD son capaces de manejar bien la base de datos parcialmente replicada
- Una base de datos no replicada almacena cada fragmento de la base de datos en un solo sitio. Por lo tanto, no hay duplicados fragmentos de base de datos en los sitios

El uso de réplica de datos posee numerosas ventajas, donde las principales son:

- Por un lado, se garantiza que el servicio ofrecido por la aplicación, no se vea interrumpido en caso de que se dé un fallo en alguna de las réplicas Además, el tiempo necesario para restablecer el servicio en la aplicación podría llegar a ser grande en algunos tipos de fallo

- Por otra parte, la capacidad de servicio se ve incrementada cuando las peticiones efectuadas por los clientes únicamente implican consultas

Sin embargo, estas ventajas tienen también un coste asociado. Cuando las peticiones atendidas impliquen una actualización en el estado de la aplicación, dicha actualización debe realizarse en todas las réplicas, y esto debe hacerse de una manera ordenada para que todas ellas mantengan un estado consistente. Esto implica que las operaciones de actualización tendrán un tiempo de servicio mayor que en el caso no replicado, pues habrá que proceder a la propagación de las actualizaciones sobre todas las réplicas y para ello será necesario emplear algún mecanismo de difusión.

1.3.1 Sentido de la réplica de datos

La réplica de datos puede aplicarse de dos maneras diferentes, de acuerdo con las necesidades propias de las instituciones que la llevan a cabo, estas son: Maestro-Esclavo y Multi-Maestro.

Maestro-Esclavo

La replicación Maestro-Esclavo permite que un solo servidor maestro pueda recibir consultas de lectura/escritura, mientras que los servidores esclavos solo pueden realizar consultas de lectura [Buretta 2009]. Los datos pueden ser modificados en múltiples ubicaciones, entonces la replicación debe procesar los cambios realizados en cada uno de los sitios de forma coordinada. El servidor maestro es el que se encarga de distribuir los cambios a todos los sitios. Los cambios realizados en los destinos fluyen hacia los otros sitios a través del servidor maestro. La Figura 1.6 muestra un escenario de la replicación Maestro-Esclavo.

Figura 1.6: Replicación Maestro-Esclavo. (Cobas 2012).

Multi-Maestro o Maestro-Maestro

La replicación Multi-Maestro permite enviar consultas de lectura/escritura a múltiples servidores. Esta capacidad tiene un considerable impacto en el rendimiento debido a la necesidad de replicar los cambios entre servidores [Buretta 2009]. Esta réplica no tiene designado un servidor maestro, cada ubicación copia los cambios desde todos los otros sitios directamente. Cada sitio es un sitio maestro, y se comunica con otros sitios maestros. Puede ser usada para mantener sitios recuperables ante posibles desastres o caídas, así como para proveer sistemas con alta disponibilidad y para balancear la carga de consultas a través de las distintas ubicaciones. La Figura 1.7 muestra un escenario de la replicación Multi-Maestro.

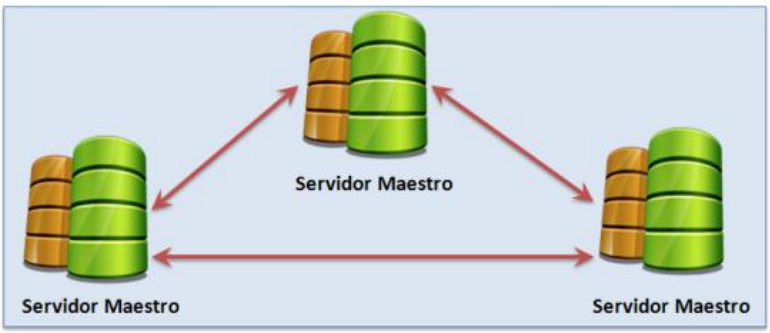

Figura 1.7: Replicación Multi-Maestro. (Cobas 2012).

En la replicación Multi-Maestro, los datos se replican entre servidores para:

- Mejorar la escalabilidad y la disponibilidad
- Almacenar datos e informes
- Integrar datos de varios sitios
- Integrar datos heterogéneos

Por lo general, a cada uno de estos entornos es asociado un modelo de distribución o sea, la forma de transmisión, que puede ser asíncrono o síncrono y cada uno se comporta de la siguiente manera:

La replicación asíncrona: las réplicas son ejecutadas en la BD origen, y los cambios hechos se mantienen registrados, y a menudo en un intervalo de tiempo predeterminado las actualizaciones llegarán en la BD destino.

La replicación síncrona: al contrario del modelo anterior, funciona actualizando primero en la BD destino, y la misma envía una confirmación de que se han actualizado en cada una de sus instancias, y es en este momento que la BD origen ejecuta los cambios.

1.4 Herramientas de bases de datos sobre entorno de software libre

En los últimos años se ha notado la existencia de varias herramientas de bases de datos sobre entornos de software libre que son ampliamente conocidas y utilizadas por varias organizaciones, pero en este trabajo se citará las más destacadas en las literaturas recientes, y que son bastantes utilizadas por las grandes organizaciones en el mercado de los sistemas de información.

PostgreSQL

PostgreSQL es el sistema de gestión de base de datos relacional de código abierto más avanzado del mundo. Es distribuida bajo licencia BSD (del inglés, *Berkeley Software Distribution*), lleva más de 15 años desarrollándose y su arquitectura goza de una excelente reputación por su fiabilidad, integridad de datos y correctitud. PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multi-hilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando [Renderos y García 2012].

Para la implementación de bases de datos distribuidas el gestor PostgreSQL, incorpora herramientas muy conocidas en el mercado, que son las que hacen el servicio de replicación de datos como el *PgCluster* para replicación multi-maestro y *PgPool* para replicación maestro-esclavo. Además PostgreSQL puede ser incorporada con otras herramientas para replicación de datos, una de ellas es el *Slony-I* también para replicación maestro-esclavo y BDR para la replicación maestro-maestro y maestro-esclavo, entre otras.

MySQL

MySQL es un sistema de gestión de bases de datos relacional, distribuido y multi-hilos. Es código abierto y el soporte es brindado por Oracle. La más reciente distribución es la versión 5.7.10 y fue lanzada el 22 de octubre de 2015. MySQL soporta replicación asíncrona unidireccional: un servidor actúa como maestro y uno o más actúan como esclavos.

La replicación en MySQL funciona de la siguiente forma: el servidor maestro escribe actualizaciones en el fichero de log binario, y mantiene un índice de los ficheros para rastrear las rotaciones de *logs*[1]. Estos *logs* sirven como registros de actualizaciones para enviar a los servidores esclavos. Cuando un esclavo se conecta al maestro, informa al maestro de la posición hasta la que el esclavo ha leído los *logs* en la última actualización satisfactoria.

El esclavo recibe cualquier actualización que ha tenido lugar desde entonces, y se bloquea y espera para que el maestro le envíe nuevas actualizaciones. Debe tenerse en cuenta que cuando se usa replicación, todas las actualizaciones de las tablas que se replican deben realizarse en el servidor maestro. De otro modo, se debe ser cuidadoso para evitar conflictos entre actualizaciones que hacen los usuarios a las tablas en el maestro y las actualizaciones que hacen en las tablas de los esclavos.

Cassandra (NoSQL)

Apache Cassandra, es un sistema de gestión de base de datos, escrita en Java, de tipo *Column Family* y NoSQL, de código abierto creada por *Facebook* en 2008, diseñada por *Avinash Lakshman* (uno de los autores de *Dynamo*[2], de Amazon) y Prashant Malik (Ingeniero de Facebook). De varias maneras se puede pensar en Cassandra como *Dynamo 2.0* o una unión de *Dynamo* y *BigTable*. Cassandra se encuentra en producción en *Facebook*.

Altamente escalable, eventualmente consistente, distribuida y almacenamiento estructurado *key-value*. Agrupa las tecnologías de sistemas distribuidos de *Dynamo* y el modelo de datos *BigTable* de Google. Como *Dynamo*, es eventualmente consistente. Como *BigTable*, provee modelo de datos basado en *ColumnFamily* más enriquecido que los sistemas comunes *key-value*. [Apache Cassandra 2012].

En Cassandra existe varias herramientas para la visualización y administración de los datos, la más destacada es *OpsCenter* que ofrece gestión y administración para

---

[1] Log: término utilizado para referirse al archivo en el que se registra toda la actividad de un sistema con funciones, principalmente de auditoria.

[2] Amazon DynamoDB: es un servicio de base de datos NoSQL totalmente administrado que proporciona un rendimiento rápido y fiable con una escalabilidad sin problemas.

los *cluster*[3], esta contiene una edición comunitaria y una empresarial que incluye características adicionales: alertas, balanceo automático de cargas, respaldos en vivo, entre otras.

Comparación entre los SGBDs

Tabla 1.4: Comparación entre los SGBD.

SGBD	Características principales
MySQL	ACID[4], tolerante a fallos, replicación de datos con pocas herramientas.
PostgreSQL	ACID, tolerante a fallos, balanceo de carga, mantiene la coherencia de los datos a través de múltiples nodos replicados, replicación de datos con varias herramientas.
Cassandra	Solo cumple dos de la propiedad CAP[5], balanceo de carga, replicación en forma de anillo.

1.5 Herramientas de réplicas de datos

Slony-I

Slony-I es un software que permite hacer replicaciones maestro/esclavo asíncrono, realizando actualizaciones en cascada.

Slony-I es un maestro “a varios esclavos” sistema de replicación en cascada de apoyo; un nodo puede alimentar a otro nodo que se alimenta de otro nodo) y de comunicación por error. El panorama para el desarrollo de Slony-I es que es un esclavo de replicación del sistema principal que incluye las características y capacidades necesarias para replicar bases de datos de gran tamaño a un número razonablemente limitado de los sistemas esclavos [Group 2009].

SymmetricDS

[3] Cluster: conjunto de varios servidores trabajando con un software y unos datos únicos para ejecutar una tarea común.

[4] ACID: Atomicity, Consistency, Isolation and Durability.

[5] CAP: Consistency, Availability and Partition Tolerance.

Es una herramienta asíncrona de datos para replicación multi-maestro y sirve de apoyo a varios suscriptores permitiendo la sincronización bidireccional. Se utiliza en internet y en las de bases de datos, para replicar tablas entre bases de datos relacionales en tiempo real. Fue diseñado a escala para un gran número de bases de datos, para trabajar a través de conexiones de baja velocidad, y soportar los periodos de interrupción de la red [Long y Henson 2010].

Admite la sincronización a través de plataformas de bases de datos diferentes por el concepto de dialectos de base de datos. Un dialecto de base de datos es una capa de abstracción con la cual interactúa SymmetricDS para aislar la lógica de sincronización de los detalles de implementación específicos de cada base de datos.

PyReplica

Es una herramienta desarrollada en lenguaje *Python*, permite réplicas maestro-esclavo y multi-maestro limitado, funciona de forma asíncrona y especialmente para el gestor PostgreSQL. Se caracteriza por ser de fácil usabilidad, es multiplataforma y permite [Garcia 2010], [Reingart 2008]:

- La replicación condicional
- La detección de conflictos
- El monitoreo de las conexiones
- Notificaciones vía email

Al igual que Slony-I, PyReplica no soporta:

- Replicación de DDL automática (pero puede usarse para propagar ordenes DDL a varios servidores)
- Replicación síncrona
- Resolución de conflictos, los cuales se pueden evitar con reglas o disparadores

BDR

La replicación bidireccional (BDR) es un sistema de replicación asíncrono multi-maestro para PostgreSQL, específicamente diseñado para permitir bases de datos distribuidas geográficamente. Con soporte de hasta 48 nodos (y posiblemente más

en el futuro), BDR es una tecnología de bajo consumo de recursos y bajo mantenimiento para bases de datos distribuidas. Diferente de otras soluciones, la BDR no depende de usar disparadores para recolectar los cambios e insertarlos en una tabla de cola. En vez de eso, procesa el WAL[6] usando el mecanismo de extracción de cambios desarrollado por 2ndQuadrant para la versión 9.4 de PostgreSQL.

Conclusiones del capítulo

Después de un análisis del marco teórico se puede concluir que para el diseño de bases de datos es esencial tener en cuenta las características establecidas por la arquitectura ANSI, según las fuentes bibliográficas consultadas. La diversidad de los modelos de replicación, tipo de fragmentación en las bases de datos distribuidas impide llegar a un consenso único en cual utilizar, para esto, en el diseño de bases de datos distribuidas se debe hacer mediante dos alternativas principales, el top-down, y el bottom-up, confirmándose la necesidad de contar con los varios métodos de fragmentación, replicación y asignación. Existe una diversidad en las herramientas para la implementación de bases de datos distribuidas, destacándose PostgreSQL, Slony, BDR, SymetricDS. Una de las herramientas que se utiliza para lograr mayor disponibilidad en las bases de datos distribuidas, es BDR (según la literatura) esto es debido a que fue desarrollada específicamente para ambientes que se encuentran geográficamente distribuidos.

[6] WAL (*Write Ahead Log*): Es una de las técnicas más utilizadas en la replicación de base de datos, en especial cuando se trata de una arquitectura de tipo Maestro-Esclavo.

CAPÍTULO 2. MODELO PARA LA IMPLEMENTACIÓN DE BASES DE DATOS DISTRIBUIDAS

En este capítulo se aplica el marco teórico establecido en el capítulo 1, que ha guiado el desarrollo del proceso de investigación, y se describen los procedimientos empleados en el diagnóstico y sus resultados, que permitieron valorar el estado que presenta la UAN al inicio de la investigación con relación al tratamiento de la información y se fundamenta un modelo para la implementación de bases de datos distribuidas en la UAN. El modelo lo integran principios, cualidades y componentes que sirven de apoyo a la implementación de la base de datos distribuida, para lograr los objetivos planteados.

