



UNAM
CAMPUS ACATLAN



1 6 2 1 ' 0 4 MAR 1 9 1 1 : 5 0

UNIDAD DE ADMINISTRACION
ESCOLAR
RECIBIDO



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



Universidad Nacional Autónoma de México

Escuela Nacional de Estudios Profesionales

Campus Acatlán

**“MIGRACIÓN Y PROPUESTA DE INSTALACIÓN DE
LAS BASES DE DATOS DEL SISTEMA DE
RECLUTAMIENTO Y SELECCIÓN DE PERSONAL DEL
SERVICIO DE ADMINISTRACIÓN TRIBUTARIA”**

SEMINARIO TALLER EXTRACURRICULAR

**QUE PARA OBTENER EL TÍTULO DE
LICENCIADO EN MATEMÁTICAS
APLICADAS Y COMPUTACIÓN**

**PRESENTA:
VERÓNICA REZA GÓMEZ**

**ASESOR:
ING. RUBÉN ROMERO RUIZ**

MARZO 2004



AGRADECIMIENTOS

Señor son tantas las cosas que tengo que agradecerle que no me alcanzaría el tiempo ni todo el papel de esta Tesis para decírtelo... pero bueno una vez más Señor Gracias por no dejarme caer en los momentos que más te necesite; por las fuerzas infiltradas para afrontar todas las adversidades que se han presentaron durante mi vida y sobre todo por la oportunidad que me brindaste de estar aquí en este momento escribiendo estas palabras.

Gracias Mamá y Papá por todo su amor, apoyo, comprensión y por todos esos regaños que me dieron el suficiente valor para enfrentar esta vida, este... mi destino.

A un gran ejemplo de fortaleza y valor, Ana... Gracias por ser mi hermana.

A mis dos hermanos Cesar y Omar por su apoyo y comprensión.

Gracias a mis tres niños David, Alex y Kevin por cada una de sus risas y sus ocurrencias que llenan día a día mi vida de alegría y armonía.

Gracias niño por la oportunidad de conocerte, de amarte, de compartir tu tiempo conmigo. Con tu apoyo y tu alegría fue mucho más fácil este camino (todavía aún me pregunto si esto hubiera sido posible si tu te hubieras ido de mi vida).

A dos grandes ejemplos de profesionalismo Ma. en C.Sara Camacho Cansino y Ma. en C.Judith Jaramillo, Gracias por compartir sus conocimientos, por enseñarme a amar la carrera de Matemáticas Aplicadas y Computación y mostrarme el Arte de las Bases de Datos.

A mi gran amigo y compañero de trabajo de estos cuatro últimos Simón Gaitan... Gracias chonito! por todo tu tiempo, por tu apoyo incondicional y sobre todo por aguantarme todos mis arranques. En verdad deseo que esta amistad perdure por siempre.

A todas y cada una de mis amigas y amigos Roxana Ordaz, Judith Nieto, Clara Garay, Adriana Ordaz, Arturo Carpio Gracias por su apoyo y sobre todo esos momentos que hemos compartido juntos. En verdad deseo que esta amistad perdure por siempre.

Gracias a mi amigo y Jefe Daniel B. Alcántara por todo su comprensión y aguante.

ÍNDICE

	PAG
INTRODUCCION	5
CAPITULO I. Organización del Secretariado Técnico del Servicio Fiscal de Carrera (STCSFC)	
1.1. Introducción al Servicio Civil de Carrera	11
1.1.1. Servicio Civil de Carrera	12
1.1.1.1. Antecedentes Históricos Internacionales	14
1.1.1.2. Antecedentes Históricos en México	17
1.1.1.3. Servicio Civil de Carrera en México	19
1.2. Servicio Fiscal de Carrera en el Servicio de Administración Tributaria	21
1.3. Estructura Organizacional del Servicio Fiscal de Carrera	22
1.3.1. Organigrama de Área	22
1.3.2. Objetivos y Funciones de las principales Áreas	23
CAPITULO II. Situación Actual del Sistema de Reclutamiento y Selección	
2.1. Introducción al Sistema de Reclutamiento y Selección	29
2.2. Descripción de los principales procesos del Sistema	29
2.3. Descripción de procesos de manteniendo del Sistema	35
2.4. Arquitectura Tecnológica	39
2.5. Sistema de Replicación de las Bases de Datos	41
2.6. Problemática Actual del Sistema de Reclutamiento y Selección	46
CAPITULO III. Reestructuración del Diseño Lógico de las Bases de Datos del Sistema de Reclutamiento y Selección	
3.1. Análisis de la Estructura de las Bases de Datos	53
3.1.1. Síntomas del Sistema Actual	53
3.1.2. Alcance de la propuesta de Reestructuración	54
3.2. Reestructuración del Diseño Lógico	54
3.2.1. Actualización de Tablas	56
3.2.1.1. Reestructuración de la Integridad de Dominio	58
3.2.1.2. Reestructuración de la Integridad de Entidad	62
3.2.1.3. Reestructuración de la Integridad Referencial	63
3.2.2. Eliminación de Tablas	67
3.2.3. Eliminación de Campos	67
3.2.4. Actualización Índices	69
3.2.5. Actualización de Stored Procedures	70
3.2.6. Actualización de Triggers	73



CAPITULO IV. Propuesta de Migración e Instalación de la Bases de Datos del Sistema de Reclutamiento y Selección.

4.1	Reestructuración del Diseño Físico de las Bases de Datos	83
4.1.1	Concepto de Diseño Físico	83
4.1.2	Actividades	83
4.2	Configuración de SQL Server	84
4.2.1	Memoria	84
4.2.2	Actividad de E/S.	85
4.3	Diseño físico de las Bases de Datos	86
4.3.1	Sistema de Discos	86
4.3.2	Definición de los Archivos de Datos	90
4.3.3	Definición de la Estructura de Datos	94
4.3.4	Migración de Datos	95
4.4	Propuesta de Mantenimiento de las Bases de Datos	102
4.4.1	Estrategia de Recuperación	102
4.4.2	Estrategia de Respaldo	105
4.4.3	Seguridad en las Bases de Datos	108
4.5	Duplicación de Información	110
4.5.1	Introducción a la Duplicación	110
4.5.2	Modelo de duplicación	110
4.5.3	Metodologías de Duplicación	113
4.5.4	Métodos de Implementación	116
4.2.3	Implementación de la Duplicación	117

CONCLUSIONES	123
---------------------	------------

ANEXOS. (CD-ROM)

- ANEXO A. Descripción de Procesos.
- ANEXO B. Diagramas Entidad – Relación del Sistema Actual.
- ANEXO C. Reportes de Modificaciones en la Estructura Actual
- ANEXO D. Publicaciones del Sistema de Reclutamiento y Selección.

BIBLIOGRAFÍA	127
---------------------	------------

INTRODUCCIÓN

En nuestra vida diaria, en cualquier tipo de organización siempre surge como una necesidad inevitable una buena administración de la información. Debido a ello, la mayoría de las organizaciones dedica un número considerable de recursos a la recolección, clasificación, procesamiento e intercambio de esta.

Innumerables organizaciones llevan a cabo su administración de la información en forma sistemática, con la finalidad de alcanzar sus objetivos de manera eficiente. Uno de estos organismos es el Sistema de Administración Tributaria (SAT) que por medio del Secretariado Técnico de la Comisión del Servicio Fiscal de Carrera (STCSFC) y la Coordinación a la Carrera Fiscal ha diseñado una herramienta capaz de automatizar el Reclutamiento y Selección de personal, mediante un proceso de aplicación y calificación de exámenes. Esta herramienta le permite contar con información veraz y oportuna relacionada con la atención de solicitudes de cobertura de vacantes, evaluaciones de los aspirantes, cartera de candidatos, etcétera.

En los últimos años, los constantes cambios en la tecnología de la computación, la subsecuente disminución de costos, las crecientes demandas de los usuarios han provocado una mejora en la administración y manipulación de la información, mediante nuevas y efectivas herramientas de desarrollo. Aunque cabe mencionar que desafortunadamente la mayoría de las organizaciones no están preparadas para estos constantes cambios, por ejemplo el STCSFC, por lo que es necesario llevar a cabo una renovación y actualización de sus herramientas para satisfacer las nuevas y crecientes necesidades de la Coordinación del Ingreso a la Carrera Fiscal.

Ante este hecho, se desarrolla el presente proyecto en el cual se ven reflejados los conocimientos adquiridos durante la Carrera de Matemáticas Aplicadas y Computación y durante mi desarrollo Profesional. El papel que desempeñe fue de Líder de Proyecto, y participe desde su fase inicial hasta su implementación; además fui elemento de Capacitación y Soporte hacia los usuarios informáticos y operativos.

El objetivo del proyecto es mejorar la operación actual y garantizar controles efectivos que permitan a la coordinación a proveer un mejor servicio al Servicio de Administración Tributaria y a todo personal que aspire a formar parte de esta entidad.

El sistema de Reclutamiento y Selección trabaja con un diseño de Bases de Datos distribuidas a través de replicaciones hacia 63 localidades distribuidas a nivel



Nacional, lo cual provocó en su primer implantación grandes pérdidas de tiempo, dinero y esfuerzos por parte de recursos humanos. Por tanto con la elaboración de este proyecto se pretende también contrarrestar las deficiencias que se presentaron en la primera implantación de este sistema.

Es importante destacar que objetivo inicial del proyecto consistía solo en la migración de una versión de SQL Server a otra más reciente, cuya finalidad era el hecho de utilizar las nuevas características de esta última versión como lo es la duplicación de mezcla (replicación bidireccional). Ahora bien, este hecho representaba llevar a cabo la reinstalación de los 63 servidores a nivel nacional, lo cual implicaba un costo tanto de recursos humanos como materiales. Ante este hecho que resultaba ser inevitable se considero que era factible llevar a cabo la optimización y reestructuración de las bases de datos en conjunto con la migración, sin embargo no era factible cambiar radicalmente el diseño de la base de datos debido a la dimensión y complejidad del sistema. Por tanto el alcance que se pretende con esta reestructuración, es el de llevar a cabo un diseño más eficiente, confiable y con mayor seguridad que se adapte al sistema sin tener que hacer cambios que impacten en el código fuente de los aplicativos, a menos que se considere realmente necesario. Quizás posteriormente se tenga la oportunidad y el tiempo para llevar a cabo también un rediseño en los aplicativos de forma que el sistema pueda aprovechar totalmente las bondades que le dan un buen diseño lógico de base de datos.

El proyecto comprende el conocimiento general del sistema, su problemática y el modelo solución, lo cual se desglosa alrededor de cuatro capítulos. En el primer apartado se describirán los antecedentes del Servicio Fiscal de Carrera, su estructura organizacional así como sus funciones y relaciones con otras áreas con las que comparte información, con la finalidad de comprender el origen, la esencia y la importancia del Sistema de Reclutamiento y Selección de Personal.

En el segundo capítulo se realiza una breve introducción a la situación actual del sistema donde se describen los modelos de proceso principales, el funcionamiento a nivel nacional, plataformas de desarrollo en las que trabaja, características de los equipos en los que se implantó. Además se incluye una pequeña evaluación del sistema, la cual nos permitirá describir la problemática actual.

El tercer capítulo se enfoca a reestructurar el modelo lógico de las bases de datos actual mediante la detección de posibles errores y la aplicación de técnicas de diseño de bases de datos con la finalidad de contrarrestar la problemática actual mediante el incremento de la integridad y consistencia de la información.

El cuarto capítulo se enfoca a reestructurar el modelo físico de las bases de datos mediante estrategias de almacenamiento, configuración, instalación y duplicación de información que aseguren un mínimo de riesgos en la instalación e incrementen el desempeño de nuestra aplicación.

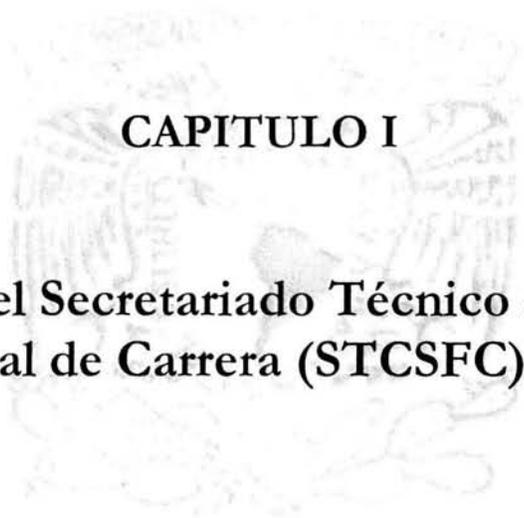


Introducción

En forma general lo expuesto en este documento va dirigido al administrador del sistema de Reclutamiento y Selección, a usuarios del sistema y a personas que se desenvuelvan en el ámbito informático especialmente en el ambiente de las Bases de Datos debido a que este comprende el uso de buenas prácticas de implementación de una base de datos desde su diseño lógico hasta su mantenimiento e instalación. Comprende además conceptos de Replicación de Base de Datos (llamado Duplicación a en las recientes versiones de SQL Server) desde la perspectiva de Microsoft.

Por otro lado puede aportar información valiosa a personal de Recursos Humanos involucrados en el Reclutamiento y Selección de Personal de una organización.

Se espera que el presente trabajo no solo sea la base para mejorar el Sistema de Reclutamiento y Selección, sino que realice aportaciones valiosas para el diseño de las bases de datos de muchos otros sistemas. Pretende además, aportar algunas ideas sobre la instalación, administración y configuración de bases de datos en SQL Server 2000.



CAPITULO I

Organización del Secretariado Técnico del Servicio Fiscal de Carrera (STCSFC)

*ALGUNOS SUEÑAN CON TENER GRANDES LOGROS, MIENTRAS OTROS
PERMANECEN DESPIERTOS Y LO REALIZAN*

ANONIMO.

CAPITULO I

ORGANIZACIÓN DEL SECRETARIADO TÉCNICO DEL SERVICIO FISCAL DE CARRERA.

En el presente capítulo se presentará una breve reseña sobre como nace el concepto de Servicio Civil de Carrera a nivel Internacional así como su surgimiento en México a fin de entender que es y como se establece el Servicio Fiscal de Carrera en la Secretaria de Hacienda y Crédito Público. Además se describe su estructura organizacional, sus funciones y sus relaciones con otras áreas con las que intercambia información.

Al final del capítulo el lector podrá comprender el origen, la esencia y la importancia del Sistema de Reclutamiento y Selección de Personal del Servicio de Administración Tributaria.

1.1 INTRODUCCIÓN AL SERVICIO CIVIL DE CARRERA

En las últimas dos décadas el tema del servicio civil de carrera o de la profesionalización en el ámbito de la administración pública y de la instauración de un servicio civil de carrera, ha sido objeto de atención y estudio por parte de una diversidad de personajes del mundo tanto académico como administrativo, aunque con muchas variantes y objetivos, pero siempre intentando propiciar y fomentar en los trabajadores estatales la vocación para el servicio y por realizar una auténtica carrera profesional dentro del sector público.

Su aceptación y práctica es una realidad que está presente en muchos países desarrollados, donde se empieza a vislumbrar esta política gubernamental que, si bien es cierto ha obedecido a una iniciativa de la parte oficial, también constituye el resultado de una política pública sustentada en una ciudadanía más exigente de ser atendida con eficacia y eficiencia, con honestidad y profesionalismo.

La función del control y de evaluación en los distintos ámbitos de la administración pública federal estatal y municipal constituye una de las nuevas tareas que conlleva a contar con un sistema de rendición de cuentas óptimo.

Actualmente con la supervisión, la vigilancia del ejercicio del gasto además de la observancia preventiva y la sanción de las conductas irregulares de los servidores públicos, han constituido un importante mecanismo institucional en el gobierno, el cual le permite exigir mayor honestidad, además de prevenir y sancionar las



acciones de corrupción que todavía se presentan en algunas de las áreas de la administración pública.

Llevar a cabo estas tareas mediante la ejecución de programas permanentes de auditorías financieras hacia las dependencias, o bien a través de amonestaciones, sanciones económicas y destitución de los funcionarios que incurran en irregularidades, indudablemente requiere de la conformación de verdaderos cuadros profesionales que garanticen mayor calidad, eficiencia y eficacia en la práctica de estas funciones.

El servicio civil de carrera se encuentra en todo tipo de países con los más variados sistemas de economías, costumbres, razas, etcétera. Por citar a algunos que cuentan con un sistema debidamente integrado están: Inglaterra, Alemania, Canadá, Estados Unidos, Suiza, Francia y Japón.

Dentro de los esquemas de administración pública de dichos países operan modelos con el fin de estimular la permanencia de los servidores públicos en sus cargos, obteniendo así la continuidad en planes y programas que permiten la generación y aprovechamiento de experiencias que se traducen en metas y objetivos a plazos más cortos y con menores recursos.

1.1.1 SERVICIO CIVIL DE CARRERA

Definición

El Servicio Civil de Carrera es el conjunto de acciones sistemáticas mediante las cuales los servidores públicos pueden ingresar, permanecer y desarrollarse profesionalmente dentro de la administración pública. Este sistema incluye diversas etapas operativas que son: reclutamiento, selección, inducción al puesto, capacitación, evaluación del desempeño, promociones e incentivos y retiro.

Propósitos

Entre los propósitos más importantes del servicio público de carrera, señalaremos los siguientes:

- 1) Se busca un mejor funcionamiento del sector público, con un enfoque claro y determinante hacia la eficiencia y calidad en la prestación de los servicios;
- 2) Ser un programa de aplicación obligatoria en todos los niveles y órdenes de gobierno, de manera que los esfuerzos sean homogéneos y exista una normatividad que lo regule;
- 3) La utilización de la capacitación constante y la especialización como instrumentos fundamentales a efecto de contar con personal idóneo y comprometido para desempeñar la diversidad de acciones que componen la actividad del estado;



- 4) El abandono y erradicación de vicios que han identificado a la corrupción como característica de toda función pública; y
- 5) Lograr la planeación estratégica de largo plazo, así como la continuidad y seguimiento de los programas de gobierno, a fin de llevarlos hasta su total conclusión, sin importar los relevos de funcionarios.

