

UNIVERSIDAD TÉCNICA DE AMBATO



FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

CENTRO DE ESTUDIOS DE POSGRADO

MAESTRÍA EN GESTIÓN DE BASES DE DATOS

Tema:

LAS BASES DE DATOS DISTRIBUIDAS Y SU IMPORTANCIA EN LA RECUPERACIÓN, ACTUALIZACIÓN Y ADMINISTRACIÓN DE LA INFORMACIÓN DEL CENTRO EDUCATIVO DIOCESANO SAN PÍO X.

Trabajo de Investigación

Previa la obtención de Grado Académico de Magister en Gestión de Bases de Datos.

Autor: Ing. Alfredo Mauricio Villacís Lizano

Director: Ing Mg. Jaime Bolivar Ruiz Banda

Ambato – Ecuador

2012.

Al Consejo de Posgrado de la UTA.

El tribunal receptor de la defensa del trabajo de investigación con el tema: LAS BASES DE DATOS DISTRIBUIDAS Y SU IMPORTANCIA EN LA RECUPERACIÓN, ACTUALIZACIÓN Y ADMINISTRACIÓN DE LA INFORMACIÓN DEL CENTRO EDUCATIVO DIOCESANO SAN PÍO X., presentado por Ing. Alfredo Mauricio Villacís Lizano y conformado por: Ing. Mg. Clay Fernando Aldas Flores, Ing. Mg. Galo Mauricio López Sevilla, Ing. Mg. Franklin Oswaldo Mayorga Mayorga. Miembros del Tribunal, Ing. Mg. Jaime Bolívar Ruiz Banda, Director del trabajo de investigación y presidido por: Ing. M.Sc. Oswaldo Paredes Ochoa, Presidente del Tribunal; Ing. Mg. Juan Garcés Chávez Director del CEPOS – UTA, una vez escuchada la defensa oral el Tribunal aprueba y remite el trabajo de investigación para uso y custodia en las bibliotecas de la UTA.

Ing. Msc. Oswaldo Paredes Ochoa
Presidente del Tribunal de Defensa

Ing. Mg, Juan Garcés Cháves
Director CEPOS

Ing. Mg. Jaime Bolívar Ruiz Banda
Director del Trabajo de Investigación

Ing. Mg. Clay Fernando Aldas Flores
Miembro del Tribunal

Ing. Mg. Galo Mauricio López Sevilla

Miembro del Tribunal

Ing. Mg. Franklin Oswaldo Mayorga Mayorga
Miembro del Tribunal

AUTORÍA DE LA INVESTIGACIÓN

La responsabilidad de las opiniones, comentarios y críticas emitidas en el trabajo de investigación con el tema: LAS BASES DE DATOS DISTRIBUIDAS Y SU IMPORTANCIA EN LA RECUPERACIÓN, ACTUALIZACIÓN Y ADMINISTRACIÓN DE LA INFORMACIÓN DEL CENTRO EDUCATIVO DIOCESANO SAN PÍO X. Nos corresponde exclusivamente a: Ing. Alfredo Mauricio Villacís Lizano, Autor y de Ing. Mg. Jaime Bolívar ruiz Banda, Director del trabajo de investigación; y el patrimonio intelectual del mismo a la Universidad técnica de

Ambato.

Ing. Alfredo Mauricio Villacís Lizano.
Autor
DERECHOS DE AUTOR

Ing. Mg Jaime Bolívar Ruiz Banda
Director

Autorizo a la Universidad Técnica de Ambato, para que haga de este trabajo de investigación o parte de él un documento disponible para su lectura, consulta y procesos de investigación, según las normas de la Institución.

Cedo los Derechos de mi trabajo de investigación, con fines de difusión pública, además apruebo la reproducción de ésta, dentro de las regulaciones de la Universidad.

Ing. Alfredo Mauricio Villacís Lizano

DEDICATORIA

A mis Padres, hermanas y sobrino
por haberme brindado su apoyo y respaldo incondicional.

Ing. Alfredo Mauricio Villacís Lizano.

AGRADECIMIENTO

Al **ETERNO CREADOR** del universo por permitirme un día mas de vida y poder contemplar sus maravillas.

A las personas que brindaron su conocimiento y sabiduría para el desarrollo de esta investigación.

Para el Ing. Mg. Jaime Bolívar Ruiz Banda mi profundo agradecimiento, ya que supo guiarme en el desarrollo de este tema de investigación.

Mi gratitud a la FISEI por impulsar esta Maestría, y el honor de formar parte de ella.

A todas las personas que de una u otra manera me han ayudado a cumplir esta meta

EL AUTOR

ÍNDICE GENERAL

PORTADA.....	i
APROBACIÓN DEL JURADO EXAMINADOR.....	ii
AUTORÍA DE LA INVESTIGACIÓN.....	iii
DERECHOS DE AUTOR.....	iv
DEDICATORIA	v
AGRADECIMIENTO	vi
ÍNDICE GENERAL.....	vii
ÍNDICE DE GRÁFICOS.....	x
ÍNDICE DE CUADROS.....	xii
ÍNDICE DE FÓRMULAS.....	xiii
ÍNDICE DE ANEXOS.....	xiv
RESÚMEN.....	xv
INTRODUCCIÓN.....	1
CAPÍTULO I.....	3
EL PROBLEMA DE INVESTIGACIÓN.....	3
PLANTEAMIENTO DEL PROBLEMA.....	3
CONTEXTUALIZACIÓN.....	3
Macro Contextualización.....	3
Meso Contextualización.....	4
Micro Contextualización.....	4
ÁRBOL DE PROBLEMAS.....	5
ANÁLISIS CRÍTICO.....	6
PROGNOSIS.....	6
FORMULACIÓN DEL PROBLEMA.....	7
INTERROGANTES DE LA INVESTIGACIÓN.....	7
DELIMITACIÓN DEL OBJETO DE LA INVESTIGACIÓN.....	7
JUSTIFICACIÓN.....	8
OBJETIVOS.....	8
Objetivo General.....	8
Objetivos Específicos:.....	9

CAPÍTULO II.....	10
MARCO TEÓRICO.....	10
ANTECEDENTES INVESTIGATIVOS.....	10
FUNDAMENTACIONES.....	11
Filosófica.....	11
Tecnológica.....	11
Administrativa.....	11
Legal.....	11
CATEGORÍAS FUNDAMENTALES.....	14
CATEGORÍAS DE LA VARIABLE INDEPENDIENTE.....	15
CATEGORÍAS DE LA VARIABLE DEPENDIENTE.....	39
HIPÓTESIS DE LA PREGUNTA DIRECTRIZ.....	48
SEÑALAMIENTO DE LAS VARIABLES DE LA HIÓTESIS.....	48
CAPÍTULO III.....	49
METODOLOGÍA	50
ENFOQUE.....	50
MODALIDAD BÁSICA DE LA	50
NIVEL O TIPO DE	51
POBLACIÓN Y	52
OPERACIONALIZACIÓN DE LAS	52
Variable	52
Variable	53
PLAN DE RECOLECCIÓN DE	55
Técnicas e	55
PLAN DE PROCESAMIENTO DE LA	55
CAPÍTULO	56
ANÁLISIS E INTERPRETACIÓN DE	56
VERIFICACIÓN DE LA	67
CAPÍTULO	69
CONCLUSIONES Y	71
Conclusiones	71
Recomendaciones	71
CAPÍTULO	73

LA	73
DATOS	73
ANTECEDENTES DE LA	73
ESTADO DEL	74
JUSTIFICACIÓN.....	74
OBJETIVOS.....	75
Objetivo	75
Objetivos	75
ANÁLISIS DE	73
FUNDAMENTACIONES.....	76
Metodología	76
Modelo	78
Plan de	79
Administración	87
Previsión de la	92
MATERIALES DE REFERENCIA	113
Referencias	113
Bibliografía.....	114
Lincografía.....	115
ANEXOS	116

ÍNDICE DE GRÁFICOS

Gráfico N° 1.1	Relación Causa – Efecto.....	5
Gráfico N° 2.1	Categorías Fundamentales.....	14
Gráfico N° 2.2	Entorno Simplificado de un Modelo de BD.....	19
Gráfico N° 2.3	Ejemplo de tipo de registro.....	28
Gráfico N° 2.4	Modelo de Red.....	29
Gráfico N° 2.5	Modelo jerárquico mediante un modelo de red.....	30
Gráfico N° 2.6	Diagrama de relaciones entre tipos de registros.....	30
Gráfico N° 2.7	Modelo E – R.....	32
Gráfico N° 2.8	Relación cuenta.....	33
Gráfico N° 2.9	Relación cuenta desordenada.....	35
Gráfico N° 2.10	Sistema Distribuido.....	37
Gráfico N° 2.11	Subcategorías de la VI.....	47
Gráfico N° 2.12	Subcategorías de la VD.....	48
Gráfico N° 4.1	Análisis de Resultados Pregunta N° 1.....	56
Gráfico N° 4.2	Análisis de Resultados Pregunta N° 2.....	57
Gráfico N° 4.3	Análisis de Resultados Pregunta N° 3.....	58
Gráfico N° 4.4	Análisis de Resultados Pregunta N° 4.....	59
Gráfico N° 4.5	Análisis de Resultados Pregunta N° 5.....	60
Gráfico N° 4.6	Análisis de Resultados Pregunta N° 6.....	61
Gráfico N° 4.7	Análisis de Resultados Pregunta N° 7.....	62
Gráfico N° 4.8	Análisis de Resultados Pregunta N° 8.....	63
Gráfico N° 4.9	Análisis de Resultados Pregunta N° 9.....	64
Gráfico N° 4.10	Análisis de Resultados Pregunta N° 10.....	65
Gráfico N° 4.11	Análisis de Resultados Pregunta N° 11.....	66
Gráfico N° 4.12	Curva de t de student – Comprobación de Hipótesis.....	69
Gráfico N° 5.1	Esquema de BD distribuido para el CED San Pio X.....	72
Gráfico N° 6.1	Entorno Distribuido.....	78
Gráfico N° 6.2	Comparación Estadística de Rendimiento.....	80
Gráfico N° 6.3	Esquema de trabajo – AdaReplicador.....	84
Gráfico N° 6.4	Esquema de trabajo – FBReplicador.....	85
Gráfico N° 6.5	Esquema de Configuración archivo .arpl.....	88

Gráfico N° 6.6	Pre – requisitos instalación AdaReplicador	89
Gráfico N° 6.7	Pre – requisitos instalación AdaReplicador	89
Gráfico N° 6.8	Pre – requisitos instalación AdaReplicador	90
Gráfico N° 6.9	Intalación AdaReplicador.....	91
Gráfico N° 6.10	Intalación AdaReplicador.....	92
Gráfico N° 6.11	Configuración AdaReplicador.....	92
Gráfico N° 6.12	Configuración AdaReplicador.....	93
Gráfico N° 6.13	Configuración AdaReplicador.....	93
Gráfico N° 6.14	Configuración AdaReplicador.....	94
Gráfico N° 6.15	Configuración AdaReplicador.....	94
Gráfico N° 6.16	Configuración AdaReplicador.....	95
Gráfico N° 6.17	Configuración AdaReplicador.....	95
Gráfico N° 6.18	Configuración AdaReplicador.....	96
Gráfico N° 6.19	Configuración AdaReplicador.....	96
Gráfico N° 6.20	Configuración AdaReplicador.....	97
Gráfico N° 6.21	Configuración AdaReplicador.....	97
Gráfico N° 6.22	Configuración AdaReplicador.....	98
Gráfico N° 6.23	Configuración Cron – Backup.....	101
Gráfico N° 6.24	Configuración Cron – Backup.....	102
Gráfico N° 6.25	Configuración Cron – Backup.....	103
Gráfico N° 6.26	Configuración Cron – Backup.....	104
Gráfico N° 6.27	Configuración Cron – Backup.....	105
Gráfico N° 6.28	Configuración Cron – Backup.....	106

ÍNDICE DE CUADROS

Cuadro N° 1	Población y muestra.....	52
Cuadro N° 2	Operacionalización de la Variable Independiente.....	53
Cuadro N° 3	Operacionalización de la Variable Dependiente.....	54
Cuadro N° 4	Análisis de Resultados Pregunta N° 1.....	56
Cuadro N° 5	Análisis de Resultados Pregunta N° 2.....	57
Cuadro N° 6	Análisis de Resultados Pregunta N° 3.....	58
Cuadro N° 7	Análisis de Resultados Pregunta N° 4.....	59
Cuadro N° 8	Análisis de Resultados Pregunta N° 5.....	60
Cuadro N° 9	Análisis de Resultados Pregunta N° 6.....	61
Cuadro N° 10	Análisis de Resultados Pregunta N° 7.....	62
Cuadro N° 11	Análisis de Resultados Pregunta N° 8.....	63
Cuadro N° 12	Análisis de Resultados Pregunta N° 9.....	64
Cuadro N° 13	Análisis de Resultados Pregunta N° 10.....	65
Cuadro N° 14	Análisis de Resultados Pregunta N° 11.....	66
Cuadro N° 15	Distribución estadística.....	68
Cuadro N° 16	Tipos de replicación.....	79
Cuadro N° 17	Comparativa de Rendimiento.....	80
Cuadro N° 18	Tipos de Fragmentación en Replicadores.....	81
Cuadro N° 19	Datos básicos configuración BD shadow.....	106

ÍNDICE DE FÓRMULAS

Fórmula N° 1	Media Aritmetica.....	69
Fórmula N° 2	Desviación Estándar.....	69
Fórmula N° 3	Valor de t de student.....	69

ANEXOS

Anexo 1	Instrumentos para la encuesta.....	116
Anexo 2	Formato de cuaderno de notas.....	117
Anexo 3	Manual de Procedimientos.....	118
Anexo 4	Formato ficha bibliográfica.....	120
Anexo 5	Tabla t de student.....	121

UNIVERSIDAD TÉCNICA DE AMBATO

CENTRO DE ESTUDIOS DE POSGRADO

MAESTRÍA EN GESTIÓN Y ADMINISTRACIÓN DE BASE DE DATOS

LAS BASES DE DATOS DISTRIBUIDAS Y SU IMPORTANCIA EN LA RECUPERACIÓN, ACTUALIZACIÓN Y ADMINISTRACIÓN DE LA INFORMACIÓN DEL CENTRO EDUCATIVO DIOCESANO SAN PÍO X.

Autor: Ing. Alfredo Mauricio Villacís Lizano.

Director: Ing. Mg. Jaime Bolívar Ruiz Banda.

RESUMEN

La investigación sobre “**Bases de Datos Distribuidas y su importancia en la recuperación, actualización y administración de la información del Centro Educativo San Pío X**”, tiene como objetivo general analizar e implementar una solución de replicación y respaldo de información para el manejo y recuperación de información en la institución Centro Educativo Diocesano (CED) San Pío X en el periodo 2011. El problema en la inexistente política de control de respaldos de información se da debido a que los equipos no cuentan con un sistema automático o manual que realice este proceso, así como respaldos de fuentes de energía alternos que proporcione su servicio en momentos de corte ocasional o planificado por parte de la empresa eléctrica de Ambato, esto hace que el sistema académico transaccional se vuelva inestable y poco confiable al momento de realizar consultas provocando pérdida de tiempo y dinero. Con esta investigación se plantea una solución a estos problemas dando respuestas y prioridades para el manejo de respaldos de seguridad, y proveer información segura y confiable a cada uno de los usuarios (clientes) de la institución.

INTRODUCCIÓN

El trabajo de investigación tiene como tema: Las Bases de Datos Distribuidas y su importancia en la recuperación, actualización y administración de la información en el Centro Educativo Diocesano San Pío X. Su importancia radica en la necesidad de contar con una configuración de información respaldada y asegurada en caso de cualquier fallo posible.

El plan de tesis está estructurado por capítulos: El capítulo I denominado El Problema contiene: contextualización, análisis crítico, prognosis, formulación del problema, interrogantes de la investigación, delimitación del objeto de investigación, delimitación de contenido, delimitación espacial, delimitación temporal, unidades de observación, justificación, objetivo general, objetivos específicos.

El capítulo II llamado Marco Teórico se estructura con los antecedentes investigativos, las fundamentaciones filosófica, tecnológica, legal, red de inclusiones contextuales, constelaciones de ideas, hipótesis con sus respectivas variables.

El capítulo III llamado Metodología contiene: investigación de campo, investigación documental-bibliográfica, investigación experimental, proyectos factibles o de intervención social, proyectos especiales, niveles o tipos de investigación, exploratorio, descriptivo, asociación de variables, explicativo, población y muestra, operacionalización de variables, operacionalización de la variable independiente, operacionalización de la variable dependiente, plan de recolección de información, plan de procesamiento de información.

El capítulo IV denominado Análisis e interpretación de resultados se presenta los datos recogidos en las encuestas buscando una solución efectiva al problema de investigación, adicionalmente se comprueba la hipótesis proponiendo varios métodos de recuperación y respaldo de información, que permitirá una mejor administración de los datos organizados en forma distribuida.

El capítulo V denominado Conclusiones y recomendaciones servirá de base a la propuesta, mediante una solución factible.

El capítulo VI denominado Propuesta plantea la solución basada en una implantación y configuración de un sistema de replicación de tipo open source o licenciado que permitirá mantener la consistencia de la información distribuida a través de los diferentes departamentos, respaldos automáticos a través de sistema operativo, y el aprovechamiento de las herramientas propias del motor de base de datos para crear espejos de información.

Finalmente se presentará la bibliografía utilizada en la presente investigación, así como también los anexos y se mostrará tanto las fichas de instrumentación como los planos de las configuraciones que forman parte de la investigación.

CAPÍTULO I

EL PROBLEMA DE INVESTIGACIÓN

TEMA DE INVESTIGACIÓN

LAS BASES DE DATOS DISTRIBUIDAS Y SU IMPORTANCIA EN LA RECUPERACIÓN, ACTUALIZACIÓN Y ADMINISTRACIÓN DE LA INFORMACIÓN DEL CENTRO EDUCATIVO DIOCESANO SAN PIO X.

PLANTEAMIENTO DEL PROBLEMA

CONTEXTUALIZACIÓN

La evolución de los sistemas de información y el aumento no planificado de la misma dentro de las organizaciones, ha dado lugar a la dispersión de los datos en sitios local o geográficamente distantes. La necesidad de integrar y compartir dichos datos implica la formación de una nueva tecnología capaz de unir de manera consistente dicha información de las organizaciones, esta originalmente se almacenada de manera centralizada, pero con el paso del tiempo las necesidades aumentaron y esto produjo ciertos inconvenientes que no era posible solucionarlos o volverlos eficientes de la forma centralizada. Estos problemas impulsaron la creación de almacenamiento distribuido, los cuales hoy en día proveen características indispensables en el manejo de información; es decir, la combinación de las redes de comunicación y las bases de datos.

Muchas aplicaciones están distribuidas naturalmente en diferentes lugares por ejemplo un banco puede tener múltiples sucursales. Es natural que las bases de datos empleadas en tales aplicaciones estén distribuidas en esos lugares. Muchos usuarios locales tienen acceso exclusivamente a los datos que están en el lugar, pero otros usuarios globales como la oficina central de la compañía puede requerir acceso ocasional a los datos almacenados en varios locales.

Cabe señalar que, por lo regular los datos en cada sitio local describen un mini mundo en ese sitio. Las fuentes de los datos y la mayoría de los usuarios y aplicaciones de la base de datos local residen físicamente en ese lugar.

Deducimos que la llamada “base de datos distribuida” es en realidad un tipo de base de datos virtual cuyas partes componentes están almacenadas en varias bases de datos “reales” distintas que se encuentran en varios sitios distintos, de hecho es la unión lógica de esas bases de datos reales. (C. J. Date, 2001, 651).

Es decir que un sistema distribuido interconecta los lugares que tienen diferentes elementos de tecnología de información, para recolectar, procesar y almacenar los datos, y ser enviados luego a otros lugares donde se requiera dicha información. Para manipular la información se utiliza un Sistema de Administración de Base de Datos Distribuida (SABDD) el cual está formado por los programas que permitan la gestión de la base de forma tal que su manejo sea imperceptible al usuario como si fuera una base de datos local.

Existen factores que han hecho que las bases de datos evolucionen a bases de datos distribuidas. En el mundo de los negocios se ha dado una globalización y a la vez las operaciones de las empresas son cada vez más descentralizadas geográficamente. También el poder de las computadoras personales aumentó y el costo de los Mainframes ya no tenía sentido. Además la necesidad de compartir datos ha hecho que crezca el mercado de las bases de datos distribuidas.

ÁRBOL DE PROBLEMAS

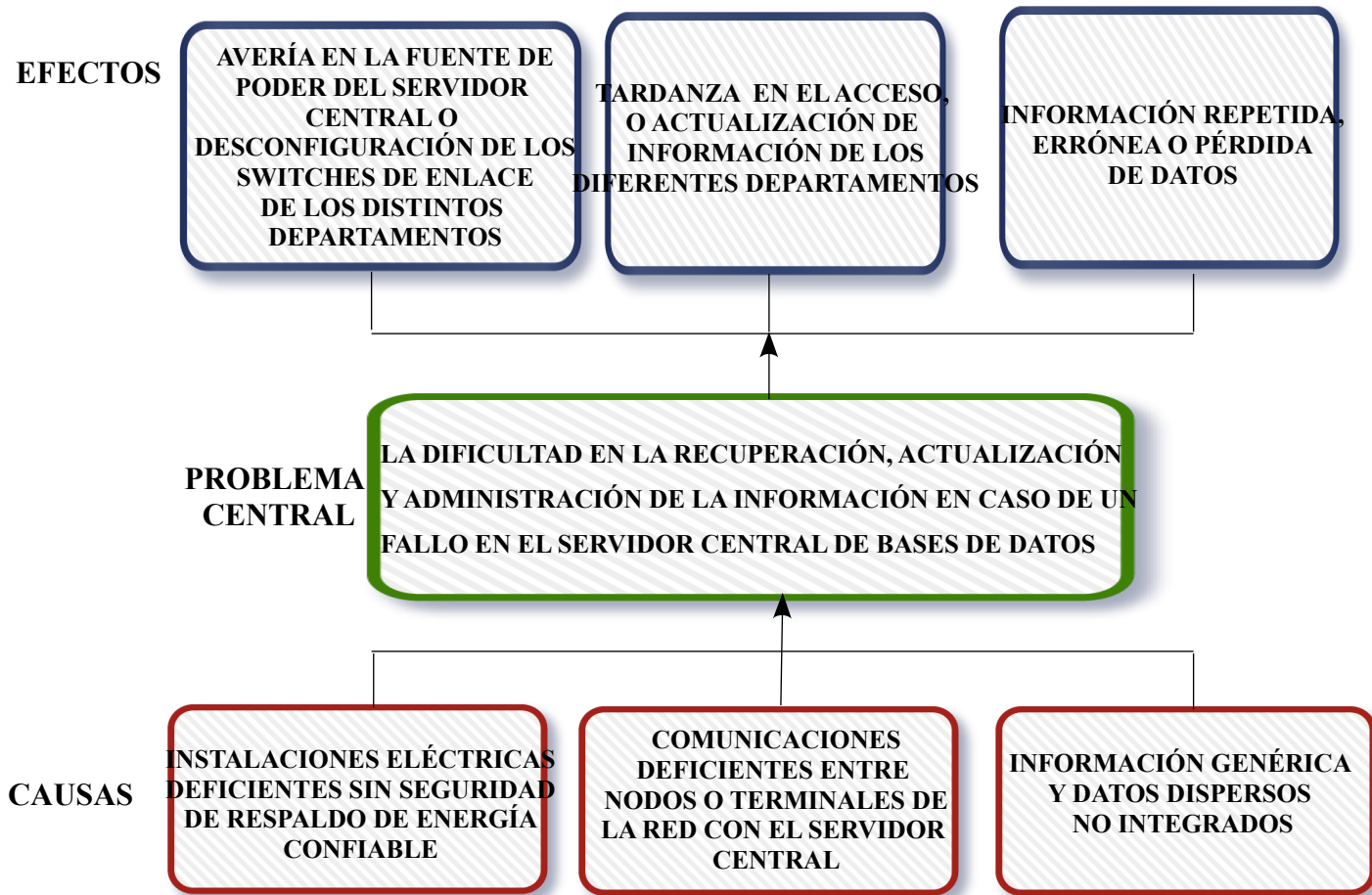


Gráfico N° 1.1: Relación Causa – Efecto
Elaborado por: Investigador

ANÁLISIS CRÍTICO

La escasa difusión de políticas y normas a seguir por parte de los organismos de control correspondientes como: la inspección, secretaría, el departamento de orientación y bienestar estudiantil (DOBE), biblioteca, etc., ha generado incertidumbre y expectativa en cuanto a la integración de la información entre los mismos para mantener una consistencia de datos y accesibilidad a los mismos debido a la falta de estándares para integrar los datos distribuidos de varios departamentos y unidades educativas, en este caso la primaria y secundaria, han provocado incompatibilidad entre la información requerida y los datos proporcionados, ya que limita la coordinación e investigación interdisciplinaria entre los diferentes departamentos dando lugar a un retraso tecnológico y desventaja competitiva con relación a las otras instituciones de la localidad.

Debido a esto la escasa posibilidad de compartir en su momento los datos con otros departamentos se podrá mantener un cierto grado de control local, pero se hace necesario que usuarios de sitios remotos tengan acceso a esa información sin mayor demora.

PROGNOSIS

De continuar con la escasa difusión de políticas y normas a seguir por parte de los organismos de control correspondientes como la inspección, secretaría, DOBE, biblioteca, etc., provocará que el acceso a la información requerida no sea inmediato, y cause duda e incertidumbre sobre la información que se tiene acceso. De persistir en la no implementación de estándares para integrar la información distribuida de los diferentes departamentos de las unidades educativas de primaria y secundaria provocará incompatibilidad e inconsistencia entre los datos registrados en dichas áreas. De no dar atención a la coordinación e investigación interdisciplinaria entre los diferentes departamentos, los datos registrados en los mismos serán irrelevantes entre ellos, y la información útil que se requiera en ciertos casos será redundante provocando retraso en los registros. De seguir con la escasa posibilidad de compartir los datos, el control local

será irrelevante ya que los datos remotos requeridos pueden volver a ser registrados provocando molestias y pérdida de tiempo a los clientes.

FORMULACIÓN DEL PROBLEMA

¿Cómo incidirá las Bases de Datos Distribuidas en la recuperación, actualización y administración de la información del Centro Educativo Diocesano San Pío X.?

INTERROGANTES (Sub problemas)

¿Qué función cumple las Bases de Datos Distribuidas? ¿Cómo se recupera la información frente a una situación crítica (corte de energía, fallo de comunicaciones, fallo en servidor central) inesperada? ¿Cómo se actualiza la información desde los diferentes nodos (departamentos) hacia los demás nodos para ser consultada o utilizada? ¿La administración de la información permite acceder y restringir de manera correcta a los diferentes usuarios de un nodo conectado? ¿Cómo se realiza el respaldo de información almacenado en la base de datos del CED San Pío X? ¿Existe opciones de solución al problema de integración de información de los diferentes departamentos en el CED San Pío X?

DELIMITACIÓN DEL OBJETO DE INVESTIGACIÓN

Delimitación de Contenido

Campo: Sistemas

Área: Gestión de Base de Datos

Aspecto: Bases de Datos Distribuidas – Almacenamiento Recuperación y Administración.

Delimitación Espacial

La investigación se desarrollará en las instalaciones del CED San Pío X.

Delimitación Temporal

El trabajo de investigación se realizará durante el periodo educativo 2011- Abril 2012.

JUSTIFICACIÓN

La investigación tendrá relevancia, porque se busca la aplicación de la distribución de bases de datos para mantener un sistema de respaldo y acceso independiente, de la información en los diferentes departamentos del CED San Pío X.

La investigación tendrá importancia, porque aplicando la distribución de bases de datos se podrá compartir información sin depender de un servicio centralizado de datos.

La factibilidad porque se cuenta con el apoyo de los directivos y ejecutivos del CED San Pío X, los recursos bibliográficos y tecnológicos necesarios para la solución planteada.

La utilidad teórica, por cuanto investigará la aplicación de las Bases de Datos Distribuidas en el entorno de las instituciones educativas de la región central de la República del Ecuador.

Los beneficiarios, serán directamente los usuarios, directivos y ejecutivos del CED San Pío X, pero además marcará una pauta para las instituciones educativas que cuenten con sucursales en varios lugares de la ciudad de Ambato o del Ecuador.

La utilidad práctica por cuanto se realizará la aplicación directamente sobre una empresa que necesitará adoptar la distribución de bases de datos para el año 2012 y tendrá el año 2011 para la transición.

OBJETIVOS

Objetivo General

Analizar y evaluar las diferentes herramientas para Bases de Datos Distribuidas, para la recuperación, actualización y administración de Información del Centro Educativo Diocesano San Pío X.