2.1 Caracterización de la UAN

La Universidad Agostinho Neto (UAN) fue la primera universidad pública en Angola. Hoy en día con el desarrollo de la Educación Superior las instituciones de la UAN en diferentes regiones del país se han convertido en instituciones independientes, excepto las que se encuentran en la capital (Luanda). Esto ha llevado a considerarla como la institución insigne en Angola. Hoy conforman la UAN 10 instituciones: siete Facultades y tres Institutos Superiores. La misma cuenta con una red de área local (LAN), distribuida entre las instituciones que componen el campo universitario (esto es, las Facultades de Ciencias y de Ingeniería).

Esta investigación se centra, en la base informativa de las Facultades de Ciencias e Ingeniería, ambas con servicios centralizados sobre los datos concernientes a los estudiantes que las mismas atienden.

Las dos instituciones han enfrentado problemas de informatización en sus actividades. La rectoría de los asuntos académicos de la UAN, que es responsable por generar la información académica concerniente a la universidad se le dificulta acceder a dicha información con alto grado de fiabilidad, lo que implica que no exista un crecimiento en calidad y cantidad de sus servicios, ya que dispone de una arquitectura centralizada. Entre las causas de este problema se destaca el siguiente:

- Cuando el sistema de la base de datos falla, se pierde toda disponibilidad de procesamiento y sobre todo de información confiada al sistema

- Dificultad en mantener la recuperación de los datos en caso de un desastre
- Las cargas de trabajo no se pueden difundir entre varias computadoras, ya que los trabajos siempre se ejecutarán en la misma máquina
- No hay un mayor poder de cómputo para una gran cantidad de datos relacionado a las instituciones
- La no existencia de una herramienta informática

Las dificultades que las instituciones enfrentan han provocado parte de los problemas que se vive actualmente en la Universidad, sí se plantease una solución que permite integrar informaciones sobre los datos de las instituciones de la UAN (de manera distribuida) para mejorar la fiabilidad y disponibilidad en la gestión de sus datos, ayudaría en resolver los problemas que la rectoría y las instituciones han enfrentado actualmente debido a falta de un proceso automatizado.

La aplicación de un modelo para la implementación de bases de datos distribuidas ayudaría a resolver mayor parte de los problemas enfrentado por la UAN actualmente brindando aún nuevas ventajas cómo tener mejor control de la gestión de sus datos, ayudando a mantener la fiabilidad y disponibilidad de sus datos.

2.2 Diagnóstico para fundamentar el desarrollo de la propuesta

Un diagnóstico es el proceso mediante el cual se lleva a cabo un análisis para recopilar información que ayude a determinar la situación actual de la organización y detectar sus áreas de mejoramiento. Mediante un diagnóstico se trata de focalizar y evaluar un conjunto de variables que juegan un importante papel en la comprensión, predicción y control del comportamiento de un fenómeno determinado [Shull et al. 2008].

Lo que concierne en el acceso a la información sobre la UAN, la misma cuenta con tres sistemas de gestión (el SIGEA, SIGEST[7], y el SIGA, este último en fase de desarrollo), los mismos utilizan como SGBD el MySQL, y el Oracle, y trabajan siempre que necesario a nivel centralizado, donde existe un servidor conectado con las estaciones de trabajo.

[7] SIGEST: Sistema de gestión de estudiantes.

Para el desarrollo del diagnóstico se aplicaron métodos científicos los que de manera general se describen a continuación.

2.2.1 Métodos aplicados en el diagnóstico

Para realizar el diagnóstico de la presente investigación se realizaron entrevistas y se aplicaron encuestas a varios especialistas con experiencia en gestión de información, desarrolladores de software y especialistas informáticos.

El objetivo principal del diagnóstico fue evaluar el estado en que se encuentra la utilización y desarrollo de aplicaciones de gestión para el soporte de los trabajos que se ejecutan en la UAN, así como valorar las posibilidades que ofrece la introducción de un modelo basado en técnicas de replicación y fragmentación para la implementación de bases de datos distribuidas, que contribuya a un incremento de la fiabilidad y la disponibilidad en el tratamiento de los datos institucionales en la Universidad Agostinho Neto.

El análisis de estos aspectos se realizó a partir de las problemáticas detectadas al aplicar los métodos y técnicas de investigación planteadas anteriormente, lo cual se describe a continuación:

- **Encuestas:** Se realizaron encuestas aplicando un cuestionario con lo que se conoció la valoración de los participantes con respecto a temas relacionados con el objeto de estudio. El grupo encuestado fue de 30 especialistas, donde se han seleccionado como muestra 20 de ellos, para asegurar la confiabilidad de las respuestas. Se evaluó la idoneidad de los especialistas en la temática abordada. La composición final de los especialistas es reflejada en la Tabla 2.1.
- **Entrevista a profundidad:** Fue realizada a varios especialistas y doctores en la especialidad de Informática con experiencias en análisis de datos y la información. La población seleccionada fue de 20 especialistas, donde se escogió una muestra, como se refleja en la Tabla 2.2.

Tabla 2.1: Composición de los especialistas involucrados en la encuesta. (Elaboración propia).

Perfil de trabajo	***Cantidad***	***Categoría científica***	***Cantidad***	***Instituciones***	***Cantidad***

Profesores	8	Máster	8	UAN	17
Desarrolladores	5	Licenciados	9	MES	3
Profesionales con labor científica	7	Doctor	3		
Total	**20**	**Total**	**20**	**Total**	**20**

2.2.1.1 Resultado de aplicación de la encuesta

Un resumen de los principales resultados se muestra a continuación:

Pregunta No. 1: ¿Considera que la estructura actual de la organización de los datos en la UAN no es ideal para brindar mayor disponibilidad y fiabilidad de los datos?

En la Figura 2.1 muestra que el 90% de los encuestados afirmaron que la estructura de la organización de los datos en la UAN no está bien estructurada, mientras que el 10% restante han afirmado que la misma está, lo que indica que hay mucha necesidad de emplear una herramienta para la implementación de bases de datos distribuidas en la organización, para lograr mayor disponibilidad y fiabilidad de la información.

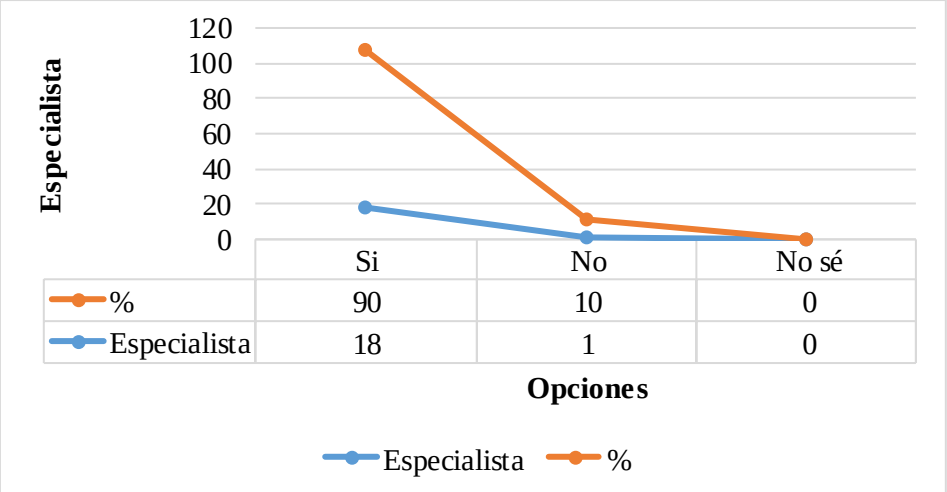

Figura 2.1: Sobre la estructura de la información en la UAN.

Pregunta No. 2: ¿Una herramienta para la implementación de bases de datos distribuidas pudiera hacer más eficaz (esto es, brindado mayor disponibilidad y fiabilidad) al tratamiento de los datos?

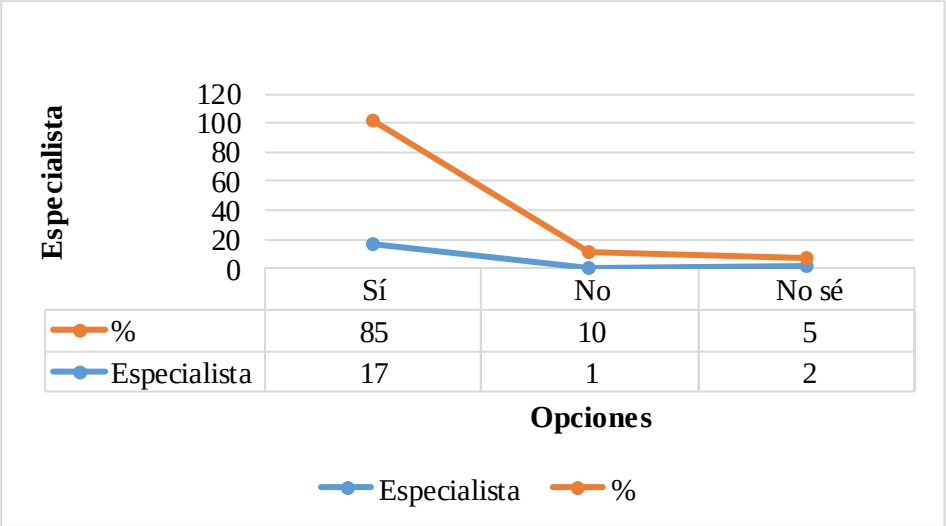

Figura 2.2: Sobre la necesidad de una herramienta para la implementación de bases de datos distribuidas.

En la Figura 2.2 se muestra que el 85% de los encuestados afirmaron que la implementación de un sistema de bases de datos distribuidas ayudaría bastante a la organización a manejar mejor su información y así hacer más eficiente el tratamiento de los datos. Esto es un elemento que indica la importancia de disponer de una herramienta que facilita la implementación de bases de datos distribuidas.

Pregunta No. 3: ¿Considera que el empleo de técnicas de replicación y fragmentación puede ser adecuado en la implementación de bases de datos distribuidas?

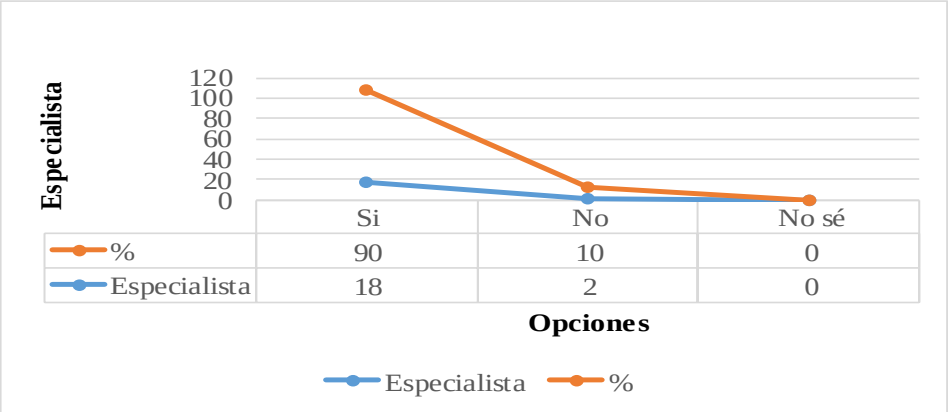

Figura 2.3: Sobre el empleo de técnicas de replicación y fragmentación.

En la Figura 2.3 se muestra que el 90% de los encuestados afirmaron que el empleo de técnicas de replicación y fragmentación puede ser adecuado en la implementación de bases de datos distribuidas en la UAN, ya que mejoraría el tratamiento de la información y la disponibilidad para cada una de las áreas que la necesiten, por lo que favorecería la toma de decisiones de la organización y contribuirá al cumplimiento de sus objetivos. Estas técnicas aportan mayor seguridad, ya que los datos no utilizados por un nodo local no son almacenados en este nodo y por lo tanto no están al alcance de usuarios sin autorización.

2.2.1.2 Resultado de la realización de entrevista a profundidad

Para conocer la opinión de los especialistas sobre la contribución del modelo para la implementación de base de datos distribuidas en la UAN, se aplicó la entrevista a profundidad con el objetivo de verificar la utilidad del modelo en la Universidad, y la necesidad y utilidad de una base de datos en la UAN utilizando el modelo. Para ello se utilizó una guía que contemplaba como aspectos principales los siguientes:

- Sobre la importancia de gestionar información en la Universidad para su socialización y su correspondencia con los objetivos estratégicos
- Si considera útil la socialización de la información y el conocimiento para apoyar la toma de decisiones
- Si considera necesario organizar y gestionar mejor los datos para apoyar la toma de decisiones

- Sobre la importancia de mejorar el tratamiento de la información en la UAN y que pueda ser gestionada de forma ágil con el empleo de una herramienta para la implementación de bases de datos distribuidas
- Sobre la necesidad de lograr mayor disponibilidad y fiabilidad de los datos distribuyendo los mismos

Fueron aplicadas para las entrevistas las preguntas citadas anteriormente, y para tal se han seleccionados 20 especialistas y una muestra de 10 especialistas, como se observa en la Tabla 2.2.

Tabla 2.2: Composición de los especialistas involucrados en la entrevista. (Elaboración propia).