En condiciones óptimas, los propósitos anteriores permiten a la administración pública obtener ventajas, tales como garantizar la estabilidad y seguridad en el empleo de los servidores públicos, sin que ello implique inamovilidad. Fomenta asimismo, la lealtad hacia las instituciones públicas, la vocación de servicio, la continuidad a la actividad gubernamental; promueve la planeación de vida y carrera de los servidores públicos y mayores grados de responsabilidad; cuenta con empleados formados desde su ingreso y capacitados permanentemente para tener calificación y competencia más elevada; propicia la transparencia en todos los puestos, reduce la corrupción y pondera el mérito como fundamento para la permanencia y la promoción.

Costos

La capacitación tanto general, como especializada, para contar con personal altamente competente, idóneo para desempeñar las distintas actividades de la gestión gubernamental implica un alto costo de inversión, tanto para capacitación, como en salarios y prestaciones para la administración pública, que no todos los gobiernos están dispuestos a erogar (No es posible tomar al servicio público de carrera como una moda o simplemente barnizar de calidad a las organizaciones; estamos en presencia de un sistema costoso con resultados a largo plazo, en el que vale la pena invertir).

Objetivo

Cualquier modelo de Servicio Civil busca enlazar los intereses del estado con los de los trabajadores, como empleador y prestadores de servicios, relaciona directamente el desarrollo individual con la obtención de los objetivos y metas institucionales, así que el Servicio Civil tiene como objetivo la congruencia de la actitud y la aptitud del servidor público.

Principios

El servicio civil debe sustentarse en dos principios fundamentales:

- 1) El sistema de valuación y aptitud en el puesto y
- 2) La garantía de seguridad y estabilidad en el cargo que incite la profesionalización de los servidores públicos.

Requisitos

Para que el servicio civil de carrera pueda ponerse en marcha, se requiere del apoyo permanente de los mandos medios y superiores de la administración pública, que exista un régimen democrático, que el servidor público vea a la Administración Pública como una opción importante para ejercer su profesión,



debe también establecerse una planificación de los recursos humanos, estructurando la administración pública conforme a sus propias necesidades, vinculando el servicio civil al desarrollo nacional siendo necesario que el diseño de la carrera se ajuste a la realidad histórica y contextual.

La eficiencia y la capacidad profesional son la base para establecer un sistema de reclutamiento y ascenso en el servicio civil. Su esencia radica en la idoneidad del candidato a ocupar el puesto de acuerdo a los requisitos que previamente se delinearón como indispensables para cada categoría ocupacional y más que propiciar expectativas de mejoramiento en el ingreso para los servidores públicos, debe representar un reto y compromiso con superación, por lo que no debe ser sólo un sistema de premios y estímulos económicos, que no generarían a largo plazo más que desconfianza y falta de compromiso de servicio por los mismos servidores.

El servicio civil busca evitar que los grupos que llegan al gobierno intenten permanecer en la acción administrativa y desterrar definitivamente la ambición por obtener beneficios solo para sus integrantes, ubicando en los puestos públicos a quienes comparten su tendencia política y aseguran un alto grado de lealtad.

En conclusión se considera que el servicio civil de carrera debe buscar su fundamento en principios y valores como son: la eficiencia, la seguridad en el empleo, la productividad del servidor público, de tal forma que para obtener una administración eficiente, debe estar presente la profesionalización, un sistema de selección, un sistema permanente de capacitación y un sistema de movilidad horizontal y vertical.

1.1.1.1 ANTECEDENTES HISTÓRICOS INTERNACIONALES

El Servicio Civil de Carrera no es un modelo o esquema reciente. Existen antecedentes en pasajes bíblicos, en la antigua Grecia, en el Imperio Romano e incluso en las sociedades precolombinas. A continuación analizaremos el servicio civil en algunos países para resaltar su importancia y vigencia internacional.

Inglaterra

En 1855, Inglaterra creó la Comisión del Servicio Civil, para supervisar los exámenes de ingreso establecidos por los Departamentos en ciertos empleos. Posteriormente la Comisión MacDonell propuso reformas entre 1912 y 1915 en las que se sugiere el ingreso de hombres de negocios para dirigir los Departamentos de Guerra los cuales ingresaron con ideas nuevas y fortalecieron el Servicio Civil de Carrera.

En 1929 la Comisión Tomlin, de acuerdo a lo establecido por el comité de Reorganización, admitió las clases que fueron tres: la “Clerical”, la “Executive” y



la “Administrative”; cada una divididas en categorías. Desde entonces se les conoce a todas ellas con el nombre de Tesorería.

Entre las características que distinguen al Servicio Civil de Inglaterra, podemos citar las siguientes:

- Unidad dentro de la variedad.
- Jerarquía basada en las diferentes funciones que desempeña cada clase.
- La permanencia de ciertas tradiciones aristocráticas, especialmente en la clase superior (“administrative class”), equivalente a los subsecretarios, directores generales y jefes de sección.
- Una basta experiencia, adquirida a través del tiempo y no tan sólo por la teoría.
- Sin derecho a huelga.
- El apoliticismo¹ de las clases superiores, en cuanto a funcionarios, bajo el argumento de conservar la imparcialidad y fidelidad tradicionales en servir a cualquier partido político que gobierne.
- Los métodos de selección de personal se basan en tres principios fundamentales:
 - Ausencia de favoritismo e influencias de partidos políticos.
 - Selección fundada ante todo en las cualidades humanas y los conocimientos generales de los candidatos.
 - Unidad del órgano de selección (Civil Service Department). Cuyos poderes son cuasi-judiciales.

Este sistema incorpora en la Administración a los que reúnen cualidades personales superiores y una cultura general amplia pues considera que la especialización puede adquirirse mediante la práctica del oficio y cursos especiales.

Estados Unidos

Por su parte el Servicio Civil Estadounidense, es un modelo basado en méritos, toda vez que promueve un nombramiento se funda en la capacidad probada, lo que asegura que el servidor público, se desempeñe cumpliendo con eficiencia sus deberes y obligaciones.

Asimismo, el Acta de Servicio Civil de 1883, estableció que la admisión a ciertos cargos dentro de la administración pública se determinaba por la competencia y exámenes abiertos a los candidatos, quienes deberían probar su capacidad para ocupar el puesto.

Entre las características más significativas del Servicio Civil, podemos citar las siguientes:

- Existe movilidad que permite ascender a las clases superiores.
- La remuneración está en función directa de la clase y la antigüedad.

¹ Apoliticismo. Actitud de indiferencia o desinterés frente a la política.



- Los servidores públicos no están organizados en sindicatos y tampoco tienen derecho a huelga.
- Cada servidor público es responsable de la disciplina que guarde ante sus superiores y se procura mantener una alta eficiencia en el personal.
- Los criterios para seleccionar personal se basan en los siguientes principios:
 - No existe favoritismo e influencia de partidos políticos.
 - La selección se basa ante todo en las cualidades humanas y las habilidades y conocimientos especializados.
 - Existe una Unidad del Órgano de Selección llamada “Service Comision”.
 - En el sistema norteamericano es muy importante la lealtad al sistema.

La legislación sobre servicio civil está contenida en el Estatuto de Servicio Civil, su Reglamento, la Ley de Salarios de la Administración Pública, el Código del Trabajo y otros decretos y disposiciones específicas que complementa la legislación.

Francia

En Francia existe un sistema de función pública de carrera y, salvo algunas excepciones, está compuesta de profesionistas, reclutados y formados para el ejercicio de una actividad al servicio del Estado, con lo cual aseguran una función pública de por vida y profesionalizada.

El reclutamiento se hace por concurso abierto a todos los ciudadanos franceses siempre que cubran los requisitos exigidos. Sus resultados son revisados por un jurado independiente de la administración y del gobierno, que después de una serie de pruebas, escogidas para evaluar las cualidades o conocimientos necesarios para ejercer las funciones emiten su dictamen. Los jurados de reclutamiento de funcionarios están compuestos por académicos universitarios, pero en una gran mayoría de profesionistas que ya desempeñan alguna función pública.

En este sistema, no puede privarse del empleo al servidor público por razones injustas, políticas o de índole personal o incluso por errores personales, pero no permanecerá en su cargo y deberá cambiar de asignación.

China

El gobierno en China era misión de un grupo de personas con características que las hacen idóneas para gobernar (contratadas a través de un examen de capacidad); se seleccionaban entre todo el pueblo, pero éste no tenía nada que opinar respecto a las medidas. “El pueblo –dice Confucio- está hecho para seguirlos (a los decretos superiores) pero no puede entenderlos”.

En el sistema absolutista chino, existían tres tipos de servidores del Imperio que se correspondían con tres niveles y grados de examen:



- 1) Los del primer nivel o shin-yün (estudiante del pueblo). Existía un número limitado de plazas y los exámenes eran evaluados por el director del distrito.
- 2) Los de nivel medio o chüjen (hombre recomendado). Se reclutaban de los del nivel inicial mediante un examen mayor ante el director provincial y formaban parte de los funcionarios menores que podían ascender a un mayor cargo que era el de director provincial.
- 3) Los de nivel superior o director provincial. Se reclutaban de los de nivel medio mediante un examen ante el director provincial.

1.1.1.2 ANTECEDENTES HISTÓRICOS EN MÉXICO

Sobre el Servicio Civil de Carrera en México encontramos antecedentes desde el Impero Azteca hasta nuestros días. A continuación se describe una breve reseña.

El Imperio Azteca

En la Ciudad de México-Tenochtitlan el régimen político se integraba de instituciones religiosas, militares y administrativas, que evolucionaban constantemente sobresaliendo la fiel observancia de sus leyes mediante una impecable, eficiente y honesta actuación de todo el engranaje gubernamental y administrativo.

Las altas clases sociales no constituían elites cerradas; “todo azteca sin distinción de clase, que demostrara su destreza para el combate, culto, comercio o vocación artística, pasaba a formar parte de la clase guerrera, sacerdotal, comerciante o de la clase de los artistas; solo los que no sobresalían en estas actividades, estaban condenados a permanecer macehualli²”.

Para la selección de cualquier servidor público, desde el Tlatoani³ hasta el último de ellos, los aztecas tomaban en cuenta las siguientes características:

- Espíritu de servicio.
- Honestidad.
- Capacidad física e intelectual para el puesto a desempeñar.
- No tener vicios.
- Un interés por el mejoramiento de la comunidad.

La Colonia

En la administración de recursos humanos del sector público, durante la Colonia no se tenía una clasificación de puestos, es decir, no estaba definida, pues los trabajadores desempeñaban diversas funciones; para la selección de servidores públicos, era requisito indispensable ser español de nacimiento para los puestos

² macehualli. Vasallo, hombre de pueblo.

³ Tlatoani.- Gobernante de las ciudades-estado mexicanas.



más importantes (virrey, alcalde, delegado, etc.). El criollo⁴ sólo tenía acceso a los puestos de menor importancia dentro del gobierno.

El aprendizaje o capacitación de los servidores públicos de la Colonia se daba dentro del desempeño de sus funciones en forma empírica.

México Independiente

Para una sociedad que se encuentra de lleno en una guerra y lucha por su independencia es complicado elaborar y aplicar un sistema de administración de personal como hoy se conoce, pero se debe indicar que en esta época no había sistemas de elección de los servidores públicos; ni clasificación de puestos, aunque se inician los intentos de su establecimiento. No se contaba con sistema de capacitación; los ascensos y/o promociones se daban comúnmente por los golpes de Estado.

La Reforma

El Presidente Benito Juárez intentó establecer orden y racionalidad a la función pública, a él se le debe el inicio de la reestructuración de la administración pública logrando en cierto modo la clasificación de puestos, aunque no en forma sistematizada.

El Porfiriato

En 1908 John Kenneth escribía: “El soborno es una institución establecida en las oficinas públicas mexicanas y reconocida como un derecho que corresponde al funcionario que ocupa el puesto”.

Existían dos funciones principales adscritas a cada puesto público:

- 1) El privilegio que consiste en usar las facultades especiales del puesto para amasar una fortuna personal.
- 2) El deber que consiste en impedir a la gente emprender cualquier clase de actividad que pueda poner en peligro la estabilidad del régimen existente.

Al analizar el sistema de Díaz en 1908, Andrés Molina Enríquez, lo resumió en una frase: “amistad personal”. En este sistema de amigos, los gobernadores estaban ligados al presidente, los jefes políticos a los gobernadores y los presidentes municipales a los jefes políticos”.

Con el tiempo Porfirio Díaz se rodeó de intelectuales y pseudo intelectuales positivistas conocidos con el nombre de científicos, los cuales influyen en las actividades de su gobierno, y por ende, en la administración de personal.

Con influencia de José Yves Limantour, Díaz estableció la carrera de empleados en la Escuela de Contaduría y Administración que deberían cursar los aspirantes a

⁴ Criollo. Descendiente de padres europeos nacido en Hispanoamérica.



empleados públicos. Con ello se dieron los primeros intentos por establecer un sistema de capacitación, sin cristalizar en nada concreto.

Por último para el sistema de ascensos y/o promociones se publicó un trabajo titulado “Guía práctica del Empleado de la República Mexicana”, en el cual se visualizan algunos lineamientos de lo que actualmente conocemos como sistema de méritos, base y sustento del ascenso y la promoción, pensión y jubilación (el servidor público no tenía seguridad social o derecho a una vejez digna y tranquila por jubilación).

1.1.1.3 SERVICIO CIVIL DE CARRERA EN MÉXICO

Actualmente en nuestro país, se han sentado las bases para el servicio civil, en específico, el artículo 123 de la Constitución Política de los Estados Unidos Mexicanos, en su apartado “B”, detalla:

- VII. La designación del personal se hará mediante sistemas que permitan apreciar los conocimientos y aptitudes de los aspirantes. El Estado organizará escuelas de Administración Pública.
- VIII. Los trabajadores gozaran de derechos de escalafón a fin de que los ascensos se otorguen en función de los conocimientos, aptitudes y antigüedad.

Bajo este esquema, el Gobierno Estatal plasma en el Plan de Desarrollo del Estado de México 1993-1999, la necesidad de aumentar la efectividad social del aparato administrativo, haciendo sus actuaciones productivas, ejerciendo un control adecuado que garantizara el pleno ejercicio de los recursos asignados.

Sin embargo, para lograr esto también se considera la necesidad de profesionalizar a los servidores públicos y de esta manera garantizar que su desempeño se lleve a cabo con legalidad, honradez, lealtad, imparcialidad y eficiencia, situaciones que en el Ejecutivo Estatal, y durante los casi seis años de la presente administración, han sido el punto de arranque de su actuación y el motor que ha permitido contar con las personas adecuadas en el lugar idóneo.

Es importante señalar, que en congruencia con lo establecido en el citado plan, una de las adecuaciones realizadas en la presente gestión gubernamental fue aprobada la Ley del Trabajo de los Servidores Públicos del Estado y Municipios, que contempla un sistema de profesionalización de los servidores públicos generales, en cuyo contenido destaca:

- La definición de un catálogo de puestos por dependencia, que debe contener el perfil, requisitos, nivel salarial y escalafonario.
- La implementación de un régimen escalafonario.
- Estructuración de programas de capacitación.



El responsable de este proceso es la comisión mixta de escalafón, presidida por el titular de cada dependencia, quien será responsable de vigilar que el sistema de profesionalización se desarrolle de acuerdo a lo establecido en la propia Ley.

Si bien el tema de la profesionalización de los servidores público en México ha sido recurrente y en el ámbito académico ha habido innumerables señalamientos sobre su importancia y necesidad, poco se ha logrado en cuanto a la instauración de un servicio civil en el ámbito federal. En cambio, ha habido importantes avances en la instauración de sistemas de carrera en organismos especializados, lo cual, si bien es limitado ha demostrado la eficacia de los servicios civiles como instrumentos para profesionalizar la función pública, mejorar los resultados y asegurar la rendición de cuentas.

La primera experiencia en este sentido fue la creación del Servicio Exterior, en 1934, dentro de la Secretaría de Relaciones Exteriores. Desde entonces se reconocía la importancia de contar con profesionales en la materia que tuvieran la posibilidad de desarrollar una carrera y, con ello, profesionalizar su trabajo y crear una memoria institucional.

Recientemente se han creado un buen número de servicios de carrera en organismos especializados (véase el cuadro 1). Es posible señalar que las características comunes de las instituciones en que se han establecido estos sistemas son, en términos generales, dos: 1) Instituciones con funciones específicas (claridad en fines y en medios); y 2) Instituciones que, en mayor o menor medida, gozan de autonomía respecto al gobierno federal, lo que les permite escapar del “laberinto jurídico” de la administración central, el cual restringe las posibilidades de innovación. Estas condiciones, claro está, son necesarias aunque no suficientes para instaurar servicios civiles en organismos públicos.

INSTITUCION	SERVICIO DE CARRERA
Comisión Nacional del Agua	Sistema de Especialistas en Hidráulica
Instituto Federal Electoral	Servicio Profesional Electoral
Instituto Nacional de Estadística, Geografía e Informática.	Sistema Integral de Profesionalización
Procuraduría Agraria	Servicio Profesional Agrario
Procuraduría General de Justicia del Distrito Federal.	Instituto de Formación Profesional
Procuraduría General de la República	Carrera de Agente del Ministerio Público Federal
Secretaría de Educación Pública	Carrera Magisterial
Secretaría de Relaciones Exteriores	Servicio Exterior Mexicano
Sistema de Administración Tributaria (Secretaría de Hacienda y Crédito Público).	Servicio Fiscal de Carrera

Cuadro 1 Servicios de Carrera en México.



Existen otros casos, como el Banco de México, donde si bien no existe formalmente un servicio civil, en la práctica se han consolidado ciertas tendencias, como la valoración de la experiencia y el mérito, que han arrojado buenos resultados.