Objetivos Específicos

- Analizar la Importancia de las Bases de Datos Distribuidas, y comprender la distribución de la información para poder integrarla coherentemente sin que existan inconsistencias.
- Determinar una adecuada arquitectura de Bases de Datos distribuidos para los distintos departamentos y unidades educativas de primaria y secundaria y evitar incompatibilidad entre los sistemas y sus datos.
- Analizar las diferentes formas de replicación existente en las bases de datos distribuidas y aplicar la que más se adecúe a la institución.
- Determinar los principales aspectos de la administración de bases de datos distribuidas y su importancia en la integridad e integración de la información.
- Diseñar y configurar Bases de Datos Distribuidas para el centro Educativo San Pío X.

CAPÍTULO II

MARCO TEÓRICO

ANTECEDENTES INVESTIGATIVOS

Realizado un recorrido por las universidades del Ecuador que ofertan la carrera de ingeniería en sistemas se encuentra que:

En la Universidad Técnica del Norte, Facultad de Ciencias Aplicadas, Escuela de Ingeniería en Sistemas Computacionales se encuentra una tesis llamada:

FRAGMENTACIÓN DE TABLAS EN BASES DE DATOS DISTRIBUIDAS

realizado por López Tulcanaza Verónica Alexandra cuya principal conclusión es que la implementación de bases de datos distribuidas mejora el rendimiento y la accesibilidad de los datos debido a una disminución del tráfico de la red y a un trabajo colaborativo de varias estaciones como servidores como si fueran uno. Además resulta importante que se puede replicar y distribuir en varios sitios cuando ciertos sitios tienen capacidades limitadas.

También en la Universidad Técnica del Norte, Facultad de Ciencias Aplicadas, Escuela de Ingeniería en Sistemas Computacionales se encuentra una tesis llamada:

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

realizado por José Luis Cisneros y Javier Jirón en la cual la principal conclusión fue que la inversión en los costos de comunicación no son significativos para una implementación de integración de información ya que era de manera esporádica y no constante.

FUNDAMENTACIONES

Fundamentación filosófica

Para realizar la investigación, el investigador se ubica en el paradigma filosófico crítico propositivo, porque cuestiona la manera de hacer investigación y por el contrario plantea una propuesta de solución al problema investigado basado en la existencia de múltiples realidades socialmente construidas.

Tecnológica

La tecnología es fundamental en esta investigación, por cuanto se trata de la aplicación de modelos de almacenamiento de Bases de Datos combinados con Motores de Bases de Datos relacionales que permitan la distribución de información en más de un sitio y que estén interconectados entre sí.

Administrativa

La fundamentación administrativa es evidente ya que la planificación, organización, dirección y control de los recursos (humanos, financieros, materiales, tecnológicos, el conocimiento, etc.), permite obtener el máximo beneficio posible al trabajar con usuarios, grupos, redes, servidores etc.

Fundamentación legal

Ley Orgánica nº 24 de 4 de mayo de 2004, de Transparencia y Acceso a la Información Pública

Ley Orgánica nº 24 de 4 de mayo de 2004 de Transparencia y Acceso a La Información Pública (Publicado en el Registro Oficial Suplemento 337 del 18 de Mayo del 2004).

EL CONGRESO NACIONAL

Considerando:

Que el artículo 81 de la Constitución Política de la República, garantiza el derecho a

acceder a las fuentes de información, como mecanismo para ejercer la participación democrática respecto del manejo de la cosa pública y la rendición de cuentas a la que están sujetos todos los funcionarios del Estado, y demás entidades obligadas por esta Ley:

Que es necesario hacer efectivo el principio de publicidad de los actos, contratos y gestiones de las instituciones del Estado y de aquellas financiadas con recursos públicos o que por su naturaleza sean de interés público;

Que la misma norma constitucional establece que no existirá reserva respecto de informaciones que reposen en archivos públicos, excepto de aquellas que por seguridad nacional no deben ser dadas a conocer;

Que la libertad de información está reconocida tanto en el artículo 19 del Pacto Internacional de Derechos Civiles y Políticos, como en el artículo 13 de la Convención Interamericana de Derechos Humanos; y,

En ejercicio de sus facultades constitucionales y legales, expide la siguiente:

LEY ORGÁNICA DE TRANSPARENCIA Y ACCESO A LA INFORMACIÓN PÚBLICA

Título Primero: Principios Generales

Artículo 1.- Principio de Publicidad de la Información Pública.

Artículo 2.- Objeto de la Ley.

Artículo 3.- Ámbito de Aplicación de la Ley.

Artículo 4.- Principios de Aplicación de la Ley.

Título Segundo: De la Información Pública y su Difusión

Artículo 5.- Información Pública.

Artículo 6.- Información Confidencial.

Artículo 7.- Difusión de la Información Pública.

Artículo 8.- Promoción del Derecho de Acceso a la Información.

Artículo 9.- Responsabilidad sobre la entrega de la Información Pública.

Artículo 10.- Custodia de la Información.

Artículo 11.- Vigilancia y Promoción de la Ley.

Artículo 12.- Presentación de Informes.

Artículo 13.- Falta de claridad en la Información.

Título Tercero: De la Información Reservada y Confidencial

Artículo 17.- De la Información Reservada.

Artículo 18.- Protección de la Información Reservada.

Título Cuarto: Del proceso administrativo para acceder a la Información Pública

Artículo 19.- De la Solicitud y sus Requisitos.

Artículo 20.- Límites de la Publicidad de la Información.

Artículo 21.- Denegación de la Información.

Título Quinto: Del Recurso de Acceso a la Información

Artículo 22.- Del Recurso de Acceso a la Información Título Sexto: De las Sanciones

Artículo 23.- Sanción a funcionarios y/o empleados públicos y privados.

Disposición General

Disposiciones Transitorias

CATEGORÍAS FUNDAMENTALES

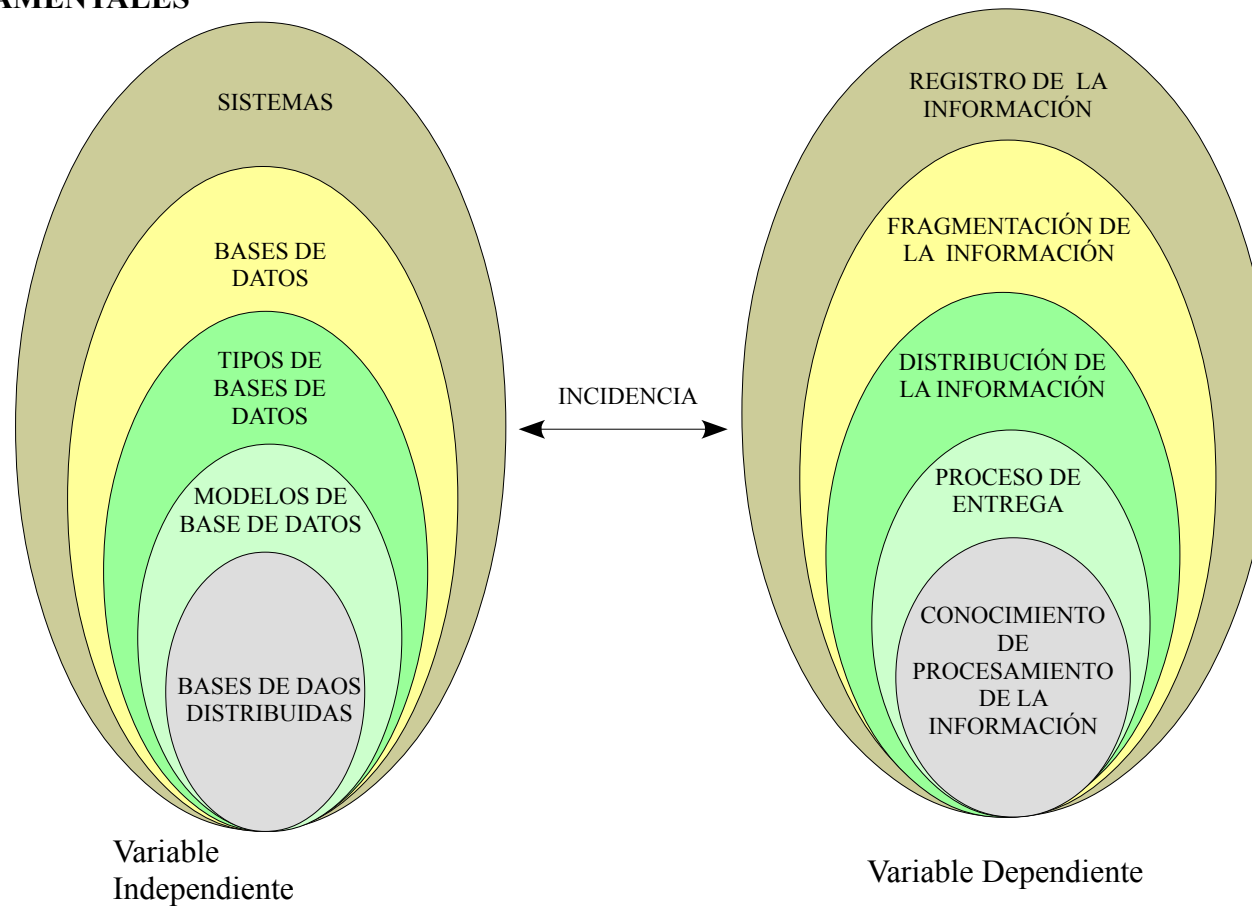


Gráfico N° 2.1: Categorías Fundamentales
Elaborado por: Investigador

SISTEMAS

Un sistema es un conjunto de partes o elementos organizadas y relacionadas que interactúan entre sí para lograr un objetivo. Los sistemas reciben datos, energía o materia del ambiente (entrada), y proveen información, energía o materia (salida).

El ambiente es el medio en externo que envuelve física o conceptualmente a un sistema. El sistema tiene interacción con el ambiente, del cual recibe entradas y al cual se le devuelven salidas. El ambiente también puede ser una amenaza para el sistema.

Un grupo de elementos no constituye un sistema si no hay una relación e interacción, que de la idea de un “todo” con un propósito.

Un sistema informático es un conjunto de partes que funcionan relacionándose entre sí con un objetivo preciso. Sus partes son: hardware, software y las personas que lo usan. Por ejemplo, una computadora, sus dispositivos periféricos y la persona que la maneja, pueden constituir un sistema informático.

Un sistema informático puede formar parte de un sistema de información; en este último la información, uso y acceso a la misma, no necesariamente está informatizada. Por ejemplo, el sistema de archivo de libros de una biblioteca y su actividad en general es un sistema de información. Si dentro del sistema de información hay computadoras que ayudan en la tarea de organizar la biblioteca, entonces ese es un sistema informático.

En un sistema informático se utilizan computadoras para almacenar, procesar y/o acceder a información. En un sistema de información se pueden utilizar computadoras, pero no es necesario. El acceso a la información puede ser físico (por ejemplo, una persona se encarga de buscar en un archivador).

Tanto el sistema informático como el sistema de información, incluyen a las personas que acceden o producen información dentro del sistema. Las personas tienen que capacitarse para entender el funcionamiento y procedimientos que soporta sistema. Ambos sistemas tienen un propósito. Por ejemplo, gestionar el acceso y distribución de

libros una biblioteca, administrar la entrada/salida de mercadería, personal y otros recursos de un comercio, etc.

BASES DE DATOS

Es una costumbre referirse a los datos de la base de datos como “persistentes” (¡aunque en realidad éstos podrían no persistir por mucho tiempo!). Por *persistentes* queremos decir, de manera intuitiva, que el tipo de datos de la base de datos difiere de otros datos más efímeros, como los datos de entrada, los datos de salida, las instrucciones de control, las colas de trabajo, los bloques de control de software, los resultados intermedios y de manera más general, cualquier dato que sea de naturaleza transitoria. En forma más precisa, decimos que los datos de la base de datos “persisten” debido en primer lugar a que una vez aceptados por el Sistema Administrador de Bases de datos (DBMS por sus siglas en inglés) para entrar en la base de datos, en lo sucesivo sólo pueden ser removidos de la base de datos por alguna solicitud explícita al DBMS, no como un mero efecto lateral de (por ejemplo) algún programa que termina su ejecución. Por lo tanto, esta noción de persistencia nos permite dar una definición más precisa del término “base de datos”:

Una **base de datos** es un conjunto de datos persistentes que es utilizado por los sistemas de aplicación de alguna empresa dada.

Una empresa podría ser un solo individuo (con una pequeña base de datos personal), toda una corporación o un gran consorcio similar (con una gran base de datos compartida) o todo lo que se ubique entre estas dos opciones. Aquí tenemos algunos ejemplos:

1. Una compañía manufacturera
2. Un banco
3. Un hospital
4. Una universidad
5. Un departamento gubernamental

Toda empresa necesariamente debe mantener una gran cantidad de datos acerca de su operación. Estos datos son los “datos persistentes” a los que nos referimos antes. En forma característica, las empresas que acabamos de mencionar incluirían entre sus datos persistentes a los siguientes:

1. Datos de producción
2. Datos contables
3. Datos de pacientes
4. Datos de estudiantes
5. Datos de planeación

Una base de datos tiene las siguientes propiedades implícitas: Una base de datos representa algún aspecto del mundo real, en ocasiones llamado **minimundo** o **universo de discurso**. Las modificaciones del minimundo se reflejan en la base de datos.

- Una base de datos es un conjunto de datos lógicamente coherente, con cierto significado inherente. Una colección aleatoria de datos no puede considerarse propiamente una base de datos.
- Toda base de datos se diseña, construye y pobla con datos para un propósito específico. Está dirigida a un grupo de usuarios y tiene ciertas aplicaciones preconcebidas que interesan a dichos usuarios.

En otras palabras, una base de datos tiene una fuente de la cual se derivan los datos, cierto grado de interacción con los acontecimientos del mundo real y un público que está activamente interesado en el contenido de la base de datos.

Las bases de datos pueden ser de cualquier tamaño y tener diversos grados de complejidad. Por ejemplo, la lista de nombres y direcciones antes mencionada puede contener apenas unas cuantas centenas de registros, cada uno de ellos con una estructura muy simple. Por otro lado, el catálogo de una biblioteca grande puede contener medio millón de tarjetas clasificadas por categorías distintas, apellido del primer autor, tema, título, etc. Y ordenadas alfabéticamente en cada categoría. Las autoridades fiscales

mantienen una base de datos todavía más grande y compleja para llevar el control de las declaraciones fiscales que presentan los contribuyentes..

SISTEMAS DE BASES DE DATOS

Un sistema de base de datos es básicamente un sistema computarizado para guardar registros; es decir, es un sistema computarizado cuya finalidad general es almacenar información y permitir a los usuarios recuperar y actualizar esa información con base en peticiones. La información en cuestión puede ser cualquier cosa que sea de importancia para el individuo u organización; en otras palabras, todo lo que sea necesario para auxiliarle en el proceso general de su administración.

Los sistemas de bases de datos (DBMS), se diseñan para gestionar grandes cantidades de información. La gestión de los datos implica tanto la definición de estructuras para almacenar la información como la provisión de mecanismos para la manipulación de la información. Además, los sistemas de bases de datos deben proporcionar la fiabilidad de la información almacenada, a pesar de las caídas del sistema o los intentos de acceso sin autorización. Si los datos van a ser compartidos entre diversos usuarios, el sistema debe evitar posibles resultados anómalos.

En general, los datos de la base de datos *por lo menos en un sistema grande* serán tanto *integrados* como *compartidos*. Los aspectos de integración y compartimiento de datos representan una ventaja importante de los sistemas de bases de datos en el entorno “grande”; y al menos, también la integración de datos puede ser importante en los entornos “pequeños”.

- Por **integrada**, queremos decir que podemos imaginar a la base de datos como una unificación de varios archivos que de otro modo serían distintos, con una redundancia entre ellos eliminada al menos parcialmente.
- Por **compartida**, queremos decir que las piezas individuales de datos en la base pueden ser compartidas entre diferentes usuarios y que cada uno de ellos puede

tener acceso a la misma pieza de datos, probablemente con fines diferentes.

Otra consecuencia de los hechos precedentes *que la base de datos sea integrada y (por lo regular) compartida* es que cualquier usuario ocupará normalmente sólo una pequeña parte de la base de datos total; lo que es más, las partes de los distintos usuarios se traslaparán de diversas formas. En otras palabras, una determinada base de datos será percibida de muchas formas diferentes por los distintos usuarios. De hecho, aun cuando dos usuarios tengan la misma porción de la base de datos, su visión de dicha parte podría diferir considerablemente a un nivel detallado

El sistema no sólo contiene la base de datos misma, sino también una definición o descripción completa de la base de datos. Esta definición se almacena en el catálogo del sistema, que contiene información como la estructura de cada archivo, el tipo y formato de almacenamiento de cada elemento de información y diversas restricciones que se aplican a los datos. A la información almacenada en el catálogo se le denomina metadatos, y éstos describen la estructura de la base de datos primaria

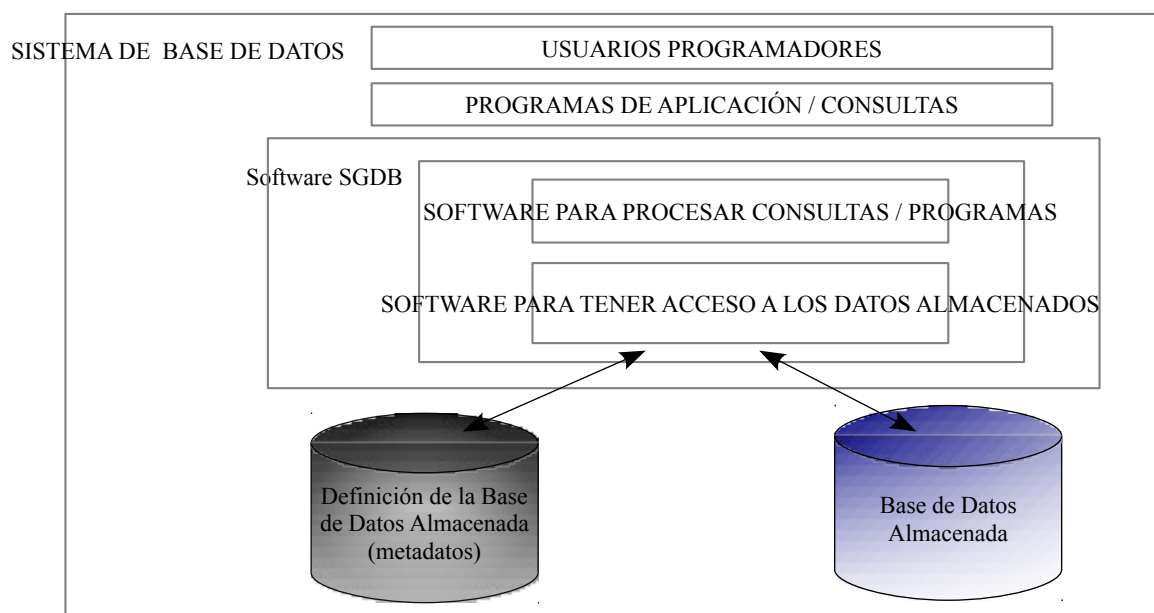


Gráfico N° 2.2: Entorno simplificado de un sistema de Bases de Datos
Fuente: ELMASARI, R, NATHE, S. (1994). *Sistemas de Bases de Datos – Conceptos Fundamentales*. Mexico: Addison – Wesley. p. 12

En esta sección analizaremos qué características son deseables en los DBMS y qué capacidades deben ofrecer. El Administrador de Bases de Datos (DBA por sus siglas en inglés) debe aprovechar estas capacidades para lograr diversos objetivos relacionados con el diseño, la administración y el empleo de una base grande de datos multiusuario

Control de la redundancia

En la creación tradicional de programas con procesamiento de archivos, cada grupo de usuarios mantiene sus propios archivos para manejar sus aplicaciones de procesamiento de datos.

Por ejemplo, la base de datos del CED San Pío X; hay dos grupos de usuarios podrían ser el personal de matriculación a cursos y la oficina de contabilidad.

Con el enfoque tradicional, cada grupo mantendría archivos independientes para cada estudiante. La oficina de contabilidad mantendría también datos relacionados con las matriculaciones y la facturación correspondiente, en tanto que la oficina de matriculación controlaría los cursos y las notas de los estudiantes. Una buena parte de los datos se almacenaría dos veces: una vez en los archivos de cada grupo de usuarios. Otros grupos de usuarios podrían duplicar parte de esos datos, o todos, en sus propios archivos. A veces, y no pocas, esta redundancia en el almacenamiento de los mismos datos provoca varios problemas.

En primer lugar, es necesario realizar una misma actualización lógica como introducir los datos de un nuevo estudiante varias veces: una vez en cada archivo en el que se registren datos de los estudiantes. Esto implica una *duplicación del trabajo*.

En segundo lugar, *se desperdicia espacio de almacenamiento* al guardar los mismos datos en varios lugares, y este problema puede ser grave si las bases de datos son grandes.

En tercer lugar es posible que los archivos que representan los mismos datos se tornen inconsistentes, quizá porque una actualización se haya aplicado a ciertos archivos pero

no a otros. Incluso la actualización digamos la adición de un nuevo estudiante se aplica a todos los archivos apropiados, persiste la posibilidad de que los datos relacionados con el estudiante sean inconsistentes porque cada grupo de usuarios aplica las actualizaciones de manera independiente.

Por ejemplo, un grupo de usuarios podría introducir como fecha de nacimiento del estudiante el valor erróneo 19-ENE-1974, en tanto que los demás grupos de usuarios introducirían el valor correcto 29-ENE-1974.

Con el enfoque de bases de datos, las vistas de los diferentes grupos de usuarios se integran durante el diseño de la base de datos. Para conservar la consistencia, debe crearse un diseño que almacene cada dato lógico “como el nombre o la fecha de nacimiento de un estudiante” en un *solo lugar* de la base de datos. Ello evita la inconsistencia y ahorra espacio de almacenamiento. En algunos casos puede convenir la redundancia controlada. Por ejemplo, podríamos almacenar de manera redundante NombreEstudiante y NúmCurso en un archivo INFORME_NOTAS porque, siempre que recuperemos un registro de INFORME_NOTAS, queremos recuperar el nombre del estudiante y el número del curso junto con la nota, el número del estudiante y el identificador de la sección. Si colocamos juntos todos los datos, no tendremos que buscar en varios archivos los datos que deseamos reunir. En tales casos, el DBMS deberá ser capaz de **controlar** esta redundancia para que no haya inconsistencias entre los archivos.

Restricción de los accesos no autorizados

Cuando muchos usuarios comparten una misma base de datos, es probable que no todos tengan la autorización para tener acceso a toda la información que contiene. Por ejemplo, es común considerar que los datos financieros son confidenciales y que sólo ciertas personas puedan tener autorización para tener acceso a ellos. Además, es posible que sólo algunos usuarios tengan permiso para recuperar datos, en tanto que a otros se les permita obtenerlos y actualizarlos. Por tanto, también es preciso controlar el tipo de

las operaciones de acceso (obtención o actualización). Por lo regular, a los usuarios o grupos de usuarios se les asignan números de cuenta protegidos con contraseñas, mismos que sirven para tener acceso a la base de datos. El DBMS debe contar con un subsistema de **seguridad y autorización** que permita al DBA crear cuentas y especificar restricciones para ellas. El DBMS deberá entonces obligar automáticamente al cumplimiento de dichas restricciones. Cabe señalar que el mismo tipo de controles se puede aplicar al software del DBMS. Por ejemplo, sólo el personal del DBA tendrá autorización para utilizar cierto software **privilegiado**, como el que sirve para crear cuentas nuevas. De manera similar, podemos hacer que los usuarios paramétricos sólo puedan tener acceso a la base de datos a través de las transacciones programadas que expresamente fueron creadas para ellos.

Almacenamiento persistente de objetos y estructuras de datos de programas Una aplicación reciente de las bases de datos consiste en ofrecer almacenamiento persistente para objetos y estructuras de datos de programas. Esta es una de las principales razones de que se hayan creado los DBMS orientados a objetos. Es común que los lenguajes de programación cuenten con estructuras de datos complejas, como los tipos de registro en PASCAL o las definiciones de clases en C++. Los valores de las variables de un programa se desechan una vez que éste termina, a menos que el programador explícitamente los almacene en archivos permanentes; para ello, suele requerirse la conversión de esas estructuras complejas a un formato adecuado para su almacenamiento en archivos. Cuando hay que leer otra vez estos datos, el programador debe convertirlos del formato de archivo a la estructura de variables del programa. Los sistemas de base de datos orientados a objetos son compatibles con lenguajes de programación del tipo de C++ y SMALLTALK, y el software del DBMS realiza automáticamente las conversiones necesarias. Así, podemos almacenar permanentemente un objeto complejo de C++ en un DBMS orientado a objetos, como ObjectStore.

Se dice que los objetos de este tipo son persistentes porque sobreviven cuando termina la ejecución del programa y después se pueden recuperar directamente mediante otro

programa en C++ .

El almacenamiento persistente de objetos y estructuras de datos de programas es una función importante para los sistemas de base de datos. Los DBMS tradicionales a menudo adolecían del llamado problema de incompatibilidad de impedancia porque las estructuras de datos proporcionadas por el DBMS eran incompatibles con las del lenguaje de programación. Los sistemas de base de datos orientados a objetos suelen ofrecer compatibilidad de las estructuras de datos con uno o más lenguajes de programación orientada a objetos.

En el cumplimiento de las restricciones de integridad La mayor parte de las aplicaciones de base de datos tienen ciertas *restricciones de integridad* que deben cumplir los datos. El DBMS debe ofrecer recursos para definir tales restricciones y hacer que se cumplan. La forma más simple de restringir la integridad consiste en especificar un tipo de datos para cada elemento de información. Por ejemplo, podemos especificar que el valor del elemento Grado dentro de cada registro de ESTUDIANTE debe ser un entero entre 1 y 5, y que el valor de Nombre debe ser una cadena de no más de 30 caracteres alfabéticos.

Otro tipo de restricción que encontramos a menudo, más complejo, implica especificar que un registro de un archivo debe relacionarse con registros de otros archivos. Por ejemplo, podemos especificar que “todo registro de SECCIÓN debe estar relacionado con un registro de CURSO”. Otro tipo de restricción especifica que los valores de los elementos de información sean únicos; por ejemplo, “cada registro de CURSO debe tener un valor único de NumCurso”. Estas restricciones se derivan de la *semántica* (o significado) de los datos y del minimundo que representa. Es responsabilidad de los diseñadores de la base de datos identificar las restricciones de integridad durante el diseño. Algunas restricciones se pueden especificar en el DBMS, el cual hará automáticamente que se cumplan; otras pueden requerir verificación mediante programas de actualización o en el momento en que se introducen los datos.^[1]

Es posible introducir erróneamente un dato sin violar las restricciones de integridad.

Por ejemplo, si un estudiante obtiene una nota de A pero se introduce C en la base de datos, el DBMS no *podrá* descubrir este error automáticamente, porque C es un valor permitido del tipo de datos de Notas. Esta clase de errores sólo puede descubrirse manualmente (cuando el estudiante reciba su boleta de notas y se queje) y corregirse después actualizando la base de datos. Por otro lado, el DBMS puede rechazar automáticamente una nota de X, porque éste no es un valor permitido para el tipo de datos de Notas.

Respaldo y recuperación

Todo DBMS debe contar con recursos para recuperarse de fallos de hardware o de software.

Para ello está el subsistema de *respaldo y recuperación* del DBMS. Por ejemplo, si el sistema falla mientras se está ejecutando un complejo programa de actualización, el subsistema de recuperación se encargará de asegurarse de que la base de datos se restaure al estado en el que estaba antes de que comenzara la ejecución del programa. Como alternativa, el subsistema de recuperación puede asegurarse de que el programa reanude su ejecución en el punto en que fue interrumpido, de modo que su efecto completo se registre en la base de datos.

Hay otras implicaciones del empleo del enfoque de bases de datos que pueden resultar provechosas para casi todas las organizaciones entre ellos :

- *Potencial para imponer normas*

Con el enfoque de las bases de datos el DBA puede definir e imponer normas a los usuarios de la base de datos en una organización grande. Esto facilita la comunicación y cooperación entre diversos departamentos, proyectos y usuarios de esa organización. Es posible definir normas para los nombres y formatos de los elementos de información, para los formatos de presentación, para las estructuras de los informes, para la terminología, etc. Es más fácil que el DBA imponga normas

en un entorno centralizado de base de datos que en un entorno en el que cada grupo de usuarios tenga el control de sus propios archivos y programas.

- *Menor tiempo de creación de aplicaciones*

Una de las características más convincentes a favor del enfoque de bases de datos es que la creación de una aplicación nueva “como la obtención de cierta información de la base de datos para imprimir un nuevo informe” requiere muy poco tiempo.

Diseñar e implementar una nueva base de datos desde cero puede tardar más que escribir una sola aplicación de archivos especializada; sin embargo, una vez que la base de datos está construida y en funciones, casi siempre se requerirá mucho menos tiempo para crear nuevas aplicaciones con los recursos del DBMS.