Perfil de trabajo	*Cantidad*	*Categoría científica*	*Cantidad*	*Instituciones*	*Cantidad*
Profesores	4	Máster	2	UAN	8
Desarrolladores	4	Licenciados	6	MES	2
Profesionales con labor científica	2	Doctor	2		
Total	**10**	**Total**	**10**	**Total**	**10**

Como resultado de las entrevistas realizadas se destaca lo siguiente: El 90% de los entrevistados fueron muy amplios con sus respuestas, la mayoría analizó sobre la posibilidad que tendría la Universidad en convertirse en un centro de excelencia guiado por los análisis que se podrían hacer a estos datos. De los 90% unos 60% analizaron también la necesidad que el tratamiento de los datos tengan al menos un esquema o un estándar, para que cualquier desarrollador de la Universidad pueda entender la estructura de los mismos, y puede usarlos para mantener la información siempre disponibles a sus instituciones.

También manifestaron la importancia de que sea necesario la incorporación de 3 planteamientos esenciales para que el modelo sea factible tanto en información como en recursos:

1. El modelo debe ser completo, simple y adaptable
2. Debe tener una estructura general
3. La aplicabilidad del mismo debe traer beneficios en la UAN

2.3 Modelo para la implementación de bases de datos distribuidas en un entorno académico

Caracterización de un entono académico

Los entornos académicos se caracterizan por disponer de sistemas de información heterogéneos y complejos. Todos estos sistemas necesitan integrarse para soportar procesos más rápidos, más precisos y proporcionar información de gestión consistente y significativa. Los sistemas deben ser fiables, escalables, seguros y gestionables eficazmente. También deben ser robustos, en este caso se refiere que deben disponer de un sistema de *backup* o respaldo, con réplicas de los datos sin descuidar el balanceo de carga.

Desde el punto de vista del acceso a la información y a los servicios es fundamental la disponibilidad 24 por 7 de los principales sistemas. Deben ser complementados con una garantía de que no hay pérdida de los datos en los diferentes sistemas. Es esencial la integración de los datos provenientes de bases de datos distribuidas por las diferentes unidades de la organización y que con frecuencia tendrán diferentes estructuras (fuentes heterogéneas).

Se puede resumir que los sistemas académicos se caracterizan por entornos de alta disponibilidad, eficaces, fiables, robustos, redundantes y escalables, con niveles de seguridad.

El término modelo proviene del italiano ***"modello"***, que significa representación de algo que se debe seguir o imitar. En la literatura también se encuentra otras definiciones válidas como:

- Una representación de un objeto, sistema o idea, de forma diferente al de la entidad misma. Con el propósito de ayudarnos a explicar, entender o mejorar un sistema. Un modelo de un objeto puede ser una réplica exacta de éste o una abstracción de las propiedades dominantes del objeto [Belloch 2012]
- La representación concisa de una situación; por eso representa un medio de comunicación más eficiente y efectivo. Cualquier modelo, debe describir al sistema con suficiente detalle para hacer predicciones válidas sobre el

comportamiento del sistema. Con características que deben corresponder a algunas características del sistema a modelar [Ríos 2011]

Un modelo permite una comprensión más plena del objeto de estudio para resolver un problema y representarlo de alguna forma. En el proceso de modelación, el eslabón que media es el modelo que actúa como función representativa sustituyendo el objeto [Febles 2012].

2.3.1 Principios, cualidades y componentes para el diseño del modelo para la implementación de bases de datos distribuidas en un entorno académico

El modelo elaborado para la implementación de bases de datos distribuidas en un entorno académico, se sustenta en los siguientes principios:

La **estandarización**. Se refiere a la utilización de los estándares ampliamente aceptados y utilizados en la práctica mundial para la implementación de bases de datos distribuidas. Entre ellos, los de modelación y arquitectura de sistemas bases de datos (*ANSI-SPARC*) explicado en el epígrafe 1.1.1 del capítulo 1. Categoría científica.

La **interoperabilidad**. Para lo cual es indispensable un diseño adecuado de los procesos, que permita que las bases de datos sean compatibles y usables sin importar la tecnología empleada para su gestión.

La **actualización.** Para garantizar la calidad de los datos mediante la retroalimentación de la información que nutre al modelo.

Flexibilidad. Que se logra utilizando componentes con funcionalidades genéricas y que se adapten a las particularidades de las bases de datos existentes para el proceso de su utilización en el entorno.

El modelo presenta las siguientes **cualidades**:

Amplitud. Que brinda la capacidad de analizar y de emplearse por aplicaciones que utilicen diferentes tipos de sistemas para la gestión de los datos.

Integralidad. Dada por los componentes del modelo que cubren de manera integrada y coherente la mayoría de los elementos necesarios para la implementación de bases de datos distribuidas en un entorno académico.

Mejora continua. El modelo constantemente se mejora con los resultados que se van obteniendo, en particular con la incorporación del componente relativo a la prueba exploratoria.

Los principales **componentes** del modelo son:

- Diseño de los datos
- Diseño de la distribución
- Diseño físico de la BDD
- Prueba exploratoria

En la Figura 2.4 se muestra un gráfico con la interacción de estos componentes, y la relación que se genera entre ellos y en el epígrafe 2.3.3 se describen cada uno de sus componentes.

2.3.2 Descripción general del modelo

El objetivo principal del modelo es brindar los pasos necesarios para la implementación de bases de datos distribuidas, lo que posibilitará a la organización académica una mejor gestión de sus datos permitiendo una mayor disponibilidad y fiabilidad de estos. Con el modelo definido será posible implementar una estructura de bases de datos distribuidas.

El modelo está formado por componentes que se relacionan entre sí, como se muestra en la Figura 2.4.

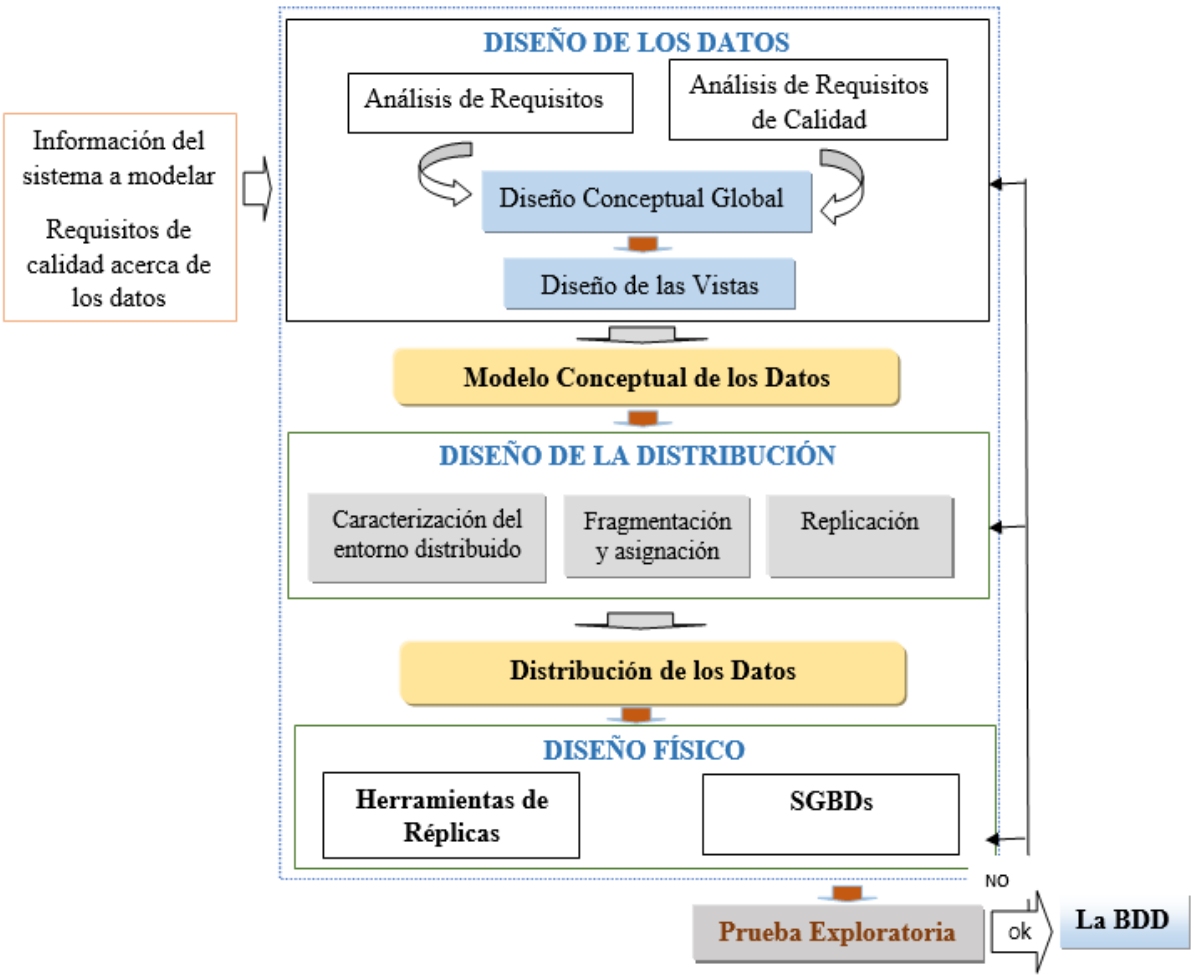

Figura 2.4: Modelo para la implementación de bases de datos distribuidas. (Elaboración propia).

La información de entrada y salida del modelo

La entrada del modelo es caracterizada con un análisis de requerimientos sobre los datos que serán utilizados por la Universidad para la gestión de la información académica de los estudiantes de las instituciones que constituyen la misma.

Si el desarrollo es en forma *top-down*, en cuanto a la información del sistema a modelar, sería la información a partir del análisis de los flujos de datos que definen el sistema que se detallaron en el análisis del sistema.

En el caso de un sistema existente que sería un *bottom-up* donde se encuentra cada base de datos local, sería la información del sistema de cada BD local como información de entrada al modelo.

También se definen los requisitos de calidad, que son los requisitos de calidad de los datos basado en la ISO 25024.

Como salida se obtiene la base de datos distribuida.

2.3.3 Descripción de los componentes del modelo

a) Componente: Diseño de los datos

En este componente el proceso se inicia con el análisis de requerimientos que define el ambiente del sistema y determina tanto las necesidades de los datos como las necesidades del proceso. Esto tiene como objetivo obtener tanto los datos como las necesidades de procesamiento de todos los posibles usuarios de la base de datos. Se fijan los requisitos del sistema, los objetivos que se deben cumplir respecto al grado de fiabilidad y disponibilidad. Para esto, se hace el análisis de todos los datos que son utilizados por las aplicaciones que tienen acceso a la base de datos. Donde se define también el esquema conceptual global de la base de datos, y el mismo define la visión global de toda BD.

El esquema conceptual global describe la estructura de toda la base de datos para la comunidad de usuarios del sistema a modelar, ocultando los detalles de las estructuras físicas de almacenamiento, y se concentra en describir entidades, tipos de datos, vínculos, operaciones de los usuarios y restricciones.

En el diseño de las vistas se construye un diagrama entidad-relación para cada grupo de usuarios, donde los requerimientos pueden tener diversos formatos como entrevistas, documentación de un sistema existente, formularios y reportes propuestos.

Técnicas: Entrevistas, Criterio de expertos

Herramientas: DB Designer, MS Visio, ER/Studio

b) Componente: Diseño de la distribución

Este componente incluye la caracterización del entorno, donde se localizan los nodos, así como el diseño de los esquemas conceptuales locales que se distribuirán a lo largo de todos los nodos de la base de datos distribuida. Después de definir los esquemas conceptuales locales y precisar la información de acceso a

los datos, se pasa a las tres actividades fundamentales de este componente: la fragmentación, asignación y la replicación.

- Fragmentación de los datos de las tablas, esto es de los datos con más accesos por cada institución (subdivisiones de relaciones)
- Asignar todos los fragmentos replicados en los nodos donde son más necesitados por cada institución, para mejorar el desempeño de las consultas, como la reducción en los tiempos de respuesta, la fiabilidad y disponibilidad de los datos para lograr un mejor control
- Replicación de los fragmentos de la base de datos por cada nodo, y también los datos que se encuentran en el nodo central para otros respectivos nodos; los procesos de fragmentación horizontal, vertical, hibrida y de replicación se hace conforme se explicó en los epígrafes 1.2.5 y 1.3 del capítulo 1

La etapa de asignación en este componente es fundamental porque cada fragmento (o cada copia de un fragmento) es asignado a un nodo determinado en el sistema distribuido. Este proceso se denomina distribución de los datos. La elección de la sede y el grado de replicación depende de los objetivos de fiabilidad y disponibilidad para el sistema, y de los tipos de frecuencias de transacciones introducidas en cada nodo.

Herramientas: MS Visio, Star UML, DB Designer

Técnicas: Fragmentación, Replicación

c) Componente: Diseño físico de la BDD

En este componente se identifica el nivel de almacenamiento de la base de datos. Aquí se definen con que SGBD se va a trabajar, si se desea un ambiente heterogéneo u homogéneo, con que SO (sistema operativo), después se definen las herramientas para el trabajo con BDD, y las herramientas de réplica y fragmentación de los datos. También se especifican la distribución de las tablas, formateo de los datos como se muestra en la Figura 2.5.