Existen, además, algunos proyectos específicos para instaurar servicios de carrera en el gobierno del Distrito Federal, en la Cámara de Diputados y en la Comisión Nacional Bancaria y de Valores, entre otros.

También en el ámbito estatal y municipal se ha reconocido la conveniencia de establecer sistemas que valoren el mérito y permitan la continuidad y profesionalización de la función pública; sin embargo, los avances, de nuevo, han sido sumamente limitados.

1.2 SERVICIO FISCAL DE CARRERA EN EL SERVICIO DE ADMINISTRACIÓN TRIBUTARIA

Como consecuencia de toda esta tendencia del Servicio Civil de Carrera nace en 1990 el Secretariado Técnico del Servicio Fiscal de Carrera bajo el nombre de Subadministración de Selección y Evaluación dentro de la Subsecretaría de Ingresos. Esta se crea como una estrategia para mejorar el ejercicio laboral, haciendo énfasis en el comportamiento ético y profesional del personal.

En el año de 1992 se integra el Despacho de Asesoría en Calidad y Competitividad en el área de consultoría e investigación de esta Subadministración, con la responsabilidad de apoyar el proceso de adecuación de las técnicas poligráficas y psicométricas de acuerdo a las necesidades de la Subsecretaría de Ingresos.

En 1993, la Subadministración se reestructura en sus aspectos administrativos y técnicos orientando la capacitación al modelo de Selección laboral mediante el análisis Poligráfico. En esta época esta área es transferida a la Dirección General de Inventoría. Las funciones de esta Subadministración se orientan primordialmente a realizar mediante el desarrollo de técnicas poligráficas y psicométricas, la selección y evaluación del personal humano tanto en los niveles operativos, en los directivos de las distintas áreas clave de la Subsecretaría de Ingresos y particulares que así lo requerían.

Dada la importancia de las funciones encomendadas a la Subadministración de Selección y Evaluación, y con la finalidad de coadyuvar a la transparencia del proceso de selección y evaluación, se demanda un sistema que apoye de forma integral los diferentes procesos para seleccionar y evaluar candidatos a ocupar los puestos claves para la Subsecretaría de Ingresos para el ejercicio de actividades relacionadas con el comercio exterior, mismo que le permitirá registrar, controlar y dar seguimiento a las evaluaciones practicadas a los candidatos, emitiendo



productos necesarios en cada uno de los procesos, asegurando la confiabilidad e Integridad de la Información registrada en cada uno de los expedientes.

En 1997 se da una desconcentración de las Subsecretarías de la Secretaría de Hacienda y Crédito Público (SHCP), por tanto la Subsecretaría de Ingresos pasa a ser un órgano autónomo bajo el nombre de Secretaria de Administración Tributaria (SAT) y con ella nace el Secretariado Técnico del Servicio Fiscal de Carrera cuya finalidad es la dotar de personal a todo el SAT.

Es así como nace el Secretariado Técnico del Servicio Fiscal de Carrera (STCSFC) con un nuevo concepto de Reclutamiento y Evaluación de Personal con la finalidad de satisfacer las nuevas necesidades de personal del SAT a nivel Nacional.

1.3 ESTRUCTURA ORGANIZACIONAL DEL SERVICIO FISCAL DE CARRERA

El Servicio de Administración Tributaria (SAT) es un órgano desconcentrado de la SHCP con el carácter de autoridad fiscal, además de contar con las atribuciones y facultades ejecutivas que señala la ley.

El Secretariado Técnico de la Comisión del Servicio Fiscal de Carrera tiene la finalidad de dotar al Servicio de Administración Tributaria de un cuerpo de funcionarios fiscales altamente calificado, profesional y especializado, el cual estará sujeto a un proceso permanente de capacitación y desarrollo integral, con base en un esquema de remuneraciones y prestaciones que coadyuven al cumplimiento óptimo de su objetivo.

1.3.1 ORGANIGRAMA DEL ÁREA

Una de las áreas del Secretariado Técnico de la Comisión del Servicio Fiscal de Carrera, es la Coordinación de Ingreso a la Carrera Fiscal a la que pertenece la Subadministración de Reclutamiento de Personal y la Subadministración de Evaluación de Personal, que tienen como propósito principal el reclutamiento y selección de personal de los Servidores Fiscales de Carrera y cuyas actividades están estrechamente ligadas al objeto de estudio de nuestro trabajo.

A continuación se muestra el organigrama con la finalidad de ubicar dentro del Secretariado Técnico a la Coordinación de Ingreso a la Carrera Fiscal, además de mostrar las relaciones que existen entre otras coordinaciones, que aunque no sean nuestro objeto de estudio servirán de referencia para explicar algunas de las diferentes actividades que se llevan a cabo por parte de la Coordinación de Ingreso a la Carrera Fiscal (Véase Figura 1.1).



1.3.2 OBJETIVOS DE LAS ÁREAS

- **COORDINACIÓN DEL INGRESO A LA CARRERA FISCAL.**
Garantizar la capacitación y selección de los mejores candidatos para cubrir las vacantes que se presenten en el Servicio de Administración Tributaria, así como asegurar la actualización de los instrumentos de evaluación que integran el sistema automatizado.
- **SUBADMINISTRACIÓN DE RECLUTAMIENTO**
Asegurar que se capten aspirantes que cubran los requisitos generales de los puestos que se convocan para ingresar al Servicio Fiscal de Carrera y obtener candidatos susceptibles, por su escolaridad, experiencia y características de personalidad, a realizar el proceso de selección de Funcionarios Fiscales de Carrera.

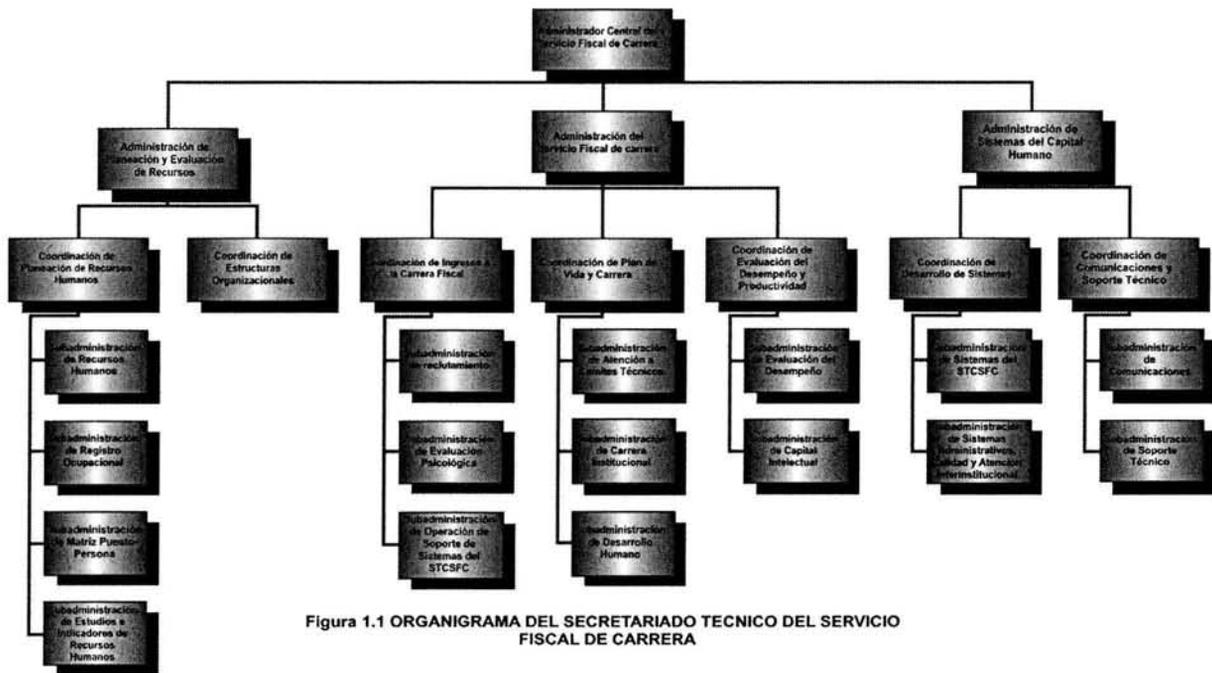


Figura 1.1 ORGANIGRAMA DEL SECRETARIADO TECNICO DEL SERVICIO FISCAL DE CARRERA

- **SUBADMINISTRACIÓN DE EVALUACIÓN PSICOLÓGICA**
Analizar en los aspirantes, sus conocimientos, experiencia, capacidades, habilidades, valores y características de personalidad que permitan contar con elementos predictivos de desempeño laboral para, decidir su ingreso al Servicio de Administración Tributaria.



- **SUBDIRECCIÓN DE OPERACIÓN Y SISTEMAS DEL SERVICIO FISCAL DE CARRERA**
Administrar, coordinar y proporcionar seguridad de la información del Sistema, además de proporcionar soporte técnico a las localidades distribuidas a nivel Nacional y a sus usuarios respecto a la operación del Sistema.

FUNCIONES DE LAS ÁREAS

- **COORDINACIÓN DEL INGRESO A LA CARRERA FISCAL.**
 - Coordinar y supervisar la realización de los procesos de reclutamiento y selección de aspirantes a puestos clave del Servicio de Administración Tributaria.
 - Dirigir el diseño, el desarrollo y el mantenimiento del sistema de reclutamiento y selección aplicable a los Funcionarios Fiscales de Carrera.
 - Asesorar técnicamente a las áreas de Administración de Recursos así como a los encargados de esta función a nivel regional y local, para optimizar la operación del área.
 - Promover la investigación de técnicas alternativas que mantengan actualizadas las baterías de pruebas psicométricas conforme las investigaciones que en estas materias se realicen.
 - Presentar indicadores estadísticos que reflejen la productividad del área.
- **SUBADMINISTRACIÓN DE RECLUTAMIENTO**
 - Recibir las solicitudes de Cobertura de Vacantes y verificar que el puesto existe en la plantilla ocupacional y en el catálogo de puestos.
 - Actualizar y verificar en la cartera de candidatos si existen personas que cubran los requerimientos del puesto solicitado.
 - Elaborar y difundir las convocatorias para allegarse aspirantes a servidores fiscales de carrera.
 - Recibir, atender y revisar que los aspirantes cumplan con lo establecido en la convocatoria, que cubran el requisito de temporalidad de evaluaciones anteriores y en caso de reingreso revisar la causa de la baja.
 - Aplicar a los aspirantes prueba de preselección.
- **SUBADMINISTRACIÓN DE EVALUACIÓN PSICOLÓGICA**
 - Coordinar el proceso de Reclutamiento y Selección a los aspirantes a ocupar puestos vacantes del Servicio de Administración Tributaria, mediante un



modelo de análisis de adecuación de las características de los aspirantes a los requerimientos de los puestos.

- Supervisar las integraciones de los reportes de evaluación del perfil de puesto persona, y en su caso realizar las adecuaciones.
 - Analizar y determinar las baterías de exámenes⁵ psicológicas que se aplicarán a los diferentes niveles de funcionarios fiscales.
 - Coordinar la automatización en el sistema de selección de todas las pruebas psicométricas y de conocimiento teórico-prácticas susceptibles de aplicar por computadora.
 - Realizar los estudios estadísticos que se requieran para actualizar la estandarización de las normas existentes a la población del Servicio de Administración Tributaria.
 - Realizar estudios de viabilidad de las técnicas y/o métodos de selección que existan en el mercado, de forma que se diseñe la creación de nuevos instrumentos psicológicos, de acuerdo a las necesidades específicas del Servicio de Administración Tributaria.
 - Realizar estudios estadísticos de los perfiles de puestos y candidatos con el propósito de constatar que correspondan a la realidad de la población evaluada o en su caso proponer los ajustes necesarios.
- SUBADMINISTRACIÓN DE OPERACIÓN Y SISTEMAS DEL SERVICIO FISCAL DE CARRERA
 - Coordinar y proporcionar seguridad de la información del Sistema.
 - Proporcionar soporte técnico a los usuarios respecto a la operación y fallas técnicas del Sistema.
 - Coordinar y Supervisar el flujo de la Información.

Es así como el lector debe darse una idea clara respecto al concepto del Servicio Fiscal de Carrera del Servicio de Administración Tributaria y con ello darse cuenta de su importancia.

⁵ Baterías de exámenes. Conjunto de exámenes que se aplican a un aspirante de acuerdo al perfil del puesto específico por el cual participa.

CAPITULO II

Situación Actual del Sistema de Reclutamiento y Selección

*ALGUNOS SUEÑAN CON TENER GRANDES LOGROS, MIENTRAS
OTROS PERMANECEN DESPIERTOS Y LO REALIZAN
ANONIMO.*

CAPITULO II

SITUACIÓN ACTUAL DEL SISTEMA DE RECLUTAMIENTO Y SELECCIÓN

En este capítulo se ilustra la situación actual del sistema, para ello se describe a nivel general los modelos principales de proceso, su funcionamiento a nivel nacional, las plataformas de desarrollo y las características de los equipos. También se realiza una pequeña evaluación a fin de identificar la problemática actual del sistema.

2.1 INTRODUCCIÓN AL SISTEMA DE RECLUTAMIENTO Y SELECCIÓN DE PERSONAL

El Sistema de Reclutamiento y Selección de Personal se concibió como una herramienta necesaria para automatizar el Reclutamiento y Selección de Personal del Servicio de Administración Tributaria. Este sistema permite a la Coordinación de la Carrera Fiscal mantener integrado, actualizado y disponible el expediente de los aspirantes, proporcionando en cada fase del proceso de selección y evaluación los productos requeridos para la toma de decisiones.

2.2 DESCRIPCIÓN DE LOS PROCESOS PRINCIPALES DEL SISTEMA

El sistema es capaz de evaluar diferentes procesos, lo cuales se pueden clasificar en procesos internos y externos.

Los procesos internos son aquellos en los se evalúa al personal para su ingreso al SAT por primera vez.

- Reclutamiento y Selección.

Los procesos externos son aquellos en los cuales se evalúa al personal que ya se encuentra elaborando actualmente en el SAT.

- Promociones.
- Detecciones.
- Evaluación posterior a la Evaluación al Desempeño.

Solo describiremos el proceso de evaluación interno, debido a que es el que provee un panorama global. Los otros procesos funcionan de manera similar, pero no incluyen todas las actividades.



Proceso de Evaluación Interno

Este proceso comprende dos fases:

1. Fase de Reclutamiento. Es la fase de evaluación mediante la cual se captan aspirantes que cubran los requisitos generales del puesto de forma de obtener candidatos susceptibles, por su escolaridad, experiencia y características de personalidad. Esta fase comprende las siguientes actividades¹:
 - 1) Recepción y trámite de la Solicitud de Cobertura de Vacante.
 - 2) Consulta de Cartera de Candidatos.
 - 3) Elaboración de Convocatorias para cubrir puestos vacantes.
 - 4) Recepción de aspirantes y revisión de documentación.
 - 5) Programación y pruebas de Preselección.
 - 6) Obtención de Resultados del aspirante.
 - 7) Programación a pruebas de Selección.

El diagrama de Procesos de Reclutamiento (Figura 2.1). Secuencia de actividades y las áreas que se encuentran involucradas.

Flujo de datos del Proceso de Reclutamiento

Como se muestra en la Figura 2.2 el proceso se inicia con la necesidad de cubrir una vacante en el Servicio de Administración Tributaria. Con base a ello la Coordinación de Plan de Vida y Carrera lleva a cabo el registro y emisión de una solicitud de cobertura de vacante, siempre que cumpla los requisitos como son: la existencia de la unidad administrativa, el puesto en plantilla así como que realmente se encuentre vacante dicha plaza. Posteriormente se consulta la cartera de candidatos con la finalidad de encontrar un candidato que cumpla con los requerimientos de la plaza, en caso de no encontrarlo se procede a elaborar y registrar una convocatoria, con base al catálogo de perfiles de puesto; catálogo de fuentes de reclutamientos además de parámetros como lugar, fecha y horas donde deberán presentarse los aspirantes. En el período de registro la Subadministración de Preselección procede a revisar la documentación del aspirante, verificando que cumpla con los requisitos establecidos en la convocatoria; en este caso que se cubra el perfil; que no exista en la bitácora de aspirantes rechazados (si existe, que su fecha de registro en el archivo de aspirantes rechazados sea mayor a 24 meses); que no exista en el inventario de recursos humanos (si existe indicar el motivo de la baja). Si el candidato no cumple con alguno de estos requerimientos se da por concluido el proceso y se anexa a la bitácora, en caso contrario se le proporciona al aspirante una solicitud, con la cual se complementa el registro del aspirante. En ese momento se realiza su programación al proceso de preselección que consiste normalmente en la aplicación de dos exámenes (un examen de conocimientos generales y uno psicológico). Por último se le hace entrega de una ficha de

¹ Ver Anexo A. Descripción detallada de las actividades de la Fase de Reclutamiento.



examen la cual le va a permitir al candidato ingresar a las instalaciones a presentar su evaluación. Una vez que el aspirante haya concluido sus exámenes se procede a calificar en forma automática y a actualizar el expediente con los resultados obtenidos, separando a los rechazados (generando para ellos su carta de agradecimiento), y los aprobados que cumplen con los requerimientos solicitados para el puesto, los cuales continúan a la siguiente fase del proceso que es la fase de Selección.