- *Flexibilidad*

En ocasiones es necesario modificar la estructura de una base de datos cuando cambian los requerimientos. Por ejemplo, podría surgir un nuevo grupo de usuarios que necesite información adicional que no se encuentra actualmente en la base de datos. Los DBMS permiten efectuar estas modificaciones en la estructura de la base de datos sin afectar los datos almacenados y los programas de aplicación ya existentes

- *Disponibilidad de información actualizada*

Los DBMS ponen la base de datos a disposición de todos los usuarios. En el momento en que un usuario actualiza la base de datos, todos los demás usuarios pueden ver de inmediato esta actualización. Esta disponibilidad de información actualizada es indispensable en muchas aplicaciones de procesamiento de transacciones, como los sistemas de reservaciones o las bases de datos bancarias, y se hace posible gracias a los subsistemas de control de concurrencia y de recuperación del DBMS.

- *Economías de escala*

El enfoque de DBMS permite consolidar los datos y las aplicaciones, reduciéndose así el desperdicio por traslapeo entre las actividades del personal de procesamiento de datos en los diferentes proyectos o departamentos. Esto permite que la organización completa invierta en procesadores más potentes, dispositivos de almacenamiento o equipo de comunicación, en vez de que cada departamento tenga que adquirir por separado su propio equipo (de menor capacidad) . Esto reduce los costos totales de operación y control.^[2]

TIPOS DE BASES DE BASES DE DATOS

Según la variabilidad de los datos almacenados Bases de datos estáticas

Éstas son bases de datos de sólo lectura, utilizadas primordialmente para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos a través del tiempo, realizar proyecciones y tomar decisiones.

Bases de datos dinámicas

Éstas son bases de datos donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualización y adición de datos, además de las operaciones fundamentales de consulta. Un ejemplo de esto puede ser la base de datos utilizada en un sistema de información de una tienda de abarrotes, una farmacia, un video club, etc.

Según el contenido

Bases de datos bibliográficas

Solo contienen un surrogante (representante) de la fuente primaria, que permite localizarla. Un registro típico de una base de datos bibliográfica contiene información sobre el autor, fecha de publicación, editorial, título, edición, de una determinada publicación, etc. Puede contener un resumen o extracto de la publicación original, pero

nunca el texto completo, porque sino estaríamos en presenciade una base de datos a texto completo (o de fuentes primarias). Como su nombre lo indica, el contenido son cifras o números. Por ejemplo, una colección de resultados de análisis de laboratorio, entre otras.

Bases de datos de texto completo

Almacenan las fuentes primarias, como por ejemplo, todo el contenido de todas las ediciones de una colección de revistas científicas

MODELOS DE BASES DE DATOS

Modelo jerárquico.

El sistema jerárquico más comúnmente conocido es el sistema IMS de IBM. Tiene como objetivo establecer una jerarquía de fichas, de manera que cada ficha puede contener a sus vez listas de otras fichas, y así sucesivamente. Por ejemplo, una ficha de clientes puede contener una lista de fichas de facturas, cada una de las cuales puede contener a su vez una lista de fichas de líneas de detalle que describen los servicios facturados.

Una base de datos jerárquica está compuesta por una secuencia de bases de datos físicas, de manera que cada base de datos física se compone de todas las ocurrencias de un tipo de registro o dicha determinada. Una ocurrencia de registro es una jerarquía de ocurrencias de segmento. Cada ocurrencia de segmento está formada por un conjunto de ocurrencias o instancias de los campos que componen el segmento. Como indica el Gráfico N° 2.3.

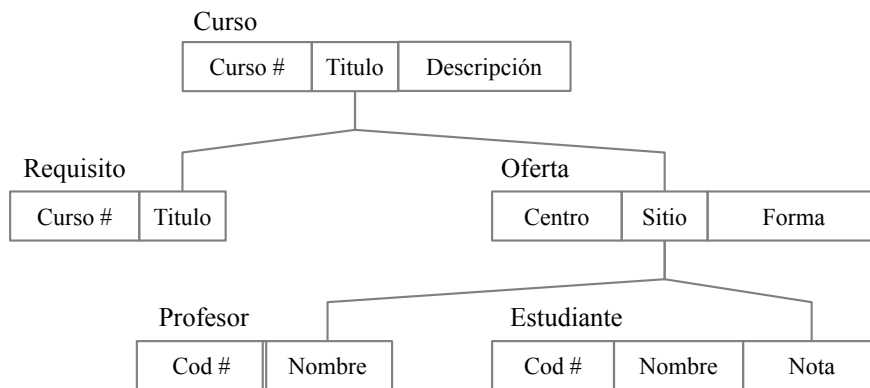


Gráfico N° 2.3: Ejemplo de tipo de registro.

Fuente: <http://www.lcc.uma.es/~galvez/ftp/bdst/Tema2.pdf>

Cabe distinguir en este punto entre el concepto de tipo de registro, y ocurrencia o instancia de registro. El tipo define la estructura general que debe poseer, o sea, los campos de cada uno de sus segmentos, y la estructura jerárquica entre ellos. Una instancia es un valor de un tipo de registro. Para que quede más claro, un tipo de registro es como un tipo de persona: blanco, negro, amarillo, aceitunado, etc., mientras que una instancia es una persona concreta perteneciente a uno de estos tipos: Pablo Picasso, Nelson Mandela, Mao Tse Tung, Toro Sentado, etc.

De esta forma, al segmento que se halla a la cabeza de un registro, se le llama segmento padre, y se llama segmentos hijo a los que dependen de él. Para moverse por un registro de estructura jerárquica lo que se hace es posicionarse inicialmente en la raíz de una instancia, e ir navegando por sus hijos según convenga consultando o modificando los datos pertinentes.

Modelo en red.

Se puede considerar al modelo de bases de datos en red como de una potencia intermedia entre el jerárquico y el relacional. Su estructura es parecida a la jerárquica aunque bastante más compleja, con lo que se consiguen evitar, al menos en parte, los problemas de aquél. Los conceptos fundamentales que debe conocer el administrador para definir el esquema de una base de datos jerárquica, son los siguientes:

- *Registro*: Viene a ser como cada una de las fichas almacenadas en un fichero convencional.
- *Campos o elementos de datos*. Son cada uno de los apartados de que se compone una ficha.
- *Conjunto*: Es el concepto que permite relacionar entre sí tipos de registro distintos.

Se puede definir los registros simplemente como fichas de un fichero. Para ilustrar el concepto de conjunto, suponga que se tiene un tipo de registro de clientes, y un tipo de registro de vuelos de avión, y suponga que quiere asociar ambas informaciones, de manera que para cada vuelo se desea saber cuáles son los pasajeros que viajan en él. La forma de hacerlo es a través de un conjunto. Un conjunto relaciona dos tipos de registro. Uno de ellos es el registro propietario del conjunto, y el otro es el miembro. El diagrama de la figura siguiente aclarará las cosas un poco más. (son los diagramas de Bachman). Cada tipo de conjunto, posee, a su vez, una serie de ocurrencias de conjunto, donde cada ocurrencia está formada por una instancia del tipo propietario, y una, varias o ninguna instancia del tipo miembro.

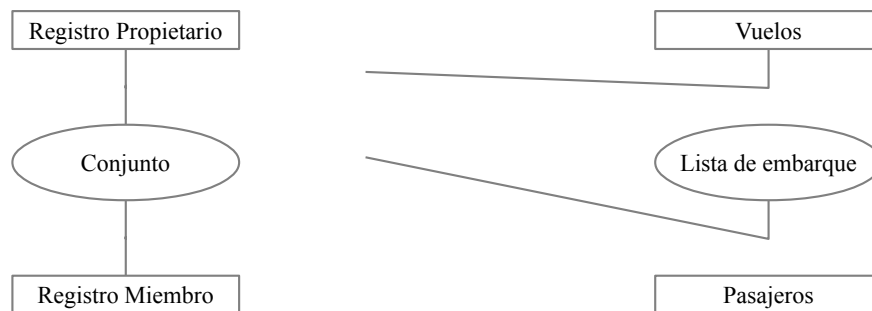


Gráfico N° 2.4: Modelo de Red

Fuente: <http://www.lcc.uma.es/~galvez/ftp/bdst/Tema2.pdf>

Una restricción bastante importante de este modelo, es que una ocurrencia de registro miembro puede pertenecer como máximo a una sola instancia de un determinado conjunto, aunque puede participar en varios tipos de conjuntos distintos. Este modelo en red es más potente que el modelo jerárquico, ya que aquél puede simularse, aplicando

una jerarquía de conjuntos en varios niveles. Por ejemplo, el ejemplo jerárquico del punto anterior quedaría ahora como:

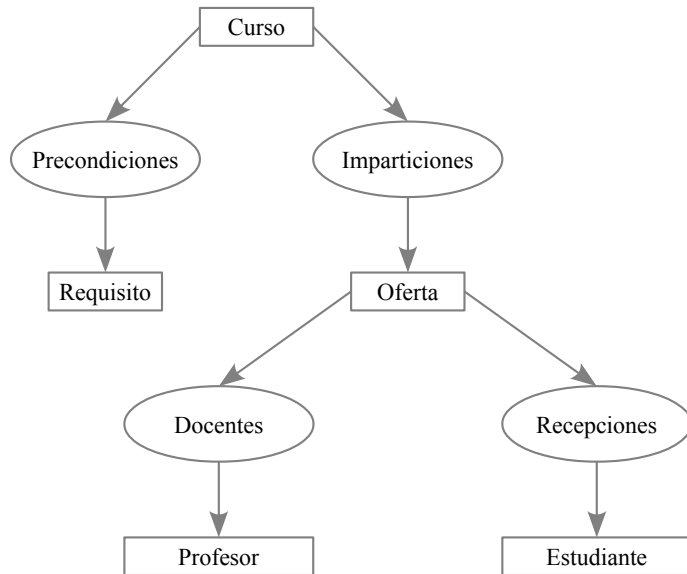


Gráfico N° 2.5: Simular un modelo jerárquico mediante un modelo de red
 Fuente: <http://www.lcc.uma.es/~galvez/ftp/bdst/Tema2.pdf>

Por otro lado, en un conjunto concreto, el tipo de registro propietario no puede ser, a su vez, el mismo que el tipo de registro miembro, o sea, un mismo tipo de registro no puede intervenir en el mismo conjunto como propietario y como miembro a la vez.

Para ilustrar por qué el modelo en red es más potente que el modelo jerárquico, basta con observar un conjunto como el siguiente:

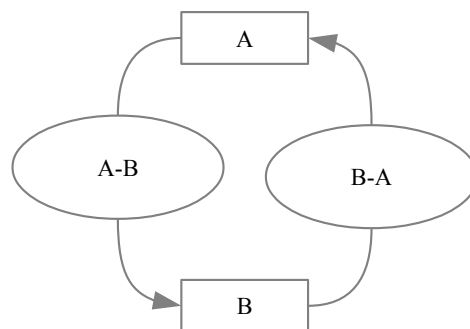


Gráfico N° 2.6: Diagrama de relaciones entre tipos de registro que no pueden expresarse según el esquema jerárquico.
 Fuente: <http://www.lcc.uma.es/~galvez/ftp/bdst/Tema2.pdf>

Aquí, un elemento de A puede poseer varios de B, mediante el conjunto A-B; a su vez, los de B pueden poseer a los de A, mediante B-A, y así sucesivamente cuantas veces se quiera. Este ejemplo no se puede hacer en el modelo jerárquico, pues el número de niveles varía dinámicamente.

Que una misma instancia de registro miembro no pueda aparecer en más de una instancia de conjunto, hace que sea difícil de expresar algunas situaciones.

Modelo Entidad – Relación

El modelo de datos entidad-relación (E-R) está basado en una percepción del mundo real que consta de una colección de objetos básicos, llamados *entidades*, y de *relaciones* entre estos objetos. Una entidad es una **cosa** u **objeto** en el mundo real que es distinguible de otros objetos. Por ejemplo, cada persona es una entidad, y las cuentas bancarias pueden ser consideradas entidades. Las entidades se describen en una base de datos mediante un conjunto de *atributos*. Por ejemplo, los atributos *número-cuenta* y *saldo* describen una cuenta particular de un banco y pueden ser atributos del conjunto de entidades *cuenta*. Análogamente, los atributos *nombre-cliente*, *calle-cliente* y *ciudad-cliente* pueden describir una entidad *cliente*. Un atributo extra, *id-cliente*, se usa para identificar unívocamente a los clientes. Se debe asignar un identificador único de cliente a cada cliente.

Una **relación** es una asociación entre varias entidades. Por ejemplo, una relación *impositor* asocia un cliente con cada cuenta que tiene. El conjunto de todas las entidades del mismo tipo, y el conjunto de todas las relaciones del mismo tipo, se denominan respectivamente **conjunto de entidades** y **conjunto de relaciones**. La estructura lógica general de una base de datos se puede expresar gráficamente mediante un *diagrama ER*, que consta de los siguientes componentes:

- **Rectángulos**, que representan conjuntos de entidades.
- **Elipses**, que representan atributos.
- **Rombos**, que representan relaciones entre conjuntos de entidades.

- **Líneas**, que unen los atributos con los conjuntos de entidades y los conjuntos de entidades con las relaciones.^[3]

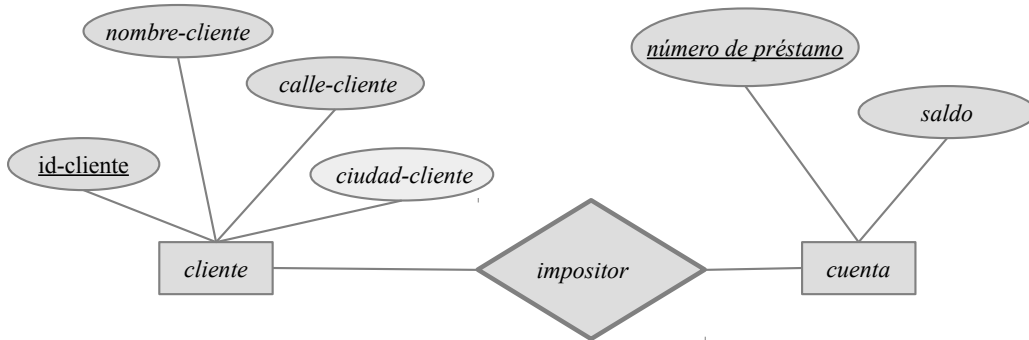


Gráfico N° 2.7 Modelo E-R

Fuente: SILBERTSHATZ, A, KORTH, H, SUDARSHAN, S. (2002). *Fundamentos de Bases de Datos*. Madrid: McGraw Hill. p. 30.

Modelo relacional.

Hemos mencionado en varias ocasiones que los sistemas relacionales se basan en un fundamento formal, o teoría, denominado *el modelo relacional de datos*. De manera intuitiva, lo que esta afirmación significa (entre otras cosas) es que en dichos sistemas:

1. **Aspecto estructural:** El usuario percibe la información de la base de datos como tablas y nada más que tablas;
2. **Aspecto de integridad:** Estas tablas satisfacen ciertas restricciones de integridad
3. **Aspecto de manipulación:** Los operadores disponibles para que el usuario manipule estas tablas “por ejemplo, para fines de recuperación de datos” son operadores que derivan tablas a partir de tablas. En particular, tres de estos operadores son importantes:

restringir, proyectar y juntar (este último operador también es conocido como combinar o reunir).

Una base de datos relacional consiste en un conjunto de **tablas**, a cada una de las cuales se le asigna un nombre exclusivo. Cada fila de la tabla representa una *relación* entre un conjunto de valores. Dado que cada tabla es un conjunto de dichas relaciones, hay una fuerte correspondencia entre el concepto de *tabla* y el concepto matemático de *relación*, del que toma su nombre el modelo de datos relacional.

Estructura básica

Considérese la tabla *cuenta* del Gráfico N° 2.8. Tiene tres cabeceras de columna: *número-cuenta*, *nombre-sucursal* y *saldo*. Siguiendo la terminología del modelo relacional se puede hacer referencia a estas cabeceras como **atributos**. Para cada atributo hay un conjunto de valores permitidos, llamado **dominio** de ese atributo. Para el atributo *nombre-sucursal*, por ejemplo, el dominio es el conjunto de los nombres de las sucursales. Supongamos que $D1$ denota el conjunto de todos los números de cuenta, $D2$ el conjunto de todos los nombres de sucursal y $D3$ el conjunto de los saldos. Como se vio en el Capítulo 2 todas las filas de *cuenta* deben consistir en una tupla $(v1, v2, v3)$, donde $v1$ es un número de cuenta (es decir, $v1$ está en el dominio $D1$), $v2$ es un nombre de sucursal (es decir, $v2$ está en el dominio $D2$) y $v3$ es un saldo (es decir, $v3$ está en el dominio $D3$).

número-cuenta	nombre-sucursal	saldo
C-101	Centro	500
C-102	Navacerrada	400
C-201	Galapagar	900
C-215	Becerril	700
C-217	Galapagar	750
C-222	Moralzarzal	700
C-305	Collado Mediano	350

Gráfico N° 2.8 La relación *cuenta*

Fuente: Fuente: SILBERTSHATZ, A, KORTH, H, SUDARSHAN, S. (2002). *Fundamentos de Bases de Datos*. Madrid: McGraw Hill. p. 55.

En general, *cuenta* sólo contendrá un subconjunto del conjunto de todas las filas posibles. Por tanto, *cuenta* es un subconjunto de $D1 \times D2 \times D2$.

En general, una **tabla** de n atributos debe ser un subconjunto de $D1 \times D2 \times \dots \times Dn - 1 \times Dn$.

Los matemáticos definen las **relaciones** como subconjuntos del producto cartesiano de la lista de dominios. Esta definición se corresponde de manera casi exacta con la definición de tabla dada anteriormente. La única diferencia es que aquí se han asignado nombres a los atributos, mientras que los matemáticos sólo utilizan «nombres» numéricos, utilizando el entero 1 para denotar el atributo cuyo dominio aparece en primer lugar en la lista de dominios, 2 para el atributo cuyo dominio aparece en segundo lugar, etcétera. Como las tablas son esencialmente relaciones, se utilizarán los términos matemáticos **relación** y **tupla** en lugar de los términos **tabla** y **fila**. Una **variable tupla** es una variable que representa a una tupla; en otras palabras, una tupla que representa al conjunto de todas las tuplas.

En la relación *cuenta* del Gráfico N° 2.8 hay siete tuplas. Supóngase que la variable tupla t hace referencia a la primera tupla de la relación. Se utiliza la notación $t[\text{número-cuenta}]$ para denotar el valor de t en el atributo *número-cuenta*. Por tanto,

$$t[\text{número-cuenta}] = \text{C-101} \text{ y}$$

$$t[\text{nombre-sucursal}] = \text{Centro.}$$

De manera alternativa, se puede escribir $t[1]$ para denotar el valor de la tupla t en el primer atributo (*número-cuenta*), $t[2]$ para denotar *nombre-sucursal*, etcétera. Dado que las relaciones son conjuntos se utiliza la notación matemática $t \in r$ para denotar que la tupla t está en la relación r . El orden en que aparecen las tuplas es irrelevante, dado que una relación es un *conjunto* de tuplas. Así, si las tuplas de una relación se muestran ordenadas como en el Gráfico N° 2.8, o desordenadas, como en el Gráfico N° 2.9 no importa; las relaciones de estas figuras son las mismas, ya que ambas contienen el

mismo conjunto de tuplas.

Se exigirá que, para todas las relaciones r , los dominios de todos los atributos de r sean atómicos. Un dominio es **atómico** si los elementos del dominio se consideran unidades indivisibles. Por ejemplo, el conjunto de los enteros es un dominio atómico, pero el conjunto de todos los conjuntos de enteros es un dominio no atómico. La diferencia es que no se suele considerar que los enteros tengan subpartes, pero sí se considera que los conjuntos de enteros las tienen; por ejemplo, los enteros que forman cada conjunto. Lo importante no es lo que sea el propio dominio, sino la manera en que se utilizan los elementos del dominio en la base de datos. El dominio de todos los enteros sería no atómico si se considerase que cada entero fuera una lista ordenada de cifras. En todos los ejemplos se supondrá que los dominios son atómicos.

número-cuenta	nombre-sucursal	saldo
C-101	Centro	500
C-215	Becerril	700
C-102	Navacerrada	400
C-305	Collado Mediano	350
C-201	Galapagar	900
C-222	Moralzarzal	700
C-217	Galapagar	750

Gráfico N° 2.9 La relación *cuenta* desordenada

Fuente: Fuente: SILBERTSHATZ, A, KORTH, H, SUDARSHAN, S. (2002).
Fundamentos de Bases de Datos. Madrid: McGraw Hill. p. 56

Es posible que varios atributos tengan el mismo dominio. Por ejemplo, supóngase que se tiene una relación *cliente* que tiene los tres atributos *nombre-cliente*, *calle-cliente* y *ciudad-cliente* y una relación *empleado* que incluye el atributo *nombre-empleado*. Es posible que los atributos *nombre-cliente* y *nombre-empleado* tengan el mismo dominio, el conjunto de todos los nombres de personas, que en el nivel físico son cadenas de caracteres. Los dominios de *saldo* y *nombre-sucursal*, por otra parte, deberían ser distintos. Quizás es menos claro si *nombre-cliente* y *nombre-sucursal* deberían tener el

mismo dominio. En el nivel físico, tanto los nombres de clientes como los nombres de sucursales son cadenas de caracteres. Sin embargo, en el nivel lógico puede que se desee que *nombre-cliente* y *nombre-sucursal* tengan dominios diferentes. Un valor de dominio que es miembro de todos los dominios posibles es el valor **nulo**, que indica que el valor es desconocido o no existe. Por ejemplo, supóngase que se incluye el atributo *número-teléfono* en la relación *cliente*. Puede ocurrir que un cliente no tenga número de teléfono, o que su número de teléfono no figure en la guía. Entonces habrá que recurrir a los valores nulos para indicar que el valor es desconocido o que no existe. ^[4]

BASES DE DATOS DISTRIBUIDAS

Sistemas Dsitribuidos

En un sistema distribuido de bases de datos se almacena la base de datos en varias computadoras. Varios medios de comunicación, como las redes de alta velocidad o las líneas telefónicas, son los que pueden poner en contacto las distintas computadoras de un sistema distribuido. No comparten ni memoria ni discos. Las computadoras de un sistema distribuido pueden variar en tamaño y función pudiendo abarcar desde las estaciones de trabajo a los grandes sistemas.

Dependiendo del contexto en el que se mencionen existen diferentes nombres para referirse a las computadoras que forman parte de un sistema distribuido, tales como sitios o nodos. Para enfatizar la distribución física de estos sistemas se usa principalmente el término sitio. En el Gráfico N° 2.10 se muestra la estructura general de un sistema distribuido. Las bases de datos distribuidas normalmente se encuentran en varios lugares geográficos distintos, se administran de forma separada y poseen una interconexión más lenta, en un sistema distribuido se dan dos tipos de transacciones, las locales y las globales. Una transacción local es aquella que accede a los datos del único sitio en el cual se inició la transacción. Por otra parte, una transacción global es aquella que, o bien accede a los datos situados en un sitio diferente de aquel en el que se inició la transacción, o bien accede a datos de varios sitios distintos.

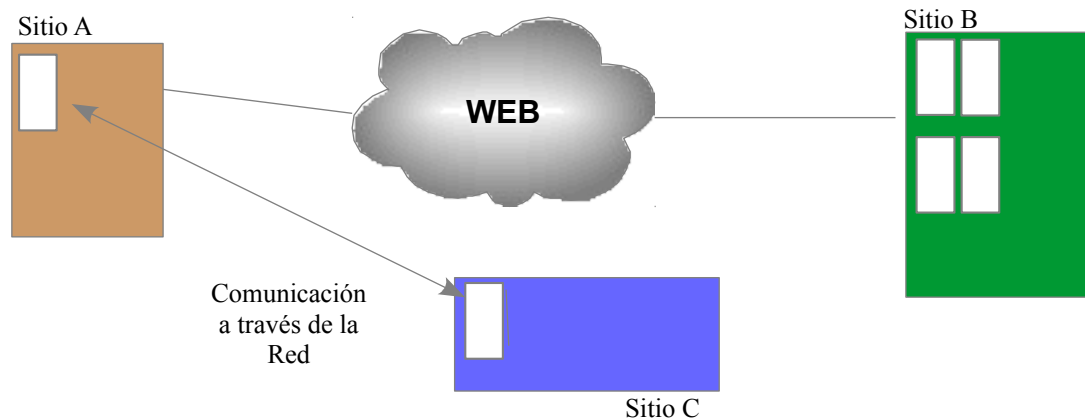


Gráfico N° 2.10 Sistema Distribuido

Fuente: SILVERTSHATZ, A. KORTH, H. SUDARSHAN, S. (2002). Fundamentos de Bases de Datos. Madrid: McGraw Hill. p. 456.

Una **base de datos distribuida** es una colección de datos que pertenece lógicamente al mismo sistema pero que está dispersa físicamente entre los sitios de una red de computadores. Varios factores han dado pie a la creación de los Sistemas de Bases de Datos Distribuidas (DDBS por sus siglas en inglés). Entre las ventajas potenciales de los DDBS están las siguientes:

- *La naturaleza distribuida de algunas aplicaciones de bases de datos:* Muchas de estas aplicaciones están *distribuidas naturalmente* en diferentes lugares. Es natural que las bases de datos empleadas en tales aplicaciones estén distribuidas en esos lugares. Muchos **usuarios locales** tienen acceso exclusivamente a los datos que están en el lugar, pero otros **usuarios globales** como la oficina central de la compañía pueden requerir acceso ocasional a los datos almacenados en varios de estos lugares. Las fuentes de los datos y la mayoría de los usuarios y aplicaciones de la base de datos local residen físicamente en ese lugar.
- *Mayor fiabilidad y disponibilidad:* Estas son dos de las ventajas potenciales de las bases de datos distribuidas que se citan más comúnmente. La **fiabilidad** se

define a grandes rasgos como la probabilidad de que un sistema esté en funciones en un momento determinado, y la **disponibilidad** es la probabilidad de que el sistema esté disponible continuamente durante un intervalo de tiempo. Cuando los datos y el software del DBMS están distribuidos en varios sitios, un sitio puede fallar mientras que los demás siguen operando. Sólo los datos y el software que residen en el sitio que falló están inaccesibles. Esto mejora tanto la fiabilidad como la disponibilidad. Se logran mejoras adicionales si se *replican*, con un criterio adecuado, los datos y el software en más de un sitio. En un sistema centralizado, el fallo de un solo sitio hace que el sistema completo deje de estar disponible para todos los usuarios.

- *Posibilidad de compartir los datos al tiempo que se mantiene un cierto grado de control local:* En algunos tipos de Sistema de Bases de datos Distribuidos (DDBS por sus siglas en inglés), es posible controlar los datos y el software localmente en cada sitio. Sin embargo, los usuarios de otros sitios remotos pueden tener acceso a ciertos datos a través del software de Sistemas Administradores de Bases de Datos Distribuidas (DDBMS por sus siglas en inglés). Esto hace posible el compartimiento *controlado* de los datos en todo el sistema distribuido.
- *Mejor rendimiento:* Cuando una base de datos grande está distribuida en múltiples sitios, hay bases de datos más pequeñas en cada uno de éstos. En consecuencia, las consultas locales y las transacciones que tienen acceso a datos de un solo sitio tienen un mejor rendimiento porque las bases de datos locales son más pequeñas. Además, cada sitio tiene un menor número de transacciones en ejecución que si todas las transacciones se enviaran a una sola base de datos centralizada. En el caso de transacciones que impliquen acceso a más de un sitio, el procesamiento en los diferentes sitios puede efectuarse en paralelo, reduciéndose así el tiempo de respuesta. La distribución produce un aumento en la complejidad del diseño y en la implementación del sistema. Para obtener las ventajas potenciales antes citadas, el software de DDBMS debe

contar con las funciones de un DBMS centralizado y *además* con las siguientes:

- ◆ La capacidad de tener acceso a sitios remotos y transmitir consultas y datos entre los diversos sitios a través de una red de comunicaciones.
- ◆ La capacidad de seguir la pista a la distribución y la replicación de los datos en el catálogo del DDBMS.
- ◆ La capacidad de elaborar estrategias de ejecución para consultas y transacciones que tienen acceso a datos de más de un sitio.
- ◆ La capacidad de decidir a cuál copia de un elemento de información replicado se tendrá acceso.
- ◆ La capacidad de mantener la consistencia de las copias de un elemento de información replicado.
- ◆ La capacidad de recuperarse de caídas de sitios individuales y de nuevos tipos de fallos, como el fallo de un enlace de comunicación.

Técnicas de fragmentación, replicación y reparto de los datos para el diseño de bases de datos distribuidas

Se puede dividir la base de datos en unidades lógicas, llamadas **fragmentos**, cuyo almacenamiento puede asignarse a los diversos sitios. Con la **replicación de los datos**, es posible almacenar ciertos datos en más de un sitio, y el proceso de **repartir** fragmentos o réplicas de fragmentos para almacenarse en los diferentes sitios. Estas técnicas se utilizan durante el proceso de **diseño de bases de datos distribuidas**. La información concerniente a la fragmentación de los datos, el reparto y la replicación se almacena en un **catálogo global del sistema** al que tiene acceso el software cliente cuando es necesario.