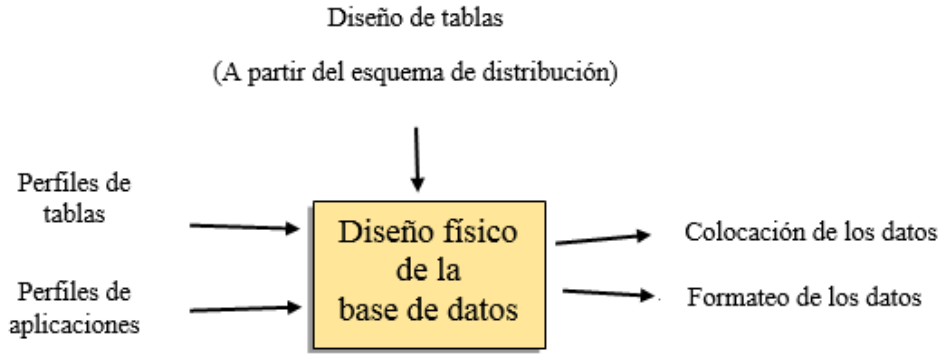

Figura 2.5: Diseño físico de la base datos. (Elaboración propia).

Para los ambientes homogéneos se debe garantizar que todos los SGBDs locales ofrezcan interfaces idénticas, deben utilizar el mismo modelo de datos, el mismo lenguaje de definición de datos (LDD) y el mismo lenguaje de manipulación de datos (LMD). Y el SGBD global debe utilizar las mismas interfaces, para que el usuario (local o global) pueda acceder tanto los datos locales o datos remotos a través del mismo LMD.

En ambientes heterogéneos se debe garantizar que los SGBDs locales utilicen modelos de datos y LMDs diferentes. Otra opción sería garantizar que el SGBD de la red ofrezca al usuario global residente en un nodo, una visión de la base de datos distribuida en el mismo modelo de datos que la base de datos local. Esta opción es esencial porque no será necesario que los usuarios residentes utilicen un nuevo LMD en un determinado nodo para que puedan acceder a datos remotos.

Herramientas: Slony-I, Reko, SymmetricDS, Magic@ Data Replication eXtensible Solution, BDR

Sistemas gestores de bases de datos: PostgreSQL, MySQL

Para la propuesta de la implementación en la UAN se utilizará las herramientas detalladas en la Tabla 2.3.

Tabla 2.3: Justificación de las herramientas seleccionadas. (Elaboración propia).

Producto	***Características***
	Posee un diseño para ambientes de alto volumen.
	Sus considerables ahorros en operación, su licencia BSD, rentabilidad en los modelos de negocio, y su extensibilidad son

	características muy atractivas.
	Es multiplataforma (compatible con Linux, Windows, y varias versiones de UNIX).
PostgreSQL	*Posee estrategias preventivas para desastres: posee una estructura adelantada de registros que evita pérdida de datos en caso de falla eléctrica, de sistema operativo o de hardware.*
	Incorpora productos software para el trabajo con el mismo: PgAdmin, PgAccess, Psql, PhpPgAdmin, PgCluster, etc.
	Utiliza el PgCluster, el Slony-I y PgPool para la replicación de datos, el primero en réplicas multi-maestro y los demás en réplicas maestro-esclavo.
	Posee buen soporte brindado por la gran comunidad de usuarios que existe en el mundo que aportan experiencias y resultados obtenidos del uso del mismo.
	Compatible con Linux, y varias versiones de UNIX).
BDR	***Es tolerante ante fallos, manteniendo los datos de réplica en un estado estable en caso de desconexión.***
	Sistema de replicación multi-maestro, diseñado específicamente para permitir bases de datos distribuidas geográficamente.
	Es una tecnología de bajo consumo de recurso y bajo mantenimiento.

a) Componente: Prueba exploratoria

Este componente representa la fase final del modelo. En este componente se revisa todo el diseño, se monta la BDD en un ambiente controlado y se realizan pruebas de carga, disponibilidad, fiabilidad, estrés y después se monitorea la BD a partir de los resultados y se va a un proceso de mejora o retroalimentación del proceso.

Esta etapa de actividades es efectuada para medir todos los procesos que se definieron en el modelo y para esto, obtener como salida la base de datos distribuida. Con la ayuda de expertos se pueden realizar las revisiones de todas las etapas del modelo y garantizar si se cumplieron con los pasos establecidos. Si se brinda el cumplimento de los procesos entonces se obtiene como salida la base de datos distribuida, si no, se hace una etapa de retroalimentación empezado desde el primer componente.

2.3.4 Implementación del modelo para la gestión académica de la UAN

Para la implementación del modelo en la UAN:

Componente 1: Diseño de los datos

Paso 1: Se realiza un análisis sobre la información de la gestión académica de la Universidad, mediante el proceso de entrevistas e históricos de documentos para obtener los datos necesarios para el análisis de requisitos de la calidad de los mismos. Partiendo del principio que no existen bases de datos ya creadas. En paralelo se realiza el análisis de los requisitos de calidad de los datos centrándose en fiabilidad y disponibilidad de los datos.

Paso 2: Se desarrolla el modelo conceptual global, que es constituido por 14 tablas. Se obtuvo un modelo de diagrama entidad relación, mostrado en el ANEXO 1.

Paso 3: Transformación del modelo lógico a un esquema relacional, ver ANEXO 2.

Paso 4: Para el diseño de las vistas se han seleccionados solo algunas relaciones del modelo conceptual global, ver ANEXO 3.

Componente 2: Diseño de la distribución

Paso 1: Caracterización del entorno distribuido

Para la distribución de los datos se ha definido un esquema general de la base de datos distribuida propuesta para la gestión académica de la Universidad como se muestra en la Figura 2.6.

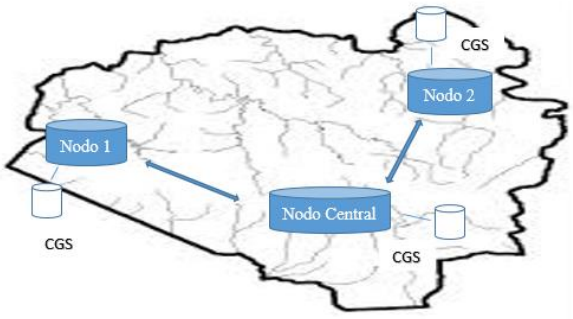

Figura 2.6: Esquema general de la BDD para la Universidad. (Elaboración propia).

El mismo esquema está constituido en 3 nodos: Nodo central, Nodo 1 y Nodo 2. Donde cada nodo contiene un SGBD local que es responsable por el control de los datos locales y el mismo mantiene los datos replicados de cada nodo. Para los SGBD de cada nodo ver Tabla 2.3 del epígrafe 2.3.3.

El esquema posee un componente de comunicación de datos que permite a todos los nodos comunicarse entre ellos, y contiene la información sobre los nodos y los enlaces. Además en el esquema se mantiene la información específica a la naturaleza distribuida del sistema: los esquemas de fragmentación, replicación y de asignación de la base de datos.

Descripción de la arquitectura de los nodos

A continuación, se especifica la comunicación entre los nodos para el procesamiento de los datos a través de la red, que es presentado en dos casos:

Caso 1: Está constituido por 3 nodos. Existe un nodo central que se encarga de la dirección de los demás nodos y tiene autoridad para insertar, modificar, suprimir o leer cualquier dato en la red. El resto de los nodos solo tienen autoridad de insertar, modificar y hacer lectura de los datos que están relacionados con sus sitios locales, en el caso de la red en su totalidad, los demás nodos tienen solamente la autoridad para leer los datos en la misma.

Caso 2: Solo el Nodo central puede modificar la base de datos. Los Nodos 1 y 2 que tienen copia de algunos datos del Nodo central, solo están autorizados para lectura. Además, todos los días los Nodos 1 y 2 son actualizados por el Nodo central.

Paso 2: Fragmentación y asignación

La fragmentación tiene como objetivo dar autonomía local a los distintos nodos, cada nodo almacena los datos que más acceden. Para ello, se analizan los datos que más se utilizan en cada nodo. De aquí se determinará una fragmentación horizontal (por selección) para cada nodo. Se ha hecho la selección de algunas relaciones para la fragmentación, para las demás fragmentaciones, ver ANEXO 4.

A través de la autonomía local que se quiere dar a los datos, se prevé almacenar la información de las tablas. IMPARTIR, GRUPOS, ASIGNATURAS, CURSOS,

INFO_PROFESORES, CLASIFICACIONES, GRUPOS, ESTUDIANTES, PLANO, PROGRAMA, DEPARTAMENTO, PROFESION, PAIS y TITULACIONES referentes a la Facultad de Ciencias en el Nodo 1, los referentes a la Facultad de Ingeniería en el Nodo 2, y los referentes al Nodo central. Es decir, todos los datos se almacenarán en el nodo de la que procedan.

Se quiere que en los Nodo 1 y Nodo 2 se maneje la información de los estudiantes que allí estudian. Por ello, se aplica una fragmentación horizontal primaria en la relación ESTUDIANTES.

ESTUDIANTES _i: $\sigma_{\text{Sede = 'i'}}$ (ESTUDIANTES)
Donde i= {Nodo 1, Nodo 2}

Se quiere también que en cada nodo se maneje la información de las titulaciones que allí se imparten. Por ello se aplica una fragmentación horizontal primaria en la relación TITULACIONES.

TITULACIONES_i: $\sigma_{\text{Sede = 'i'}}$ (TITULACIONES)
Donde i= {Nodo Central, Nodo 1, Nodo 2}

Para tener la información completa sobre las titulaciones de cada nodo necesitamos conocer que curso se imparten y los grupos formados para cada curso, por lo que tendremos, tanto para la relación CURSOS como para la relación GRUPOS, que realizar una fragmentación derivada.

CURSOS_i: $\ltimes_{\text{id\_titulacion}}$ TITULACIONES_i

GRUPOS_i: GRUPO $\ltimes_{\text{id\_titulacion, curso}}$ CURSOS_i
Donde i= {Nodo central, Nodo 1, Nodo 2}

De la misma forma asignamos las asignaturas impartidas en las distintas titulaciones a los nodos correspondientes realizando la fragmentación horizontal derivada de la relación de asignaturas con relación a los fragmentos los cursos correspondientes a cada nodo (CURSOS_i).

ASIGNATURAS_i: ASIGNATURAS $\ltimes_{\text{id\_titulacion,curso}}$ CURSOS_i
Donde i= {Nodo central, Nodo 1, Nodo 2}

Asignación de los fragmentos a los nodos.

Tabla 2.4: Asignación de los fragmentos.

Relación	*Nodo1*	*Nodo2*	*Nodo Central*
Estudiantes	Estudiantes_nodo1	Estudiantes_nodo2	
Titulaciones	Titulaciones_nodo1	Titulaciones_nodo2	Titulaciones_nc
Cursos	Cursos_ndo1	Cursos_nodo2	Cursos_nc
Grupos	Grupos _nodo1	Grupos_nodo2	Grupos_nc
Asignaturas	Asignaturas_nodo1	Asignaturas_nodo2	Asignaturas_nc
Impartir	Impartir_nodo1	Impartir _nodo2	
Profesores	Info_Profesores_nodo1	Info_Profesores_nodo2	
	Nominas_Profesores		

Paso 2: Replicación

En el caso de la replicación, las relaciones CLASIFICACIONES, DEPARTAMENTOS, PROGRAMAS, INSTITUCIONES, PLANOS, PROVINCIAS, ESTUDIANTES, ASIGNATURAS, CURSOS, GRUPOS y PROFESIONES se van a replicaren todas en los nodos. La información contenida en la tabla CLASIFICACIONES es tan elemental y las replicaciones se realizarán todos los días.

Esquema de replicación: A continuación, en la Tabla 2.5 se pueden ver las relaciones replicadas que se indicaron en el proceso anterior. A principio se hizo una replicación total para conocer el nivel de disponibilidad y fiabilidad de los datos de la base de datos.

Tabla 2.5: Esquema de replicación de algunas tablas. (Elaboración propia).

Relación	*Nodo 1*	*Nodo 2*	*Nodo central*
CLASIFICACIONES	RN1_ CLASIFICACIONES	RN2_ CLASIFICACIONES	RNC_ CLASIFICACIONES
DEPARTAMENTO	RN1_ DEPARTAMENTO	RN2_ DEPARTAMENTO	RNC_ DEPARTAMENTO
INSTITUCION	RN1_INSTITUCION	RN2_INSTITUCION	RNC_INSTITUCION
PROGRAMA	RN1_PROGRAMA	RN2_PROGRAMA	RNC_PROGRAMA
PLANO	RN1_PLANO	RN2_PLANO	RNC_PLANO
PROVINCIA	RN1_PROVINCIA	RN2_PROVINCIA	RNC_PROVINCIA
PROFESION	RN1_PROFESION	RN2_PROFESION	RNC_PROFESION
ESTUDIANTE	RN1_ESTUDIANTE	RN2_ESTUDIANTE	RNC_ESTUDIANTE
CURSO	RN1_CURSO	RN2_CURSO	RNC_CURSO
ASIGNATURA	RN1_ASIGNATURA	RN2_ASIGNATURA	RNC_ASIGNATURA

Componente 4: Prueba exploratoria

Las pruebas fueron ejecutadas en cada uno de los casos (caso 1 y caso 2) del entorno distribuido anteriormente descrito (ver epígrafe 2.3.3) y las mismas arrojaron resultados satisfactorios.

A continuación se presenta cada una de las pruebas:

Disponibilidad de los datos: se insertaran 300 tuplas en el Nodo central, y las mismas fueron replicadas en cada uno de los nodos (Nodo 1 y Nodo 2). Donde el tiempo de ejecución de la consulta realizada en el momento de replicación fue de 26 Ms.

La misma operación se realizó nuevamente, insertando 500 y 700 tuplas. El siguiente gráfico muestra el comportamiento del tiempo de ejecución de la consulta antes y después de ser insertadas las tuplas, en el entorno.