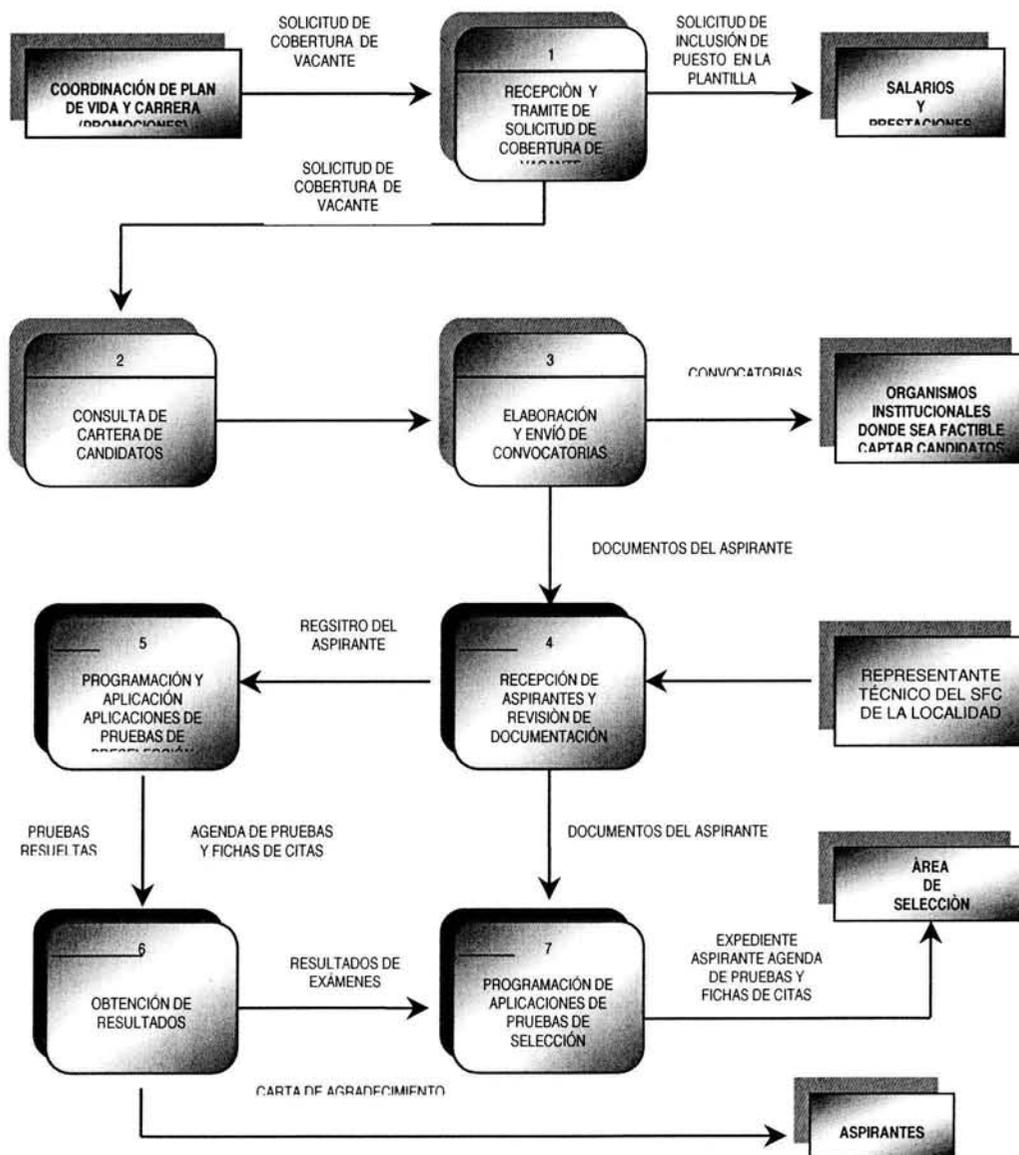


Figura 2.1 Diagrama de Procesos de Reclutamiento.



Migración y propuesta de Instalación de las Bases de Datos del Sistema de Reclutamiento y Selección de Personal del Servicio de Administración Tributaria

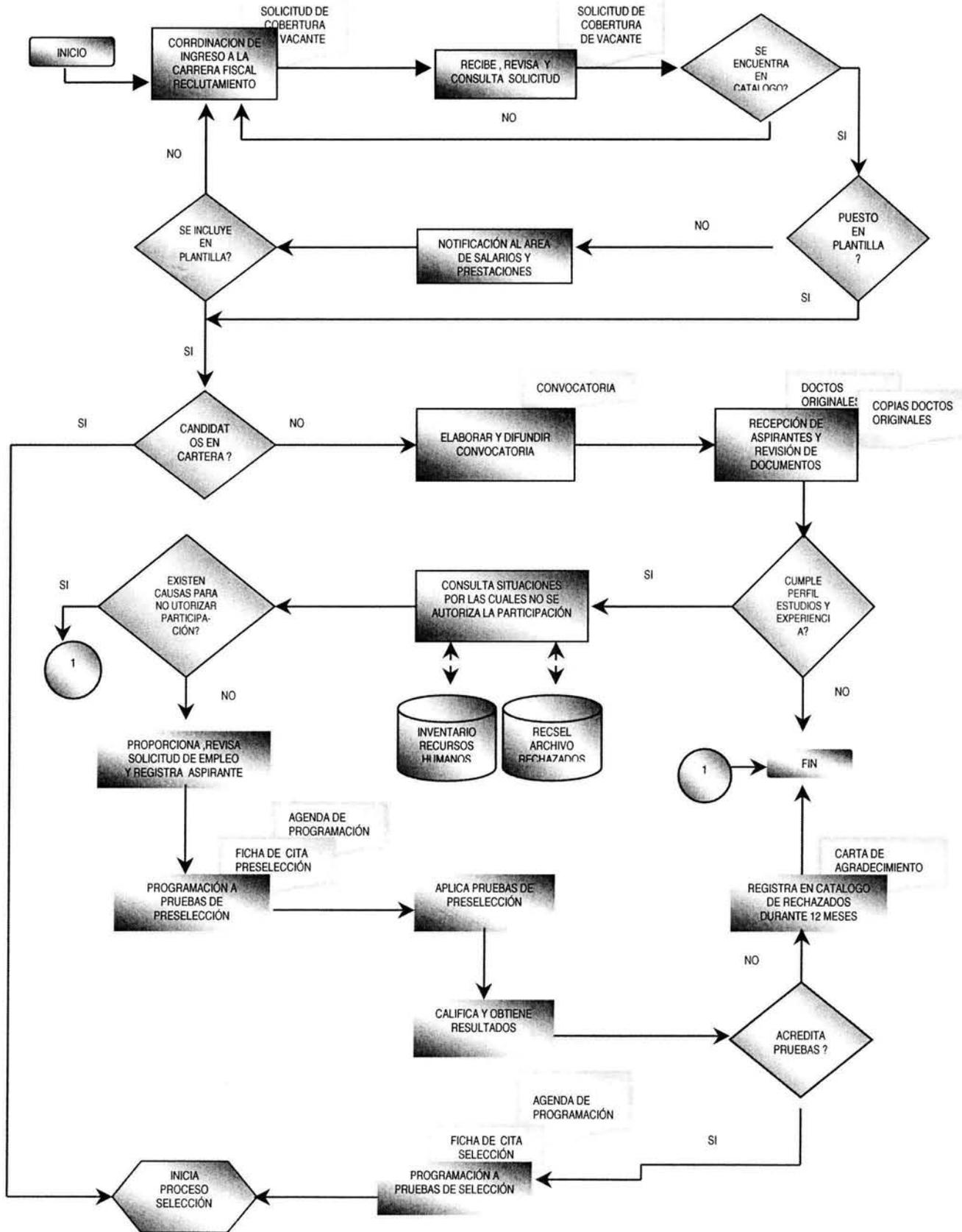


Figura 2.2 Diagrama de Flujo del Proceso de Reclutamiento.



2. Fase de Selección (Evaluación Psicológica). Es la fase de evaluación mediante la cual se analizan los aspirantes que hayan aprobado el proceso de preselección, sus conocimientos, experiencia, capacidades, habilidades, valores y características de personalidad que permitan contar con elementos predictivos del desempeño laboral para decidir su ingreso al SAT con base en la calificación, grado de adecuación al perfil del puesto, obtenidos a través de instrumentos objetivos y confiables. Esta fase comprende las siguientes actividades²:
 - 1) Verificación de agenda de aspirantes programados para prueba de selección.
 - 2) Presentación de exámenes psicológicos y de conocimiento de acuerdo al nivel del puesto y/o área solicitante.
 - 3) Evaluación del resultado de los exámenes con base en el perfil del puesto y el perfil de la persona. En su caso realización de los ajustes correspondientes e identificación de áreas de riesgo.
 - 4) Asignación de plaza a los (2) candidatos mejor calificados para cubrir el puesto vacante.
 - 5) Entrevista de los dos mejores candidatos.
 - 6) Tramitación de examen médico.
 - 7) Tramitación de encuesta socioeconómica.

El diagrama de Procesos de Selección (Figura 2.3) muestra la secuencia de actividades y las áreas que se encuentran involucradas.

Flujo de datos del Proceso de Selección

Como se muestra en la figura 2.4 el proceso de selección del aspirante se inicia aplicando la batería correspondiente, esta batería consta aproximadamente de 5 a 7 exámenes psicológicos dependiendo del nivel y área del puesto además de un examen de conocimientos específicos. Una vez que el aspirante concluye con sus exámenes se procede en forma automática a calificar y a determinar el nivel de adecuación del perfil psicológico del aspirante con el perfil de requerimientos del puesto.

Posteriormente en la Subadministración de Evaluación realizan la interpretación psicológica de los exámenes, en esta se detectan fortalezas, riesgos, observaciones y los datos para la tarjeta amarilla. Registrado lo anterior el sistema realiza la evaluación global tomando en consideración los resultados de los exámenes psicológicos, las observaciones del psicólogo y el examen de conocimientos específicos y determina si el aspirante se integra a la cartera de candidatos o se rechaza. En este último caso se entrega al candidato una carta de agradecimiento y se registra en la bitácora de rechazos por un período de 24 meses.

² Ver Anexo A. Descripción detallada de las actividades de la Fase de Selección.

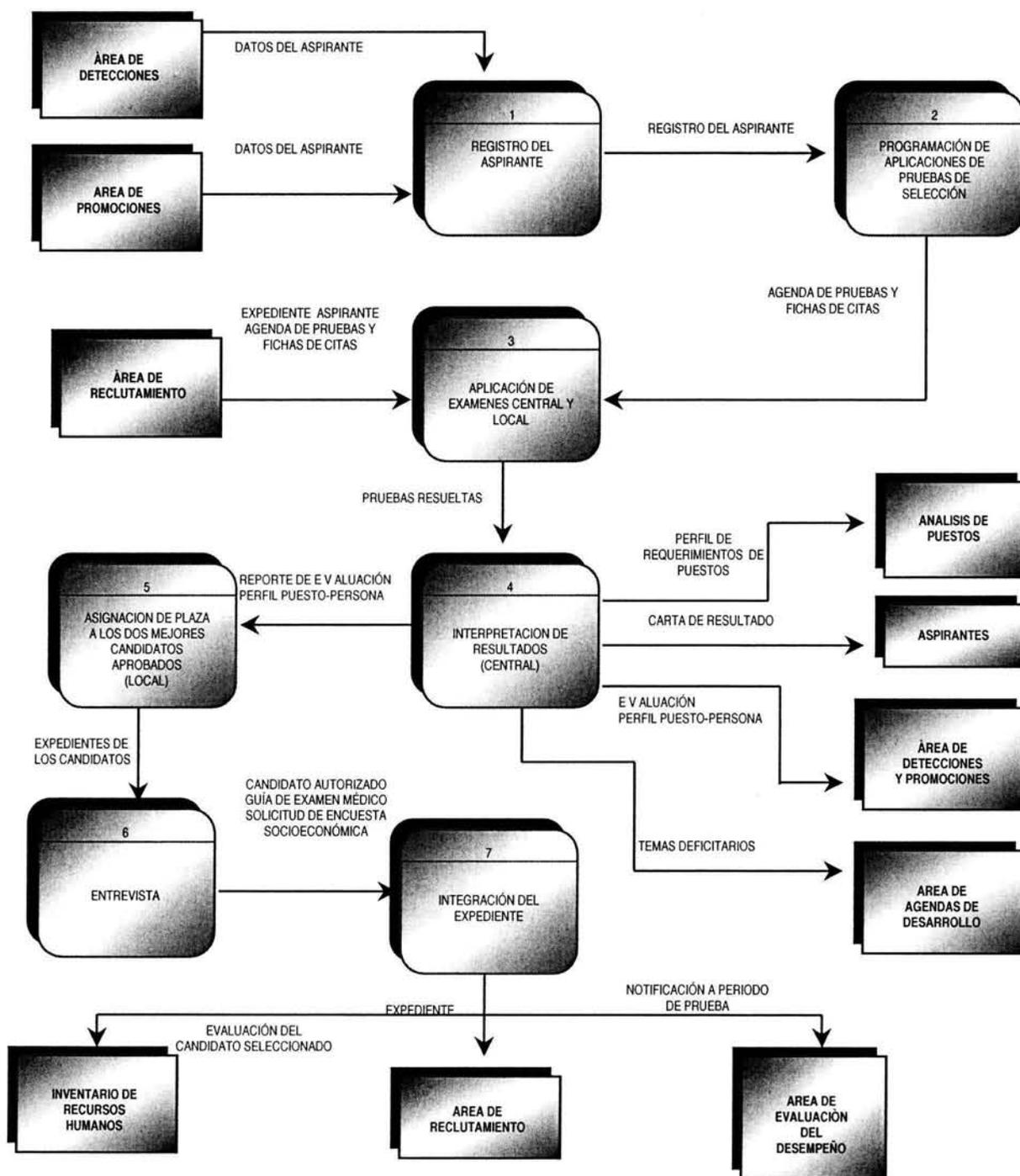


Figura 2.3 Diagrama de Procesos de Selección.



En la cartera de candidatos se asigna la plaza temporalmente a los dos mejores aspirantes y se envían a entrevista con su jefe inmediato. Si el candidato es aceptado por el jefe se procede a integrar su expediente, de forma contraria el aspirante regresa a ser parte de la cartera de candidatos para otra posible plaza con las mismas características. La integración de su expediente consta de un examen médico y una encuesta socioeconómica, si se observa alguna anomalía en alguna de estas evaluaciones el candidato será rechazado por un período determinado dependiendo de la gravedad, en caso contrario el candidato pasará a un periodo de prueba ya en el puesto con duración de seis meses, al final su jefe inmediato registrará una evaluación en un sistema externo llamado Evaluación del Desempeño. Este sistema dispara una última evaluación llamada POST EDD, la cual consta de un examen de conocimientos específicos que contiene preguntas de conocimiento específico del puesto y de conocimiento institucional. Calificado el examen se realiza una integración final de sus evaluaciones lo cual va permitir decidir si el aspirante ocupa definitivamente la plaza o se rechaza.

2.3 DESCRIPCIÓN DE PROCESOS DE MANTENIMIENTO DEL SISTEMA

Ahora bien, en el Sistema de Reclutamiento y Selección existen muchos procesos que retroalimentan y dan mantenimiento a la información. Estos procesos se deben en gran medida al gran número de necesidades y actividades de la Coordinación de Ingreso a la Carrera Fiscal y la Coordinación de Plan de Vida y Carrera.

A continuación se da una breve explicación de cada uno de ellos para dar un panorama general de todas las actividades que engloba este sistema.

ADMINISTRACIÓN DEL SISTEMA

El proceso de Administración del Sistema engloba el mantenimiento de los sitios donde se aplicarán exámenes (centros de aplicación), del mantenimiento de las estaciones donde se realizará la aplicación, de las ciudades, del tipo de evaluación a realizar, y de las unidades administrativas pertenecientes a un centro de aplicación. También controla el mantenimiento de los parámetros utilizados en el sistema, como los días festivos y especiales, la bitácora de rechazados en un periodo de tiempo, la depuración del histórico de rechazados en preselección y selección, la consulta del estado de los exámenes aplicados en cada centro, la depuración de respuestas de exámenes y por último las actividades de cambio de puestos específicos, unidades y/o R.F.C's.

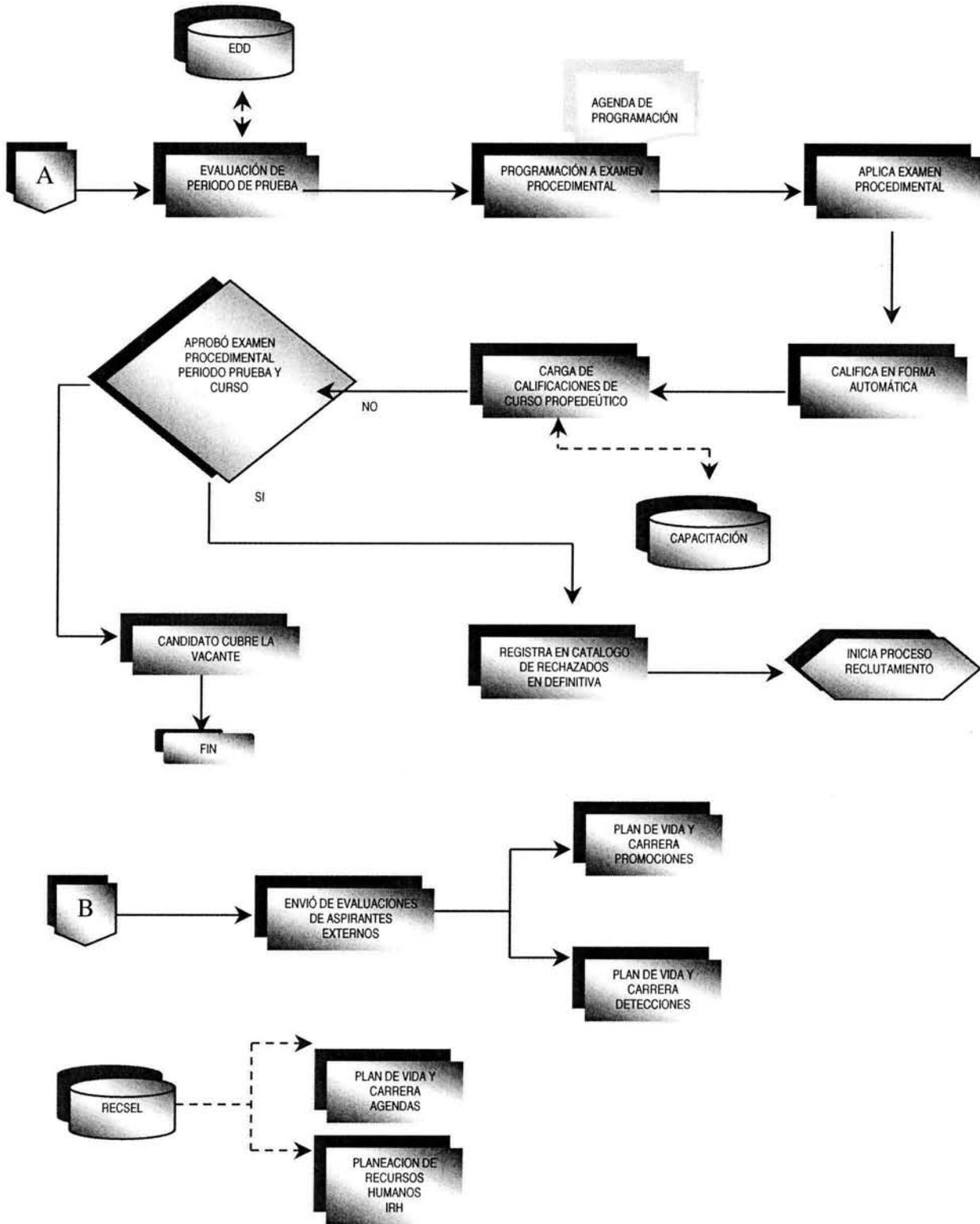


Figura 2.4 Diagrama de Flujo del Proceso de Selección.