Fragmentación de los datos

En un DDBS, es preciso tomar decisiones respecto a los sitios en los que se almacenarán las porciones de la base de datos. Por ahora se supondrá que *no hay replicación*, esto es, que cada archivo o porción de archivo se almacenará en un solo sitio.

Antes de decidir cómo distribuir los datos, se debe determinar las *unidades lógicas* de la base de datos que se van a distribuir. Las unidades lógicas más simples son las relaciones mismas; es decir, cada relación *completa* se almacenará en un sitio específico. Se debe decidir en qué sitio almacenar cada una de las relaciones. En muchos casos, empero, es posible dividir una relación en unidades lógicas más pequeñas para su distribución.

Se considera la base de datos de una compañía, suponiendo que hay tres sitios de computador, uno para cada departamento de la compañía. Desde luego, en una situación real, habrá muchas más tuplas en las relaciones. Tal vez se quiera almacenar la información referente a cada departamento en el sitio de computador correspondiente a ese departamento. Para ello, se necesitará dividir cada relación mediante una técnica llamada fragmentación horizontal.

Fragmentación horizontal. Un fragmento horizontal de una relación es un subconjunto de las tuplas de esa relación. Las tuplas que pertenecen al fragmento horizontal se especifican mediante una condición sobre uno o más de los atributos de la relación. Con frecuencia, sólo interviene un atributo. Por ejemplo, se puede definir tres fragmentos horizontales.

La fragmentación horizontal divide una relación "horizontalmente" agrupando filas para crear subconjuntos de tuplas, donde cada subconjunto tiene un cierto significado lógico. Estos fragmentos pueden entonces asignarse a diferentes sitios en el sistema distribuido.

Fragmentación vertical. Un fragmento vertical de una relación mantiene sólo ciertos

atributos de la relación.

Esta fragmentación vertical no es del todo apropiada porque, si ambos fragmentos se almacenan por separado, no podremos juntar otra vez las tuplas, ya que *no existe un atributo común* entre los dos fragmentos. Es necesario incluir el *atributo de clave primaria* en *todo* fragmento vertical para que sea posible reconstruir la relación completa a partir de los fragmentos.

La fragmentación vertical divide una relación "verticalmente" por columnas. Observe que cada fragmento horizontal de una relación R se puede especificar con una operación $\sigma_{Q}(R)$ del álgebra relacional. Un conjunto de fragmentos horizontales cuyas condiciones C_1, C_2, \dots, C_n incluyen todas las tuplas de R esto es, todas las tuplas de R satisfacen $(C_1 \text{ OR } C_2 \text{ OR } \dots \text{ OR } C_n)$ se denomina **fragmentación horizontal completa** de R.

En muchos casos una fragmentación horizontal completa es además **disjunta**; es decir, ninguna tupla de R satisface $(C_i \text{ AND } C_j)$, para cualquier $i \neq j$. Para reconstruir la relación R a partir de una fragmentación horizontal *completa*, necesitamos aplicar la operación UNIÓN a los fragmentos.

Un fragmento vertical de una relación R puede especificarse con una operación $\pi_{L}(R)$ del álgebra relacional. Un conjunto de fragmentos verticales cuyas listas de proyección L_1, L_2, \dots, L_n incluyen todos los atributos de R, pero sólo comparten el atributo de clave primaria de R, se denomina **fragmentación vertical completa** de R. En este caso, las listas de proyección satisfacen estas dos condiciones:

- $L_1 \cup L_2 \cup \dots \cup L_n = \text{ATR}(R)$.
- $L_i \cap L_j = \text{CLP}(R)$ para cualquier $i \neq j$, donde $\text{ATR}(R)$ es el conjunto de atributos de R y $\text{CLP}(R)$ es la clave primaria de R.

Para reconstruir la relación R a partir de una fragmentación vertical *completa*, aplicamos la operación de UNIÓN EXTERNA a los fragmentos. Cabe señalar que

también podría aplicar la operación de REUNIÓN EXTERNA COMPLETA y obtener el mismo resultado que con una fragmentación vertical completa.

Fragmentación mixta. Podemos entremezclar los dos tipos de fragmentación, para obtener una fragmentación mixta. En este caso la relación original se puede reconstruir aplicando operaciones de UNIÓN y UNIÓN EXTERNA (o REUNIÓN EXTERNA) en el orden apropiado. En general, un **fragmento** de una relación R se puede especificar con una combinación de operaciones SELECCIONAR – PROYECTAR $wL(oc(R))$. Si $C = TRUE$ y $L = ATRS(R)$, obtendremos un fragmento vertical, y si $C \neq TRUE$ y $L = ATRS(R)$, obtendremos un fragmento horizontal. Por último, si $C \neq TRUE$ y $L \neq ATRS(R)$ obtendremos un fragmento mixto. Cabe señalar que una relación completa puede considerarse un fragmento con $C = TRUE$ y $L = ATRS(R)$. En el análisis que sigue, utilizamos el término *fragmento* para referirnos a una relación o a cualquiera de los tipos de fragmentos mencionados.

Un **esquema de fragmentación** de una base de datos es una definición de un conjunto de fragmentos que incluye *todos* los atributos y tuplas de la base de datos y satisface la condición de que la base de datos completa se puede reconstruir a partir de los fragmentos mediante alguna secuencia de operaciones UNIÓN EXTERNA (o REUNIÓN EXTERNA) y UNIÓN.

En ocasiones también es útil aunque no necesario que todos los fragmentos sean disjuntos, excepto la repetición de las claves primarias entre los fragmentos verticales (o mixtos). En este último caso, toda la replicación y distribución de los fragmentos se especifica claramente en una etapa subsecuente, aparte de la fragmentación.

Un **esquema de reparto** describe el reparto de fragmentos entre los sitios del DDBS; por tanto, es una correspondencia que especifica el sitio o sitios donde se almacena cada fragmento. Si un fragmento se almacena en más de un sitio, se dice que está **replicado**.

La replicación resulta útil para mejorar la disponibilidad de los datos. El caso más extremo es la replicación de *toda la base de datos* en todos los sitios del sistema

distribuido, creando así una base de datos distribuida **totalmente replicada**.

Esto puede mejorar la disponibilidad notablemente porque el sistema puede seguir operando mientras, por lo menos, uno de los sitios esté activo. También mejora el rendimiento de la obtención de datos en consultas globales, porque el resultado de semejante consulta se puede obtener localmente en cualquier sitio; así, una consulta de obtención de datos se puede procesar en el sitio local donde se introduce, si dicho sitio cuenta con un módulo servidor. La desventaja de la replicación completa es que puede reducir drásticamente la rapidez de las operaciones de actualización, pues una sola actualización lógica se deberá ejecutar con todas y cada una de las copias de la base de datos a fin de mantener la consistencia. Esto es especialmente cierto si existen muchas copias de la base de datos. Con la replicación completa, las técnicas de control de concurrencia y recuperación se tornan más costosas de lo que serían sin replicación.

El extremo opuesto a la replicación completa es **no tener ninguna replicación**; esto es, cada fragmento se almacena exactamente en un sitio. En este caso todos los fragmentos *deben* ser disjuntos, con excepción de la repetición de claves primarias entre los fragmentos verticales (o mixtos). Esto se denomina también **reparto no redundante**. Entre estos dos extremos, tenemos una amplia gama de **replicación parcial** de los datos; es decir, algunos fragmentos de la base de datos pueden estar replicados y otros no. El número de copias de cada fragmento puede ir desde una hasta el número total de sitios en el sistema distribuido. En ocasiones se llama **esquema de replicación** a una descripción de la replicación de los fragmentos. Cada fragmento o cada copia de un fragmento se debe asignar a un sitio determinado en el sistema distribuido. Este proceso se denomina **distribución de los datos** (o **reparto de los datos**). La elección de sitios y el grado de replicación dependen de los objetivos de rendimiento y disponibilidad para el sistema y de los tipos y frecuencias de transacciones introducidas en cada sitio. Por ejemplo, si se requiere una alta disponibilidad, y si las transacciones se pueden introducir en cualquier sitio y si la mayoría de ellas son de obtención de datos, entonces una base de datos completamente replicada será una buena opción. Sin embargo, si por lo regular ciertas transacciones que tienen acceso a partes específicas de la base de datos

se introducen en un solo sitio, se podría asignar el conjunto de fragmentos correspondiente exclusivamente a ese sitio. Los datos que se utilizan en múltiples sitios se pueden replicar en esos sitios. Si se efectúan muchas actualizaciones, puede ser conveniente limitar la replicación. Encontrar una solución óptima, o siquiera buena, para el reparto de datos distribuidos es un problema de optimización muy complejo.

Tipos de sistemas de bases de datos distribuidas

El término *sistema de gestión de bases de datos distribuidas* puede describir diversos sistemas que presentan muchas diferencias entre sí. El punto principal que todos estos sistemas tienen en común es el hecho de que los datos y el software están distribuidos entre múltiples sitios conectados por alguna especie de red de comunicaciones. En esta sección analizaremos varios tipos de DDBMS y los criterios y factores que distinguen a algunos de estos sistemas.

El primer factor que consideraremos es el grado de homogeneidad del software de DDBMS. Si todos los servidores (o DBMS locales individuales) utilizan software idéntico y todos los clientes emplean software idéntico, se dice que el DDBMS es homogéneo; en caso contrario se le caracteriza como heterogéneo. Otro factor relacionado con el grado de homogeneidad es el grado de autonomía local. Si todo acceso al DDBMS debe hacerse a través de un cliente, el sistema no tiene autonomía local. Por otro lado, si se permite a las transacciones locales *acceso directo* a un servidor, el sistema tendrá cierto grado de autonomía local.

En un extremo de la gama de autonomía tenemos un DDBMS que da al usuario la impresión de ser un DBMS centralizado. Sólo hay un esquema conceptual, y todo acceso al sistema se hace a través de un cliente, de modo que no hay autonomía local. En el otro extremo nos encontramos con un tipo de DDBMS denominado DDBMS federado (o sistema de multibases de datos). En un sistema así, cada servidor es un DBMS centralizado independiente y autónomo que tiene sus propios usuarios locales, transacciones locales y DBA, y por ende posee un alto grado de *autonomía local*. Cada

servidor puede autorizar el acceso a porciones específicas de su base de datos definiendo un esquema de exportación, el cual especifica la parte de la base de datos a la cual puede tener acceso una cierta clase de usuarios no locales.

En esencia, un cliente en un sistema así es una interfaz adicional para varios servidores (DBMS locales) que permite a un usuario de multibases de datos (o global) tener acceso a información almacenada en varias de estas *bases de datos autónomas*. Observe que un sistema federado es un híbrido entre los sistemas distribuidos y los centralizados; es un sistema centralizado para los usuarios autónomos locales y es un sistema distribuido para los usuarios globales.

En un sistema heterogéneo de multibases de datos, un servidor puede ser un DBMS relacional, otro un DBMS de red y un tercero un DBMS jerárquico; en tal caso, es necesario contar con un lenguaje de sistema canónico e incluir traductores del lenguaje en el cliente a fin de traducir las subconsultas del lenguaje canónico al lenguaje de cada servidor.

Un tercer aspecto que puede servir para clasificar las bases de datos distribuidas es el grado de transparencia de la distribución o, de manera alternativa, el grado de integración de los esquemas. Si el usuario percibe un solo esquema integrado sin información alguna relativa a la fragmentación, replicación o distribución, se dice que el DDBMS tiene un *alto grado de transparencia de distribución* (o de integración de esquemas). Por otro lado, si el usuario puede ver toda la fragmentación, el reparto y la replicación, el DBMSD no *tiene transparencia de distribución* ni integración de esquemas. En este caso, el usuario debe hacer referencia a copias específicas de fragmentos en sitios específicos cuando formule una consulta, anexando el nombre del sitio como prefijo de cada nombre de relación o de fragmento.

Esto forma parte del complejo problema de dar nombres en los sistemas distribuidos. Hay casos en que el DDBMS no ofrece transparencia de distribución; si es así, es responsabilidad del usuario *especificar sin ambigüedad* el nombre de una copia de

relación o de fragmento en particular. La tarea es más difícil en un sistema de multibases de datos, porque es de suponer que cada servidor (DBMS local) se creó de manera independiente, y en consecuencia es muy posible que se hayan usado nombres conflictivos en diferentes servidores. En el caso de un DDBMS que provee un esquema integrado, en cambio, la asignación de nombres se convierte en un problema interno del sistema, porque el usuario percibe un solo esquema sin ambigüedades. El DDBMS debe almacenar en el catálogo de distribución *todas las correspondencias* entre los objetos del esquema integrado y los objetos distribuidos en los diversos procesadores de bases de datos. ^[5]

Podemos expresar las categorías de la variable independiente como: Los fundamentos, el alcance, la seguridad en respaldo de información, la fragmentación y replicación de las Bases de datos Distribuidas, así como los modelos de las mismas.

Las categorías de la variable dependiente podemos resumirlas como: el desempeño de los servidores de almacenamiento de bases de datos, la estructura de la red de datos así como la comunicación, la identificación de los modelos de datos, su dispersión y redundancia.

Podemos resumir esto en los siguientes gráficos que se presentan a continuación.

CATEGORÍAS DE LA VARIABLE INDEPENDIENTE

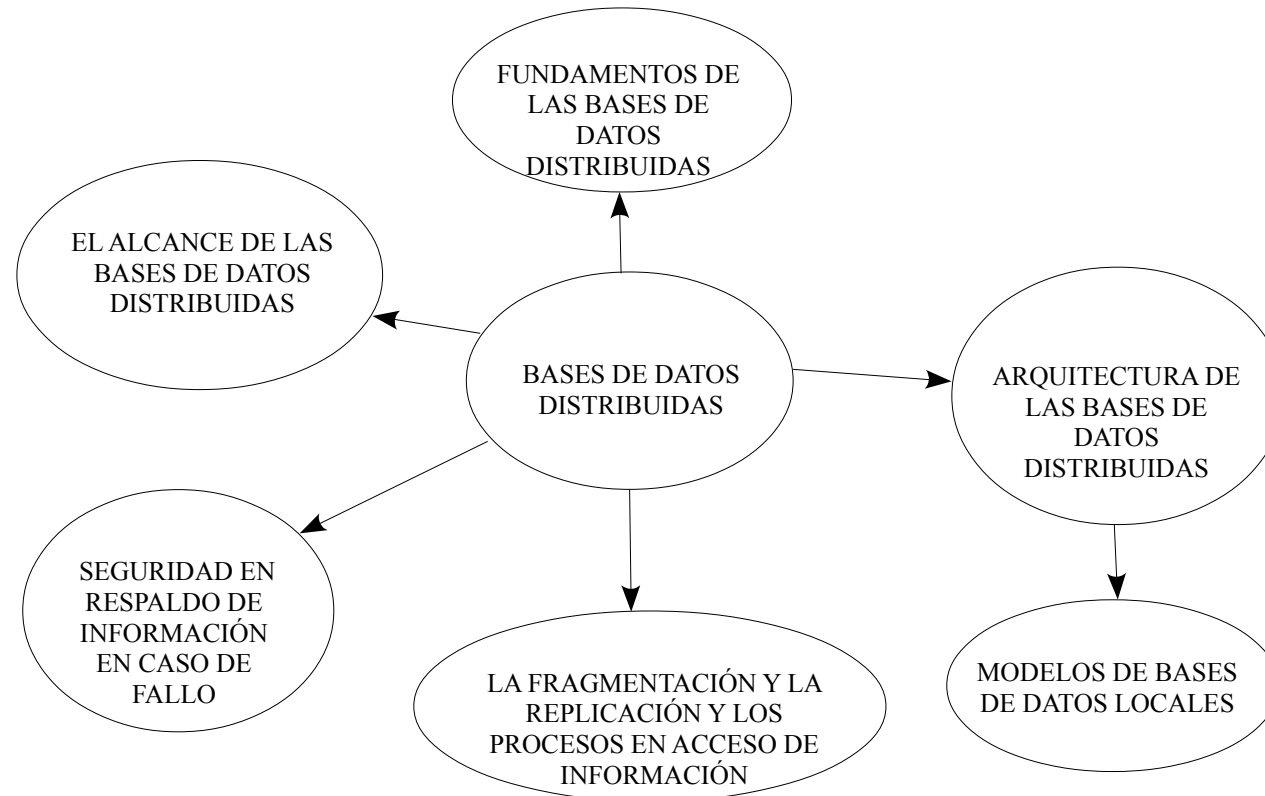


Gráfico N° 2.11: Subcategorías de la VI
Elaborado por: Investigador

CATEGORÍAS DE LA VARIABLE DEPENDIENTE

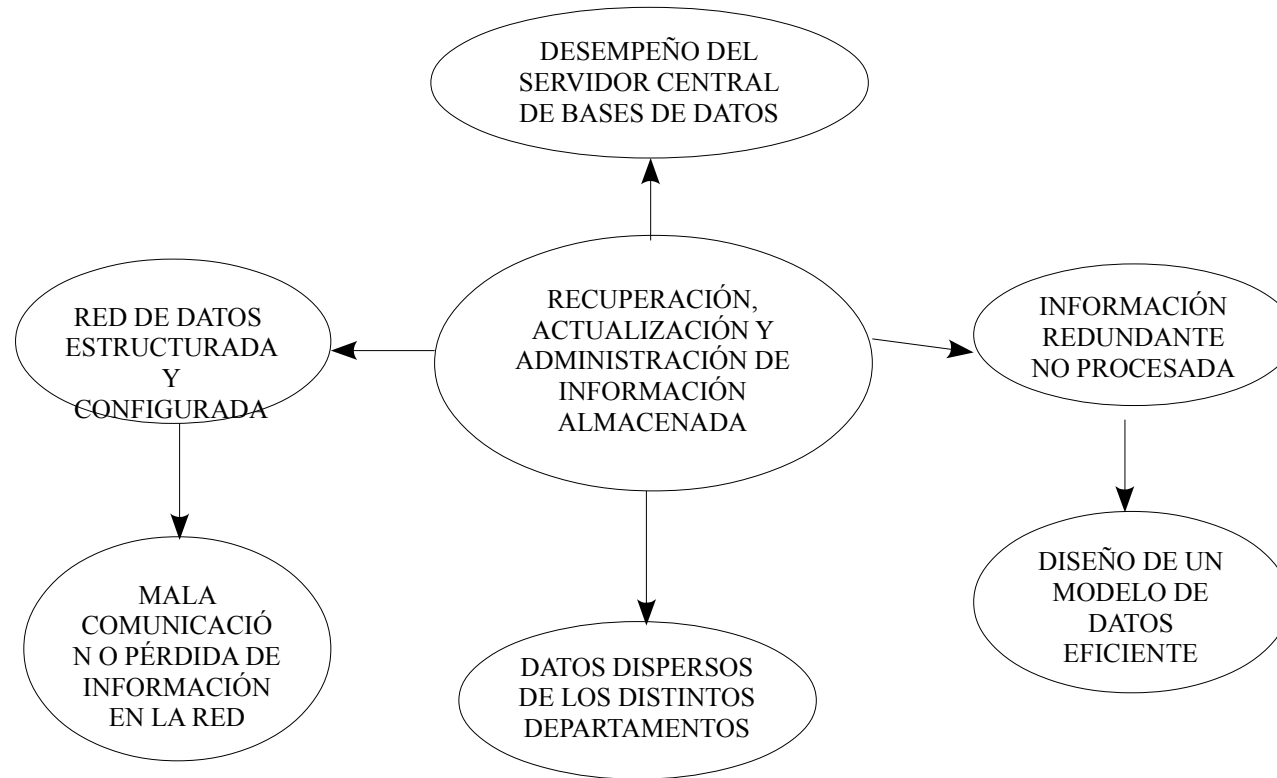


Gráfico N° 2.12: Subcategorías de la VD
Elaborado por: Investigador

HIPÓTESIS DE LA PREGUNTA DIRECTRIZ

El Análisis y la evaluación de diferentes herramientas para Bases de Datos Distribuidas incidirán significativamente en la recuperación, actualización y administración de la información del Centro Educativo Diocesano San Pío X en el período educativo 2011

SEÑALAMIENTO DE LAS VARIABLES DE LA HIPÓTESIS

Variable Independiente

Bases de Datos Distribuidas

Variable Dependiente

Recuperación, Actualización, y Administración de la Información

Término de Relación

Incidencia

CAPÍTULO III

METODOLOGÍA

ENFOQUE

La presente investigación tendrá un enfoque de investigación de Ciencias Sociales ya que permite utilizar varios métodos predominantemente cualitativos, que se caracterizan por buscar la comprensión de los fenómenos sociales, a través de una observación naturalista cuyas conclusiones son contextualizadas, además está orientado al descubrimiento de las hipótesis para llegar a propuestas concretas, enfatiza en el proceso y asume una realidad dinámica.

MODALIDAD BÁSICA DE INVESTIGACIÓN

Investigación de campo

La investigación tendrá la modalidad de campo porque el investigador acudirá al lugar en donde se producen los hechos para obtener información relacionada con los objetivos del trabajo de grado.

Investigación Documental Bibliográfica

Con el propósito de fortalecer la investigación se recurrirá a obtener información teórica de diferentes autores obtenidas en fuentes secundarias, libros revistas especializadas publicaciones, internet, otros y de ser necesario fuentes de información primario a través de documentos válidos y confiables.

Proyectos factibles de Intervención Social

El trabajo de grado responde a un proyecto factible de intervención social porque se planteara una propuesta viable de solución al problema investigado dentro de un contexto determinado.

Proyectos especiales

Con la finalidad de dar solución a los problemas de un contexto determinado la investigación tiene la modalidad de proyectos especiales porque con la utilización de la tecnología se implementará un software libre; de ser el caso, orientado al almacenamiento y manejo de información distribuida en el CED San Pío X en el periodo educativo 2011 -2012 para el respaldo y acceso de información confiable.

NIVEL O TIPO DE INVESTIGACIÓN

Exploratorio

La investigación pasará por el nivel de investigación exploratorio porque sondeará un problema poco investigado o desconocido en un contexto determinado, especialmente para definir el tema de investigación

Descriptivo

Es descriptivo porque se buscará informar los resultados obtenidos de la investigación entre la comparación de dos variables tomando en cuenta criterios de coherencia interna y pertinencia

Asociación de variables

Porque permite establecer y estudiar las tendencias de comportamiento entre variables en un contexto determinado

POBLACIÓN Y MUESTRA

La población universo esta conformado por 13 personas distribuidos de la siguiente manera:

Encuestados	Cantidad
Autoridades	3
Administrativos	7
Sistemas	3

Cuadro N° 1 Población y Muestra
Elaborado por: Investigador

Se utilizó para la investigación toda la población debido a que su número es inferior a 100

Las variables se operacionalizan en los cuadros N° 2 y N° 3 de la s siguientes páginas.

OPERACIONALIZACIÓN DE VARIABLES

Variable independiente

Variable: Bases de Datos Distribuidas

CONCEPTUALIZACIÓN	DIMENSIONES	INDICADORES	ÍTEMS BÁSICOS
<p>Bajo el nombre de distribución de datos se engloba todo un conjunto de técnicas encaminadas al almacenamiento masivo de datos en distintas computadoras mostrándolas como una sola</p>	<p>Conjunto de técnicas</p>	<p>BDD</p> <ul style="list-style-type: none"> • Homogéneas • Heterogéneas 	<p>¿Cómo se puede determinar la técnica más conveniente para validar el proceso de un modelo exitoso?</p>
	<p>Extracción de conocimiento procesable</p>	<p>fragmentación Replicación Transparencia</p> <p>Bases de Datos en sistema</p>	<p>¿Contribuye significativamente la distribución de datos en múltiples aplicaciones del conocimiento?</p>

Cuadro N° 2 Operacionalización de la Variable Independiente
Elaborado por Investigador

Variable dependiente

Variable: Recuperar, Actualizar y Administrar la Información del Centro Educativo Diocesano San Pío X

CONCEPTUALIZACIÓN	DIMENSIONES	INDICADORES	ÍTEMS BASICOS	TÉCNICAS E INSTRUMENTOS
Se dará seguridad al recuperar actualizar y administrar la información distribuida en caso de falla del servidor central	Recuperacion y Actualización de Información	Por usuario Por departamento Por nodo de red Por demanda	¿Es posible crear a un repositorio general de datos lo más automatizada posible para el manejo de la información?	Encuesta Cuestionario
	Información distribuida en caso de fallos	Por usuario Por departamento	¿Se pueden determinar la información más utilizadas en función de los registro ingresados en los distintos departamentos?	Entrevista Guía de la entrevista

Cuadro N° 3 Operacionalización de la Variable Dependiente
Realizado por: Investigador

PLAN DE RECOLECCIÓN DE INFORMACIÓN

Técnicas e instrumentos

Encuesta:

Dirigido a los departamentos de, inspecciones, DOBE, biblioteca cuyo instrumento es un cuestionario elaborado con preguntas cerradas para recabar información sobre la información que genera y requiere de los alumnos y profesores.

Entrevista

Dirigido a (vicerrector, secretaria) cuyo instrumento será la guía de la entrevista elaborado con preguntas abiertas

Validez y confiabilidad

La validez de los instrumentos vendrá dado por la “técnica de juicios de los expertos” mientras que la confiabilidad se lo hará aplicando una prueba piloto a una muestra pequeña para detectar errores y corregirlos a tiempo.

PLAN DE PROCESAMIENTO DE INFORMACIÓN

Tabulación de los datos registrados en los cuestionarios hechos en los distintos departamentos del personal administrativo, autoridades y sistemas del Centro Educativo San Pío X.

CAPÍTULO IV

ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

Pregunta N° 1. ¿Conoce Usted como se almacena la información de los alumnos de la Institución?

Respuesta	Frecuencia	Porcentaje
SI	8	61.54
No	5	38.46

Cuadro N° 4

Fuente: Encuestas
Elaborado por: Investigador

¿Conoce Uste como se almacena la Información de los Alumnos en la Intitución?

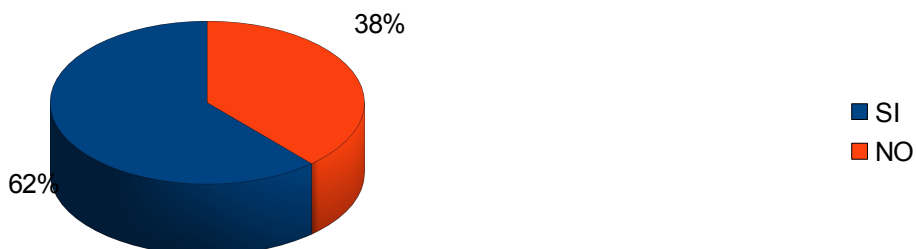


Gráfico N° 4.1

Fuente: Encuestas
Elaborado por: Investigador

Mediante las encuestas realizadas al personal se obtiene que el 62% de ellos sabe como se almacena la información mientras que un 38% no sabe como se hace dicho procedimiento, en ete caso se puede apreciar que hay desconocimiento de como se almacena la información en la institución.

Pregunta N° 2. ¿Conoce si los equipos de computo con los que cuenta la institución son los apropiados para el almacenamiento seguro de información?

Respuesta	Frecuencia	Porcentaje
SI	6	46.15
No	7	53.85

Cuadro N° 5
Fuente: Encuestas
Elaborado por: Investigador

2.¿Conoce si los equipos de computo con los que cuenta la institución son los apropiados para el almacenamiento seguro de información?

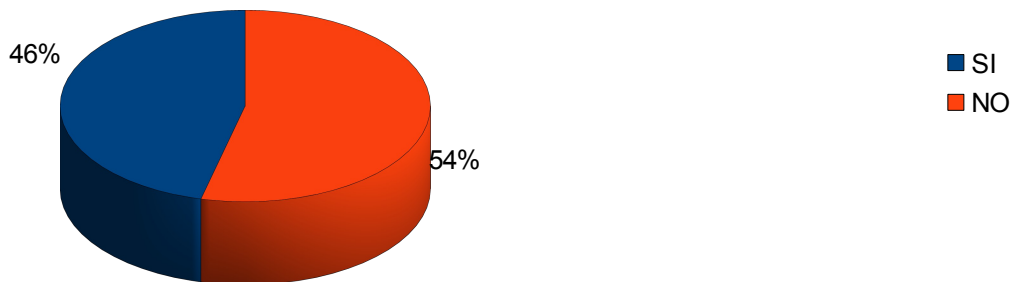


Gráfico N° 4.2
Fuente: Encuestas
Elaborado por: Investigador

Mediante las encuestas realizadas al personal se obtiene que el 45% de ellos sabe que los equipos con los que cuenta la institución son los apropiados mientras que un 54% no sabe con que equipos cuenta y si son los apropiados para este proceso, en ete caso se puede apreciar que se ha hecho una buena inversión en equipos de cómputo para almacenar la información en la institución.

Pregunta N° 3. ¿Los equipos de la institución cuenta con un respaldo de energía en caso de cortes o alteraciones de tensión inprevistos?

Respuesta	Frecuencia	Porcentaje
SI	3	23.08
No	10	76.92

Cuadro N° 6

Fuente: Encuestas

Elaborado por: Investigador

3.¿Los equipos de la institución cuenta con un respaldo de energía en caso de cortes o alteraciones de tensión inprevistos?