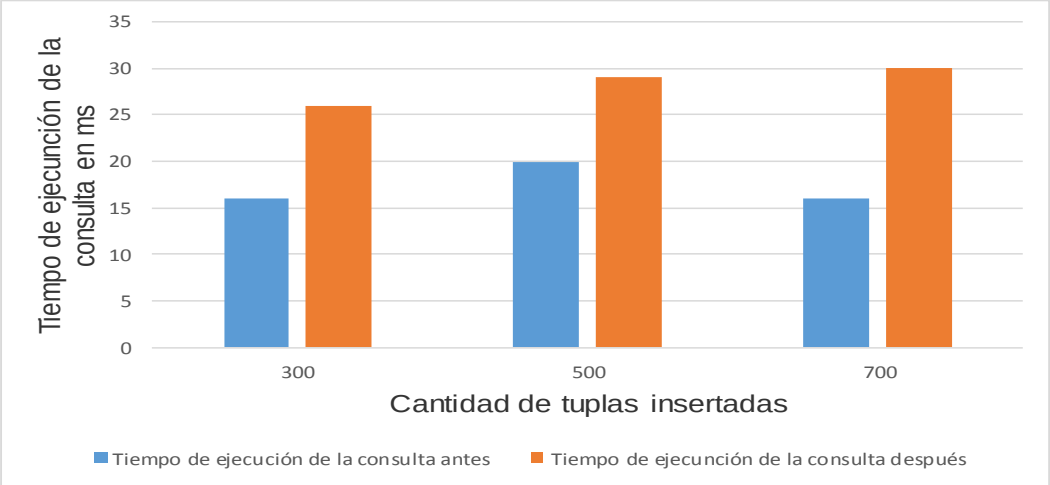

Figura 2.7: Prueba de disponibilidad.

Los tiempos de ejecución de la consulta no tuvieron una diferencia significativamente grande al aumentar la cantidad de tuplas insertadas. Se comprobó que el sistema de réplica no afecta en gran medida los tiempos de respuesta del servidor donde se encuentra funcionando.

Tolerancia ante falos: El sistema soporta que puedan existir fallos en uno de los nodos, como por ejemplo la pérdida de conexión en el momento de la replicación por parte de alguno de los nodos, manteniendo el correcto comportamiento de la base de datos. Fue realizada un proceso de replicación enviando una carga de 500

tuplas desde el Nodo Central al Nodo 1 y Nodo 2. En medio del proceso se detuvo el Nodo 1 y Nodo 2 y luego de 5 minutos se inició. Al encender los Nodos receptores los datos fueron recibidos en sus tablas correspondientes en el mismo orden en que fueron enviados.

Se comprobó que el sistema de réplica funciona correctamente al ser instalado en varios niveles, ubicados en distintos nodos. Se evidenció la réplica multi-maestra, ya que ambos servidores se comportan como maestros, al poder realizar cambios en la base de datos.

Conclusiones del capítulo

En el presente capítulo se han expuestos los conceptos fundamentales para el diseño del modelo para la implementación de base de datos distribuidas en un entorno académico.

A partir del diagnóstico realizado en el capítulo 1 se identificaron los problemas que permitieron definir el modelo con 4 principios, 3 cualidades y 4 componentes.

Se presenta el esquema general del modelo y se fundamentan sus componentes. Al aplicar el modelo en la gestión académica de la UAN se obtuvo 14 tablas, 7 fragmentos distribuidos en los nodos conforme fueron especificados, y replicados en los nodos de la base de datos distribuida implementada, para lograr mayor disponibilidad y fiabilidad de los datos, todo implementado sobre tecnología de software libre que servirá de base a la validación del modelo.

Se han definidos los conceptos importantes que sostienen el modelo, como la definición de los procesos que constituyen cada componente. También fue definido un esquema de referencia para la base de datos distribuida a ser implementada en la Universidad.

CAPÍTULO 3. ANÁLISIS DE RESULTADOS

Introducción

En el capítulo se valida la propuesta del modelo para la implementación de base de datos distribuidas en la UAN, mediante diferentes métodos. Se explica, además, la forma en que fueron aplicados, teniendo en cuenta el propósito de la validación. En el proceso intervinieron expertos y usuarios potenciales para la aplicación del modelo. Debido a las características de la investigación se aplicaron métodos cuantitativos y cualitativos, que fueron posteriormente triangulados para lograr una mayor precisión y objetividad de las comprobaciones.

Técnica de Iadov: esta técnica se aplica para medir el índice de satisfacción de los usuarios con la propuesta del modelo para la implementación de bases de datos distribuidas.

Entrevista a profundidad: con el objetivo de verificar la posibilidad de la utilidad del modelo diseñado en las instituciones de la UAN, debido a sus características.

Cuasi experimento: para comprobar las variables de fiabilidad y disponibilidad de los datos con la aplicación del modelo en un entorno real, teniendo como caso de estudio la gestión académica de la UAN.

Triangulación metodológica: con el objetivo de minimizar el sesgo de la investigación.

En la UAN la aplicación de este modelo es necesaria por los beneficios que aporta, debido fundamentalmente a necesidades de tener una mayor fiabilidad y disponibilidad en relación al tratamiento de los datos. Para esto se arribó luego de sostener entrevistas con especialistas y procesar una encuesta a funcionarios de las instituciones que constituyen la UAN y a profesionales de la rectoría de la misma. El diseño de la encuesta puede ser consultada en el ANEXO 5.

3.1 Evaluación de la satisfacción por usuarios potenciales aplicando la técnica de Iadov

Para el conocimiento del estado de la satisfacción del usuario respecto a la elaboración de un modelo basado en la integración de técnicas de fragmentación y replicación de datos para la implementación de bases de datos distribuidas que

contribuirá a un incremento de la fiabilidad y la disponibilidad en el tratamiento de los datos institucionales en la Universidad Agostinho Neto, es de gran utilidad para la toma de decisiones y en la validación de la propuesta.

La técnica de Iadov constituye una vía para el estudio del grado de satisfacción de los implicados en el proceso objeto de análisis. La misma ha sido aplicada para valorar la satisfacción en múltiples campos y como parte de diagnósticos y validaciones en diferentes investigaciones [Febles 2012]. Para el desarrollo de esta técnica se aplicó un cuestionario detallado en el ANEXO 6 el cual fue aplicado a 16 profesionales con experiencia en el diseño de bases de datos distribuidas, analistas de datos encuestados en la UAN, que permitió conocer el grado de satisfacción sobre el modelo elaborado para la implementación de bases de datos distribuidas, ver el ANEXO 7.

El cuestionario aplicado posee una estructura interna que sigue una relación entre tres preguntas cerradas y dos abiertas. La relación entre las preguntas cerradas queda establecida a través del denominado "cuadro lógico de Iadov". La tabla 3.1 muestra el cuadro lógico modificado con las preguntas cerradas empleadas en la evaluación.

Tabla 3.1: Cuadro lógico de Iadov.

	1. ¿Considera usted que se deba implementar bases de datos distribuidas sin contar con una herramienta que oriente el proceso?								
	No			**No sé**			**Sí**		
3. ¿Le satisface la representación de este modelo para implementar bases de datos distribuidas?	2. ¿Si usted necesitara implementar bases de datos distribuidas usaría el modelo propuesto?								
	Sí	**No sé**	**No**	**Sí**	**No sé**	**No**	**Sí**	**No sé**	**No**
No me gusta mucho	1	2	6	2	2	6	6	6	6
No me gusta tanto	2	2	3	2	3	3	6	3	6
Me da lo mismo	3	3	3	3	3	3	3	3	3
Me disgusta más de lo que me gusta	6	3	6	3	4	4	3	4	4
No me gusta nada	6	6	6	6	4	4	6	4	5
No sé qué decir	2	3	6	3	3	3	6	3	4

Para obtener el índice de satisfacción grupal (ISG) se trabaja con los diferentes niveles de satisfacción que se expresan en la escala numérica que oscila entre +1 y -1 de la siguiente forma:

Tabla 3.2: Escala de calificación del nivel de satisfación.

+1	***Máximo de satisfacción***
0,5	Más satisfecho que insatisfecho
0	No definido y contradictorio
-0,5	Más insatisfecho que satisfecho
-1	Máxima insatisfacción

Donde la escala de satisfacción es la siguiente: clara satisfacción (A), más satisfecho que insatisfecho(B), no definida (C), más insatisfecho que satisfecho (D), clara insatisfacción (E) y contradictoria (C). La cantidad de respuestas por categoría, es empleada para calcular el Índice de Satisfacción Grupal (ISG), donde se calcula mediante la siguiente formula:

$$ISG = \frac{\mathbf{A(+1) + B(+0.5) + C(0) + D(-0.5) + E(-1)}}{N}$$

Donde N representa el número total de sujetos del grupo.

El índice grupal varía entre los valores +1 y -1. Los valores que se encuentran comprendidos entre -1 y -0,5 indican insatisfacción; los comprendidos entre - 0,49 y +0,49 evidencian contradicción y los que caen entre 0,5 y 1 indican que existe satisfacción.

La Figura 3.1 muestra los resultados de la técnica de Iadov. Fue obtenido para ISG el valor de **0,90** lo que significa una clara satisfacción de usuarios potenciales con la propuesta del modelo elaborado.

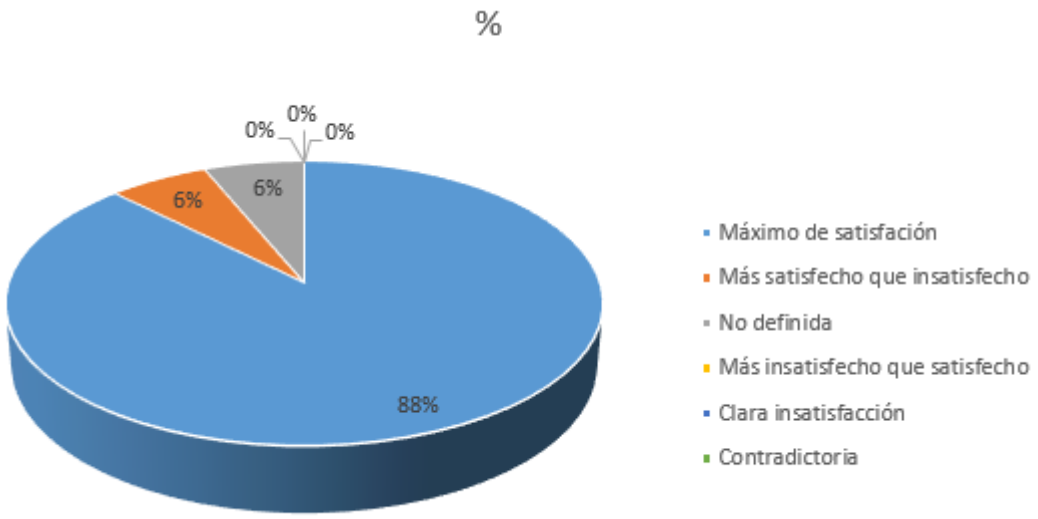

Figura 3.1: Nivel de satisfacción de usuarios potenciales. (Elaboración propia).

La técnica de Iadov contempla además dos preguntas complementarias de carácter abierto que son de mucha importancia. Estas permiten profundizar en las causas que originan los diferentes niveles de satisfacción. Las respuestas dadas plantearon sugerencias de utilidad para la presente y futuras investigaciones entre las que se destacan:

- Necesidad de preparar a especialistas para la aplicación del modelo
- Definir metodologías para la aplicación del modelo
- Añadir componentes para futuras situaciones que puedan ser incorporados, como la utilización de bases de datos no relacionales (NoSQL)

3.2 Entrevista a profundidad

La entrevista a profundidad es un método que permite, en un ambiente de confianza y confidencialidad, desarrollar una entrevista amplia. El entrevistador identifica información a profundidad en personas que por sus características cuentan con datos relevantes y que en otras condiciones no estarían dispuestos a compartir [Valles 2003]. Para la validación se realizó con el objetivo de identificar algunos aspectos importantes al utilizar el modelo elaborado para mejorar los procesos del tratamiento de los datos dentro de la UAN.

Se presentan las siguientes preguntas que fueron el guion de la entrevista:

- ¿Considera que la implementación de bases de datos distribuidas en la institución sea adaptable a las características de la misma?
- ¿Considera que la descentralización de la base de datos institucional traerá un procesamiento más económico y una mayor autonomía en el tratamiento de los datos?
- ¿Considera que al replicar los datos en más de una ubicación se logra una apreciable mejora en la disponibilidad de los mismos?
- ¿Considera que la posibilidad que brinda el modelo, a su vez puede elevar el grado de la disponibilidad y fiabilidad de los datos en la Universidad?
- ¿Considera factible que la propuesta pueda ser extendida en todas las instituciones de la UAN?
- Sobre la importancia de adaptar los sistemas TICs en la UAN para soportar los nuevos paradigmas de su uso

Los entrevistados fueron profesionales de Informática con conocimientos de bases de datos en general y analistas de datos:

- Rectoría de la universidad: por ser responsable por el procesamiento de todos los datos que constituyen el trabajo de la Universidad
- Facultades de la UAN: por ser las áreas constituidas por profesionales que brindan servicios de soporte a la gestión académica

Durante las entrevistas se dio respuesta a las preguntas antes citadas. Todas las preguntas fueron aplicadas después del autor haber realizado una presentación formal de la propuesta del modelo elaborado para la implementación de bases de datos distribuidas. Fueron realizadas de esta forma para comprobación de las variables fiabilidad y disponibilidad en el tratamiento de los datos. Los entrevistados concordaron con el modelo elaborado para la implementación de bases de datos distribuidas, ya que estas soportan un suave crecimiento para la Universidad, con un mínimo impacto en las unidades existentes en las instituciones. Otros criterios que los entrevistados dieron como positivo, el factor de tener un ambiente distribuido permite la reducción de la sobrecarga de tráfico en la red ya que los sitios locales, van a poseer los fragmentos más usados por sus aplicaciones locales teniendo mayor disponibilidad y fiabilidad de los datos.