SEGURIDAD

El proceso de Seguridad comprende la administración de los usuarios encargados de manejar los diferentes módulos del sistema, es decir, permite definir las funciones a realizar por parte de los usuarios y grupos, también tiene la capacidad de definir la pertenencia de un usuario a un grupo. Cuenta con un control de auditoría encargado de registrar cada uno de los movimientos realizados por los usuarios en los diferentes módulos como altas, bajas, modificaciones, consulta, etc., con la flexibilidad de modificar los diferentes niveles de auditoría que estos registrarán.

BATERÍAS

Las actividades que engloba el proceso de generación de baterías (grupo de exámenes por aplicar de acuerdo al nivel del puesto) son: la definición del conjunto de exámenes para una batería por tipo de proceso, el establecimiento de un orden de aplicación de exámenes. Así mismo la asignación a la batería de un tipo de evaluación y un nivel, es decir que cada batería se define de acuerdo al nivel del puesto, el proceso de evaluación y del tipo de una unidad (local o central).

GENERACIÓN DE EXÁMENES PSICOLÓGICOS Y DE CONOCIMIENTOS GENERALES

Este proceso engloba la creación de exámenes psicológicos y de conocimientos generales que serán aplicados al aspirante. En este se definen los nombres de los exámenes por aplicar, los reactivos que lo componen, las respuestas correctas a los

reactivos, las combinaciones de una o varias respuestas para obtener un puntaje dado, los reactivos y respuestas que se utilizarán en el ejercicio de práctica del aspirante que le permite familiarizarse con la resolución del examen, etc.

MANTENIMIENTO A TEMAS

Este proceso es el encargado de definir el Temario de un Puesto específico, es decir, los temas, subtemas, subsubtemas y subsubsubtemas que correspondan a cada uno de los puestos. El proceso también comprende la administración de los Temarios a nivel Unidades y Subunidades Administrativas. Además da mantenimiento a catálogos como el de Tipo de Tema y Niveles de Dominio los cuales son parte del Temario.

DEFINICIÓN DE EXÁMENES DE CONOCIMIENTOS ESPECÍFICOS

El proceso de generación de exámenes de conocimientos específicos es el encargado de definir los tipos de casos o de preguntas que serán asignados a un tema para aplicar a un aspirante. En el se define el número máximo y mínimo de preguntas a aplicar así como el peso (valor) que tendrán los niveles de dominio de un tema.

APLICACIÓN DE EXÁMENES

Este proceso es el que nos permite aplicar exámenes a los aspirantes. En el se presentan los textos e imágenes de reactivos correspondientes a cada uno de los exámenes definidos por el usuario en el módulo de generación de exámenes.



Además internamente comprende la administración de respuestas del aspirante para su posterior calificación.

CALIFICACIÓN

Este proceso es el encargado de calificar los exámenes presentados que tengan categoría de terminado. En caso de algún percance también existe un proceso que nos permite deshacer la calificación de los exámenes que hayan quedado inconclusos de calificar.

DEFINICIÓN DE PERFILES

Este proceso engloba la definición de los parámetros o factores psicológicos que permitirán al sistema identificar, cuando un aspirante cumple con el perfil requerido para un puesto dado; también se controla la adecuación de rangos para los puestos y los ajustes de factores psicológicos permitidos.

SISTEMA DE EVALUACIÓN POST EDD (Posterior a Evaluación al desempeño)

En este proceso se engloba las actividades que nos permiten obtener la calificación del curso propedéutico tomado por el aspirante, obtener la calificación de la evaluación del desempeño y la calificación del examen procedimental (examen de conocimientos específicos que contiene pregunta de tipo académico y preguntas de procedimiento interno de la institución). Así como la emisión de una carta de agradecimiento al participante o la emisión de una carta de aceptación para ocupar el puesto.

APLICACIÓN DE EXÁMENES SQL (CAPTURA DE EXÁMENES IMPRESOS)

Este proceso permite la captura de exámenes pendientes de los aspirantes que no hayan sido calendarizados para aplicar dichos exámenes a través de un equipo de computo y por lo tanto hayan resuelto el examen de forma manual, de esta forma se incorpora al sistema el resultado de su aplicación de exámenes.

IMPRESIÓN DE EXÁMENES EN CENTRAL

Este proceso nos permite la impresión masiva de exámenes de conocimientos específicos para aspirantes con tipos de evaluación de detecciones, promociones y post edd. Esta aplicación normalmente se usa cuando hay procesos masivos en los cuales no existe suficiente equipo de computo para llevar a cabo la aplicación en PC's.

2.4 ARQUITECTURA TECNOLÓGICA

Actualmente el Servicio de Administración Tributaria cuenta con varias redes locales con Microsoft Windows NT como servidor de Red, así como una red amplia WAN que cubre el país. Esta red WAN es administrada por la Administración General de Tecnología de Información (AGTI) y es la que permite



al STCSFC cubrir sus necesidades a nivel nacional de todos sus departamentos encargados de administrar el personal del SAT.

El Sistema de Reclutamiento y Selección se diseño para trabajar bajo una arquitectura Cliente/Servidor con las siguientes características:

Servidor

Software

Sistema Operativo

Windows NT Enterprise 4.0

Manejador de Base Datos Relacional

SQL Server Enterprise 6.5

SQL Server 6.5 Service Pack 5 a

Otros Componentes

- Windows N.T 4.0 Option Pack 4

Hardware (mínimo).

Memoria.

128 Mb mínimo

Componentes.

- Procesador PentiumII 400 Mhz
- Disco duro con más de 10GB
- Monitor VGA o de resolución superior
- Unidad de disco flexible de 3.5 pulgadas
- CD-ROM
- Tarjeta de red
- Mouse

Cliente

Software

Sistema Operativo

Windows 95, 98 y WorkStation

Otros Componentes

- DCOM 95
- Office 97

Hardware (mínimo).

Memoria.

32 Mb mínimo



Componentes.

- Procesador Pentium a 166 Mhz o superior
- Disco duro con más de 400 MB disponibles para la aplicación
- Monitor VGA o cualquiera de mayor resolución
- Tarjeta de Vídeo con resolución de 800x600 Mínimo
- Unidad de disco flexible de 3.5 pulgadas
- Tarjeta de red
- Mouse

Componentes Opcionales.

- CD-ROM
- Módem de 28.8Kb o superior

NOTA: Aunque en algunos equipos los requerimientos y características suelen cambiar debido a factores externos al STCSFC, la mayoría cumple con lo que se ha mencionado.

2.5 SISTEMA DE REPLICACIÓN DE LAS BASES DE DATOS

La tecnología de replicación es un mecanismo para distribuir automáticamente copias de datos desde un servidor hacia uno o más servidores remotos.

En SQL Server 6.5 la replicación se basa en las transacciones, es decir, no se mandan directamente los datos de servidor a servidor, si no que ejecutan las mismas transacciones en ambos servidores, partiendo de la base de que dichos servidores tienen la misma información (en los datos que se replican).

El Sistema de Reclutamiento y Selección cuenta con 63 centros de aplicación de exámenes (CAE's) y un servidor central, los cuales hacen uso de la tecnología de replicación. Los centros se encuentran distribuidos a nivel nacional y se localizan en 8 zonas geográficas: Noroeste, Noreste, Norte Centro, Centro, Occidente, Sur, Golfo Pacifico, y Área Metropolitana (Figura 2.5 y Figura 2.10). El servidor central se encuentra ubicado en el área metropolitana y su objetivo es concentrar la información referente a las evaluaciones realizadas a nivel nacional.

El Sistema de Reclutamiento y Selección hace uso de cuatro tipos de replications:

- De servidor central a servidor regional.
- De regional a estación de trabajo(Access).
- De estación de trabajo a servidor regional.
- De servidor regional a servidor central.



SQL CENTRAL A SQL REGIONAL

Existen dos clases de Replicación de Central a Regional, la *incondicional* y la *condicional*.

Replicación *incondicional* se aplica para datos que tienen que estar en todos los centros con servidores SQL. A esta clase de replicación corresponden los exámenes y subtests ya que esta información es generada a nivel central y es necesario enviarla a nivel nacional. Para realizar esta replicación es necesario suscribir a todos los CAES a la publicación del servidor central.

Replicación *condicional* es aplicada para los casos en los que se requiere enviar información específica a CAES, pero no enviar la misma información a todos ellos, por ejemplo, a cada CAE son enviadas únicamente las unidades administrativas correspondientes a su localidad. La figura 2.6 nos muestra una parte de la topología actual en la cual se observa como el servidor Central replica a cada uno de los centros de aplicación de exámenes.

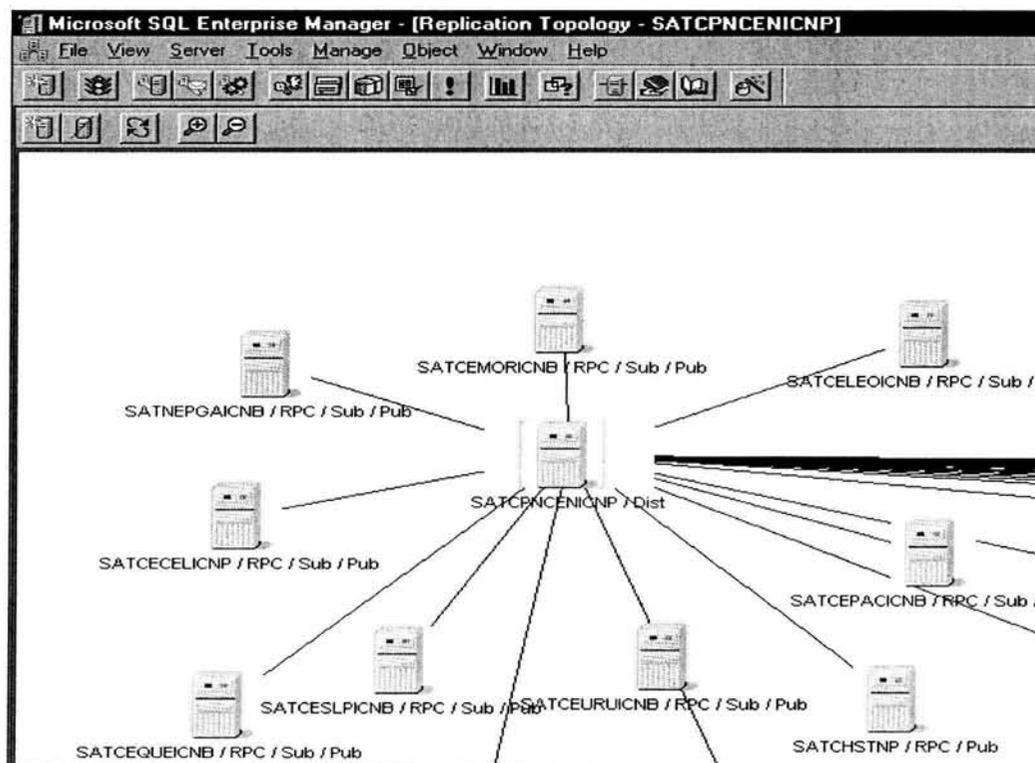


Figura 2.6 Topología de Replicación Actual del Servidor Central a los Centros de aplicación de exámenes(CAES).

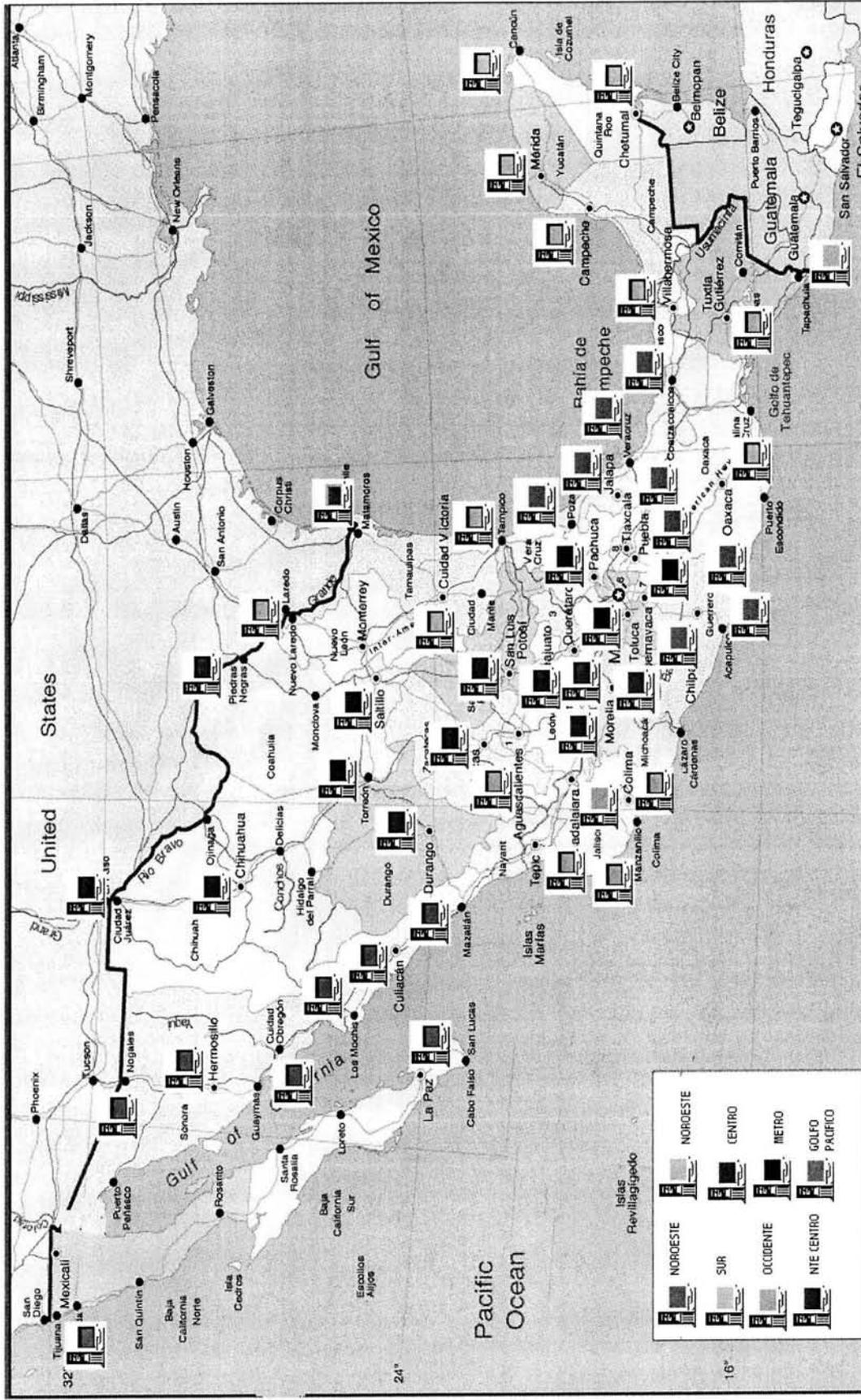


Figura 2.5 Distribución de Centros de Aplicación a nivel nacional.



SQL REGIONAL A SQL CENTRAL

Esta replicación es de tipo incondicional desde los CAE's hacia el servidor central. Esta se aplica para los datos que se generan en todos los CAE's. A esta clase de replicación en general corresponden el registro de aspirantes y los resultados de las evaluaciones de exámenes aplicadas. La figura 2.7 nos muestra la topología actual que existe en el centro de aplicación de exámenes del área metropolitana el cual replica su información al servidor central.

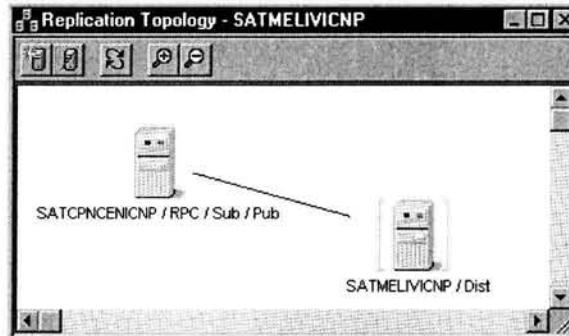


Figura 2.7 Topología de Replicación Actual del Centro de aplicación de exámenes (Metropolitana) al servidor central.