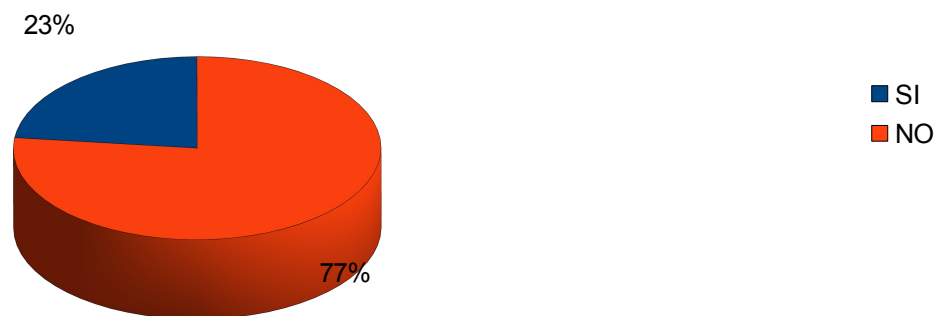


Gráfico N° 4.3

Fuente: Encuestas

Elaborado por: Investigador

Mediante las encuestas realizadas al personal se obtiene que el 15% de ellos sabe que los equipos con los que cuenta la institución cuenta con un sistema de respaldo de energía, mientras que un 85% desconoce de este particular, en este caso se puede apreciar que no han previsto cortes o alteraciones de energía los cuales podrían afectar el buen funcionamiento de los equipos de la institución.

Pregunta N° 4. ¿Conoce Uste si existe algun procedimiento para el respaldo de información en la institución en caso de corte de energía?

Respuesta	Frecuencia	Porcentaje
SI	4	30.77
No	9	69.23

Cuadro N° 7

Fuente: Encuestas

Elaborado por: Investigador

3.¿Conoce Uste si existe algun procedimiento para el respaldo de información en la institución en caso de corte de energía?

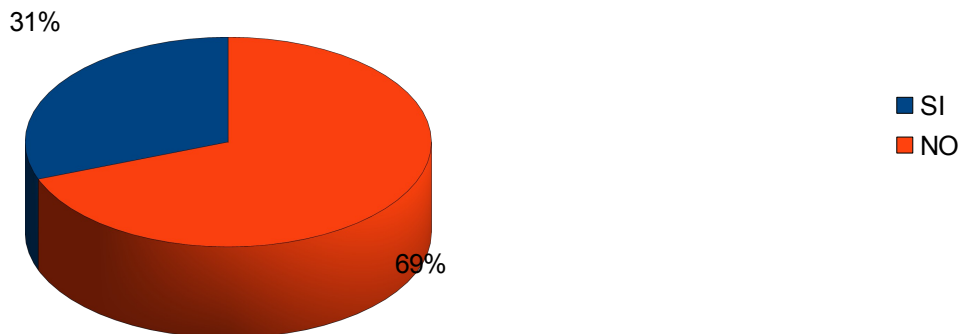


Gráfico N° 4.4

Fuente: Encuestas

Elaborado por: Investigador

Mediante las encuestas realizadas a las Autoridades se obtiene que el 31% de ellos sabe que la información se respalda de forma esporádica, un 69% no sabe que se realiza el procedimiento de respaldo de información, en este caso se puede apreciar que no se han coordinado y definido una política para asegurar la información que se genera en la institución.

Pregunta N° 5. ¿En caso de deterioro o daño de un equipo de computo de la institución, existe algun procedimiento para la recuperación de la información?

Respuesta	Frecuencia	Porcentaje
SI	3	23.08
No	10	76.92

Cuadro N° 8

Fuente: Encuestas

Elaborado por: Investigador

5.¿En caso de deterioro o daño de un equipo de computo de la institución, existe algun procedimiento para la recuperación de la información?

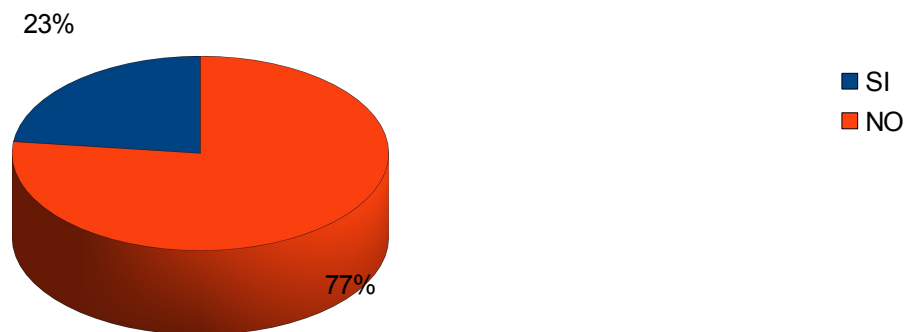


Gráfico N° 4.5

Fuente: Encuestas

Elaborado por: Investigador

Mediante las encuestas realizadas al personal se obtiene que el 19% de ellos sabe como recuperar la información que se haya respaldado en caso de daño del equipo principal y un 81% desconoce que exista un proceso de recuperación de información, en este caso se puede apreciar que no saben como manejar una contingencia de este tipo y como poner un proceso temporal de captación de información hasta recuperar los datos del o los equipos averiados.

Pregunta N° 6. ¿Se beneficiará la empresa con la implementación de bases de datos distribuidas para la recuperación o respaldo de información?

Respuesta	Frecuencia	Porcentaje
SI	8	61.54
No	5	38.46

Cuadro N° 9

Fuente: Encuestas

Elaborado por: Investigador

¿Se beneficiará la empresa con la implementación de bases de datos distribuidas para la recuperación o respaldo de información?

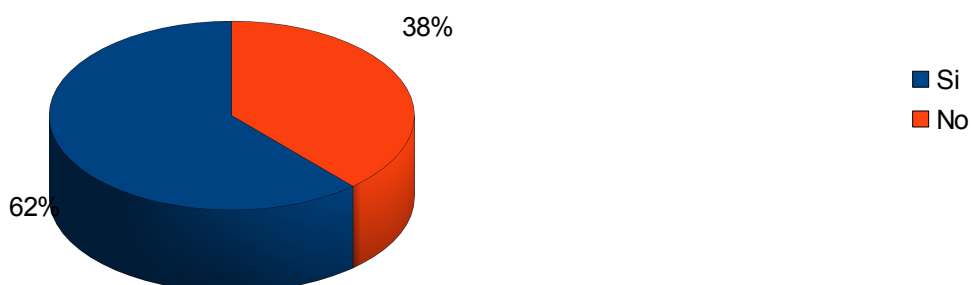


Gráfico N° 4.6

Fuente: Encuestas

Elaborado por: Investigador

Mediante las encuestas realizadas al personal se obtiene que un 38% cree que se beneficiará la institución implementando bases de datos distribuidas, otro 62% considera que no se beneficiará,, en este caso hay falta de información, capacitación y comunicación por parte del personal de sistemas sobre el uso de la tecnología de bases de datos distribuidas, sus ventajas y beneficios.

Pregunta N° 7. ¿Cree usted que una implementación de bases de datos distribuidas ayude a minimizar el tiempo en la recuperación de información en caso de fallo?

Respuesta	Frecuencia	Porcentaje
SI	11	84.62
No	2	15.38

Cuadro N° 10

Fuente: Encuestas

Elaborado por: Investigador

¿Cree usted que una implementación de bases de datos distribuidas ayude a minimizar el tiempo en la recuperación de información en caso de fallo?

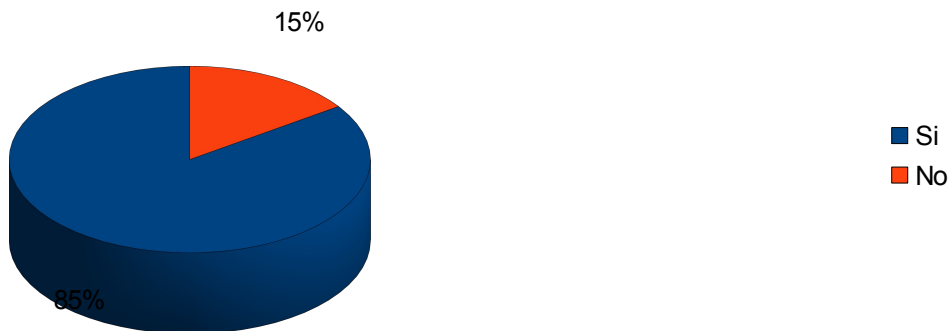


Gráfico N° 4.7

Fuente: Encuestas

Elaborado por: Investigador

Mediante las encuestas realizadas al personal se obtiene que el 85% de ellos considera que las bases de datos distribuidas minimizará el tiempo de recuperación de la información en caso de fallo, un 15% cree que el tiempo será mayor al deseado para recuperar información de una base de datos distribuida en caso de fallo, en este caso se puede ver una buena disposición del personal a cambiar el método y los procesos de respaldo mediante bases de datos distribuidas y como recuperar la información en un lugar determinado.

Pregunta N° 8. ¿Cree Usted que la distribución de datos mejorará el rendimiento de la red y el acceso a la información?

Respuesta	Frecuencia	Porcentaje
SI	6	46.15
No	7	53.85

Cuadro N° 11

Fuente: Encuestas

Elaborado por: Investigador

¿Cree usted que la distribución de datos mejorará el rendimiento de la red y el acceso a la información?

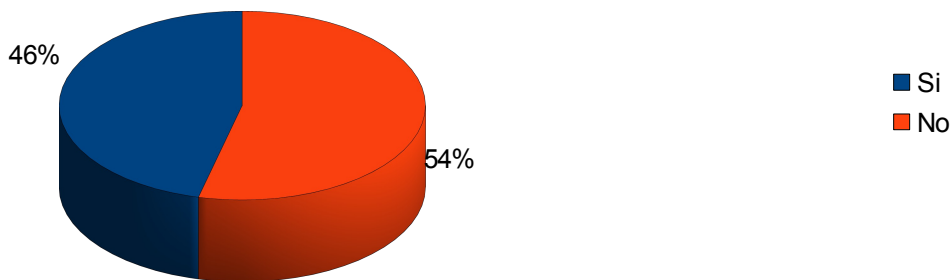


Gráfico N° 4.8

Fuente: Encuestas

Elaborado por: Investigador

Mediante las encuestas realizadas al personal se tiene que el 46% de ellos considera que las bases de datos distribuidas mejorará el acceso de los datos al sistema y bajará considerablemente el tráfico en la red, y un 54% considera que el tráfico de la red se mantendrá como esta actualmente y el acceso seguirá siendo lento, para este caso es claro para el personal que teniendo bases de datos locales que se reflejen cada cierto tiempo a una base de datos central no minimizará el tiempo de acceso a la información que cada cliente requiera, por tanto el tráfico de la red no disminuirá en forma considerablemente.

Pregunta N° 9. ¿Cree usted que las bases de datos distribuidas es un buen mecanismo de seguridad de la información en caso de un ataque de seguridad?

Respuesta	Frecuencia	Porcentaje
SI	7	53.85
No	6	46.15

Cuadro N° 12

Fuente: Encuestas

Elaborado por: Investigador

¿Cree usted que las bases de datos distribuidas es un buen mecanismo de seguridad de la información en caso de un ataque de seguridad?

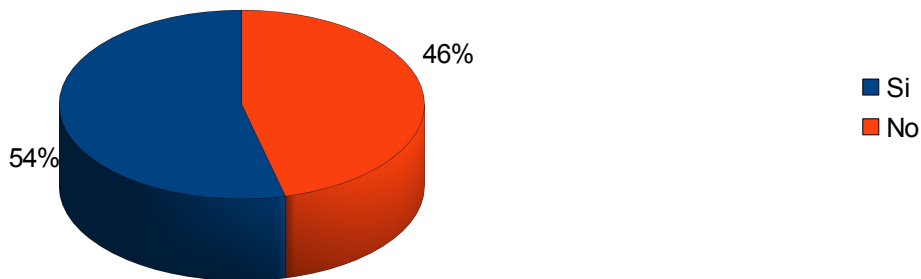


Gráfico N° 4.9

Fuente: Encuestas

Elaborado por: Investigador

Mediante las encuestas realizadas al personal se tiene que el 54% de ellos considera que las bases de datos distribuidas es un buen mecanismo de seguridad, en caso de ataque, y un 46% desconoce que se pueda realizar un ataque a los equipos por parte de piratas informáticos, para este caso solo cierto personal está conciente que se puede estar expuesto a un ataque informático y que la información es suceptble a perderse, la mayoría desconoce como se podría llevar a cabo y como las bases de datos distribuidas pueden ayudar a recuperar la información si fuera este el caso

Pregunta N° 10. ¿Cree Usted que la Fragmentación de Datos afectará en la coherencia y consistencia de la información?

Respuesta	Frecuencia	Porcentaje
SI	10	76.92
No	3	23.08

Cuadro N° 13

Fuente: Encuestas

Elaborado por: Investigador

¿Cree Usted que la Fragmentación de Datos afectará en la coherencia y consistencia de la información?

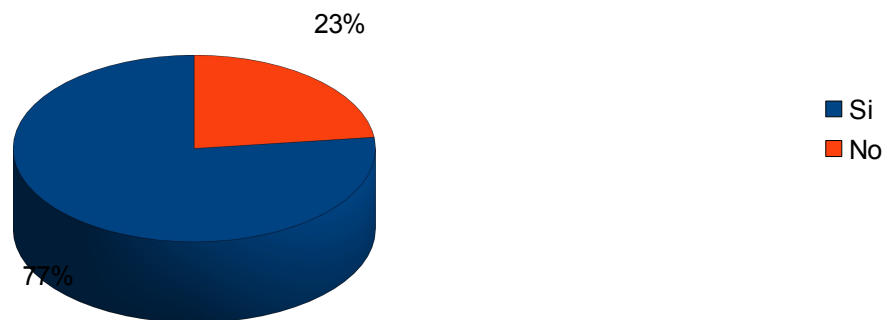


Gráfico N° 4.10

Fuente: Encuestas

Elaborado por: Investigador

Mediante las encuestas realizadas al personal de sistemas se tiene que el 77% de ellos considera que la fragmentación de datos si afectará a la coherencia y consistencia de la información y un 23% considera que la fragmentación de información, no afectará la información que se va a replicar desde varias bases de datos a una sola base de datos central, tenga coherencia entre los mismos.

Pregunta N° 11. ¿Cree Uste que los recursos de Hardware y Software con los que cuenta la Institución son suficientes para una implementación de Bases de Datos Dsistribuidas?

Respuesta	Frecuencia	Porcentaje
SI	9	69.23
No	4	30.77

Cuadro N 14

Fuente: Encuestas
Elaborado por: Investigador

¿Cree Uste que los recursos de Hardware y Software con los que cuenta la Institución son suficientes para una implementación de Bases de Datos Dsistribuidas?

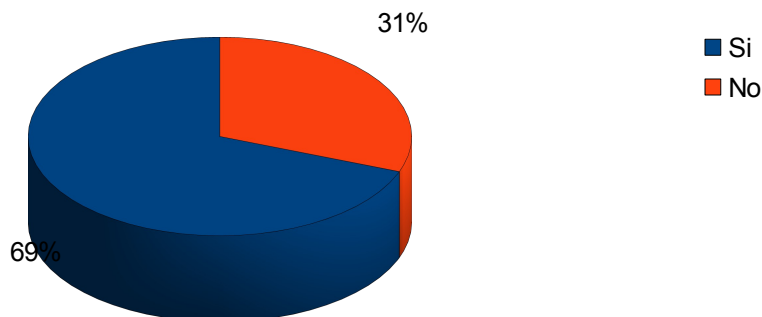


Gráfico N° 4.11

Fuente: Encuestas
Elaborado por: Investigador

Mediante las encuestas realizadas al personal se tiene que el 69% de ellos considera que tanto los sistemas operativos, y equipos son lo bastante buenos para poder hacer una implementación de bases de datos distribuidas, mientras que el 31% considera que se debería hacer actualizaciones periódicas de equipos y que si es necesario hacer instalaciones de respaldo de energía o configuración de redes, se haran según sean los requerimientos para un óptimo desempeño.

VERIFICACIÓN DE LA HIPÓTESIS

Modelo Lógico:

Formulación de la hipótesis

- H_0 = Hipótesis nula
- H_1 = Hipótesis alterna

H_0 = El Análisis y la evaluación de diferentes herramientas para Bases de Datos Distribuidas **NO** incidirán significativamente en la recuperación, actualización y administración de la información del Centro Educativo Diocesano San Pío X

H_1 = El Análisis y la evaluación de diferentes herramientas para Bases de Datos Distribuidas **SI** incidirán significativamente en la recuperación, actualización y administración de la información del Centro Educativo Diocesano San Pío X

Modelo Matemático:

Modelo Estadístico:

Las pruebas t de student es una prueba estadística para evaluar hipótesis con muestras pequeñas (menos de treinta casos), el caso de la determinación de t_t se lo hace en base de grados de libertad..

Prueba de Hipótesis:

Elección de la prueba estadística

Para la verificación de la hipótesis se escogió la prueba estadística de t de student para muestras relacionadas.

Simbología

\bar{X} = Valor Promedio

gl = Grados de libertad

σ = Desviación Estándar (Anexo)

t = Valor t de Student

α = Nivel de Significación

El nivel de significación escogido para la investigación es del 5%.

$t_i = \alpha = 0.05$

$gl = n-1 \Rightarrow 11 - 1 = 10$

$t_i = \pm 1.81$ segun la tabla de *t* de Student

Regla de Decisión: Se acepta la H_1 , es decir si hay incidencia de las bases de datos distribuidas en la recuperación, actualización y administración de la información si el valor de *t* a calcularse se encuentra en el intervalo de -1.81 a 1.81

Cuadro de distribución estadística

CUESTIONARIO PREGUNTA	Respuesta		X	X - \bar{X}	(X - \bar{X}) ²	
	SI	NO				
1	8	5	3	2.36	5.59	
2	6	7	-1	-1.64	2.68	
3	3	10	-7	-7.64	58.31	
4	4	9	-5	-5.64	31.77	
5	3	10	-7	-7.64	58.31	
6	8	5	3	2.36	5.59	
7	11	2	9	8.36	69.95	
8	6	7	-1	-1.64	2.68	
9	7	6	1	0.36	0.13	
10	10	3	7	6.36	40.50	
11	9	4	5	4.36	19.04	
			$\bar{X} =$	0.64	$\Sigma =$	294.55

Cuadro N° 15 Distribución estadística

Elaborado por: Investigador

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n}$$

Formula N° 1
Calculo de la Media
Aritmética

$$\bar{X} = \frac{7}{11}$$

$$\bar{X} = 0.64$$

$$\bar{X} = 0.64$$

$$\mu = 0$$

$$\sigma = \frac{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2}}{n-1}$$

Formula N° 2
Calculo de la
Desviación Estandar

$$\sigma = \frac{\sqrt{294.55}}{11}$$

$$\sigma = 5.17$$

$$t = \frac{\bar{X} - \mu}{\frac{\sigma}{\sqrt{n}}}$$

Formula N° 3
Calculo del valor de
t de Student

$$t = \frac{0.64 - 0}{\frac{5.17}{\sqrt{11}}}$$

$$t = 0.41$$

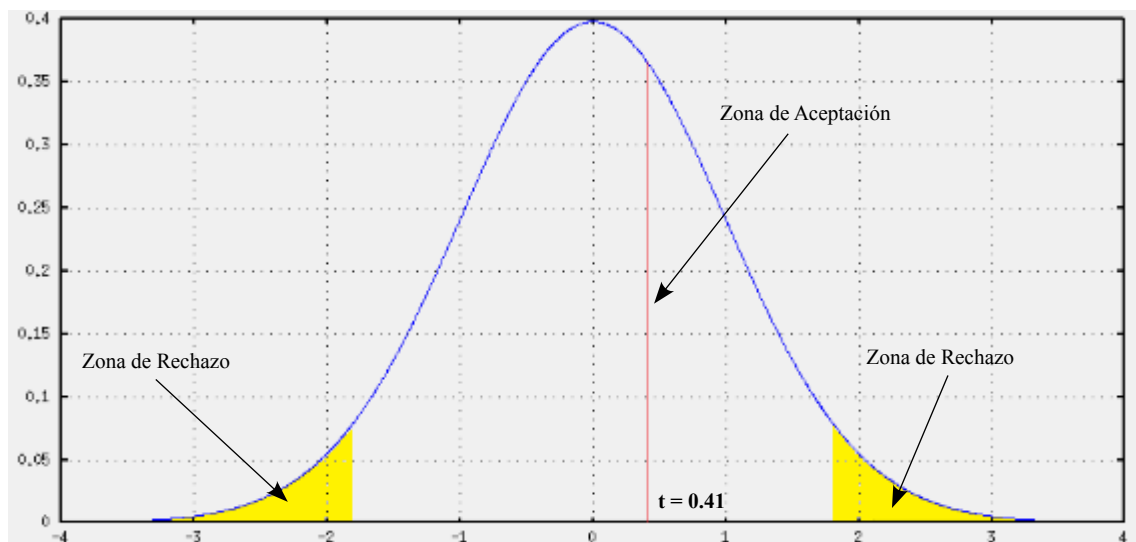


Gráfico N° 4.12 Curva de t de student – Comprobación de Hipótesis

Fuente: <http://www.google.com.ec/imgres?>

q=Curva+de+t+de+Student&um=1&hl=es&biw=1600&bih=794&tbm=isch&tbnid=J5w9eWpd3lyv5M:
&imgrefurl=http://electronicayciencia.blogspot.com/2010/09/inferencia-estadistica-ii-
introduccion.html&docid=IGxwPRAjxn8zDM&imgurl=http://4.bp.blogspot.com/_QF4k-
mng6_A/TJDcbu6jUGI/AAAAAAAAAYE/Id_DW4VIAu8/s1600/t_normalizada_m1m2.png&w=865&h
=408&ei=eS_ZT7jsCNCYhQeYt_jzAw&zoom=1&iact=hc&vpx=147&vpy=319&dur=2389&hovh=154
&hovw=327&tx=236&ty=82&sig=113806513197113439477&page=1&tbnh=85&tbnw=180&start=0&n
dsp=32&ved=1t:429,r:8,s:0,i:91

Decisión:

El valor de $t = 0.41 > t_{\alpha} -1.81 \wedge < 1.81$

Se encuentra dentro del intervalo Por consiguiente se acepta la hipótesis alterna, es decir, que El Análisis y la evaluación de diferentes herramientas para Bases de Datos Distribuidas SI incidirán significativamente en la recuperación, actualización y administración de la información del Centro Educativo Diocesano San Pío X.

Y se rechaza la hipótesis nula.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

Concluido el trabajo de campo y confrontados los resultados con el marco teórico estructurado se ha podido verificar la hipótesis planteada de acuerdo al análisis de las encuestas se han llegado a las siguientes conclusiones

- En el Centro Educativo Diocesano San Pío X, los equipos (servidores de bases de datos, concentradores de señales, clientes) no cuentan con un sistema de respaldo de energía que pueda soportar un corte o alteración de tensión eléctrica que pueda provocar pérdida de información vital para la institución
- El Software (Sistema Operativo) del Servidor principal no cuenta con las suficientes seguridades de acceso a usuarios, tampoco cuenta con un sistema de control de virus informáticos volviendose muy vulnerable al acceso de cualquier índole
- No hay un procedimiento formal para realizar respaldos periódicos de información, teniendo una única base de datos, que se respalda esporádicamente en el mismo equipo servidor, quedando estos archivos expuestos a ser borrados accidental o intencionalmente o copiados a facilidad por terceros.
- No existe una política o un manual de contingencia para la recuperación de datos en el caso de una fallo de los equipos en los que se almacena la información por perdida de energía, o por borrado accidental o intencional de las bases de datos.

Recomendaciones

- Se recomienda implementar bases de datos distribuidas para mantener la información de la institución integrada y coherente, de fácil acceso y manipulación.

- Se recomienda implementar una arquitectura de replicación para bases de datos homogéneas que integren los departamentos de Colecturía, Secretaría, Inspección, Vicerrectorado, DOBE, y Biblioteca con el siguiente esquema:
- Se hace necesario aplicar una replicación de tipo asíncrona con intervalos de tiempo definidos cortos.
- Se deberá generar archivos de configuración que sean fácilmente modificables que permitan mantener una distribución y replicación de la información de los diferentes departamentos a integrarse.
- Se recomienda implementar el siguiente esquema de distribución de Bases de Datos

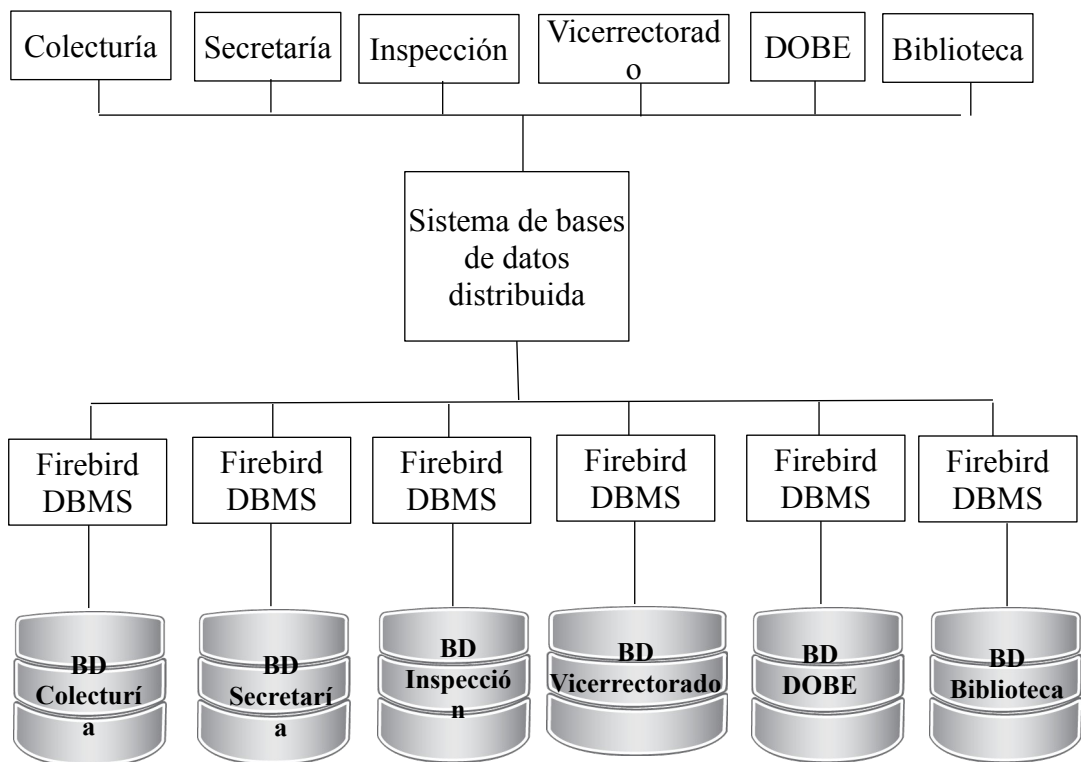


Gráfico N° 5.1 Esquema de Base de Datos Distribuidas

Elaborado por: Investigador

CAPÍTULO VI

LA PROPUESTA

DATOS INFORMATIVOS

La siguiente propuesta se realizara en el “Centro Educativo Diocesano San Pío X” ubicado en el barrio de Atocha de la Ciudad de Ambato.

El CED San Pío X es una institución educativa sin fines de lucro con mas de 30 años de servicio a la comunidad del centro del país formando bachilleres de la república cuya misión es la excelencia académica.

ANTECEDENTES DE LA PROPUESTA

El CED San Pío X al generar información muy importante como el rendimiento e historial académico de sus alumnos lo ha estado controlando con un software académico desarrollado bajo sus requerimientos y que hace algunos años lo ha manejando de forma regular.

El principal problema que se ha encontrado en los últimos 5 años ha sido la poca seguridad en el mantenimiento de un flujo de información constante debido a que el servicio eléctrico de nuestro país en ciertas épocas del año no ha sido permanente, existiendo cortes de luz, unas veces programadas y otras inesperadas.

Si bien es cierto que la información no se perdió debido a la robustez de la base de datos, eso provocaba que los equipos de computo como el servidor principal vaya sufriendo deterioro y daño a largo plazo, lo cual ocurrió en el período lectivo 2009 – 2010, deteniendo todo el proceso de matriculación que se desarrollaba en ese momento.

Esta situación se presenta ya que la institución no cuenta con un sistema y políticas de respaldo y acceso seguro a la información, ya que los métodos actuales no son suficientes para poder simplificar o corregir el problema.

ESTADO DEL ARTE

Luego de haber realizado la investigación acerca del problema podemos darnos cuenta que la situación actual del CED San Pío X no posee un sistema que soporte la utilización de bases de Datos Distribuidas, debido a esto la información es llevada y manejada en un solo servidor de Bases de datos con arquitectura Cliente – Servidor por lo que no existe un procedimiento manual ni automático de respaldo de información.