3.3 Cuasi experimento para validar las variables fiabilidad y disponibilidad en el modelo propuesto

Basado en el modelo diseñado, se implementó la base de datos distribuida en un entorno controlado utilizando las herramientas expuestas en la Tabla 2.3 del epígrafe 2.3.3 del capítulo 2. También se hicieron pruebas de rendimiento utilizando la plataforma *pgbench*, y pruebas de carga de trabajo personalizado utilizando la plataforma *benchmark*.

Las variables que se midieron en el cuasi experimento fueron disponibilidad y fiabilidad. Y las mismas fueron medidas aplicándolas en dos ambientes. Para ello se utilizaron las siguientes formulas:

Disponibilidad= TMDF/ (TMDF+TMDR)*100% (1)

Fiabilidad= (TMDF+TMDR) (2)

En que:

$$TMDF = (\sum_{i=1}^{n} Tfi)/n \quad (3)$$

$$TMDR = (\sum_{i=1}^{n} Tri)/n \quad (4)$$

Dónde: TMDF es el tiempo promedio entre dos fallos consecutivos sufridos por el sistema. Y TMDR es el tiempo promedio necesario para reparar los fallos ocurridos en el sistema [García 2013], [García at al 2015].

Para el ambiente distribuido: para una prueba de 600min; se repartieron en dos intervalos en cada uno tuvo un tiempo promedio 18min de reparación del fallo.

Aplicando la fórmula 1. Disponibilidad = 270min/ (270min + 18min) x 100 % = 93,75%. Que significa un alto grado de disponibilidad.

Aplicando la fórmula 2. Fiabilidad = (270min+18min) = 288min. Que significa un alto grado de fiabilidad.

Para el ambiente centralizado: para una prueba de 600min en la primera reparación se tuvo hasta 60min, mientras en la segunda se tuvo una reparación de 18min. Lo que quedaría:

Disponibilidad = 261min/ (261min + 39min) = 87%. Lo que significa la disponibilidad aquí es menor en relación a la base de datos distribuida.

Fiabilidad = (261min+39min) = 300min. Lo mismo aconteció con la fiabilidad, demostrando que no sucedió lo planeado.

Se comprobó que, con la técnica de replicación y la fragmentación utilizada para la base de datos distribuida implementada (usando el modelo) se obtuvo mayor disponibilidad: porque no introducía cuellos de botella, permitió obtener de esta forma, un esquema más robusto entre los nodos y facilitó la distribución de cargas entre los diferentes nodos. Donde la configuración de los nodos permitió la comunicación en las réplicas e intercambio de actualizaciones.

Se hizo una replicación total de la BD en los nodos, para mejorar notablemente la disponibilidad de la información, dado que es altamente probable que algún nodo siempre permanezca activo. También se mejora el rendimiento de las consultas que se hacen sobre la BD, dado que cualquier nodo dispone de información necesaria para contestar a la requisitoria.

Se hizo una prueba de rendimiento con uso de la plataforma *pgbench*, y pruebas de *benchmark* con carga de trabajo personalizada se obtuvo una latencia de <2 segundos en las consultas donde implicaba 300 tuplas, y revelaron ventajas significativas al usar la plataforma BDR.

Ver ANEXO 8, sobre la comunicación entre los nodos del entorno distribuido.

3.4 Triangulación metodológica de los métodos aplicados

A partir de la aplicación de los métodos anteriores se procedió a aplicar una triangulación metodológica, donde se contrastan los resultados para analizar las coincidencias y divergencias y así minimizar el sesgo de la investigación.

[Kimchi et al 1991] proponen cuatro tipos básicos de triangulación, dentro de esta clasificación está la triangulación metodológica, que puede ser empleada en el

diseño o durante la recolección de datos. Existen dos tipos de triangulación metodológica, dentro de un método y entre métodos, la primera realiza múltiples interpretaciones de conjuntos similares, aplicando la misma técnica y la segunda se basa en la aplicación de diferentes métodos.

La hipótesis formulada en la investigación fue confirmada después de aplicada la triangulación metodológica, y se obtuvieron los siguientes resultados:

- ✓ El modelo para la implementación de bases de datos distribuidas elaborado, supera las principales deficiencias encontradas en el sistema actual del tratamiento de los datos y la información, y ofrece mayor capacidad para contribuir en el incremento de la disponibilidad y fiabilidad de los datos por parte de las instituciones
- ✓ Se ratifica la utilidad del modelo, así como el impacto al ser implementado en la Universidad para ayudar en la toma de decisiones

Conclusiones del capítulo

Con la aplicación de los métodos científicos con el objetivo de validar la propuesta, se pudo confirmar la pertinencia del modelo, y se reconoce además el valor científico del mismo. También contribuyó a demostrar la aplicabilidad del modelo y su contribución a la mejora en el incremento de la disponibilidad y fiabilidad en el tratamiento de los datos. La triangulación metodológica realizada permitió constatar los resultados que por separado arrojaran el criterio de expertos (con la técnica de Iadov), la entrevista a profundidad el cuasiexperimento respecto a la validez, pertinencia y viabilidad del modelo elaborado.

Conclusiones

La investigación realizada permite llegar a las siguientes conclusiones:

- A partir de la sistematización de los principales referentes teóricos y prácticos más actuales de las tecnologías de información en lo referido a las bases de datos distribuidas, se diseñó un modelo para la implementación de bases de datos distribuidas, validado con métodos científicos, que contribuya a un incremento de la fiabilidad y disponibilidad en el tratamiento de los datos institucionales en la Universidad Agostinho Neto.
- El modelo propuesto incluye principios, cualidades, componentes y relaciones entre los diferentes elementos que conforman el mismo, como el proceso de fragmentación, asignación y replicación de datos que son aspectos muy esenciales a que se tuvo en cuenta a la hora de la implementación de la base de datos distribuida en el entorno, lo que garantiza mayor fiabilidad y disponibilidad de los datos.
- El conjunto de métodos científicos utilizados para la validación de la propuesta (técnica de Iadov, entrevista a profundidad, cuasiexperimento y la triangulación) permite comprobar que:
    - La implementación de la base de datos utilizando el modelo propuesto garantiza un alto grado de disponibilidad y fiabilidad de los datos institucionales del entorno académico
    - Existe una alta satisfacción de los usuarios actuales y potenciales con respecto a necesidad, utilidad y actualidad del modelo propuesto

Recomendaciones

El autor recomienda:

- Generalizar el modelo para implementar servicios con herramientas analíticas para plataformas de Big Data y de bases de datos no estructurados (NoSQL);
- Aplicar en futuras instalaciones los sistemas de actualización de réplicas con esquema de propagación *Lazy*; esto para brindar mayor performance el en tamaño de las transacciones, limitando también a actualizar las copias primarias de los datos.

Referencias bibliográficas

1. Date, C. J. An introduction to database systems, 7ª edition. Addison Wesley, 2004. 960 p. ISBN: 968-444-419-2.
2. Elmasri, R. AND Navathe, Sh. B. Fundamentals of database systems. 6ª edition. Addison Wesley, 2004.
3. Connolly, Th. AND Begg C. Database Solutions: A step-by-step guide to building databases. 2ª edition. Addison Wesley, 2004. 1427 p. ISBN 0 321 21025 5.
4. Coronel, C., Morris, S. AND Rob, P. Database Systems: Design, implementation, and management. 9ª edition. Boston. Cengage learning, 2011. 724 p. ISBN-13: 978-0-538-74884-1.
5. Connolly, Th. AND Begg C. Database systems: A practical approach to design, implementation, and management. 4ª edition. Addison Wesley, 2005.
6. Paulraj P. Data modelling fundamentals. John Wiley, 2007. 461 p. ISBN-13: 978-0-471-79049-5, ISBN-10: 0-471-79049-4.
7. Elmasri, R. AND Navathe, Sh. B. Fundamentals of database systems. 6ª edition: Addison Wesley, 2008.
8. Coulouris, G., Dollimore, J., Kindberg, T. And Blair G. Distributed systems: concepts and desining. 5ª edition. Addison Wesley, 2012. 1067 p. ISBN 10: 0-13-214301-1, ISBN 13: 978-0-13-214301-1.
9. Ozsu, M. T. AND Valduriez P. Principles of distributed database systems. 3ª edition. Springer, 2011.866 p. ISBN 978-1-4419-8833-1.
10. Coulouris, G. Sistemas distribuidos: Conceitos e projecto. Bookman, 2000.
11. Bortolini,C. A. Um prototipo de banco de datos distribuidos. Chapecó, 2008.
12. Brito, M. S. Arquitectura de um sistema para integración de bancos de dados com suporte a replicacão utilizando tecnologías de grades computacionais. Sao Paulo, 2009.
13. Ghosh, Sukumar. Distributed systems: An algorithmic approach. 2ª edition: CRC Press, 2015. 526 p. ISBN 978-1-4665-5297-5.
14. Belloch, C. Las Tecnologías de la Información y Comunicación en el aprendizaje. (2012) .
15. Elmasri, R. AND Navathe, S. B. Fundamentals of database systems; Diaz José M. (trad). Madrid: 5th Ed. Redwood City. Addison-Wesley, 2007. 1012 p. ISBN 978-84-7829-085-7.

16. Garcia-Molina, H.; Ullman, J.; Widom, J. Database systems: The complete book. 2nd Ed. London: Prentice Hall. 2008. 570 p.
17. Heuser, C. A. Projeto de banco de dados. 6ª Ed. Porto Alegre: Bookman, 2009.
18. Guimarães, C.C. Fundamentos de bancos de dados: modelagem, projeto e linguagem SQL. Campinas. Editora da Unicamp. 2003.
19. Korth, H. F.; Silberschatz, M.; Sudarshan, S. Database systems concepts**.** NewYork. McGraw Hill, 2006.
20. Ramakrishnan, R. AND Gehrke, J. Database management systems. 3rd Ed. New York. McGraw-Hill, 2003.
21. Bertino, E. AND Zarri, G.P.; Catania, B. Intelligent database systems. Redwood City. Addison-Wesley, 2001.
22. Tanenbaum, A. S. AND Steen, M. S. Sistemas distribuídos. 2ª Ed. São Paulo. Prentice-Hall, 2007.
23. Carballal, F. M. Bases de datos distribuídas. 2007.
24. Huacuja, F. AND Joaquín, H. Automatización del diseño de la fragmentación horizontal en bases de datos distribuidas.
25. Reingart, M. PyReplica, sistema de replicación simple para PostgreSQL programado en Python. 2012.
26. García, J. L. Réplica bidireccional basada en control de cambios. La Habana : s.n.
27. Buretta, M. Data replication: tools and techniques for managing distributed information. s.l.: Wiley, 2009.
28. Ricardo C. M. Base de datos; Campos Olguín, Víctor; Enríquez Brito, Javier (trad). Mexico. McGraw-Hill. 2009. 666 p. ISBN 0-7637-3314-8.
29. Mannino, M. V. Administración de bases de datos: Diseño y desarrollo de aplicaciones; Guerrero, Ekatenia; Diaz, José (trad). 3ª Ed. Mexico. McGraw-Hill. 2007. 739 p. ISBN 978-970-10-6109-1.
30. Ramos, M. AND J. Ramos, A. Montero, F. Sistemas gestores de bases de datos. Madrid. McGraw-Hill. 2006. 460 p. ISBN 84-481-4879-7.
31. Mistry, R. AND Misner, S. Introducing Microsoft SQL server 2008 R2. Redmond. Microsoft press. 2010. 236 p.
32. Darwen, H. SQL: A comparative survey. 2ª Ed. Bookboon. 2014. 169 p. ISBN 978-87-403-0778-8.
33. Date, C. J. AND Darwen H. A guide to the SQL standard. 4ª Ed. Adisson Wesley. 20003.

34. Date, C. J. SQL and relational theory: How to write accurate SQL code. 2ª Ed. O´Reilly. 2012.
35. Abadi, D. J. Data management in the cloud: Limitations and opportunities. Q. Bull. IEEE TC on Data Eng. (2009). 32(1):3–12. p. 746, 747, 762.
36. Abadi, D. J., Madden, S., AND Hachem, N. Column-stores vs. row-tores:how different are they really? In Proc. ACM SIGMOD Int. Conf. on management of data. (2008). p. 967–980. 753.
37. Aggarwal, C., Han, J., Wang, J., AND Yu, P. S. A framework for clustering evolving data streams. In Proc. 29th Int. Conf. on Very Large Data Bases. (2003). p. 81–92. 743.
38. Aggarwal, C. C. Data Streams: Models and algorithms. Springer. (2007). p 762.
39. Antonioletti, M. et al. The design and implementation of grid database services in OGSA-DAI. Concurrency — Practice & Experience, 17(2-4):357–376. (2005). p 750.
40. O'Neil, P. AND E. O'Neil. Database principles, programming, performance. 2a. ed.). Morgan Kaufmann, 2003.
41. Amjad, U. Distributed database management systems: issues and approaches, University of Michigan.
42. Guillermo, Á. C. Integración de esquemas en bases de datos heterogéneas fuertemente acopladas.
43. Antoniazzi, F. L. Bases de datos distribuidas multiplataforma. Corrientes-Argentina, 2004.
44. Lightstone, S.; Teory T.; Nadeau T. Physical database design. Morgan Kaufmann, 2007.
45. Vargas, J., Mendoza, M. et al. Fundamentos teóricos de bases de datos distribuidas. 2015.
46. Emil Sit. Storing and Managing Data in a Distributed Hash Table. PhD thesis, Massachusetts Institute of Technology. June 2008.
47. Bonifati, A., Chrysanthis, P. K., Aris M. Ouksel, AND Kai-Uwe Sattler. Distributed databases and peer-to-peer databases: past and present. SIGMOD Rec., 37(1): p 5–11. March 2008.
48. Neves, S. F. Banco de dados distribuídos: Um estudo de caso em um banco de dados homogêneo distribuído visando a alta disponibilidade de dados. Brasilia 2012.