SQL AL CLIENTE ACCESS

La replicación contempla dos procesos (Exámenes y Aspirantes). La replicación de Aspirantes (Rep_S_AA) consiste en el envío de los datos generales del aspirante a una base de datos de Access en una PC local (Figura 2.8a). Al terminar, invoca a la replicación de exámenes (Rep_S_AE) la cual se encarga del envío del conjunto de exámenes que va aplicar el aspirante(Figura 2.8b).



Figura 2.8a Replicación del Servidor SQL Server al Cliente Access (Aspirantes).

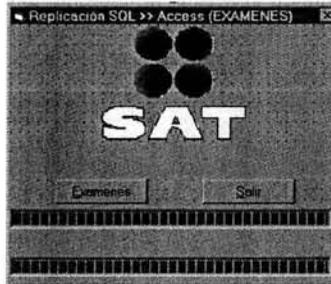


Figura 2.8b Replicación de Servidor SQL Server al Cliente Access (Exámenes)

CLIENTE ACCESS AL SERVIDOR SQL

Esta replicación se ha ligado a la aplicación de exámenes, de manera que cuando un aspirante termina sus exámenes se ejecuta de manera automática (Figura 2.9), aunque también puede operar de manera independiente. Esta consiste en enviar las respuestas de los exámenes de cada aspirante al Servidor Regional a fin de que sean calificadas.



Figura 2.9 Replicación del Cliente Access al servidor SQL

Un ejemplo claro de la aplicación de la replicación en forma conjunta es cuando un aspirante se somete a un proceso de evaluación. Primeramente se debe pasar la información de los exámenes del Servidor Central al Servidor Regional, posteriormente cuando el aspirante tiene su evaluación se replican los exámenes al cliente de Access en una PC local. Una vez que el aspirante termina su evaluación, las respuestas se replican del cliente al Servidor Regional para su calificación, al finalizar el resultado es replicado del Servidor Regional al Servidor Central a fin de que un psicólogo interprete los resultados.



2.6 PROBLEMÁTICA ACTUAL DEL SISTEMA DE RECLUTAMIENTO Y SELECCIÓN

La problemática actual del Sistema de Reclutamiento y Selección de personal recae principalmente en la inconsistencia de información. Estos hechos han provocado que el usuario pierda la confiabilidad. Además se han observado errores técnicos de instalación, configuración de equipos y errores manuales que van desde una mala planeación hasta la captura incorrecta de datos, los cuales causan que la información sea inconsistente, llegue a destiempo o en algunos casos extremos se pierda.

Una de las principales causas de la problemática actual es el diseño deficiente de la base de datos, en particular la definición de los objetos, sus relaciones y las restricciones. La integridad referencial se ha ido perdiendo en un porcentaje considerable, debido en gran medida a los cambios constantes en los requerimientos del usuario, los tiempos acordados, y la deficiente planeación por parte del equipo de trabajo durante el desarrollo del sistema. Por ejemplo:

Podríamos citar el caso en donde el usuario determinó que el examen de Coeficiente Intelectual debía redefinirse esto incluía la eliminación de algunos reactivos. Posteriormente al querer hacer una aclaración de examen³ de un funcionario, el cual había aplicado este examen antes de ser modificado, ya no se encontraron todas las preguntas. La causa de este problema fue que la entidad de respuestas, no tiene definida una referencia sobre la entidad de reactivos en la cual se encuentran definido el banco de preguntas. La solución parcial a este problema fue bajar un respaldo en otro servidor de forma que se lograran recuperar las preguntas.

La frecuencia con la que se presentan este tipo de problemas es relativo, la mayoría de los casos esta en función de la demanda de evaluaciones y de los errores técnicos y administrativos que se presentan. Otro ejemplo que podemos citar es: El registro de 200 funcionarios, podría ser que uno de ellos no se encuentra en la localidad que le corresponde (pongamos Chetumal). El problema se pudo haber presentado por diferentes razones: 1) que a nivel administrativo se haya determinado que esta persona aplicara en otra localidad, por lo que por obvias razones no aparecerá en la localidad de Chetumal; 2) que no existe comunicación con el centro de aplicación lo cual es común en esta localidad, además de que no estar en nuestras manos la solución debido a que es un problema de comunicaciones; 3) que la replicación no esta activa o que haya ocurrido un problema previo a ella, el cual no se ha detectado y por tanto no este permitiendo se que continúe con los siguientes procesos.

³ Aclaración de examen. Reporte que permite revisar la evaluación de un funcionario y constatar su calificación.

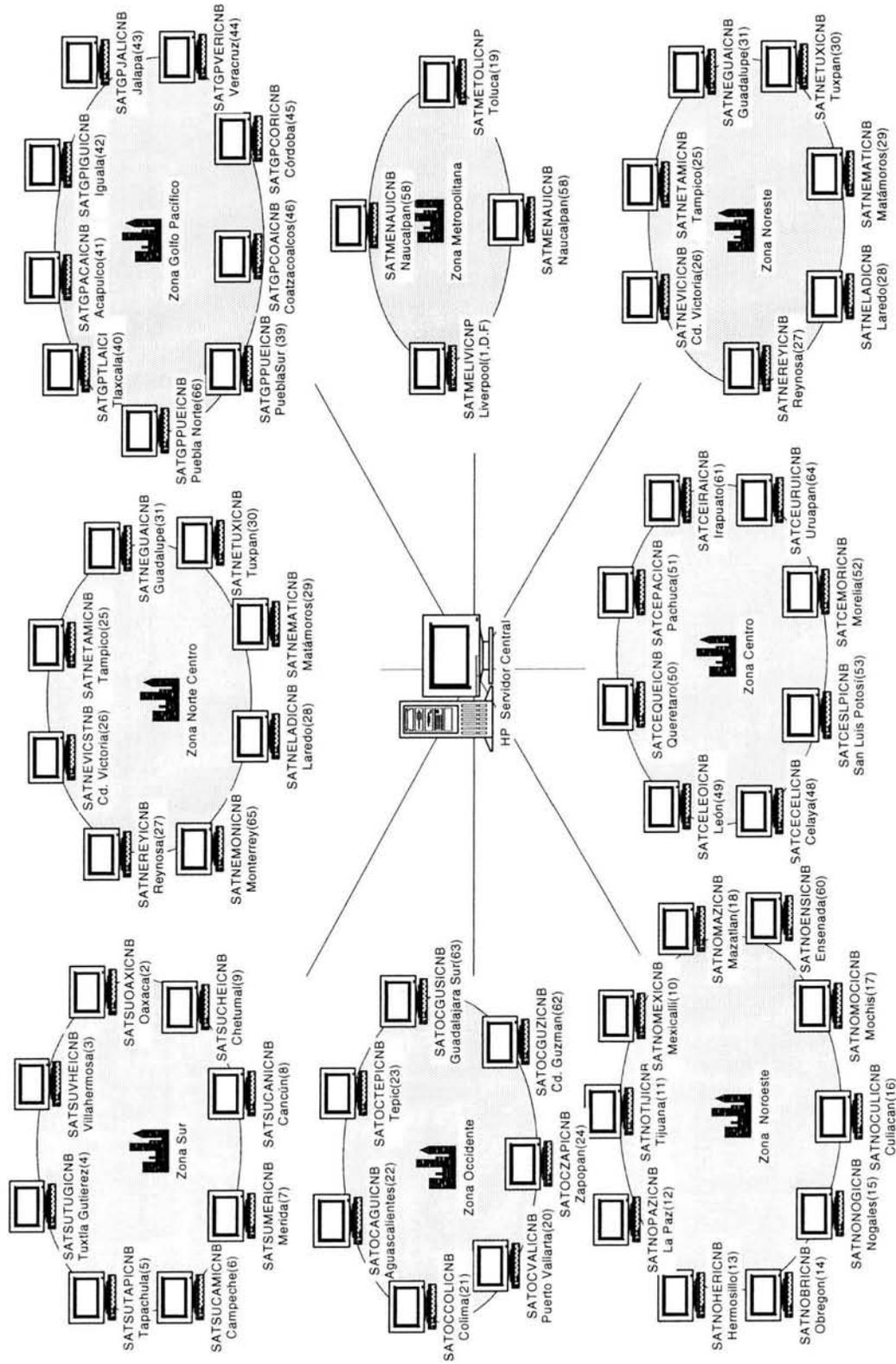


Figura 2.10 Sistema de Replicación Actual.



Por otro lado tenemos que en la última década en la mayoría de las organizaciones de carácter público los recursos económicos han sido limitados en una porción considerable, lo que ha influido en la escasa capacitación del personal, el uso de equipo obsoleto y en la carencia de herramientas que ayuden a supervisar los servidores. Todo esto trae como consecuencia el bajo desempeño de nuestro aplicativo. Sabemos de antemano que este factor es muy difícil de atacar debido a que obedece a políticas internas de la organización que están fuera de nuestro alcance, sin embargo es posible contrarrestarlas haciendo un buen diseño de la base de datos y óptimo uso de las herramientas con las que se cuenta actualmente.

El siguiente diagrama (Figura 2.11) muestra la problemática actual de una forma gráfica, así como los factores que están involucrados en ella.

Con la problemática actual del sistema se da por concluido este capítulo, esperando se haya comprendido en forma general su funcionamiento, las principales actividades de la Coordinación de Evaluación que fueron automatizadas e implementadas así como las deficiencias que presenta.



12/06/2003
SAT / Problemática Actual

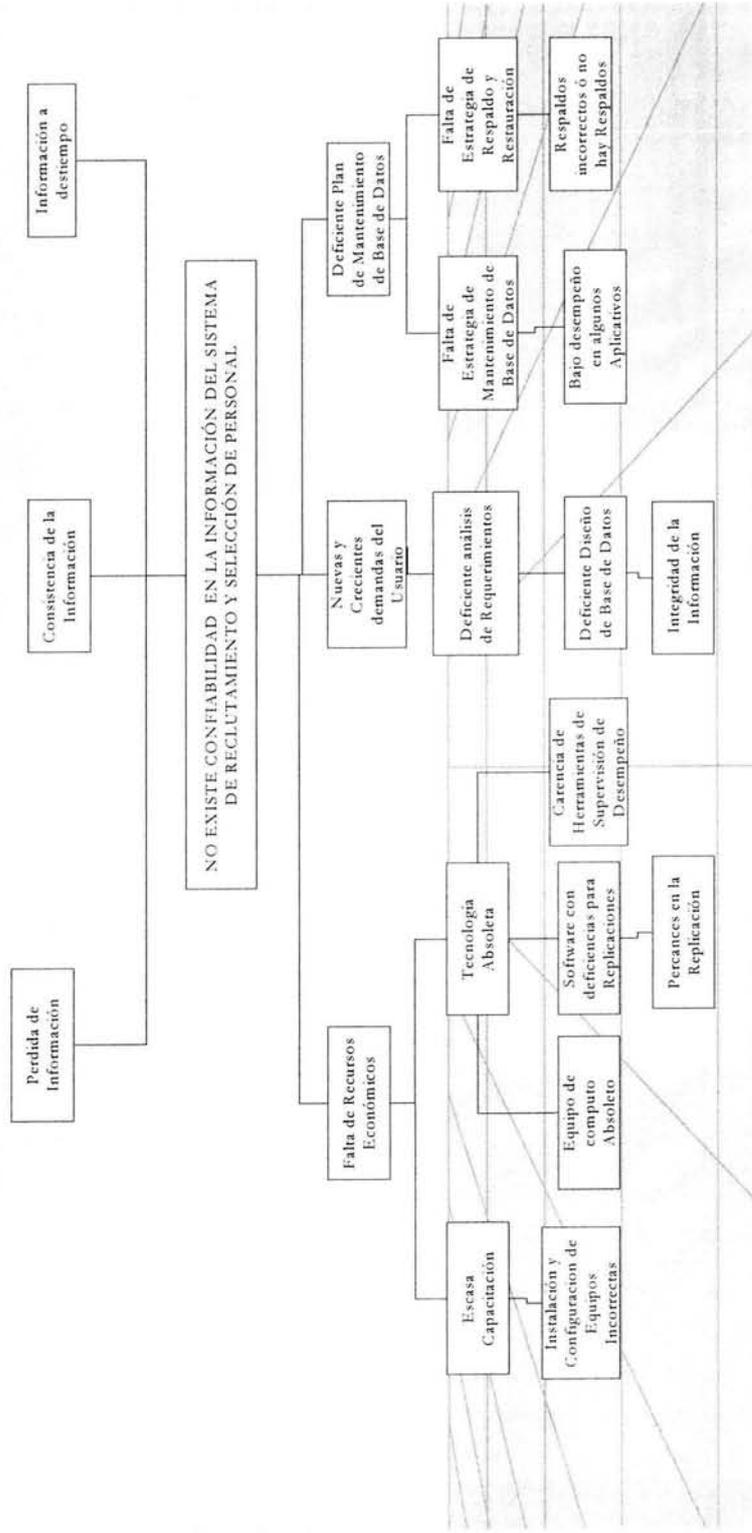


Figura 2.11 Gráfica de la problemática Actual del sistema de Reclutamiento y Selección.

CAPITULO III

Reestructuración del Diseño Lógico de las Bases de Datos del Sistema de Reclutamiento y Selección

COMPARTE LO QUE SEPAS Y APRENDE LO QUE NO SEPAS

ANONIMO.

CAPITULO III

REESTRUCTURACIÓN DEL DISEÑO LÓGICO DE LAS BASES DE DATOS DEL SISTEMA DE RECLUTAMIENTO Y SELECCIÓN.

Este capítulo se enfoca a detectar y analizar los posibles errores del modelo de datos actual, para poder determinar el nuevo diseño lógico para las Bases de Datos del Sistema de Reclutamiento y Selección de personal, con la finalidad de reducir inconsistencias e incrementar la integridad de la información.

3.1 ANÁLISIS DE LAS ESTRUCTURAS DE LAS BASES DE DATOS ACTUALES

3.1.1 SÍNTOMAS DEL SISTEMA ACTUAL

Una de las principales causas de la problemática actual del sistema se debe al deficiente diseño de la base de datos que es generado a raíz de los frecuentes cambios en los requerimientos de los usuarios y de la adaptación de nuevas funciones al sistema (procesos de evaluación como Promociones, Detecciones y PostEDD¹, o procesos de Mantenimiento y Generación de Exámenes de conocimientos específicos).

Este diseño deficiente tiene como consecuencia lo siguiente:

- 1) Información inconsistente.
- 2) Altos tiempos de Respuesta.
- 3) Pérdida de la información.

Un punto importante por mencionar antes de pasar a la reestructuración es el hecho de que el sistema hace uso de dos diferentes estructuras de Bases de Datos, una que da servicio a nivel central y el otro tipo lo conforma la estructura que provee el servicio en cada una de las localidades distribuidas a nivel nacional. La estructura de la base de datos Central tiene como objetivo concentrar toda la información que se genera a nivel nacional, así como proveer al resto de las localidades de información respecto a actualización catálogos, definición de exámenes, baterías etc. La otra base de datos es la Regional que tiene como objetivo concentrar toda la información referente a las evaluaciones aplicadas en la localidad, así como el envío de resultados a nivel Central.

¹ POSTEDD. Proceso de Evaluación posterior a la Evaluación del Desempeño.



3.1.2 ALCANCE DE LA PROPUESTA DE REESTRUCTURACIÓN

Muchas veces, el diseño de la base de datos es fijo como en nuestro caso y su remodelación resulta inevitable. No obstante, incluso en estos casos pueden aplicar técnicas de diseño de bases de datos que garanticen el incremento de la integridad y consistencia de la información con la finalidad de lograr un eficiente funcionamiento del sistema.

Es importante destacar que es imposible cambiar radicalmente el diseño de la base de datos. El alcance que se pretende con esta reestructuración, es el de llevar a cabo un diseño más eficiente, confiable y con mayor seguridad que se adapte al sistema sin tener que hacer cambios que impacten en el código fuente de los aplicativos, a menos que se considere realmente necesario. Quizás posteriormente se tenga la oportunidad y el tiempo para llevar a cabo también un rediseño en los aplicativos de forma que el sistema pueda aprovechar totalmente las bondades que le dan un buen diseño lógico de base de datos.

Ahora bien, el diseño de bases de datos comprende dos elementos: 1) el diseño lógico de la base de datos (como las tablas y restricciones) y; 2) el diseño físico de la base de datos (los sistemas de discos, la colocación de objetos, etc.). Por lo cual este capítulo se enfocará a la reestructuración del diseño lógico de la base de datos y en el siguiente capítulo se retomará lo referente a la reestructuración del diseño físico.



3.2 REESTRUCTURACIÓN DEL DISEÑO LÓGICO.

El diseño lógico de la base de datos incluye la conceptualización de tablas y sus relaciones, por ende se considera un elemento clave de una base de datos relacional aceptablemente optimizada, que es a su vez influye en forma considerable en el rendimiento óptimo de una aplicación.

ACTIVIDADES

Mediante las siguientes actividades y aplicando algunas técnicas de diseño recomendadas por los citados autores en la bibliografía, se pretende atacar las deficiencias del diseño lógico de las bases de datos actuales:

A. Actualización de Tablas

Incremento de la integridad de:

- 1) Entidad.
 - Asignar de manera adecuada identificadores a cada registro de una tabla.
- 2) Dominio.



- Asignar de manera adecuada los tipos de datos.
- 3) Referencial.
 - Redefinir relaciones entre entidades y crear nuevas.
- B. Eliminar tablas o entidades que no sean utilizadas.
- C. Eliminar campos de las entidades que no sean utilizados.
- D. Redefinir Índices.
 - 1) Determinara índices agrupados² para campos por los que más frecuente se hagan las búsquedas.
 - 2) Determinara índices no agrupados³ para campos por los que más frecuente se hagan JOINS.
- E. Actualización de Stored Procedures.
 - 1) Evitar el uso de subconsultas.
 - 2) Evitar vistas que hagan referencia a otras vistas.
 - 3) Deber hacer referencia únicamente a la información que se requiera.
 - 4) Evitar pasar grandes cantidades de información por la red a menos que sea necesario.
- F. Actualización de Triggers.
 - 1) Evitar el uso de subconsultas.
 - 2) Evitar vistas que hagan referencia a otras vistas.
 - 3) Deber hacer referencia únicamente a la información que se requiera.
 - 4) Evitar pasar grandes cantidades de información por la red a menos que sea necesario.