JUSTIFICACIÓN

Es **importante** realizar esta propuesta de implementar Bases de Datos Distribuidas y mecanismos alternativos de copia y respaldo de información de la base de datos principal en servidores alternos y/o remotos con el propósito de mantener, recuperar y levantar la información en caso de un fallo general, como corte de energía o daño permanente del equipo servidor de base de datos que soporta el acceso de usuarios internos y externos garantizando un acceso rápido a los datos almacenados en el mismo.

El proponer la implementación de Bases de Datos Distribuidas al CED San Pío X, dará la oportunidad de permitir acceder y generar información en forma local en cada uno de los departamentos sin tener que depender de un servidor principal, o esperar que este sea reparado si ha habido algún fallo para poder revisar los datos requeridos.

Es importante realizar esta propuesta ya que la Distribución de Bases de Datos Distribuidas permitirá crear ciertas políticas de seguridad de información y normalizar los procedimientos de respaldo y recuperación de la misma. Se debe tomar en cuenta que las Bases de Datos Distribuidas ayudarán a una administración de la información mucho mas eficiente minimizando la utilización de recursos.

Los beneficios que aportará la la implementación de Bases de Datos Distribuidas favorecerá a los usuarios de los diferentes departamentos ya que el tráfico de la red se minimizará ostensiblemente, evitando de esa manera pérdida de tiempo en acceso de información específica, además, las solución se adaptan a las necesidades y requisitos

específicos de la institución.

OBJETIVOS

Objetivo General

- Implementar un sistema de replicación freeware para configurar y administrar Bases de Datos Distribuidas en el Centro Educativo Diocesano San Pío X en el período 2011.

Objetivos Específicos

- Evaluar dos aplicaciones de tipo freeware y código abierto de replicación de bases de datos distribuidas que se adapte a las necesidades de la institución.
- Generar y mantener un archivo de replicación para cada uno de los nodos de los diferentes departamentos a integrarse.
- Tener un conocimiento básico de la herramienta de replicación seleccionada, y el sistema operativo en el que se instala el DBMS.
- Seleccionar un sistema de respaldo de energía UPS para mantener el servidor de Bases de Datos central replicado con alimentación constante y se mantenga encendido y funcionando de forma confiable y segura.

ANÁLISIS DE FACTIBILIDAD

El Centro Educativo Diocesano San Pío X cuenta con los recursos necesarios para implementar la propuesta de solución a los problemas de acceso, respaldo y recuperación de información detallado a continuación:

- Sistema académico transaccional para el control de notas, matrículas, pensiones, e inspecciones.
- Base de datos open source Firebird con licencias IPL (Interbase Public Licence)

- Servidor Principal HP Proliant 350 G5 procesador Core 2 Duo Xeon
- Equipos clientes en los distintos departamentos donde se replicará la información necesaria que le corresponda administrar.
- Capacitación al personal que estará a cargo de seguir el manual de procedimientos y recuperar la información en caso de fallo.

FUNDAMENTACIONES

Filosófica Para realizar la investigación el investigador se ubica en el paradigma filosófico crítico propositivo porque cuestiona la manera de hacer investigación y por el contrario plantea una propuesta de solución al problema investigado basado en la existencia de múltiples realidades socialmente construidas.

Además se utilizará el método de investigación comparativo normativo ya que en el análisis normativo el criterio principal es evaluativo como: la satisfacción, la utilidad, el rendimiento, el costo, etc. Y el objetivo del estudio es precisar el mejor de dos tipos de software de replicación para bases de datos distribuidas, uno de ellos de tipo libre sin acceso a su código fuente, y el otro también de tipo libre pero con acceso a su código fuente (open source).

Cabe mencionar que se busco la posibilidad de integrar en el estudio un tercer software de pago de licencia, pero no existiendo demos de tiempo limitado y solamente funcionales con la adquisición de las mismas y debido a los altos costos en la adquisición de los permisos (en este caso licencias de uso), para realizar las diferentes evaluaciones y comparaciones no ameritaba hacer este tipo de inversión con la posibilidad de no obtener los resultados deseados en el proceso, convirtiéndose en un gasto infructuoso.

En base a la experiencia adquirida en el análisis de los problemas se ha planteado la siguiente metodología para la implementación de un sistema de replicación de bases de datos distribuidas la cual está compuesta por las siguientes fases:

Definición del Proyecto

Análisis de fuentes internas y externas

- Evaluación de la aplicación.- Comparación de las dos alternativas en base a criterios como: rendimiento, conectividad, soporte de tipos de datos, facilidad de configuración, tiempos de respuesta etc.
- Instalación y puesta en marcha.- Implementación del software elegido para el proceso de replicación.

- Alternativas en respaldo de seguridad.- Varios procedimientos alternativos de respaldo de información por medio de sistema operativo o la propia base de datos.
- Procedimientos de recuperación.- Manual de proceso para restaurar copias de seguridad después de un fallo grave de la base de datos principal.

Científico Técnico

Esta propuesta está basada en el análisis comparativo normativo para determinar la mejor opción de software de replicación para bases de datos distribuidas, la metodología utilizada se detalla a continuación:

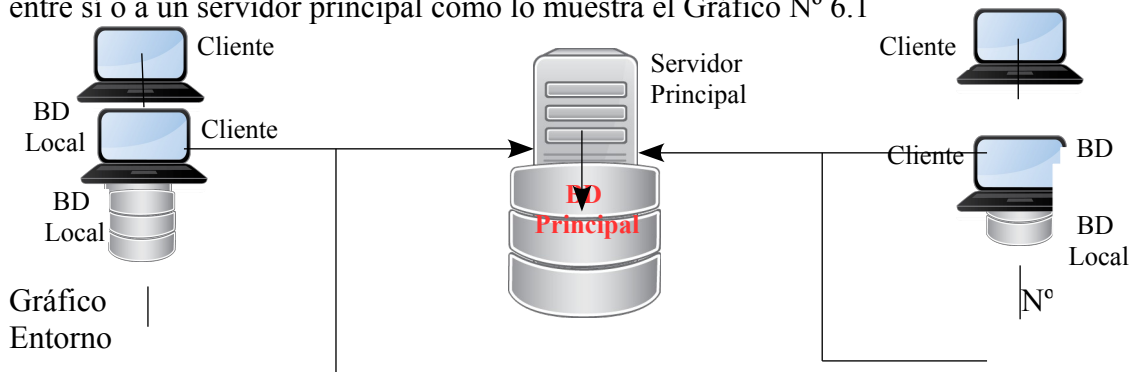
Análisis Comparativo Normativo

Definimos 7 criterios de comparación para la evaluación y aprobación del software de replicación que mejor se ajuste a las necesidades de la institución los cuales son:

- Tipo de replicación
- Rendimiento
 - ◆ Tiempo de respuesta en conexión
 - ◆ Tiempo de respuesta en mapeo de estructura de datos
 - ◆ Tiempo de respuesta en generación automática de código (triggers, generadores, etc)
 - ◆ Tiempo de respuesta en replicación
 - ◆ Tiempo de respuesta en recolección de errores
- Compatibilidad de tipos de datos
- Fragmentación de Datos
- Facilidad de Configuración
- Reacción en corte de señal
- Integración con el sistema operativo

Modelo Operativo

La representación básica de un sistema de bases de datos distribuidas está orientado a tener varios clientes remotos a través de una red local o a través del internet, conectadas entre sí o a un servidor principal como lo muestra el Gráfico N° 6.1



Distribuido

Elaborado por: Investigadosr

Plan de Acción

Tipo de Replicación

Las aplicaciones escogidas son compatibles con los DBMS Firebird e Interbase. Para este caso hay dos tipos básicos de replicación: síncronos y asíncronos

Con la replicación sincrónica, todas las copias o réplicas de los datos se mantienen exactamente sincronizada y coherente. Si cualquier copia se actualiza los cambios se aplicarán inmediatamente a todos los otras bases de datos en la misma transacción. La replicación síncrona es apropiada cuando esta consistencia exacta es importante para la aplicación empresarial.

Con asíncrona, o almacenar y replicar hacia adelante, las copias de los datos se vuelven temporalmente fuera de sincronización con los demás. Si una copia se actualiza, el cambio se propagará y se aplica a las copias de otros, como segundo paso, dentro de las transacciones independientes que pueden aparecer en segundos, minutos, horas o incluso días más tarde. Copias por lo tanto, puede estar temporalmente fuera de sincronía, pero con el tiempo los datos deben converger con los mismos valores en todos los sitios.

Software Replicador	Tipo de Replicación	
	Síncrona	Asíncrona
AdaReplicator	X	X

Cuadro N° 16 Tipos de Replicación

Realizado por: Investigador

Para este caso la replicación es de tipo Homogenea es decir BD1 Firebird – BD2 Firebird o BD1 Interbase – BD2 Interbase. Sería posible un tipo de replicación Heterogenea BD1 Firebird – BD2 Interbase o BD1 Interbase – BD2 Firebird siempre y cuando los tipos de datos en ambas BD sean los mismos, es decir exista compatibilidad en los mismos, caso contrario no se puede dar dicho tipo de replicación.

Rendimiento

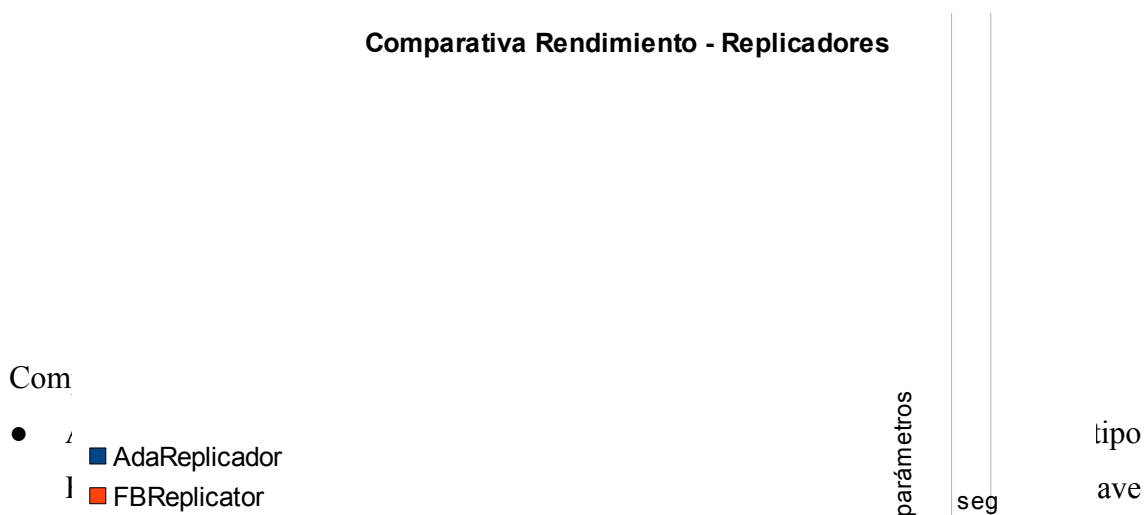
Item	Medidor	Software	
		AdaReplicador	FBReplicator
1	Tiempo de respuesta en conexión	2.25	1.85
2	Tiempo de respuesta en mapeo de estructura de datos	11.78	14.25
3	Tiempo de respuesta en generación automática de código (triggers, generadores, etc)	39.75	29.73
4	Tiempo de respuesta en replicación (promedio 20000 registros)	230	265
5	Tiempo de respuesta en recolección de errores	50	38.95

Se hicieron las pruebas tomando el tiempo en cuenta en segundos

Cuadro N° 17 Comparativa de Rendimiento

Elaborado por: Investigador

Comparativa Rendimiento - Replicadores



que pueden ser solamente de tipo entero o caracter o una combinación de los mismo.

- *FBReplicator*.- Permite replicar todos los tipos de datos inclusive datos de tipo Blob que no excedan los 128000 bytes, la definición de los campos clave pueden ser unicamente de tipo smallint, integer, y bigint mas no de tipo caracter, y vienen definidos por defecto como los campos clave de la tabla a replicarse, si poderse definir como el usuario desee, limitando posibles combinaciones.

Fragmentación de Datos

Cuadro N° 18 T

Software Replicador	Fragmentación		
	Horizontal	Vertical	Mixta
AdaReplicador	X	X	-

Realizado por: Investigador

Facilidad de Configuración

Se tomó en cuenta:

- Tipo de archivo de configuración
 - ◆ *AdaReplicador*.- Genera y utiliza un archivo propio de extensión .arpl en el cual se definen los esquemas de replicación como el Publicador (origen) y el Subscriptor (destino), tambien permite definir si se es replicación unidireccional o bidireccional., se valida el acceso a las BD super usuario de la BD, en este caso SYSDBA y la clave respectiva.
 - ◆ *FBReplicator*.- Utiliza un archivo de Base de Datos Interbase/Firebird de nombre *replicate.fdb*; proporcionada por el fabricante. En esta BD se definen los esquemas de replicación: Publicador y Subscriptor como registros de una tabla. Definiendo la ubicación de las bases de datos, el Super Usuario de conexión, el usuario de replicación, y el puerto de conexión que presta el servicio.
- Usuarios específicos para la replicación
 - ◆ *AdaReplicador*.- Requiere que se cree un usuario replicador de nombre ARPL y se le otorgue los permisos respectivos de acceso a las tablas que se van a replicar desde el origen al destino o viceversa en forma manual.

- ◆ *FBReplicator*.- Requiere que se cree un usuario replicador de nombre REPL, al igual que el anterior programa es necesario otorgar los permisos respectivos sobre los objetos a replicar (tablas y campos) de forma manual o mediante un programa de administración de bases de datos.
- Interfaces de Configuración.
 - ◆ *AdaReplicador*.- Utiliza dos aplicaciones el **AdaReplicador – Gerenciador de Configuraciones** en el cual se genera el archivo de configuración y los respectivos parámetros para la replicación, esta aplicación genera los distintos triggers asociados a configuración de la replicación y los crea dentro de las bases de datos tanto de origen como destino los cuales, también crea dentro de la base de datos origen dos tablas: la tabla ARPL\$LOG la cual almacena todos los registros generados por los triggers de inserción, actualización y borrado, estos registros son borrados automáticamente luego de que se realiza el proceso y si la replicación es exitosa. Se crea también la tabla ARPL\$SEPARADOR que guarda los nombres de las tablas que están siendo replicadas.

El **AdaReplicador – Servidor de Replicación**, que es un programa residente que se ejecuta cuando arranca el sistema operativo (Windows XP o Windows 7) y en el cual debemos definir que archivo de configuración debe utilizar para que trabaje la replicación, en este programa podemos definir el intervalo de tiempo para realizar la replicación (desde un segundo – hasta horas o días (expresado en segundos), se puede configurar un servidor de correo para hacer llegar un informe de errores si se diera el caso de que ocurra en un momento dado.

Se puede configurar de una lista de posibles errores, el presentarlos o ignorarlos dentro del proceso de replicación.

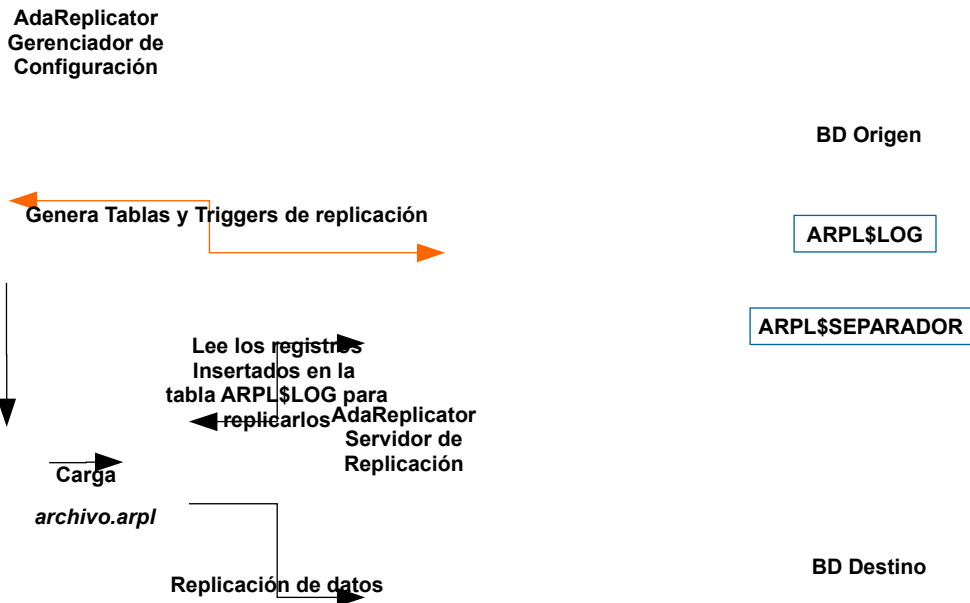
- ◆ *FBReplicator*.- Utiliza también dos aplicaciones el **FBReplication Manager** que es similar al software anterior salvo que este registra su configuración

dentro de una base de datos en tres diferentes tablas. La tabla que guarda el origen de los datos, la tabla que guarda las definiciones de las acciones a realizar, en el caso de la replicación selección, inserción, actualización y borrado, adicionalmente se crea una tabla en la base de datos origen llamada REPL_TABLES que guarda los nombres de las tablas a replicarse, crea también una tabla llamada CHANGES en la que guarda que tabla se afectó y el tipo de transacción mediante un código clave (inserción, actualización o borrado), estos registros son borrados inmediatamente después de que el proceso de replicación se ejecuta exitosamente.

Similar al otro sistema también cuenta con una aplicación llamada **FBReplicator Manager** la cual es instalada como un servicio de windows y es configurable similar a la anterior con la diferencia de que los intervalos de replicación se definen en minutos, en este programa se puede definir el número de errores que se puede tolerar sin que se detenga la replicación, crea un archivo de registro llamado ***FBReplicator.log***, también se puede definir envío de archivo de errores por correo configurando los datos del servidor de email.

- Esquema de trabajo de AdaReplicador

AdaReplicador genera un archivo de configuración *.arpl* en el cual se define si la



replicación va a ser unidireccional o bidireccional, se configura los publicadores y suscriptores, así como los intervalos de replicación.

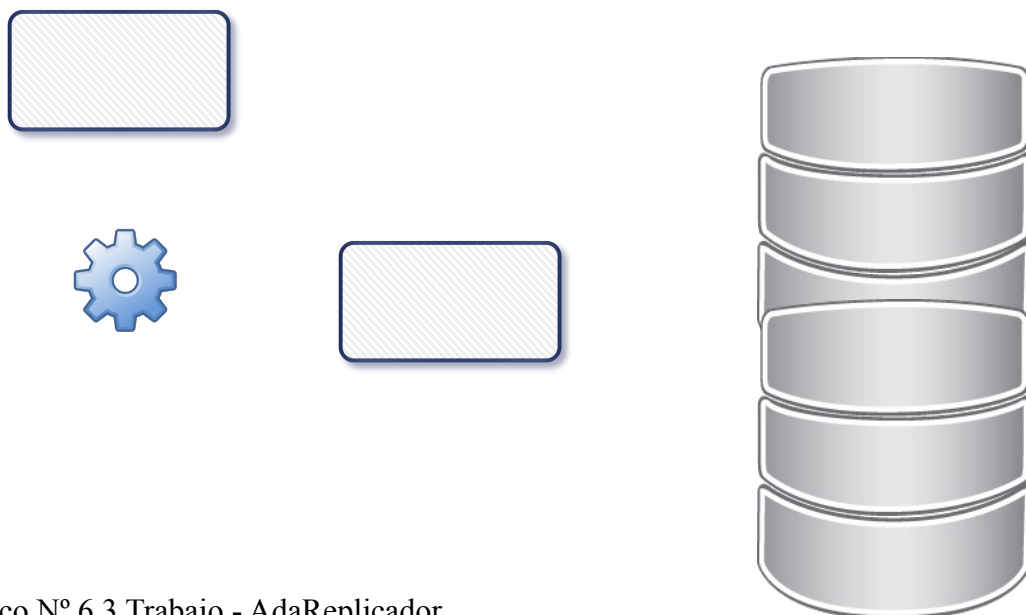


Gráfico N° 6.3 Trabajo - AdaReplicador
Realizado por: Investigador

- Esquema de Trabajo de FBReplicator

FBReplicator maneja un archivo de base de datos Firebird llamado *replicate.fdb* del cual hay una plantilla cuando se instala el software, en esta base de datos se registra el publicador o publicadores y el suscriptor o suscriptores, los intervalos de replicación se configuran en el mismo FBReplicator Manager.

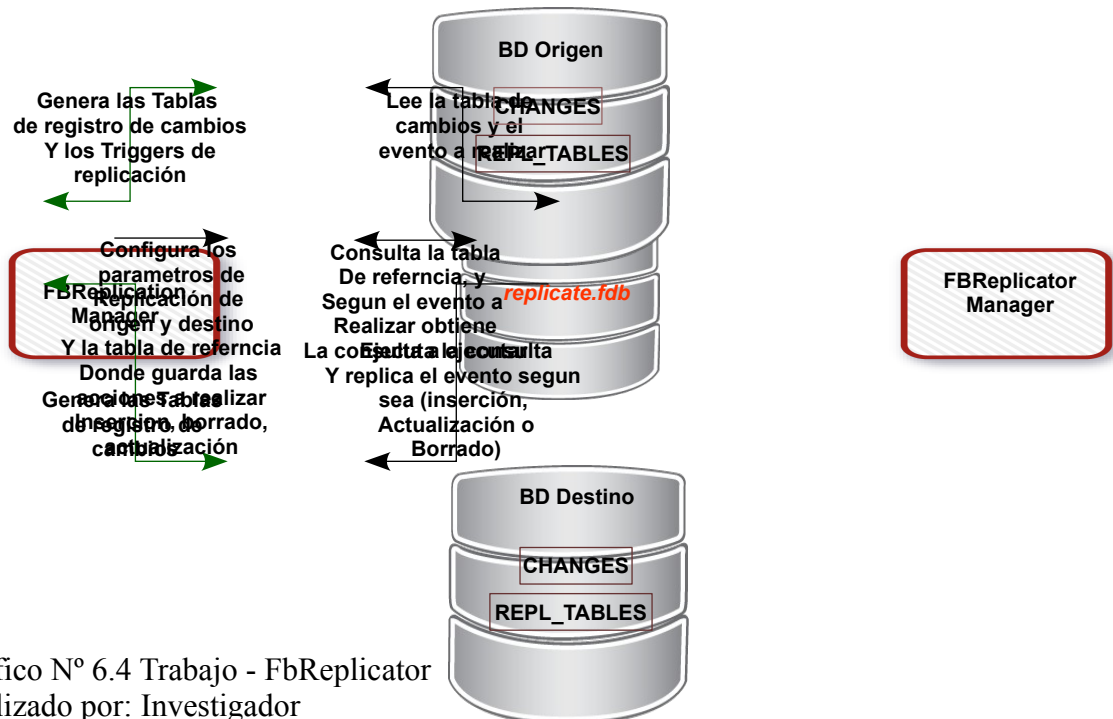


Gráfico N° 6.4 Trabajo - FbReplicator
Realizado por: Investigador

Reacción en corte de señal

Tanto *AdaReplicador* como *FBReplicator* trabajan de la misma forma, al crearse los triggers en los tres tipos de eventos (insert, update y delete) estos van guardando tanto en la tabla **ARPL\$LOG** como en la tabla **CHANGES** los registros que se vayan insertando, modificando o borrando de las tablas previamente configuradas a replicar, al momento de existir un corte de señal por cualquier motivo que este fuera (corte de energía, fallo en la red suspensión del servicio, etc. Los trigger seguirán trabajando normalmente hasta que se habilite el servicio de replicación.

Cuando este servicio se restaura *AdaReplicador* ejecuta su acción esperando el intervalo de configuración envía los datos hacia la base de datos destino, y se borran de la tabla de configuración **ARPL\$LOG**, mientras que *FBReplicator* lo ejecuta instantáneamente después de levantar el servicio y borra los datos de la tabla **CHANGES** de la BD

Origen.

Integración con el sistema operativo

Estos dos sistemas de replicación se integran solamente con Sistemas Operativos Windows.

AdaReplicador instala un sistema que se ejecuta al iniciar computador, el cual carga el archivo de configuración que lo hayamos definido por defecto, y trabaja en modo silencioso (background). Cabe mencionar que para Sistemas Operativos Windows XP service pack 2 y anteriores es necesario actualizar al service pack 3, y adicionalmente instalar el Microsoft .NET Framework 2.0

FBReplicator desarrollado en Delphi es de tipo open source con algunas restricciones, aunque el código es accesible, su compilación y posterior desarrollo es muy complicada ya que depende de objetos VCL (Visual Component Library) que si bien es cierto se pueden encontrar en internet, algunos son de pago y varían entre versión y versión haciendo muy difícil hacer un seguimiento para realizar posibles mejoras. *FbReplicator* instala un servicio en el sistema operativo que se puede configurar su ejecución desde el programa de servicios de Windows XP o desde una aplicación que se instala en Panel de Control.

En esta aplicación es posible configurar los tiempos de intervalo, emails de alertas etc. Que ya se mencionaron previamente.

Administración

Elección e implementación de la herramienta a Utilizar.

Luego de relizar este estudio comparativo se llegó a la conclusión de que la herramienta que mas se ajusta a los requerimientos actuales del integración de Bases de datos Distribuidas es *AdaReplicador* debido principalmente a la facilidad en la definición y configuración del archivo de replicación, la generación automática de los trigger respectivos, los tipos de datos en claves, el rendimiento y la velocidad en la replicación, hacen de esta herramienta la más viable para su implementación.

Sin embargo como las dos herramientas son libres, *FBReplicator* es una muy buena alternativa en sistemas opeartivos Windows XP, o Windows 2003 server que no cuenten con los requerimientos de instalacion del otro software como el .NET Framework, o los service pack.

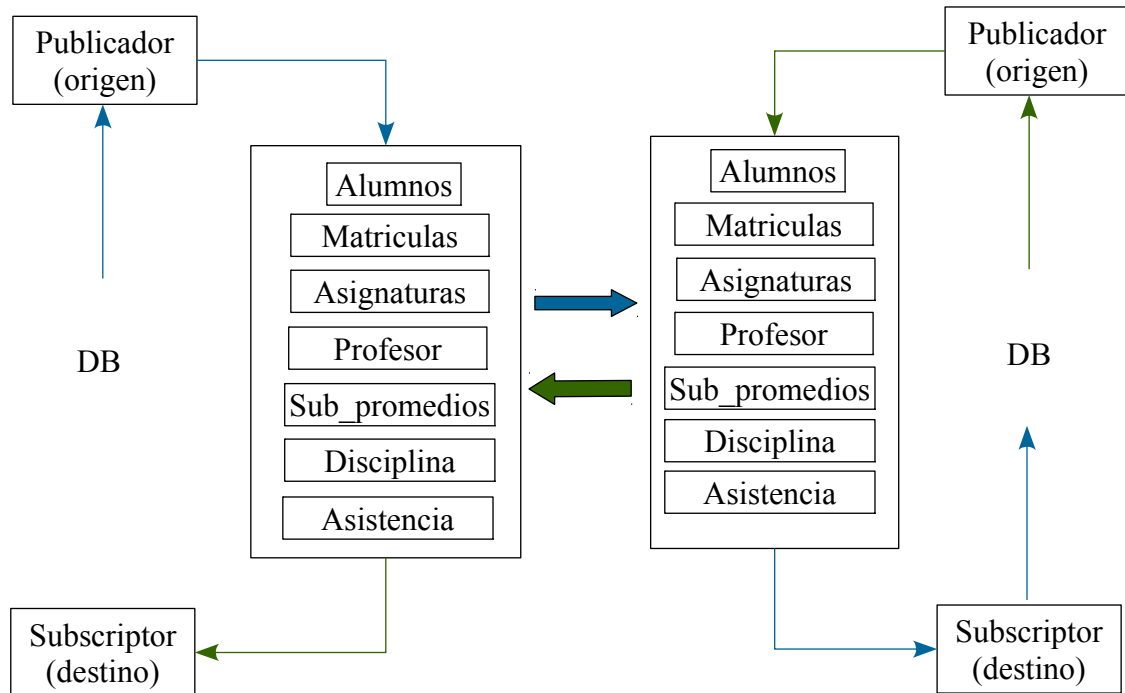
AdaReplicador utiliza un esquema de configuración basado en un archivo, el cual controla y administra los publicadores y subscriptores de replicación, el esquema de configuración para cada nodo de replicación del CED San Pio X.

El publicador genera los triggers necesarios en la base de datos los cuales se encargarán de registrar las transacciones que se generen como: inserción, actualización o borrado, y se almacenarán en la tabla ARPL\$LOG.

Es subscriptor mapea las tablas de la base y se definen que tablas van a interactuar, los campos claves que controlarán la integridad y los campos que se replicaran o serán recibidos.

El programa residente *AdaReplication Server* utiliza este archivo de configuración y en intervalos de tiempo utiliza la configuración del publicador, consulta la tabla ARPL\$LOG de la BD origen y verifica que haya registros, los cuales serán enviados hacia el destino, utilizando la configuración del subscriptor e insertando, actualizando o

borrando los registros que sean necesario procesar.