49. Mendoza, A. Z. Replicación y fragmentación de bases de datos distribuidas. Veracruzana. 2010.
50. Suarez, D., Palacio, Y. S. Propuesta de procedimiento para la prestación del servicio de replicación de datos del centro de tecnologías y gestión de datos. Universidad de las Ciencias Informáticas, La Habana, 2010.
51. Cobas, R. S. Propuesta de solución de réplica multi-maestra asíncrona para bases de datos. Universidad de las Ciencias Informáticas, La Habana, 2012.
52. Abadi, D. Consistency Tradeoffs in Modern Distributed database system design. Yale University. IEEE. 2012.
53. M. Stonebraker, “Errors in database systems, eventual consistency, and the CAP Theorem,” blog, Comm. ACM, 5 Apr. 2010; http://cacm.acm.org/blogs/blog-cacm/83396-errors-in-database-systems-eventualconsistency-and-the-cap-theorem. [Consulta: 22 de Mayo de 2015].
54. Lakshman and P. Malik. “Cassandra: structured storage system on a P2P Network,” Proc. 28th ACM Symp. Principles of distributed computing (PODC 09), ACM, 2009, article no. 5; doi:10.1145/1582716.1582722.
55. Reingart, M. PyReplica. Sistema de replicación simple para PostgreSQL programado en Python. 2008.
56. Long, E. AND Henson, Ch. Chris. 2007-2010. SymmetricDS. [En línea] 2007-2010. [Citado el: 6 de diciembre de 2014.] http://symmetricds.codehaus.org/.
57. García, J. L. Réplica bidireccional basada en control de cambios "Magic@Data Replication eXtensible solution". Ciudad Habana: s.n., 2010.
58. Group, C. Slony-I. Enterprise-level replication system. [En línea] 2007-2009. [Citado el: 25 de enero de 2015.] http://www.slony.info/documentation.
59. Freddy A. Poll García, “Métricas y pruebas de validación para sistemas de almacenamiento,” ISPJAE, La Habana, Cuba, 2013.
60. García, F., González, A., Perellada, L., Hernández, A. Métricas y pruebas de validación para sistemas de almacenamiento. Revista Telem@tica. Vol. 14. No. 1, 2015, p. 24-38. ISSN 1729-3804. La Habana: ISPJAE. 2015.
61. Carranza Athó, F. Bases de datos distribuidas. Perú: Escuela Académico Profesional de Informática, 2006.
62. Díaz, O. F. Modelo para el desarrollo de aplicaciones compuestas basadas en arquitecturas orientadas a servicios. La Habana. 2012.

63. Martínez L. Diseño y Construcción de bases de datos distribuidas heterogéneas sobre Oracle Y SQL Server. Universidad Carlos III de Madrid Escuela Politécnica Superior Ingeniería Técnica en Informática. 2011.
64. Paderni, M del C. Y otros. Bases de datos distribuidas para aplicaciones médicas en el Sistema Nacional de Salud. Revista Cubana de Informática Médica versión ISSN 1684-1859. vol.6 no.2 Ciudad de la Habana. 2014.
65. El Modelo Cliente-Servidor. Herramientas _ Web para la enseñanza de protocolos de comunicación. (Consultado: 7 de febrero de 2014). http://neo.lcc.uma.es/evirtual/cdd/tutorial/aplicacion/cliente-servidor.html
Gutiérrez, S., García, D., Infante, E. Registro de Trabajadores de la Salud. VIII Congreso Internacional de Informática en la Salud, 2011.
66. Martos-Rodríguez, P. Ejecución de una base de datos distribuida sobre un entorno de Cloud Computing. Tesis de master en ingeniería de computadores. Universidad Complutense de Madrid. http://eprints.ucm.es/9889/1/Memoria.pdf. 2009

Anexos

Anexo 1: Modelo entidad-relación de la base de datos de la gestión académica de la UAN.

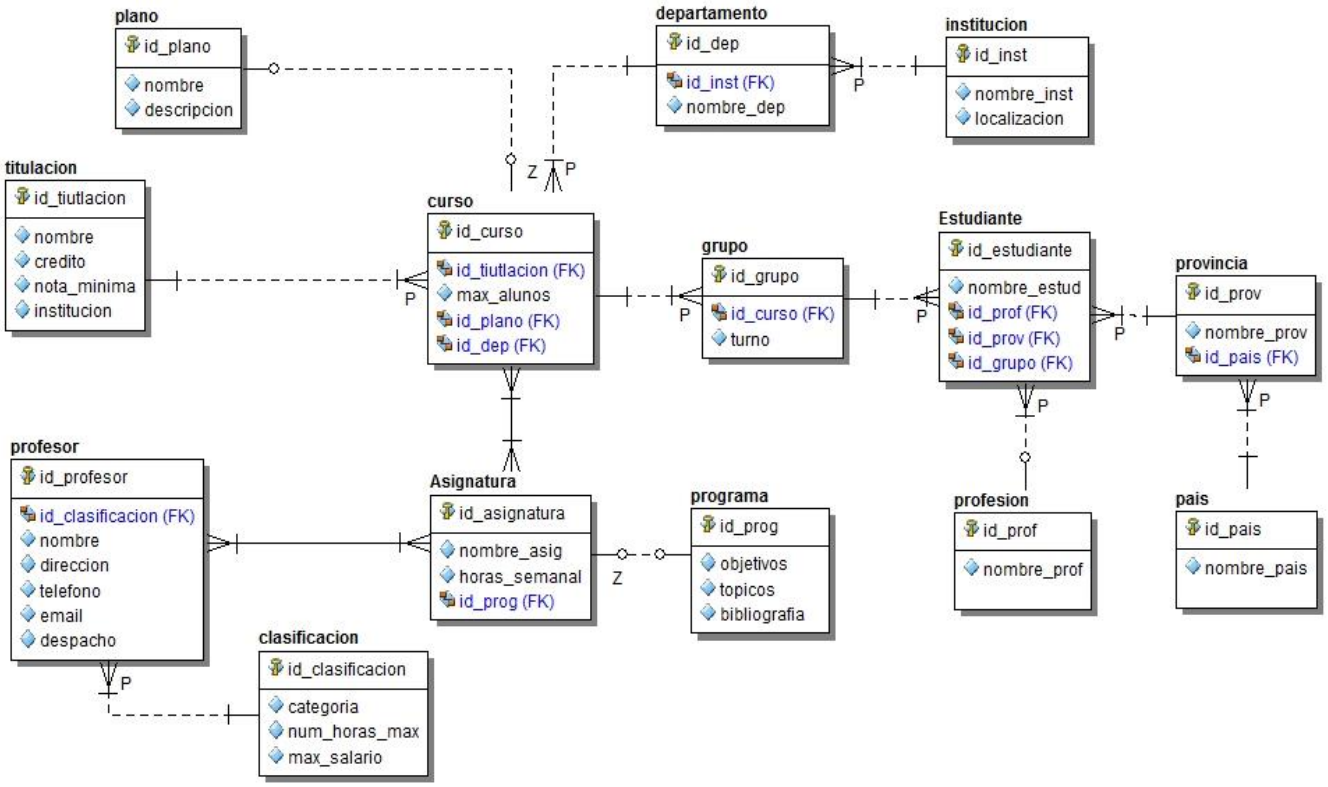

Figura 1: Modelo entidad relación de la base de datos.

Anexo 2: Modelo relacional de la base de datos de la gestión académica de la UAN.

asignatura (id_asignatura, nombre_asig, horas_semanal, id_prog)

impartir (id_asignatura, id_profesor, num_horas)

estudiante (id_estudiante, nombre_estud, id_prof, id_prov, id_grupo)

clasificacion (id_clasificacion, categoría, num_horas_max, max_salario)

curso (id_curso, max_alunos, id_plano, id_titulacion, id_dep)

pertencer (id_curso, id_asignatura, num_asig)

departamento (id_dep, id_inst, nombre_dep)

grupo (id_grupo, id_curso, turno)

institucion (id_inst, nombre_inst, localizacion)

país (id_pais, nombre_pais)

plano (id_plano, nombre, descripcion)

profesion (id_profesion, nombre_prof)
profesor (id_profesor, nombre, direccion, telefono, email, despacho, id_clasificacion)
programa (id_prog, objetivos, tópicos, bibliografia)
provincia (id_prov, nombre_prov, id_pais)
titulacion (id_titulacion, nombre, nota_minima, institucion)

Anexo 3: **Diseño de vistas seleccionadas a partir de algunas relaciones.**

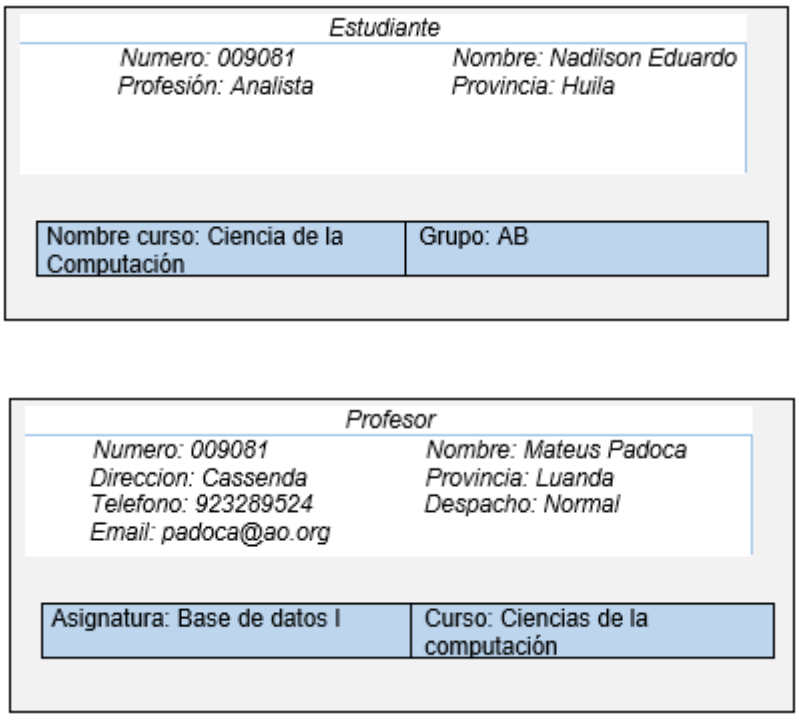

Figura 2: Vistas de las relaciones Estudiante-Curso y Profesor-Asignatura.

Anexo 4: **Fragmentación de algunas relaciones de la base de datos.**

Para conocer los profesores que dan clases en las distintas instituciones realizamos otra fragmentación derivada de la relación que almacena la impartición docente de los profesores con respecto a los fragmentos de las asignaturas correspondientes a cada institución.

IMPARTIR_i: IMPARTIR $\ltimes_{id\_asignatura}$ ASIGNATURAS_i
Donde i: {Nodo1, Nodo2}

Para mantener en Nodo1 los datos de gestión de nóminas y contratación del profesorado, primero se realiza una fragmentación vertical para ubicar en dicho nodo solo la información que interesa.

INFO_PROFESORES: $\Pi_{id\_profesor, nombre, email, despacho}$ (PROFESORES)
NOMINAS_PROFESORES: $\Pi_{id\_profesor, direccion, telefono, categoria}$ (PROFESORES)

Para luego realizar una fragmentación horizontal derivada y de esta forma obtener fragmentos con los atributos de los profesores necesarios en cada una de las instituciones.

INFO_PROFESORES_i= $id\_ \ltimes_{asignatura}$ IMPARTIR_i
Donde i= {Nodo1, Nodo2}

ANEXO 5. Encuesta aplicada a especialistas y a funcionarios de las instituciones que constituyen la UAN y a profesionales de la rectoría para valorar la adopción del modelo para la implementación de bases datos distribuidas.

Para conocer la valoración que tienen los especialistas, profesionales y funcionarios de la UAN, con respecto a la utilización del modelo para la implementación de bases de datos distribuidas en la UAN, se aplicó la encuesta que se destaca a continuación:

1. ¿En la institución, es muy importante la gestión de la información para su socialización y su correspondencia con los objetivos estratégicos?

Sí___ En cierta medida____ No____

2. ¿Deberías considerar útil la socialización de la información y el conocimiento para apoyar la toma de decisiones de la organización?

Sí____ No_____

ANEXO 6. Encuesta aplicada para evaluar la satisfacción de usuarios potenciales del modelo para la implementación de bases de datos distribuidas elaborado.

Tabla 1: Cuadro lógico de ladov.