Cabe mencionar que se desarrollo un pequeño programa en Visual Basic (ver figura 3.1) a fin de poder detectar más rápidamente algunas de estas deficiencias en el modelo de datos actual como son: tablas y campos que no se utilizan, campos a los que se les asigno un tipo de datos incorrecto o que en la mayoría de los casos tienen valores nulos, etc. Este programa nos generó un reporte (se incluye en el

² Índice agrupado.- Índice en el que el orden lógico de los valores clave determina el orden físico de las filas correspondientes de la tabla.

³ Índice no agrupado.- Índice cuyo orden lógico es distinto del orden físico almacenado de las filas en el disco.



CD que se anexa) describiendo algunas de las deficiencias mencionadas de tal manera que formo parte de la base para la reestructuración de los objetos de la base de datos.



Figura 3.1. Programa de Revisión del Diseño Lógico de una Base de Datos

3.2.1 ACTUALIZACIÓN DE TABLAS

Un método eficaz para actualizar una tabla consiste en definir todo lo que se necesita en la tabla al mismo tiempo, incluidas las restricciones para los datos y los componentes adicionales como son índices, valores predeterminados, reglas y otros objetos.

Entre las cosas que deben tomarse en cuenta cuando se diseña una tabla, se incluyen:

- Las columnas de la tabla y los tipos de datos para cada columna (así como su longitud, si es preciso).
- Las columnas que aceptan valores NULL.
- La gestión de los valores NULL (ausencia de información), debido a que este valor en muchos procesos puede provocar un mensaje de error.
- Las restricciones o valores predeterminados y reglas.
- Los tipos de índices requeridos donde sea necesario, qué columnas son claves principales y cuáles claves externas.
- Los métodos formales que establece la normalización.
- Asegurar el acceso a cada valor de la base de datos, esto es, usar claves principales que identifiquen de manera única los valores en una base de datos.



Un elemento relevante dentro del diseño de las tablas es la integridad de los datos. Por ello es importante determinar cómo se va a implementar. A continuación describiremos algunos conceptos acerca de la integridad de los datos antes de indicar como se reestructura el modelo de datos actual.

Integridad de los datos

La integridad de los datos garantiza la consistencia y calidad de la información de la base de datos, es decir, que los valores de los datos almacenados en la base de datos son correctos y confiables. Para el diseño de bases de datos relacionales hay tres tipos de integridad de datos:

Integridad de dominio

Establece los valores válidos para una columna específica y determina si los valores nulos son permitidos. Entre los componentes que nos ayudan a especificar que valores son válidos para una columna, están: tipos de datos, la nulidad y restricciones como tipos de datos definidos por el usuario, restricciones check, reglas y valores predeterminados.

Integridad de entidad

Establece que todos los renglones en una tabla tienen un único identificador, es decir, que cada fila de una tabla se puede identificar en forma única. Los componentes que nos permiten establecer la integridad de entidad es el uso de un índice único o mediante la entidad declarativa (restricciones de clave principal o única).

Integridad referencial

Conserva las relaciones definidas entre las filas de datos en varias tablas. La integridad referencial se establece mediante llaves foráneas (foreign keys) y llaves primarias (primary keys) o llaves foráneas y llaves únicas (unique keys). También se puede exigir la integridad referencial con desencadenadores y controlar esta funcionalidad por medio de programación, pero hacer esto requiere de un trabajo más laborioso.

Además de los tres tipos de integridad de datos, hay dos métodos para implementarla, estas son la integridad declarativa y la integridad procedimental. La primera define el criterio que debe reunir el dato como parte de la declaración del objeto, es decir son constraints, defaults y reglas que se definen directamente sobre tablas y columnas de la base de datos. En la segunda se utilizan estructuras de programación y objetos por separado para exigir la integridad de los datos. Esta última puede ser implementada sobre el cliente o el servidor mediante triggers o stored procedures no sin mencionar que su uso es recomendable limitarlo a excepciones o a la implementación de complejas reglas de negocios.

Cabe mencionar que el método más usual es la integridad declarativa, sin embargo para decidir cual es el método más apropiado es necesario involucrar el análisis de varios factores, incluyendo la funcionalidad y el de la aplicación (Ver tabla 3.1).



Componentes de integridad de datos.	Impacto	Funcionalidad	Costo	Antes o después transacción
Constraints	Define con una tabla y valida los datos antes de que comience la transacción resultando un mejor performace.	Medio	Menor	Antes
Defaults y reglas	Implementa la integridad de datos mediante objetos separados	Menor	Medio	Antes
Triggers	Provee de funcionalidad adicional tal como un borrado en cascada o la implementación de una lógica compleja de una aplicación	Alto	Alto	Después

Tabla 3.1. Componentes para implementar la integridad de datos

Nota: Se pueden utilizar tipos de datos definidos por el usuario, valores predeterminados y reglas como mecanismos para exigir la integridad, sin embargo es recomendable utilizar restricciones de integridad de ANSI siempre que sea posible. Estas restricciones son por lo general más flexibles. Además otro beneficio de las restricciones de ANSI es que el optimizador de SQL Server comprende de mejor manera las restricciones ANSI y puede utilizarlas para la mejor toma de decisiones en cuanto a los planes de acceso a los datos.



3.2.1.1 REESTRUCTURACIÓN DE LA INTEGRIDAD DE DOMINIO

El objetivo en el uso de un tipo de datos determinado es expresar de la mejor forma los datos que describe a una entidad de la realidad. La causa principal de algunos problemas de las bases de datos se deben a que cierta información no se almacena de forma correcta. Casi cualquier tipo de datos se puede expresar como una cadena de caracteres. Sin embargo el uso único de un tipo de datos nos puede llevar a ineficiencias. Por ejemplo los enteros de rango entre 0 y 255, se pueden almacenar en un único byte si se almacenan como números. En cambio cuando se almacenan como caracteres la mayor parte de estos números necesitaría dos o tres bytes de almacenamiento puesto que se debe almacenar dos o tres caracteres.

Un modo distinto de ineficiencia es creer que es más rápido almacenar una fecha como una cadena de ocho caracteres que utilizar un valor de tipo fecha, esto debido a que se puede ahorrar algunos ciclos de reloj del tiempo total de procesamiento en una situación específica, pero no se considera el hecho de que el significado de los valores se pierde, además de que se incrementa el trabajo en cuanto a la manipulación y validación de estos datos.



En conclusión mediante el uso adecuado de los tipos de datos, no se producirá una pérdida significativa en el rendimiento, en cambio se ganará en términos de garantizar que los datos son válidos y adecuados.

Estrategia para reestructurar la Integridad de Dominio

Con base en lo anterior se aplicó el siguiente proceso para redefinir los tipos de datos con el propósito de obtener un mayor significado en los datos, reducir espacio de almacenamiento y contribuir a un mejor rendimiento en la obtención de la información:

- Se identificó la longitud máxima o el valor máximo para todos los campos de la base de datos.
- Posteriormente se procedió a comparar su valor máximo con el tipo de datos que anteriormente tenía y si este era considerado como incorrecto se proponía un nuevo tipo de datos para este campo, el cual se consideró más adecuado con base en el conocimiento del dominio (rango) de cada uno de los campos de una tabla.
- Se utilizó el tipo de datos **char** solo cuando se aseguraba que todos los datos de la columna iban a tener prácticamente el mismo tamaño, en caso contrario y por tanto en la mayoría de las veces se utilizó el tipo de datos **varchar**.
- Se cambió los tipos de datos text por un tipo de datos varchar y se les asignó una longitud de acuerdo al dominio del campo, esto porque SQL Server 2000 ya permite utilizar longitudes mayores a 255 en tipos de datos varchar.

Algunas sugerencias adicionales son:

- Modificar los campos de tipo carácter que describen fechas por los tipos de dato `smalldatetime` y `datetime` de acuerdo al dominio que manejen.
- Eliminar los tipos de datos definidos por el usuario.

Un ejemplo que muestran lo anterior es una de las tablas más importantes del sistema “*solicitud_de_empleo*”, en la cual el log que generó el programa de Visual Basic detectó lo que se muestra en el figura 3.2, en el se observa que campos como: `DATPER_EDO`, `DATPER_CODIGO_POSTAL`, `ESCMAX_GRADO_MAX`, `NUM_ENTREVISTAS`, `CLAVE_DE_PLAZA`, `ID_RANGO_ADEC`, `ID_TIP_EVAL`, `ID_CENTRO_APP`, `HORARIO`, `NIVEL_PUESTO`, `SOLICITUD_ACTUAL`, `CVE_ORGANIZACIONAL`, y `DESC_ORGANIZACIONAL` fueron modificados asignándoles nuevos tipos de datos. Estos cambios lograron reducir en alrededor de 50 bytes cada registro de la tabla (ver figura 3.3). En consecuencia se reduce el tiempo de procesamiento y de



respuesta al obtener información, además de evitar pasar menos paquetes de información por la red.

```
***** TABLA: SOLICITUD_DE_EMPLEO *****
CAMPO: DATPER_EDO STATUS: CHECAR LA LONGITUD DEL TIPO ***
MAX CAMPO: 2 TIPO ORIGINAL: varchar(20)
CAMPO: DATPER_CODIGO_POSTAL STATUS: CHECAR LA LONGITUD DEL TIPO ***
MAX CAMPO: 5 TIPO ORIGINAL: char(10)
CAMPO: DATPER_EDO_CIVIL *** CHECAR USO ***
CAMPO: DATPER_EDO_CIVIL STATUS: CHECAR LA LONGITUD DEL TIPO ***
MAX CAMPO: 0 TIPO ORIGINAL: char(1)
CAMPO: ESCMAX_GRADO_MAX STATUS: CHECAR TIPO CAMBIO TYNINT
MAX CAMPO: 9 TIPO ORIGINAL: int
CAMPO: CAL_FINAL_SEL *** CHECAR USO ***
CAMPO: GRADO_MAX_ESTUDIO *** CHECAR USO ***
CAMPO: GRADO_MAX_ESTUDIO STATUS: CHECAR TIPO CAMBIO BIT
MAX CAMPO: 0 TIPO ORIGINAL: smallint
CAMPO: POR_ADEC_PUESTO *** CHECAR USO ***
CAMPO: IND_CONDICIONADO *** CHECAR USO ***
CAMPO: IND_RECHAZADO *** CHECAR USO ***
CAMPO: IND_REQ_ESOCIO *** CHECAR USO ***
CAMPO: FEC_CONDICIONADO *** CHECAR USO ***
CAMPO: NUM_ENTREVISTAS *** CHECAR USO ***
CAMPO: NUM_ENTREVISTAS STATUS: CHECAR TIPO CAMBIO BIT
MAX CAMPO: 1 TIPO ORIGINAL: smallint
CAMPO: CALIF_EXAM_CONOC *** CHECAR USO ***
CAMPO: PORC_ADEC_PUESTO *** CHECAR USO ***
CAMPO: ID_RANGO_ADEC *** CHECAR USO ***
CAMPO: ID_RANGO_ADEC STATUS: CHECAR TIPO CAMBIO TYNINT
MAX CAMPO: 5 TIPO ORIGINAL: int
CAMPO: IND_COND_MEDICO *** CHECAR USO ***
CAMPO: IND_EN_ENTREVISTA *** CHECAR USO ***
CAMPO: IND_EN_ENTREVISTA STATUS: CHECAR LA LONGITUD DEL TIPO ***
MAX CAMPO: 0 TIPO ORIGINAL: char(1)
CAMPO: CLAVE_DE_PLAZA *** CHECAR USO ***
CAMPO: CLAVE_DE_PLAZA STATUS: CHECAR LA LONGITUD DEL TIPO ***
MAX CAMPO: 13 TIPO ORIGINAL: char(20)
CAMPO: ID_TIP_EVAL STATUS: CHECAR TIPO CAMBIO TYNINT
MAX CAMPO: 6 TIPO ORIGINAL: int
CAMPO: ID_SOLICITUD_DE_PER *** CHECAR USO ***
CAMPO: ID_SOLICITUD_DE_PER STATUS: CHECAR TIPO CAMBIO TYNINT
MAX CAMPO: 111 TIPO ORIGINAL: int
CAMPO: ID_CENTRO_APP STATUS: CHECAR TIPO CAMBIO TYNINT
MAX CAMPO: 54 TIPO ORIGINAL: int
CAMPO: FEC_INI_PER_PRUEBA *** CHECAR USO ***
CAMPO: PFI_CVE_TIPO_PERFIL *** CHECAR USO ***
CAMPO: PFI_CVE_TIPO_PERFIL STATUS: CHECAR TIPO CAMBIO BIT
MAX CAMPO: 0 TIPO ORIGINAL: int
CAMPO: HORARIO STATUS: CHECAR TIPO CAMBIO TYNINT
MAX CAMPO: 5 TIPO ORIGINAL: int
CAMPO: NIVEL_PUESTO STATUS: CHECAR TIPO CAMBIO TYNINT
MAX CAMPO: 5 TIPO ORIGINAL: int
CAMPO: SOLICITUD_ACTUAL STATUS: CHECAR TIPO CAMBIO BIT
MAX CAMPO: 1 TIPO ORIGINAL: tinyint
CAMPO: CVE_ORGANIZACIONAL STATUS: CHECAR LA LONGITUD DEL TIPO ***
MAX CAMPO: 10 TIPO ORIGINAL: char(11)
CAMPO: DESC_ORGANIZACIONAL STATUS: CHECAR LA LONGITUD DEL TIPO ***
MAX CAMPO: 94 TIPO ORIGINAL: char(100)
```

Figura 3.2 Log del Programa de Visual Basic:
Tabla Solicitud_de_Empleo



ANTERIOR

NUEVA

SOLICITUD_DE_EMPLEO		SOLICITUD_DE_EMPLEO	
	Nombre de columna	Tipo comprimido	Nombre de columna
	ID_SOLICITUD_EMPLEO	T_DID:int	ID_SOLICITUD_EMPLEO
	ID_DEL_ASPIRANTE	T_DID:int	ID_DEL_ASPIRANTE
	FEC_SOLICITUD	T_DFECHA:datetime	FEC_SOLICITUD
	DATPER_CALL	T_DDESC:varchar(60)	DATPER_CALL
	DATPER_COL	T_DDESC:varchar(60)	DATPER_COL
	DATPER_DELEGACION_MUNICIPIO	T_DDESC:varchar(60)	DATPER_DELEGACION_MUNICIPIO
	DATPER_CIUDAD	T_DDESC:varchar(60)	DATPER_CIUDAD
	DATPER_EDO	varchar(20)	DATPER_EDO
VARCHAR(2)	DATPER_CODIGO_POSTAL	char(10)	DATPER_CODIGO_POSTAL
VARCHAR(5)	DATPER_TEL	T_DTELEFONO:varchar(30)	DATPER_TEL
	DATPER_EDO_CIVIL	char(1)	DATPER_EDO_CIVIL
VARCHAR(10)	DATPER_NUMEXT	char(10)	DATPER_NUMEXT
VARCHAR(10)	DATPER_NUMINT	char(10)	DATPER_NUMINT
TINYINT	ESCMAX_GRADO_MAX	int	ESCMAX_GRADO_MAX
	CAL_FINAL_SEL	float(53)	CAL_FINAL_SEL
	IND_TITULADO	char(1)	IND_TITULADO
	GRADO_MAX_ESTUDIO	smallint	GRADO_MAX_ESTUDIO
TINYINT	EDO_SOL_EMP	T_DID:int	EDO_SOL_EMP
TINYINT	ORIGEN_SOL_EMP	char(1)	ORIGEN_SOL_EMP
XXXXXX	POR_ADEC_PUESTO	float(53)	POR_ADEC_PUESTO
	IND_CONDICIONADO	T_DINDICADOR:char(1)	IND_CONDICIONADO
	IND_RECHAZADO	T_DINDICADOR:char(1)	IND_RECHAZADO
	IND_HA_LABORADO	T_DINDICADOR:char(1)	IND_HA_LABORADO
	IND_TIENE_FAM	T_DINDICADOR:char(1)	IND_TIENE_FAM
	IND_REQ_ESOCIO	T_DINDICADOR:char(1)	IND_REQ_ESOCIO
VARCHAR(250)	FAMILIARES	varchar(255)	FAMILIARES
VARCHAR(6)	CVE_PUESTO	char(6)	CVE_PUESTO
	FEC_CONDICIONADO	T_DFECHA:datetime	FEC_CONDICIONADO
	FEC_RECHAZADO	T_DFECHA:datetime	FEC_RECHAZADO
	FEC_CONDICIONADO_ASP	T_DFECHA:datetime	FEC_CONDICIONADO_ASP
SMALLINT	ID_DE_LA_UNIDAD	T_DID:int	ID_DE_LA_UNIDAD
TINYINT	ID_PAIS	T_DID:int	ID_PAIS
TINYINT	NUM_ENTREVISTAS	smallint	NUM_ENTREVISTAS
	CALIF_EXAM_CONOC	float(53)	CALIF_EXAM_CONOC
	PORC_ADEC_PUESTO	float(53)	PORC_ADEC_PUESTO
TINYINT	ID_RANGO_ADEC	int	ID_RANGO_ADEC
	IND_COND_MEDICO	char(1)	IND_COND_MEDICO
	IND_EN_ENTREVISTA	char(1)	IND_EN_ENTREVISTA
VARCHAR(20)	CLAVE_DE_PLAZA	char(20)	CLAVE_DE_PLAZA
	CONOC_SOL_EMP	varchar(255)	CONOC_SOL_EMP
	EXP_SOL_EMP	varchar(255)	EXP_SOL_EMP
TINYINT	ID_TIP_EVAL	int	ID_TIP_EVAL
TINYINT	ID_SOLICITUD_DE_PER	int	ID_SOLICITUD_DE_PER
	ID_CENTRO_APP	int	ID_CENTRO_APP
	FEC_INI_PER_PRUEBA	datetime	FEC_INI_PER_PRUEBA
	PFL_CVE_TIPO_PERFIL	int	PFL_CVE_TIPO_PERFIL
TINYINT	HORARIO	int	HORARIO
TINYINT	NIVEL_PUESTO	int	NIVEL_PUESTO
	PUEDE_VIAJAR	varchar(1)	PUEDE_VIAJAR
	CAMBIAR_RESIDENCIA	varchar(1)	CAMBIAR_RESIDENCIA
	FEC_EXAM_CONOCIMIENTOS	datetime	FEC_EXAM_CONOCIMIENTOS
	FEC_EXAM_PSICOLOGICOS	datetime	FEC_EXAM_PSICOLOGICOS
BIT	SOLICITUD_ACTUAL	tinyint	SOLICITUD_ACTUAL
VARCHAR(11)	CVE_ORGANIZACIONAL	char(11)	CVE_ORGANIZACIONAL
VARCHAR(100)	DESC_ORGANIZACIONAL	char(100)	DESC_ORGANIZACIONAL

Figura 3.3 Diagrama entidad-relación. Comparación de la tabla de SOLICITUD_DE_EMPLEO.