Los intervalos de tiempo se definen en este programa y se ejecutan desde 1 segundo pudiendo simular una replicación sincrona



Gráfico N° 6.5 Estructura de Configuración de Archivo .arpl
Elaborado por: Investigador

Instalación de AdaReplicador

Instalando los Prerequisitos – Microsoft .NET Framework 2

Gráfico N° 6.6
Elaborado por: Inv

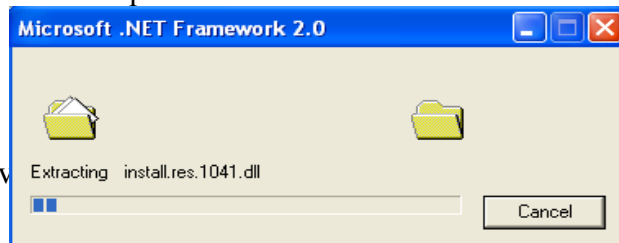


Gráfico N° 6.7
Elaborado por: I

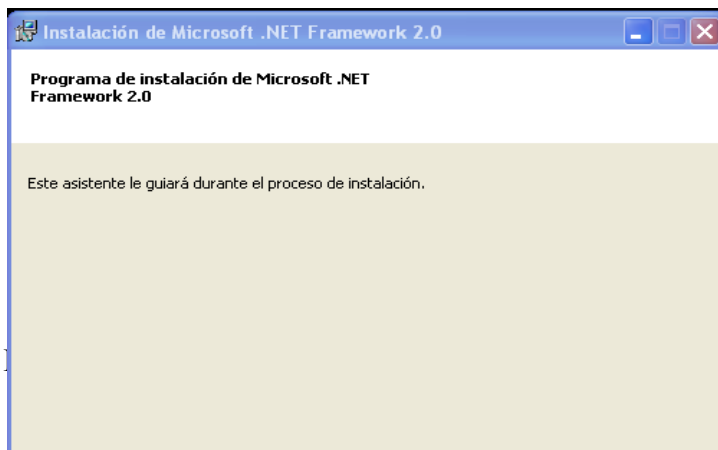
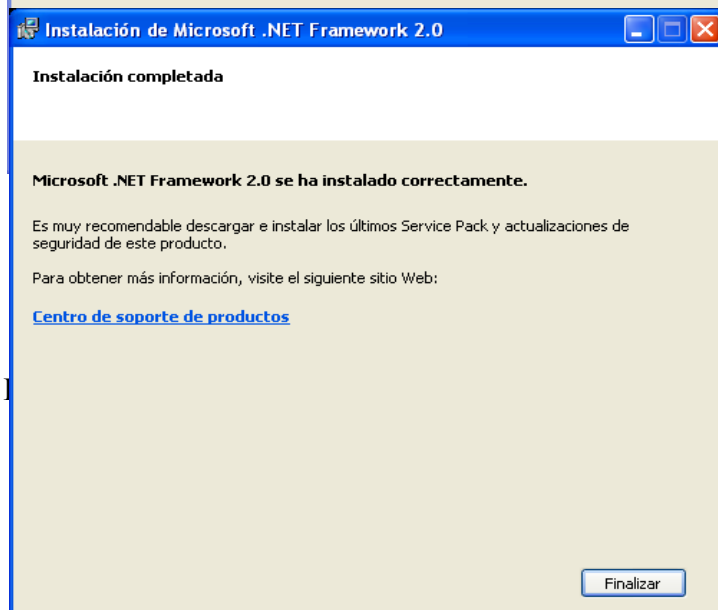


Gráfico N° 6.8
Elaborado por: I



Metodología de Implementación de Replicación

1. Se definen y configuran los sitios o nodos a ser replicados
2. Se instala el programa AdaReplicador
3. Se crea el usuario ARPL en las bases de datos que van a intervenir en la replicación de datos, se asigna los permisos necesarios de selección, inserción, actualización y borrado a las tablas necesarias.

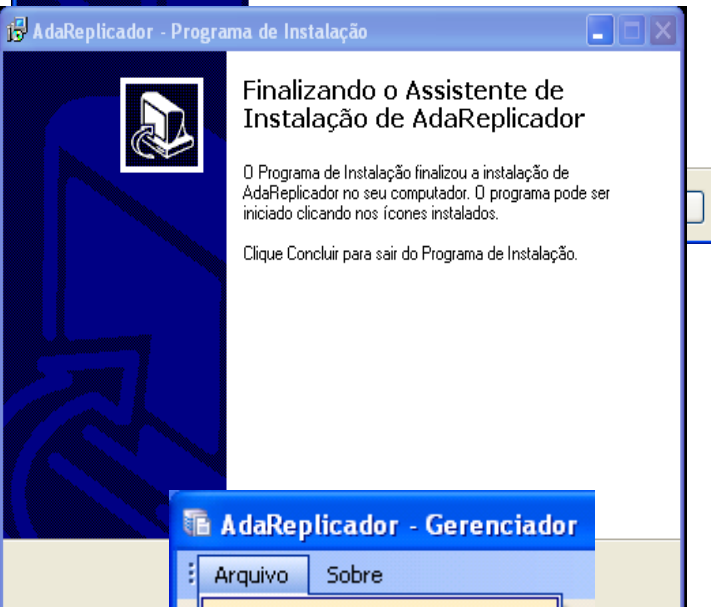
4. Se abre la aplicación AdaReplicador Gerenciador de Configuración y se configura el archivo **.arpl** para definir publicadores (origen) y subscriptores (destino), así como las tablas, campos claves, y campos a ser replicados.
5. Se ejecuta el programa AdaReplicador Server, se carga el archivo **.arpl** y se define el intervalo de tiempo en el que se ejecutará la replicación.

Instalando AdaReplicador

Gráfico N° 6.9
Elaborado por:

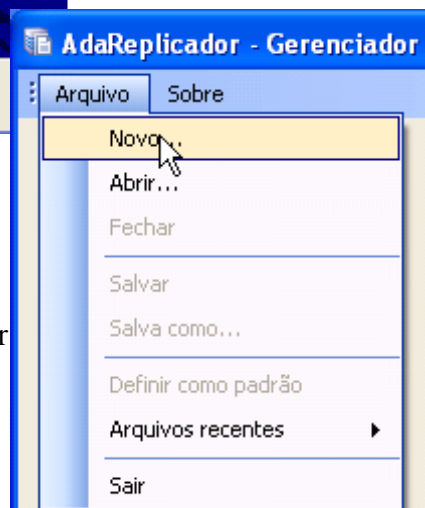


Gráfico N° 6.10
Elaborado por:



Crea un nuevo

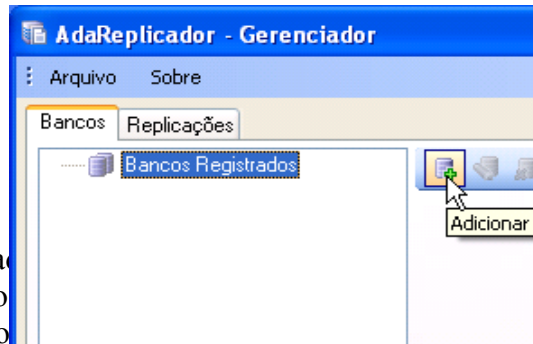
Gráfico N° 6.11
Elaborado por: Investigador



Añade un nuevo registro de Base de Datos para replicar

Gráfico N° 6.12

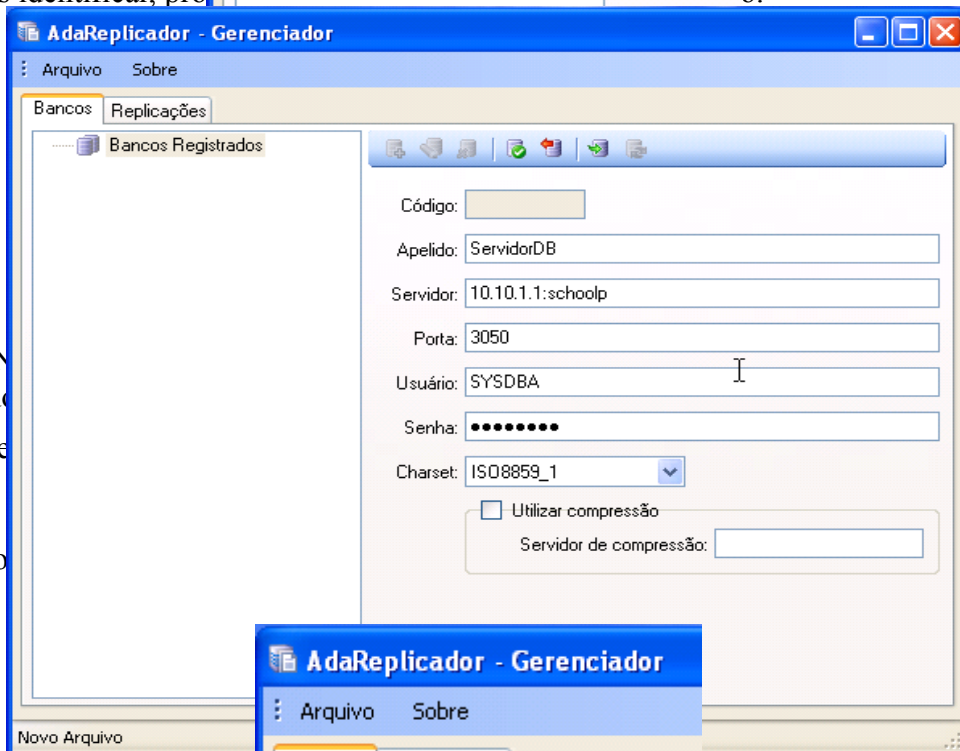
Elaborado por: Investigador
Se registra los parámetros que deseamos identificar, pro



de replicación que
0.

Gráfico N° 6.13

Elaborado por: Investigador
Luego de configurar los parámetros, vamos a guardarlos. Luego de guardarlos, vamos a configurar los parámetros de replicación que vamos a utilizar.



car,
ales

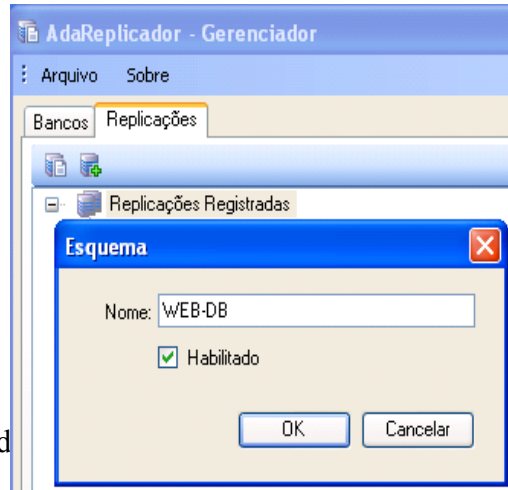
Gráfico N° 6.14

Elaborado por: Investigador

En la pestaña de Replicaciones, definimos los esquemas de replicación que vamos a configurar. Podemos definir una replicación **unidireccional**, es decir solo de recepción de información hacia el servidor principal, o una replicación **bidireccional**, desde un

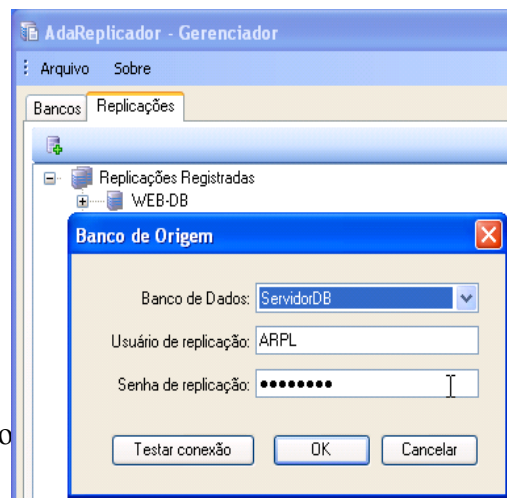
cliente - servidor principal y servidor principal - cliente

Gráfico N° 6.15
Elaborado por: Investigad



Escogemos de nuestro listado de BD, la que origina la información, es importate el usuario porque es el unico que esta autorizado a ejecutar los triggers de replicación

Gráfico N° 6.16
Elaborado por: Investigado



Escogemos de nuestro listado de BD, la que recibe los datos, al igual que en el caso anterior el usuario ARPL es el único autorizado a realizar los procesos de replicación, adicionalmente definimos cada cuantos registros se realizará la validación de registros (commit)

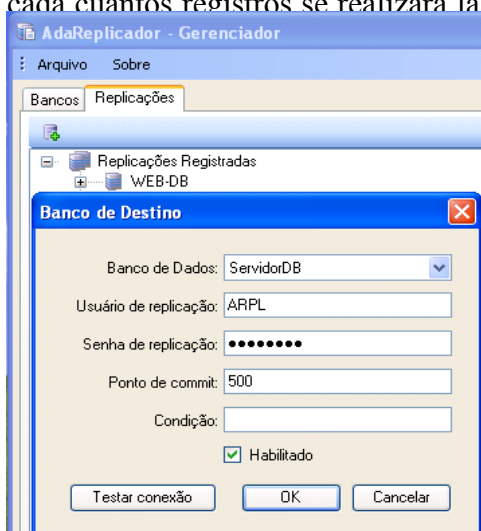


Gráfico N° 6.17

Elaborado por: Investigador

Definidas las BD que haran las funciones de Publicador y Subscriptor, con el icono de flecha verde hacia arriba, generamos los objetos que almacenarán la información, que será replicada en la base de datos origen, es decir la tabla **ARPL\$LOG** y la tabla **ARPL\$SEPARADOR**

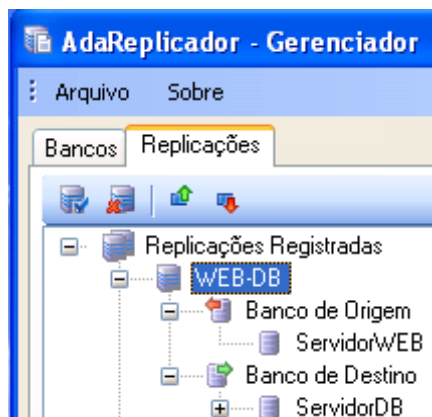


Gráfico N° 6.18

Elaborado por: Investigador

Se escoge las tablas que se desea replicar

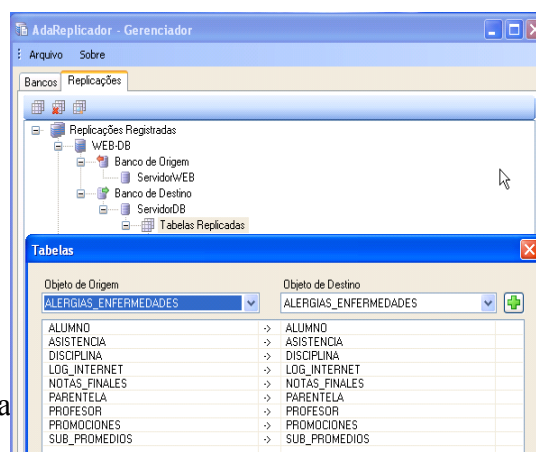
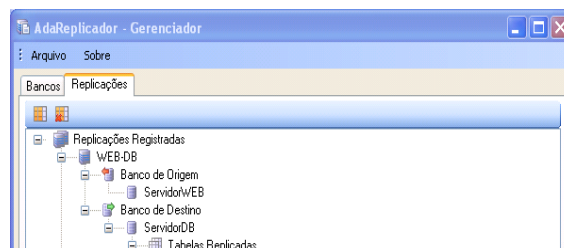
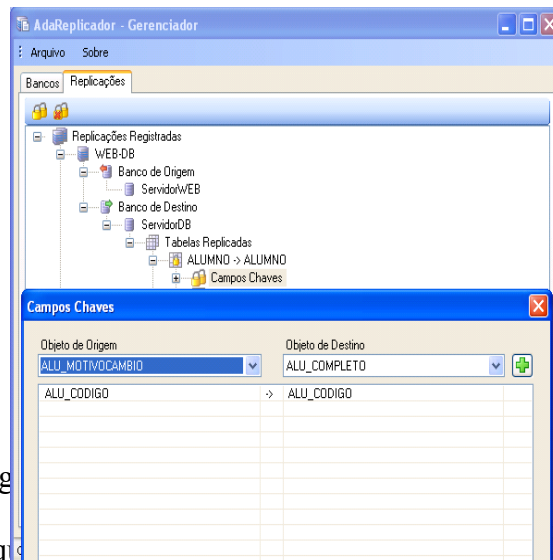


Gráfico N° 6.19

Elaborado por: Investiga

Se define los campos claves que controlarán la replicación en la tabla activa.

Gráfico N° 6.20
 Elaborado por: Investig
 Se define los campos que



C
E
P
E
c
l

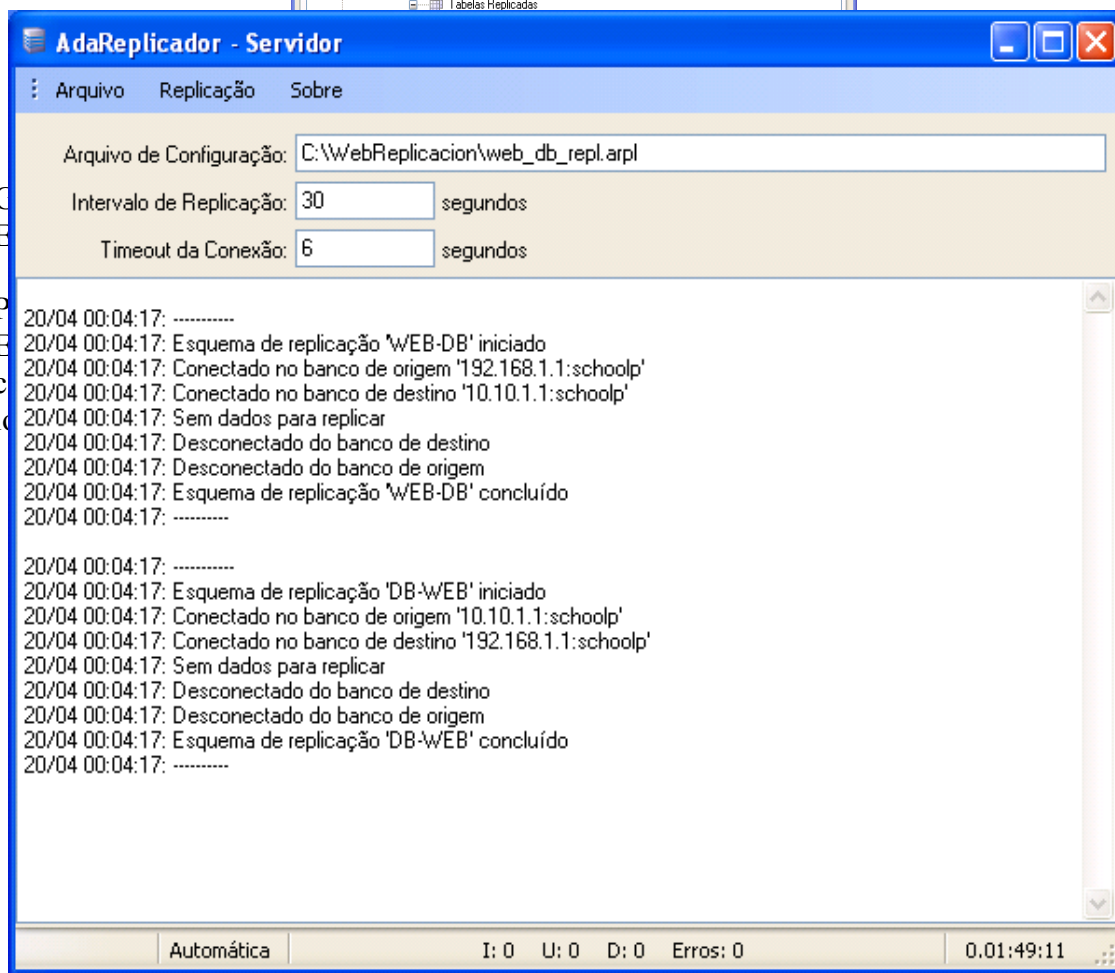


Gráfico N° 6.22

Elaborado por: Investigador

En el ejemplo mostrado el programa *AdaReplicador* genera los siguientes objetos dentro de la base de datos **Origen**

```
CREATE TABLE ARPL$LOG (  
  CODIGODOESQUEMA    INTEGER NOT NULL,  
  CODIGODOBANCODEORIGEM  INTEGER NOT NULL,  
  CODIGODOBANCODEDESTINO INTEGER NOT NULL,  
  SEQUENCIA          INTEGER NOT NULL,  
  CODIGODORELACIONAMENTO INTEGER NOT NULL,  
  TIPODEREPLICACAO    CHAR(1),  
  CHAVEANTIGA        VARCHAR(100),  
  CHAVENOVA          VARCHAR(100)  
);
```

```
ALTER TABLE ARPL$LOG ADD CONSTRAINT  
ARPL$LOG_PK  
PRIMARY KEY (CODIGODOESQUEMA, CODIGODOBANCODEORIGEM,  
CODIGODOBANCODEDESTINO, SEQUENCIA);
```

```
CREATE TRIGGER ARPL$LOG_BI0 FOR ARPL$LOG  
ACTIVE BEFORE INSERT POSITION 0  
AS BEGIN NEW.SEQUENCIA = GEN_ID(ARPL$GENERATOR, 1); END;
```

```
CREATE GENERATOR ARPL$GENERATOR;  
SET GENERATOR ARPL$GENERATOR TO 291772;  
CREATE TABLE ARPL$SEPARADOR (  
  CODIGODOESQUEMA    INTEGER NOT NULL,  
  CODIGODOBANCODEORIGEM  INTEGER NOT NULL,  
  CODIGODOBANCODEDESTINO INTEGER NOT NULL,
```

```

CODIGODORELACIONAMENTO INTEGER NOT NULL,
TABELADEORIGEM    VARCHAR(50) NOT NULL
);

```

```

ALTER TABLE ARPL$SEPARADOR ADD CONSTRAINT
ARPL$SEPARADOR_PK
PRIMARY KEY (CODIGODOESQUEMA, CODIGODOBANCODEORIGEM,
CODIGODOBANCODEDESTINO, CODIGODORELACIONAMENTO);

```

Para el ejemplo de la tabla ALUMNO se generan los siguientes disparadores (triggers) de replicación

```

CREATE TRIGGER ARPL$4_1_2_75_I FOR ALUMNO
ACTIVE AFTER INSERT POSITION 32700
AS
BEGIN
    IF (USER <> 'ARPL') THEN
        BEGIN
            INSERT INTO ARPL$LOG (CODIGODOESQUEMA,
                CODIGODOBANCODEORIGEM,
CODIGODOBANCODEDESTINO,
                CODIGODORELACIONAMENTO, TIPODEREPLICACAO, CHAVENOVA)
                VALUES (4, 1, 2, 75, 'I', NEW.ALU_CODIGO);
        END
    END;

```

```

CREATE TRIGGER ARPL$4_1_2_75_U FOR ALUMNO
ACTIVE AFTER UPDATE POSITION 32700
AS
BEGIN
    IF (USER <> 'ARPL') THEN
        BEGIN
            INSERT INTO ARPL$LOG (CODIGODOESQUEMA,
                CODIGODOBANCODEORIGEM,
CODIGODOBANCODEDESTINO,
                CODIGODORELACIONAMENTO, TIPODEREPLICACAO,
                CHAVEANTIGA, CHAVENOVA) VALUES (4, 1, 2, 75, 'U',
                OLD.ALU_CODIGO, NEW.ALU_CODIGO);
        END
    END;

```

```

CREATE TRIGGER ARPL$4_1_2_75_D FOR ALUMNO
ACTIVE AFTER DELETE POSITION 32700
AS
BEGIN

```

```

IF (USER <> 'ARPL') THEN
    BEGIN
        INSERT INTO ARPL$LOG (CODIGODOESQUEMA,
            CODIGODOBANCODEORIGEM,
CODIGODOBANCODEDESTINO,
            CODIGODORELACIONAMIENTO, TIPODEREPLICACAO,
            CHAVEANTIGA) VALUES (4, 1, 2, 75, 'D', OLD.ALU_CODIGO);
        END
END;

```

Este mismo tipo de disparadores se generan en las distintas tablas que se configuran para su replicación

El archivo .arpl maneja la base de datos Destino, es decir que dentro de él se administra la información de como se debe replicar, direcciona a que tabla y que registros se procesan, ya que no se hace ningun tipo de alteración en dicha base para controlar la replicación en la recepción de datos.

Mecanismos alternativos de respaldo de información

El motor de Base de Datos Firebird, permite la instalación en varias plataformas como HP-UX Unix, Sun Solaris, Mac OS, Linux y Windows. Para este caso se ha utilizado el sistema operativo Linux versión CenOS 5.6 el cual permite programar tareas que se ejecutan periodicamente mediante el demonio (servicio) *cron*. Se ha procedido a realizar este control como una medida alternativa de seguridad para el respaldo de infromación permanente, sin intervención del factor humano.

A continuación se describe el procedimiento y su implementación.

Comando *crontab -e* para abrir un editor de texto *vi* y añadir las reglas de la tarea

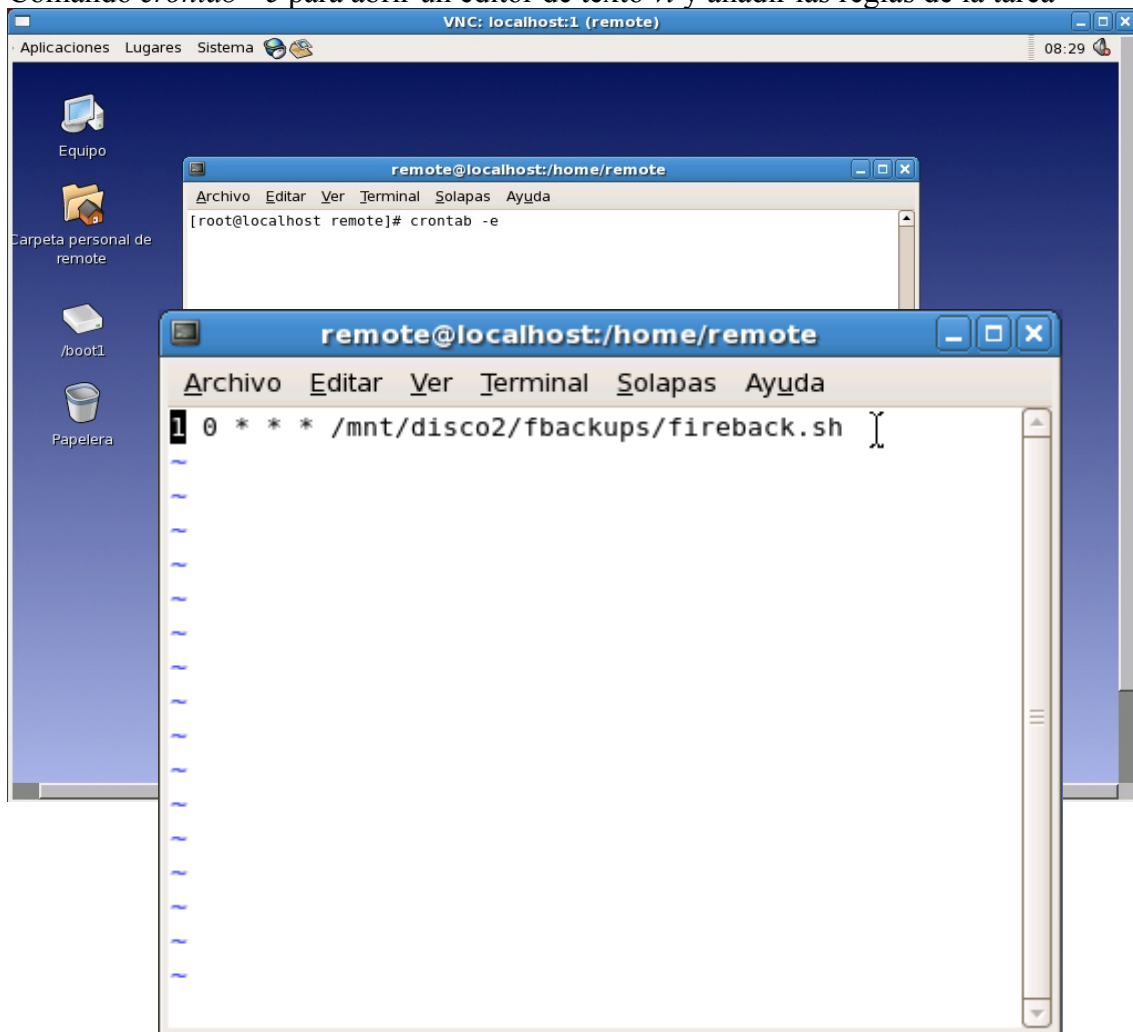


Gráfico N° 6.24

Elaborado por: Investigador

1 => minuto 0-59

0 => hora 0-23

* => día del mes 0-31

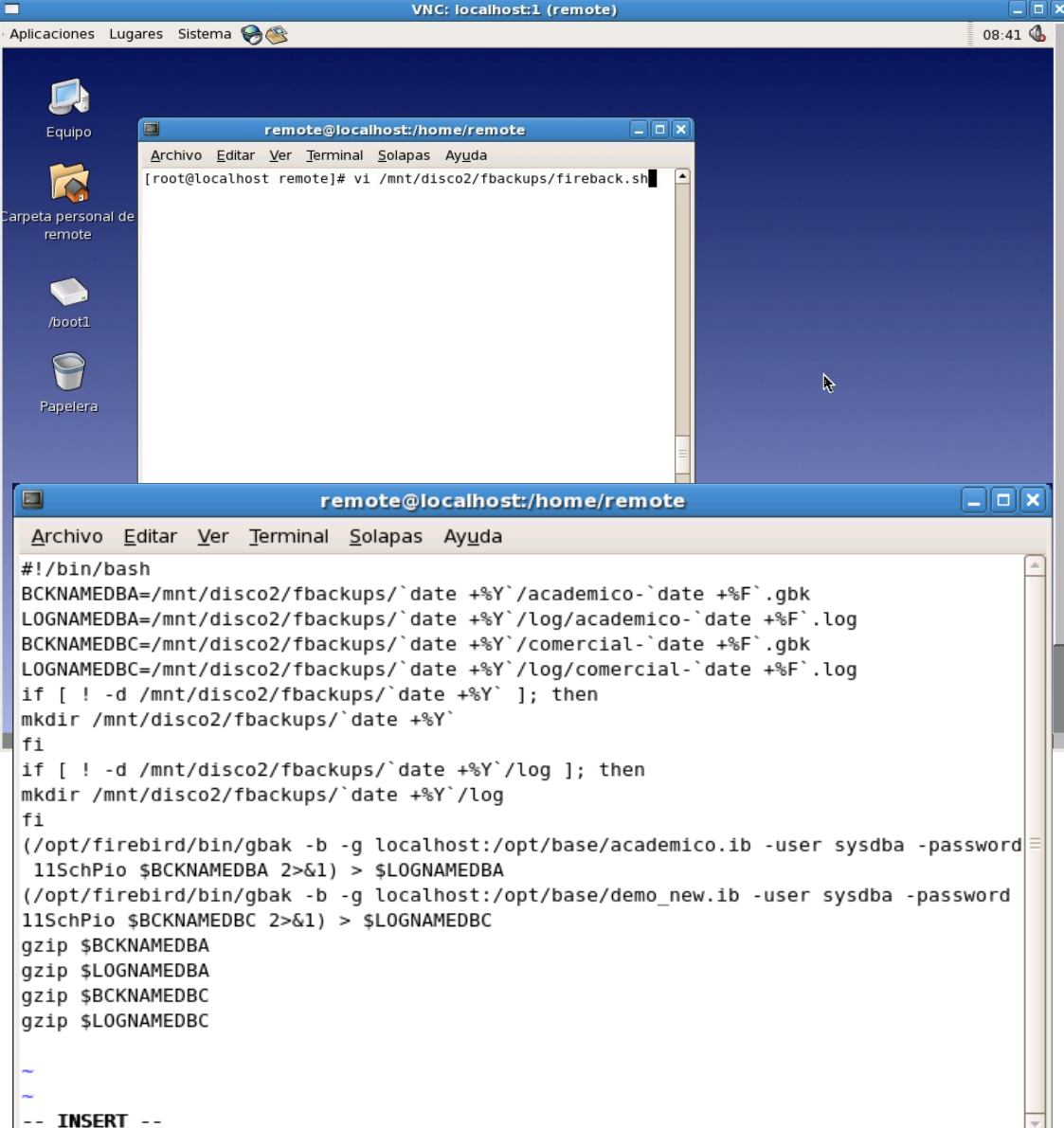
* => mes 0-12 (o sunombre con las tres primeras letras en inglés)

* => día semana 0-7 (0 or 7 indica domingo, o su nombre con las tres primeras letras en inglés)

Un asterisco (*) indica todos los posibles valores.