EVALUACION DEL MODELO PARA LA IMPLEMENTACION DE BASE DE DATOS DISTRIBUIDAS ELABORADO		
No	**Preguntas**	**Respuestas**
1	¿Considera usted que se deba implementar bases de datos distribuidas sin contar con una herramienta que oriente el proceso?	Si__No__No sé__
2	¿Considera adecuados los componentes del modelo propuesto?	Si__No__No sé__
3	¿Si usted necesitara implementar bases de datos distribuidas usaría el modelo propuesto?	Si__No__No sé__
4	¿Considera que las técnicas de fragmentación y replicación son adecuadas para el modelo propuesto?	Si__No__No sé__
5	¿Le satisface la representación de este modelo para implementar bases de datos distribuidas?	Si__No__No sé__
6	¿Le satisface la forma en que fueron integrados y representado en el modelo los principios que los sustentan, para contribuir en la implementación de base de datos distribuidas?	Me gusta Mucho _____ No me gusta tanto_____ Me da lo mismo _____ Me disgusta más de lo que me gusta _____ No me gusta nada_____ No SÉ qué decir _____
7	¿Tiene algunas sugerencias para el desarrollo e implantación de este modelo?	Entre 0 y 10 _____
8	¿Sugiere otros componentes que serían necesarios para incorporar al modelo?	Entre 0 y 10 _____

CUESTIONARIO PARA LA VALIDACIÓN DEL MODELO

1. ¿Considera usted que se deba implementar bases de datos distribuidas sin contar con una herramienta que oriente el proceso? No_____ Sí_____ No sé_____
2. ¿Considera adecuados los componentes del modelo propuesto? No_____ Sí_____ No sé_____

3. ¿Si usted necesitara implementar bases de datos distribuidas usaría el modelo propuesto? No_____ Sí_____ No sé_____
4. ¿Considera que las técnicas de fragmentación y replicación son adecuadas para el modelo propuesto? No_____ Sí_____ No sé_____
5. ¿Le satisface la representación de este modelo para implementar bases de datos distribuidas? : marque con una cruz la que considere conveniente.

    Me gusta mucho _____
    No me gusta tanto_____
    Me da lo mismo_____
    Me disgusta más de lo que me gusta_____
    No me gusta nada_____
    No sé qué decir_____

6. ¿Sugiere otros componentes que serían necesarios para incorporar al modelo?
7. ¿Tiene algunas sugerencias para el desarrollo e implantación de este modelo?

La fórmula con los valores calculados:

$$ISG = \mathbf{A(+1) + B(+0.5) + C(0) + D(-0.5) + E(-1)}/N$$

ISG = 14(+1) + 1(+0.5) + 1(0) + 0(-0.5) + 0(-1)/16 = 0,90

Como se aprecia, el índice de satisfacción grupal es 0,93 lo que significa una clara satisfacción con la propuesta y reconocimiento de su utilidad para lograr un incremento de la fiabilidad y disponibilidad de los datos institucionales.

Anexo 7: Resultado del grado de satisfación de los usuarios potenciales.

Tabla 2: Resultado de la técnica de Iadov.

Resultado	***Cantidad***	***%***
Máximo de satisfación	14	87,50
Más satisfecho que insatisfecho	1	6,25
No definida	1	6,25
Más insatisfecho que satisfecho	0	-
Clara insatisfacción	0	-
Contradictoria	0	-

Anexo 8: Replicación de las tablas en los nodos establecidos.

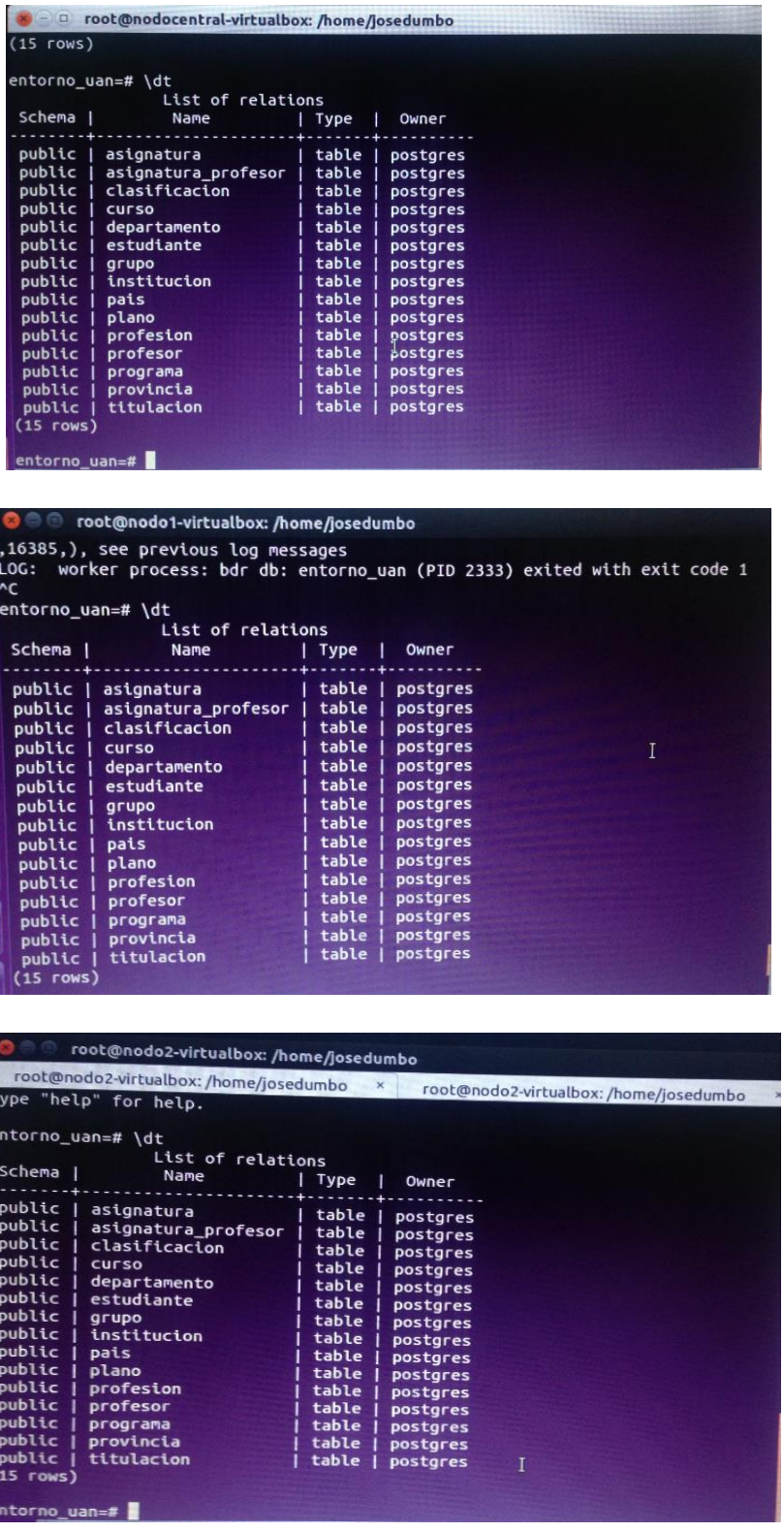

Figura 3: Replicación de las tablas en el entorno distribuido.

Selección de una relación a partir de los 3 nodos

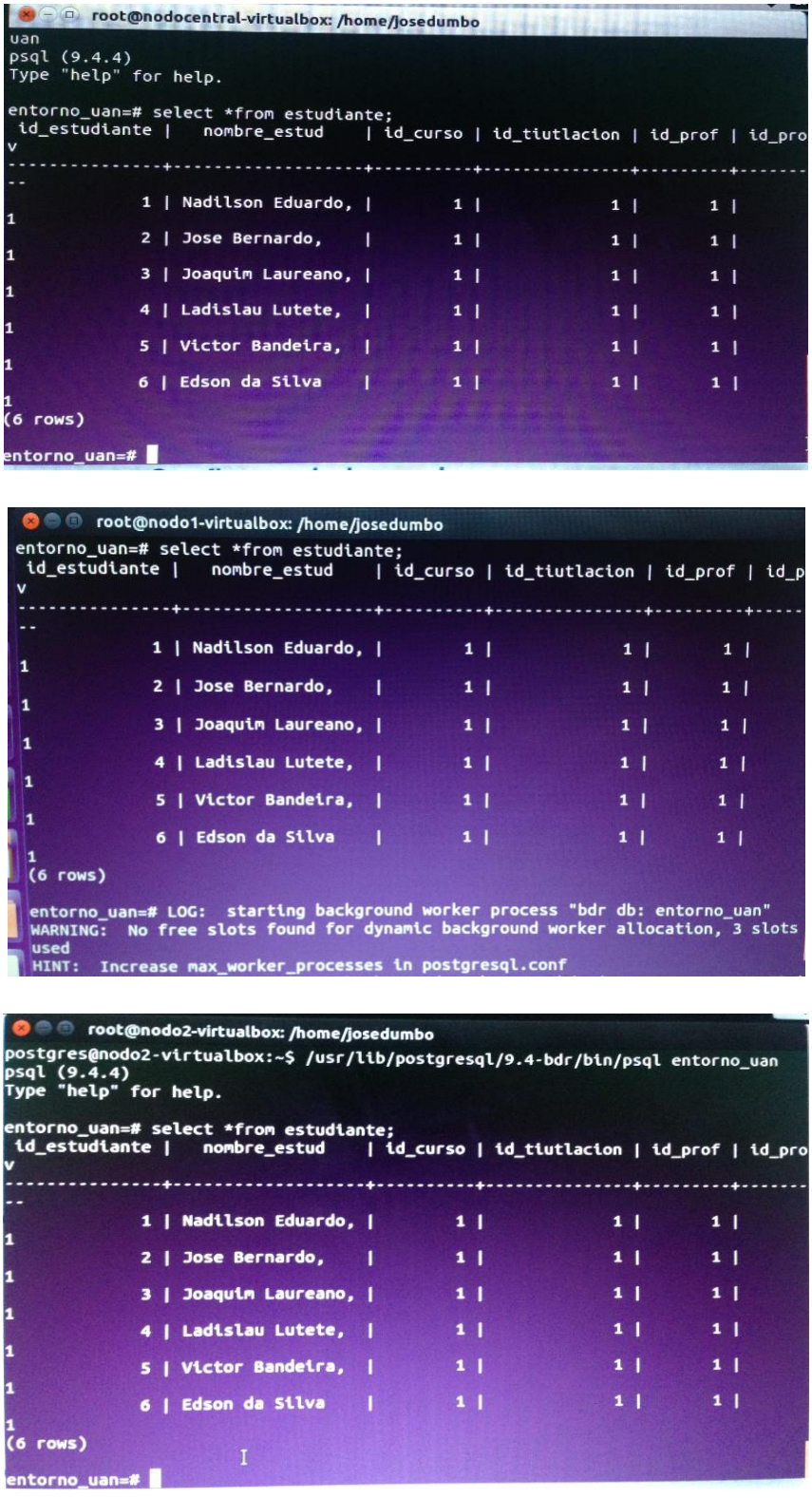

Figura 4: Selección de una relación en cada nodo del entorno distribuido.

Inserción de datos a partir del Nodo Central, y su respectiva consulta en el Nodo2

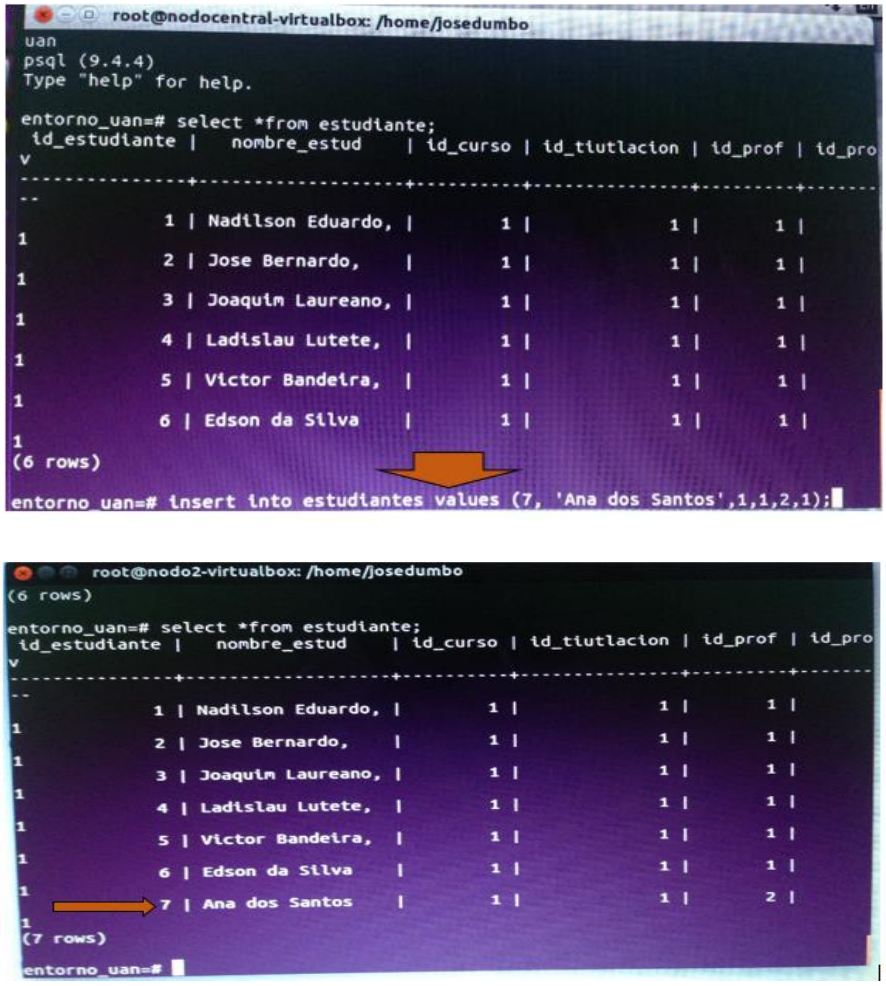

Figura 5: Consulta de una relación, insertada a partir del otro nodo.