/mnt/disco/fbackups/fireback.sh = camino y archivo .sh que se ejecutará de acuerdo al intervalo programado.

Creación del script .sh con las reglas de ejecución para el respaldo automático de las bases de datos mediante el comando **vi** para abrir un editor de textos



The screenshot shows a VNC window titled "VNC: localhost:1 (remote)". Inside, there is a desktop environment with a terminal window open. The terminal window shows the command `vi /mnt/disco2/fbackups/fireback.sh` being executed. The script content is as follows:

```
#!/bin/bash
BCKNAMEDBA=/mnt/disco2/fbackups/`date +%Y`/academico-`date +%F`.gbk
LOGNAMEDBA=/mnt/disco2/fbackups/`date +%Y`/log/academico-`date +%F`.log
BCKNAMEDBC=/mnt/disco2/fbackups/`date +%Y`/comercial-`date +%F`.gbk
LOGNAMEDBC=/mnt/disco2/fbackups/`date +%Y`/log/comercial-`date +%F`.log
if [ ! -d /mnt/disco2/fbackups/`date +%Y` ]; then
mkdir /mnt/disco2/fbackups/`date +%Y`
fi
if [ ! -d /mnt/disco2/fbackups/`date +%Y`/log ]; then
mkdir /mnt/disco2/fbackups/`date +%Y`/log
fi
(/opt/firebird/bin/gbak -b -g localhost:/opt/base/academico.ib -user sysdba -password
11SchPio $BCKNAMEDBA 2>&1) > $LOGNAMEDBA
(/opt/firebird/bin/gbak -b -g localhost:/opt/base/demo_new.ib -user sysdba -password
11SchPio $BCKNAMEDBC 2>&1) > $LOGNAMEDBC
gzip $BCKNAMEDBA
gzip $LOGNAMEDBA
gzip $BCKNAMEDBC
gzip $LOGNAMEDBC
```

At the bottom of the terminal window, there is a prompt `-- INSERT --`.

Gráfico N° 6.26

Elaborado por: Investigador

#bin bash => cabecera del archivo

BCKNAMEDBA=/mnt/disco2/fbackups/`date +%Y` /academico -`date +%F`.gbk
=> Nombre del archivo el que se guarda en una carpeta en el año activo con la variable **date +%Y**, se le añade la fecha con la variable **date +%F** para identificar el día que se guardó y la extensión **.gbk** que es la extensión que da fairebird a sus archivos de respaldo

LOGNAMEDBA=/mnt/disco2/fbackups/`date +%Y` /log/academico -`date +%F`.log => Nombre del archivo de registro el que se guarda en una carpeta en el año activo dentro de otra carpeta **log** con la variable **date +%Y**, se le añade la fecha con la variable **date +%F** para identificar el día que se guardó y la extensión **.log** de registro de respaldo.

if [! -d /mnt/disco2/fbackups/`date +%Y`]; then

mkdir /mnt/disco2/fbackups/`date +%Y`

fi => verifica si existe la carpeta del año activo si no, la crea.

if [! -d /mnt/disco2/fbackups/`date +%Y`]; then

mkdir /mnt/disco2/fbackups/`date +%Y`

fi => verifica si existe la carpeta de registro log si no, la crea.

(/opt/firebird/bin/gback -b -g localhost:/opt/base/academico.ib -user SYSDBA

-password xxxxxx \$BCKNAMEDBA 2 > &1) > \$LOGNAMEDBA => comando que hace un backup de la base de datos y los guarda en las variables de respaldo y de log o registro.

gzip \$BCKNAMEDBA => empaqueta el archivo de respaldo de Base de Datos.

gzip \$LOGNAMEDBA => empaqueta el archivo de registro o log del respaldo de la Base de Datos.

Listamos los archivos con el comando ls

```
remote@localhost:/home/remote
Archivo Editar Ver Terminal Solapas Ayuda
academico-2012-03-13.gbk.gz  comercial-2012-03-15.gbk.gz
academico-2012-03-15.gbk.gz  comercial-2012-03-16.gbk.gz
academico-2012-03-16.gbk.gz  comercial-2012-03-18.gbk.gz
academico-2012-03-18.gbk.gz  comercial-2012-03-19.gbk.gz
academico-2012-03-19.gbk.gz  comercial-2012-03-20.gbk.gz
academico-2012-03-20.gbk.gz  comercial-2012-03-22.gbk.gz
academico-2012-03-22.gbk.gz  comercial-2012-03-23.gbk.gz
academico-2012-03-23.gbk.gz  comercial-2012-03-24.gbk.gz
academico-2012-03-27.gbk.gz  comercial-2012-03-27.gbk.gz
academico-2012-03-28.gbk.gz  comercial-2012-03-28.gbk.gz
academico-2012-03-29.gbk.gz  comercial-2012-03-29.gbk.gz
academico-2012-04-04.gbk.gz  comercial-2012-04-04.gbk.gz
academico-2012-04-06.gbk.gz  comercial-2012-04-06.gbk.gz
academico-2012-04-10.gbk.gz  comercial-2012-04-10.gbk.gz
academico-2012-04-11.gbk.gz  comercial-2012-04-11.gbk.gz
academico-2012-04-12.gbk.gz  comercial-2012-04-12.gbk.gz
academico-2012-04-13.gbk.gz  comercial-2012-04-13.gbk.gz
academico-2012-04-14.gbk.gz  comercial-2012-04-14.gbk.gz
academico-2012-04-17.gbk.gz  comercial-2012-04-17.gbk.gz
academico-2012-04-18.gbk.gz  comercial-2012-04-18.gbk.gz
academico-2012-04-20.gbk.gz  comercial-2012-04-20.gbk.gz
comercial-2012-01-05.gbk.gz  log
[root@localhost remote]#
```

Gráfico N° 6.28
Elaborado por: Investigador

Un método relativamente barato de conseguir replicación de una base de datos Firebird es crear una shadow o sombra en un disco externo de red local. Nos protege contra desastres en el disco donde se aloja la base de datos principal, aunque, evidentemente, si borramos datos de la principal, automáticamente se borrarán también de la copia “*shadow*”, por lo que este es un sistema válido únicamente contra daños físicos, incendios, robos, etc.

Aunque las pruebas realizadas han sido en un servidor con CentOS 5.6 y Firebird 2.5, sirve cualquier distribución Linux. La base de datos (*academico.ib*) está alojada en el directorio */opt/base/* del mismo servidor. El disco externo es de conexión por red (Ethernet). Ambos están conectados a un switch, aunque sirven igualmente un router o un hub.

	IP	Directorio
Servidor		<i>/opt/base/</i>
Disco Externo	192.168.2.1	<i>/opt/shadowdatos/</i>

Cuadro N° 19 datos iniciales para BD “*shadow*”
Realizado por: Investigador

Los pasos a seguir son los siguientes: Crear en el directorio */root* el fichero oculto *smbcredentials*

>edit /root/.smbcredentials (se puede usar cualquier editor de textos)

Añadir las siguientes líneas:

username=firebird

password=clavedelroot El password (*clavedelroot*) es el del root del servidor.

Crear, en el disco de red, el directorio para alojar la shadow de la base de datos, por ejemplo: */shadowdatos*

Crear el directorio para montar el disco de red, normalmente en */opt*, creamos, por

ejemplo: *sombra*.

```
>mkdir /optt/sombra
```

Añadir la siguiente línea al final del fichero */etc/fstab*

```
//192.198.0.125/shadowdatos/mnt/sombra/smbfs  
credentials=/root/smbcredentials,dmask=777,fmask=777 0 0
```

Hay que recargar */etc/fstab* para que monte la nueva entrada:

```
>mount -a (también puede reiniciar el servidor)
```

Ahora sólo queda crear la shadow. Deberá conectar a la base de datos del servidor con cualquier utilidad como *ibexpert*, *isql*,

ibadmin, *marathon*, etc. y ejecutar la siguiente sentencia sql:

```
>create shadow 1 auto "/optt/sombra/basedatosprueba.shd"
```

"basedatosprueba.shd" puede ser cualquier nombre válido para una base de datos Firebird.

Esta sentencia tardará en ejecutarse según el tamaño de la base de datos y la velocidad de la red local.

Desde este momento, todo lo que se haga en la base de datos principal, quedará reflejado en la shadow. En caso de necesidad, se puede usar como cualquier otra base de datos Firebird, aunque antes debemos activarla de la siguiente forma:

```
>gfix -activate -user sysdba -password masterkey "/mnt/sombra/basedatosprueba.shd"
```

Para deshabilitar la shadow, use la siguiente sentencia sql:

```
>drop shadow 1 (Tenga cuidado, también se borrará la base de datos shadow)
```

Manual de procedimientos para recuperación de Información en caso de fallo.

Preventivo

1. Respaldar físicamente los archivos de backup almacenados en el directorio */mnt/disco2/fbackups/* en discos externos, DVD's o CD's periódicamente, por lo menos una vez al mes.

Correctivo

2. Instalar en una máquina temporal el motor de base de datos Firebird V 2.5
3. Configurar la red con las siguiente dirección para el acceso de los clientes
 - 10.10.1.1 como servidor principal
 - Crear los usuarios necesarios con un administrador para Bases de datos Firebird como SQLHammer (de licencia libre), y asignar los permisos de acceso, inserción, borrado o actualización según sea el caso.
 - Editar el archivo *alias.conf* que se encuentra en el directorio *c:\Archivos de Programa\Firebird\Firebird_2_5* y añadir las siguiente líneas

schoolp = c:\base\academico.ib

comercial = c:\base\demo_new.ib
4. Buscar el los respaldos físicos del último backup automático que realizó el servicio *cron*
5. Desempaquetar los archivos *.gbk* en un directorio temporal
6. Restaurar las Bases de Datos con la herramienta de Administración SQLHammer
7. Copiar la base de datos dentro del directorio *c:\base* y renombrarlos como *academico.ib* y *demo_new.ib* respectivamente.

Implementación: representa el correcto funcionamiento de la tecnología, los datos y las aplicaciones de usuarios finales accesibles para el usuario del negocio.

Mantenimiento y crecimiento: se basa en la necesidad de continuar con las actualizaciones de forma constante para así lograr la evolución de las metas por conseguir.

Gestión del proyecto: asegura que todas las actividades del ciclo de vida se lleven a cabo de manera sincronizada.

Pruebas Se realizaron las pruebas para verificar la veracidad y confiabilidad del sistema dando como resultado la misma información tanto en el origen como en el destino en el caso de la replicación, los respaldos automáticos y las copias de seguridad.

Conclusiones y Recomendaciones

Conclusiones

- El estudio realizado de los dos sistemas de replicación provee una solución efectiva a los problemas de respaldo y recuperación de información con todas las características necesarias para administrar y mejorar los procesos de control de Bases de Datos Distribuidas, reducirá el tiempo en el acceso o recuperación de información en caso de un fallo grave cualquiera que este sea, preveyendo de forma oportuna la seguridad de los datos almacenados.
- Es necesario aprovechar las bondades que brinde un *Sistema Operativo* para poder contar con un respaldo automático de los archivos de Bases de Datos y evitar la intervención del factor humano, solo delegándolo hacer los respaldos a medios físicos como CD's o DVD's.
- Es indispensable contar con documentación formal como manuales de procesos y procedimientos para el respaldo y recuperación de la información para mantener el acceso a la misma de forma segura y confiable.
- Con el sistema de replicación implementado de datos y las tareas programadas de respaldo se mantiene la información mas precisa, exacta y confiable para una toma de decisiones rápida y oportuna

El paquete AdaReplicador provee un ambiente amigable, de fácil uso, al mismo tiempo que cuenta con todas las herramientas necesarias para la realización de la replicación de datos de manera adecuada y confiable y así reducir tiempo en la recuperación y presentación de datos.

Recomendaciones .

- Se recomienda buscar actualizaciones del software seleccionado, tratando de obtener mejoras a las prestaciones.
- Se recomienda hacer un mantenimiento periódico del o los archivos de configuración para la replicación de las Bases de datos, añadiendo o eliminando campos o tablas, que hayan sido modificados en las bases de datos publicadoras o subscriptoras.
- Se hace necesario también realizar una capacitación básica al personal de sistemas tanto en el manejo de la herramienta de replicación, como en la administración de sistemas operativos Linux para un mejor funcionamiento del DBMS, y que los resultados sean confiables y exactos.
- Para que la solución sea completamente efectiva es recomendable mantener un sistema de respaldo de energía confiable y constante, así como el mantenimiento de los equipos por lo menos unas dos veces al año para mejorar el rendimiento y alargar la vida útil de los mismos..

MATERIALES DE REFERENCIA

Referencias Bibliográficas

[1] DATE, C.J. (2001). *Introducción a los Sistemas de Bases de Datos*. Mexico: Pearson Education. p. 5 – 19.

[2] ELMASARI, R. NAVATHE, S. (1994). *Sistemas de Bases de Datos – Conceptos Fundamentales*. Mexico: Addison Wesley. p. 19.

[3] SILVERTSHATZ, A. KORTH, H. SUDARSHAN, S. (2002). *Fundamentos de Bases de Datos*. Madrid: McGraw Hill. p. 19 – 26.

[4] SILVERTSHATZ, A. KORTH, H. SUDARSHAN, S. (2002). *Fundamentos de Bases de Datos*. Madrid: McGraw Hill. p. 52 – 58.

[5] ELMASARI, R. NAVATHE, S. (1997). *Sistemas de Bases de Datos – Conceptos Fundamentales*. Mexico: Addison Wesley. p. 702 – 720.

Bibliografía

BORRIE, H. (2004). *The Firebird Book: A Reference for Database Developers*. New York: Apress

DATE, C. J. (2001). *Introducción A Los Sistemas De Base De Datos*. Mexico: Pearson

Education.

ELMASARI, R, NATHE, S. (1994). *Sistemas de Bases de Datos – Conceptos Fundamentales*. Mexico: Addison – Wesley.

HERNÁNDEZ, J, LANDÁZURI, A. (.1999). *Corrientes, Métodos y Técnicas de Investigación*. Quito:..EB-PRODEC.

HERNÁNDEZ, R.(2000). *Metodología de la Investigación*. Mexico: McGraw-Hill

HERRERA, L. (2008). *Tutoría de la Investigación Científica*. Quito: Diemerino Editores.

IZQUIERDO, E. (2003). *Investigación Científica*. Loja.: COSMOS.

MARTEENS, I. (2002). *La Cara Oculta de Delphi 6* . Madrid: Intuitive Sight

RAHAMI, S, HAUG, F. (2010). *Distributed Database Managment System – A Practical Approach*. New Jersey: John Willey & Sons.

SILBERTSHATZ, A, KORTH, H, SUDARSHAN, S. (2002). *Fundamentos de Bases de Datos*. Madrid: McGraw Hill.

Lincografía

PADILLA. V (2002). Sistemas de Bases de Datos Distribuidos. Recuperado el 10 de mayo del 2011 del sitio web de la Universidad de Colima.

http://docente.ucol.mx/vpc1052/public_html/Expo%20SBDD.doc.

ROJAS Sergio. (2010), Tipos de Bases de Datos. Recuperado el 20 de julio del 2011 del sitio web del Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga <http://www.lcc.uma.es/~galvez/ftp/bdst/Tema2.pdf>.

CHINCHILLA. A, (2003). Fragmentacion de Datos en Bases de Datos Dstribuidas. [en línea]. Universidad de Costa Rica – Centro Centroamericano de Población. Disponible en: <http://ccp.ucr.ac.cr/bvp/pdf/desarrollohumano/bd_distribuidas-rca.pdf> [Consulta 12 septiembre 2011]

Congreso Nacional Ecuatoriano (2004), Ley de Transparencia y Acceso a la Información. Recuperado el 21 de noviembre del 2011 del sitio web de la Escuela Politécnica del Litoral ESPOL. http://www.transparencia.espol.edu.ec/documentos/L_acceso.pdf

HOPKINS. Matt (19988). Data Replication With Interbase. Recuperado el 15 de diciembre del 2011. del sitio web de IBPhoenix de la Borland Developers Conference. http://www.ibphoenix.com/resources/documents/how_to/doc_316

NAVAS Ignacio. (2007). Ingeniería Colaborativa y sincronización de bases de datos entre oficinas internacionales de IBERNINCO. Recuperado el Consulta: 10 febrero 2012. del sitio web de la Universidad Pontificia Comillas. Disponible <http://www.iit.upcomillas.es/pfc/resumenes/4687e9a993c8e.pdf>.

ANEXOS

Anexo 1

INSTRUMENTOS PARA LA ENCUESTA.

ENCUESTA DIRIGIDA AL PERSONAL DEL CENTRO EDUCATIVO DIOCESANO SAN PIO X Verificar la necesidad de la implementación de un sistema de replicación para manejar respaldos de información

Estamos trabajando en un estudio que servirá para elaborar una tesis profesional acerca del grado de aceptación de desarrollo de la Gestión de Base de Datos en el Centro Educativo Diocesano San Pío X. Sus respuestas serán confidenciales y anónimas.

DATOS GENERALES:

Fecha de la Encuesta.....

DATOS ESPECÍFICOS:

Marque con X en el paréntesis de su elección

Nº	Pregunta	Respuesta	Cod.
1	¿Conoce Usted como se almacena la información de los alumnos de la Institución?	Si No	1() 2()
2	¿Conoce si los equipos de computo con los que cuenta la institución son los apropiados para el almacenamiento seguro de información?	Si No	1() 2()
3	¿Los equipos de la institución cuenta con un repaldo de energía en caso de cortes o alteraciones de tensión imprevistos?	Si No	1() 2()
4	¿Conoce Uste si existe algun procedimiento para el respaldo de información en la institución en caso de corte de energía?	Si No	1() 2()
5	¿En caso de deterioro o daño de un equipo de computo de la institución, existe algun procedimiento para la recuperación de la información?	Si No	1() 2()
6	¿Se beneficiara la empresa con la implementación de bases de datos distribuidas para la recuperación o respaldo de información?	Si No	1() 2()
7	¿Cree usted que una implementación de bases de datos distribuidas ayude a minimizar el tiempo en la recuperación de información en caso de fallo?	Si No	1() 2()
8	¿Cree usted que la distribución de datos mejorara el rendimiento de la red y el acceso a la información?	Si No	1() 2()
9	¿Cree usted que las bases de datos distribuidas es un buen mecanismo de seguridad de la información en caso de un ataque de seguridad?	Si No	1() 2()
10	¿Cree Usted que la Fragmentación de Datos afectará en la coherencia y consistencia de la información?	Si No	1() 2()
11	¿Cree Uste que los recursos de Hardware y Software con los que cuenta la Institución son suficientes para una implementación de Bases de Datos Dsitribuidas?	Si No	1() 2()

Gracias por su colaboración

Anexo 3

Manual de Procedimientos inicialización de Servidores

- 1.** Preparar discos duros de servidores
- 2.** Instalar sistema operativo Linux CentOS 5.6 o CentOS 5.7
- 3.** Configurar las direcciones IP de red
- 4.** Configurar los servicios de proxy
- 5.** Configurar los servicios y tareas de cron
- 6.** Configurar servicios de postfix
- 7.** Instalar motor de base de datos Firebird
- 8.** Copiar librerías UDF IBCPP.so en /opt/firebird/UDF/
- 9.** Instalar librerías PHP necesarias para conectar a Firebird
- 10.** Configurar servidor WEB apache
- 11.** Iniciar servicios, squid, httpd, cron, smtp, dovecot

Manual de Procedimientos Respaldo Físico

1. Iniciar sesión en el servidor principal como superusuario
2. Copiar la carpeta 2012 ubicada en /mnt/fbackups/ en una memoria flash
3. Grabar el contenido de esa carpeta en un CD o DVD
4. Archivar el DVD en un lugar seguro.

Manual de procedimientos recuperación backup

1. Copiar el archivo de fecha mas reciente de extensión *.tar.gz* respaldado en la carpeta 2012
2. Ejecutar la sentencia de desempaquetar el archivo
3. Ejecutar la sentencia de de restaurar el archivo *.fbk* desempaquetado
4. renombrar el archivo al nombre original *academico.fdb* en el caso de la base de datos de la aplicación escolástico y *demo.fdb* en el caso de la base de datos comercial.

Anexo 4

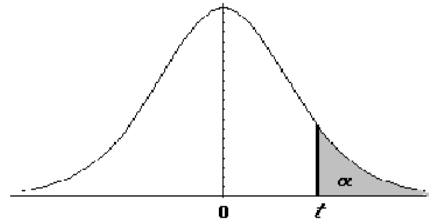
Formato de Ficha Bibliográfica

Ficha bibliográfica	
Autor: _____	Editorial: _____
Título: _____	País: _____
Año: _____	
Resumen del Contenido: _____ _____ _____ _____ _____ _____ _____	
Numero de impresión o edición: _____	
Traductor: _____	

Anexo 5

Tabla de la t de student

Contiene los valores tales que $p[T > t] = \alpha$,



dónde n son los grados de libertad

n / α	0.3	0.25	0.2	0.1	0.05	0.025	0.01	0.005	0.0025	0.001	0.0005
1	0.7265	1	1.3764	3.0777	6.3137	12.7062	31.821	63.6559	127.3213	318.3088	636.6192
2	0.6172	0.8165	1.0607	1.8856	2.92	4.3027	6.9645	9.925	14.089	22.3271	31.5991
3	0.5844	0.7649	0.9785	1.6377	2.3534	3.1824	4.5407	5.8408	7.4533	10.2145	12.924
4	0.5686	0.7407	0.941	1.5332	2.1318	2.7765	3.7469	4.6041	5.5976	7.1732	8.6103
5	0.5594	0.7267	0.9195	1.4759	2.015	2.5706	3.3649	4.0321	4.7733	5.8934	6.8688
6	0.5534	0.7176	0.9057	1.4398	1.9432	2.4469	3.1427	3.7074	4.3168	5.2076	5.9588
7	0.5491	0.7111	0.896	1.4149	1.8946	2.3646	2.9979	3.4995	4.0293	4.7853	5.4079
8	0.5459	0.7064	0.8889	1.3968	1.8595	2.306	2.8965	3.3554	3.8325	4.5008	5.0413
9	0.5435	0.7027	0.8834	1.383	1.8331	2.2622	2.8214	3.2498	3.6897	4.2968	4.7809
10	0.5415	0.6998	0.8791	1.3722	1.8125	2.2281	2.7638	3.1693	3.5814	4.1437	4.5869
11	0.5399	0.6974	0.8755	1.3634	1.7959	2.201	2.7181	3.1058	3.4966	4.0247	4.437
12	0.5386	0.6955	0.8726	1.3562	1.7823	2.1788	2.681	3.0545	3.4284	3.9296	4.3178
13	0.5375	0.6938	0.8702	1.3502	1.7709	2.1604	2.6503	3.0123	3.3725	3.852	4.2208
14	0.5366	0.6924	0.8681	1.345	1.7613	2.1448	2.6245	2.9768	3.3257	3.7874	4.1405
15	0.5357	0.6912	0.8662	1.3406	1.7531	2.1315	2.6025	2.9467	3.286	3.7328	4.0728
16	0.535	0.6901	0.8647	1.3368	1.7459	2.1199	2.5835	2.9208	3.252	3.6862	4.015
17	0.5344	0.6892	0.8633	1.3334	1.7396	2.1098	2.5669	2.8982	3.2224	3.6458	3.9651
18	0.5338	0.6884	0.862	1.3304	1.7341	2.1009	2.5524	2.8784	3.1966	3.6105	3.9216
19	0.5333	0.6876	0.861	1.3277	1.7291	2.093	2.5395	2.8609	3.1737	3.5794	3.8834
20	0.5329	0.687	0.86	1.3253	1.7247	2.086	2.528	2.8453	3.1534	3.5518	3.8495
21	0.5325	0.6864	0.8591	1.3232	1.7207	2.0796	2.5176	2.8314	3.1352	3.5272	3.8193
22	0.5321	0.6858	0.8583	1.3212	1.7171	2.0739	2.5083	2.8188	3.1188	3.505	3.7921
23	0.5317	0.6853	0.8575	1.3195	1.7139	2.0687	2.4999	2.8073	3.104	3.485	3.7676
24	0.5314	0.6848	0.8569	1.3178	1.7109	2.0639	2.4922	2.797	3.0905	3.4668	3.7454
25	0.5312	0.6844	0.8562	1.3163	1.7081	2.0595	2.4851	2.7874	3.0782	3.4502	3.7251
26	0.5309	0.684	0.8557	1.315	1.7056	2.0555	2.4786	2.7787	3.0669	3.435	3.7066
27	0.5306	0.6837	0.8551	1.3137	1.7033	2.0518	2.4727	2.7707	3.0565	3.421	3.6896
28	0.5304	0.6834	0.8546	1.3125	1.7011	2.0484	2.4671	2.7633	3.0469	3.4082	3.6739
29	0.5302	0.683	0.8542	1.3114	1.6991	2.0452	2.462	2.7564	3.038	3.3962	3.6594
30	0.53	0.6828	0.8538	1.3104	1.6973	2.0423	2.4573	2.75	3.0298	3.3852	3.646
40	0.5286	0.6807	0.8507	1.3031	1.6839	2.0211	2.4233	2.7045	2.9712	3.3069	3.551
80	0.5265	0.6776	0.8461	1.2922	1.6641	1.9901	2.3739	2.6387	2.887	3.1953	3.4163