El reporte muestra cada uno de los campos que fueron modificados en la base de datos, esto se detalla específicamente en el ANEXO C, cabe señalar que estos reportes se hicieron con ayuda de la herramienta de Crystal Reports.



3.2.1.2 REESTRUCTURACIÓN DE LA INTEGRIDAD DE ENTIDAD

La integridad de entidad define una fila como entidad única para una tabla determinada. La integridad de entidad fuerza la integridad de la columna o columnas que forma parte del identificador de la tabla o de la clave principal de una tabla (mediante índices, restricciones UNIQUE, restricciones PRIMARY KEY o propiedades IDENTITY).

Cuando se establece una clave principal también se crea un índice sobre la clave que proporciona un acceso rápido a los valores de las claves principales, normalmente este será el índice agrupado, el cual es el que afecta el orden de almacenamiento de la tabla.

Estrategia para reestructurar la Integridad de Entidad.

En conclusión se aplicaron las siguientes recomendaciones para la reestructuración de llaves primarias con la finalidad de mejorar lo referente a la integridad de entidad.

- Se detectaron todas aquellas tablas que carecían de una llave primaria, esto mediante la siguiente consulta:

```
SELECT TABLA.name  
FROM RECSEL..sysobjects TABLA  
WHERE TABLA.type= 'U' AND TABLA.id NOT IN  
      (SELECT PK.parent_obj From RECSEL..sysobjects PK WHERE  
      PK.xtype = 'PK')  
ORDER BY TABLA.name
```

- Se detectó el campo que identificaba de manera única cada uno de los registros de estas tablas.
- Se asignó una llave primaria sobre el campo detectado como identificador para cada una de las tablas que se detectaron.
- En otros casos la llave primaria fue modificada debido a factores como la eliminación de un campo.

Un ejemplo de las tablas (ver Figura 3.4) las cuales reflejan este tipo de cambio son tablas ESCOLARIDADES, BATERIA_PROC_NIVEL_REGIONAL, cabe señalar que fueron pocos los casos con este tipo de deficiencias.



SE AGREGO LLAVE PRIMARIA

ESCOLARIDADES			
Nombre de columna	Tipo comprimido	Anulable	
KEY_ESCOLARIDAD	int	NOT NULL	
DESC_ESCOLARIDAD	char(60)	NOT NULL	

ESCOLARIDADES			
Nombre de columna	Tipo comprimido	Anulable	
KEY_ESCOLARIDAD	tinyint	NOT NULL	
DESC_ESCOLARIDAD	varchar(60)	NOT NULL	

Figura 3.4(a) Diagrama entidad-relación. Tablas que fueron afectadas en la llave primaria(Antes)

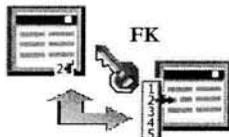
MODIFICACION DE LLAVE PRIMARIA DEBIDO A QUE SE ELIMINO UNA CAMPO

BATERIA_PROCC_NIVEL_REGIONAL			
Nombre de columna	Tipo comprimido	Anulable	
KEY_NIVEL	T_DID:int	NOT NULL	
IND_DE_SUPERVISION	T_DINDICADOR:char(1)	NOT NULL	
ID_TIP_EVAL	T_DID:int	NOT NULL	
ID_BATERIA_EXAM	T_DID:int	NOT NULL	
TIPO_DE_UNIDAD_ADM	char(1)	NOT NULL	
NUM_APLIC_BAT_TIPEVAL_NIV_UBIC	int	NOT NULL	

BATERIA_PROCC_NIVEL_REGIONAL			
Nombre de columna	Tipo comprimido	Anulable	
KEY_Nivel	tinyint	NOT NULL	
IND_DE_SUPERVISION	T_DINDICADOR:char(1)	NOT NULL	
ID_TIP_EVAL	tinyint	NOT NULL	
ID_BATERIA_EXAM	tinyint	NOT NULL	
TIPO_DE_UNIDAD_ADM	char(1)	NOT NULL	

Figura 3.4 (b) Diagrama entidad-relación. Tablas que fueron afectadas en la llave primaria

El reporte (ver figura 3.5) que muestra cada una de las llaves primarias que fueron definidas en la base de datos, se detalla en el ANEXO C.



3.2.1.3 REESTRUCTURACIÓN DE LA INTEGRIDAD REFERENCIAL

Como ya se menciona la integridad referencial se refiere a un conjunto de reglas que se aplican a todas las filas en una tabla. Las reglas se especifican a nivel columna. La integridad referencial protege las relaciones definidas entre las tablas cuando se crean o se eliminan registros. La integridad referencial se basa en las relaciones entre claves externas y claves principales o entre claves externas y claves exclusivas (mediante restricciones FOREIGN KEY, restricciones CHECK). La integridad referencial garantiza que los valores clave sean coherentes en las distintas tablas. Para conseguir esa coherencia, es preciso que no haya referencias a valores inexistentes y que, si cambia el valor de una clave, todas las referencias a ella se cambien en consecuencia en toda la base de datos.



ID OBJETO	NOMBRE OBJETO
1458820259	CLAVES_DE_CAPACITACION
1666821000	CONVOCATORIA_PROMO
495340829	ESCOLARIDADES
511340886	ESTADOS
1071342881	MSLAST_JOB_INFO
1247543508	OPCIONES_LARGAS
1327543795	PARAMAPLICA
1343343850	PARAMDEFEXAM
1359343907	PARAMETROS
1375343964	PARAMPERFIL
1391344021	PARAMREC
1407344078	PARAMSEG
1423344135	PARAMSEL
1659868980	PERMISOS
1935345959	RES_EVAL_PSI_X_FAC
1951346016	RFC_INCONSISTENTE_EDD
292912115	RPT_TABLAS_SIN_PK
1999346187	RSROTACION
1528392514	SOL_RECHAZADA
443864648	TMP_FUNDAMENTO_LEGAL

Figura 3.5 Reporte de Tablas sin llave primaria.

Según Joe Celko, un notorio autor sobre SQL y un miembro del comité de estándares de SQL, las restricciones de integridad referencial son la característica menos aplicada debido a que un número considerable de programadores considera que el uso de estas impacta en el rendimiento de la base de datos, sin embargo él comenta que no se ha encontrado ninguna clara evidencia de que esto sea cierto.

Cuando se fuerza la integridad referencial, se impide a los usuarios:

- Agregar registros a una tabla relacionada si no hay ningún registro asociado en la tabla principal.
- Cambiar valores en una tabla principal de manera que queden registros huérfanos en una tabla relacionada.
- Eliminar registros de una tabla principal cuando hay registros relacionados coincidentes.



Estrategia para reestructurar la Integridad de Referencial

Tomando en cuenta esto se detecto:

- Todas aquellas relaciones que no estaban establecidas entre las entidades, con ayuda de las siguientes consultas en el diccionario de datos y la experiencia y conocimiento en la información que contiene cada una de las tablas.
 - ```
SELECT c.name, count(c.name)
FROM sysobjects o INNER JOIN syscolumns c ON (o.id = c.id)
WHERE o.type= 'U' AND o.uid = 1
GROUP BY c.name
HAVING COUNT(c.name) > 1
```
  - `SP_HELPCONSTRAINT @TABLA`
  - ```
SELECT *
FROM SYSCONSTRAINTS
WHERE ID = OBJECT_ID(@TABLA)
```
- También se consideraron todas aquellas tablas que no tenía relación alguna, es decir se encontraban totalmente aisladas. Posteriormente se consulto su función y se determino sobre la base de esto su relación con otras tablas.

Un ejemplo muy importante de este caso es el diagrama entidad-relación el cual muestra el diseño lógico referente a los procesos de Mantenimiento a Temarios de conocimientos específicos, en el cual en su etapa de diseño no se estableció ninguna relación entre las tablas (Note las diferencias entre la figura 3.6a y la figura 3.6b). Esto provoco que se tuvieran muchas inconsistencias en la información referente a temarios y preguntas de exámenes de conocimientos específicos, por ejemplo, hay repuestas de las cuales ya no existe la definición de la pregunta debido a que esta ya había sido eliminada, o preguntas que no tienen ningún temario asignado etc.

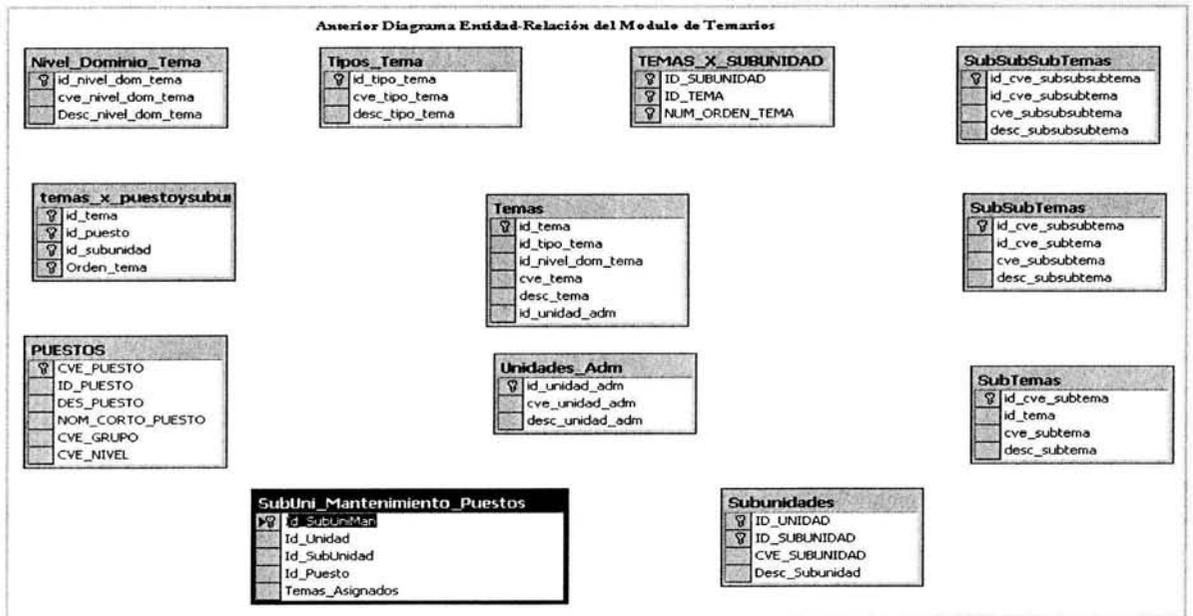


Figura 3.6a Anterior Diagrama Entidad-Relación del Módulo de Temarios

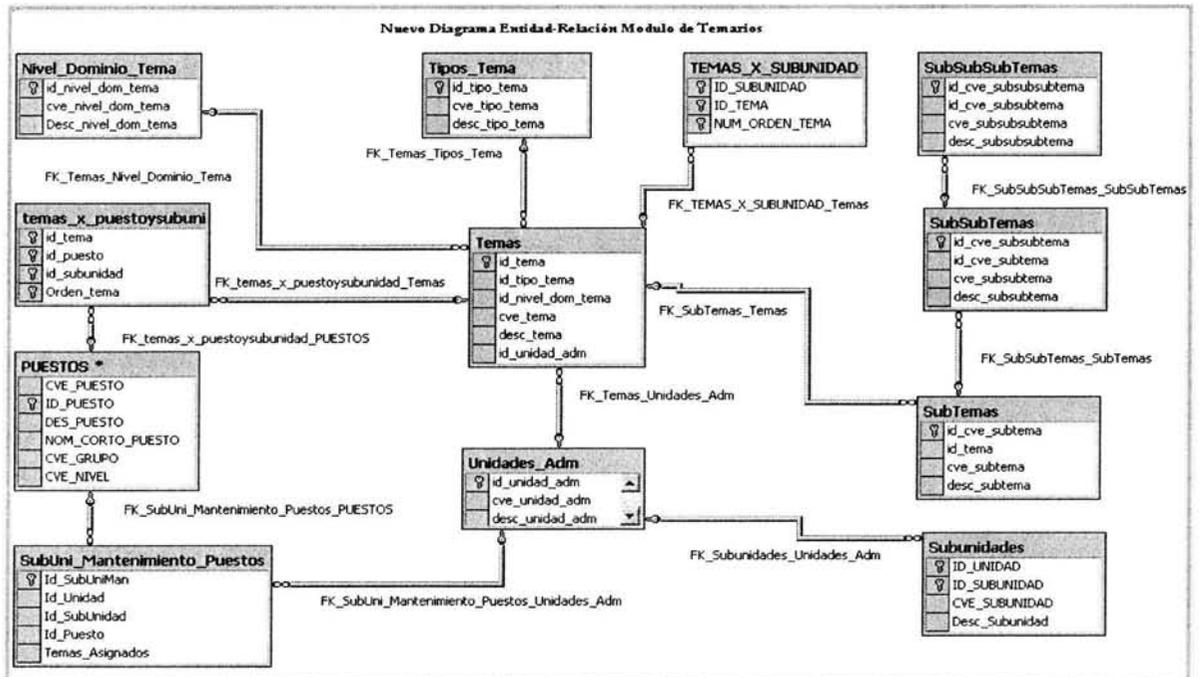


Figura 3.6b Nuevo Diagrama Entidad-Relación del Módulo de Temarios



3.2.2 ELIMINAR TABLAS O ENTIDADES QUE NO SEAN UTILIZADAS

En lo que se refiere a eliminación de tablas se establecieron los siguientes puntos como métricas para analizar las tablas y así poder determinar si su uso actual es indispensable dentro del sistema:

- Se detectó todas aquellas tablas que no almacenarán ningún dato.
- Se detectaron aquellas tablas que sirvieron como catálogos temporales en cierto momento en el desarrollo del sistema.

El reporte (ver figura 3.7) que muestra cada uno de las tablas que fueron eliminadas de la base de datos se detalla en el ANEXO C.



3.2.3 ELIMINAR CAMPOS DE LAS ENTIDADES QUE NO SEAN UTILIZADOS

En lo que se refiere a eliminación de campos se establecieron los siguientes puntos como métricas para analizar los campos y así poder determinar si era factible eliminar los campos.

- Se detectaron aquellas columnas que en un 85% o más de sus registros tuvieran el mismo valor. Después de acuerdo al conocimiento y experiencia se determinó si era o no necesario conservar el campo. En caso de que fuera posible eliminarlo así se procedía en caso de pudiera afectar varios aplicativos se conservó (no sin antes especificarlo como un posible cambio en un rediseño del aplicativo posterior).
- Se detectaron también todos aquellos campos que en todos sus valores fueran NULL y al igual que los anteriores se procedió a analizar si era posible o no su depuración.



SISTEMA DE RECLUTAMIENTO Y SELECCION DE PERSONAL
Reporte Tablas Eliminadas de la Base de Datos

ID OBJETO	NOMBRE OBJETO
485576768	AREAS
462624691	BATERIA_EXAMENES_PUESTO
158623608	BORRAME
1099150961	INT_LUSCHER
450100644	MAX_ASP_X_PUESTO
498100815	NECESIDADES
530100929	NIVEL
562101043	NOTA_EVAL_DESEMP
738101670	PARAMAPLICA
754101727	PARAMDEPEXAM
786101841	PARAMPERFIL
802101898	PARAMREC
818101955	PARAMSEG
834102012	PARAMSEL
318624178	PERFIL_IDEAL
1211151360	RANGO_ADECUACION_ASPIRANTE
1275151588	RANGO_DE_ADECUACION
1346103836	RES_EVAL_PSI_X_FAC
1883153754	RPT_TABLAS_SIN_PK
1490104349	SOLICITANTE_DE_COBERTURA
1842105603	TEMAS_FOR_PUESTO
62623266	UNIDADES_ADMVAS

NUM OBJETOS 22

Records: 22 100%

Figura 3.7 Reporte de Tablas Eliminadas de la Base de Datos.

El reporte que muestra (ver figura 3.8) cada uno de los campos que fueron eliminadas de la base de datos se detalla en el ANEXO C.

SISTEMA DE RECLUTAMIENTO Y SELECCION DE PERSONAL
Reporte cambios de Procedimientos Almacenados y Desencadenadores

ID COLUMNA	NOMBRE TABLA	NOMBRE COLUMNA
6	BATERIA_PROC_NIVEL_REGIONAL	NUM_APLIC_BA1_IIEVAL_NIV_UBIC

NUM OBJETOS

Records: 1 100%

Figura 3.8 Reporte de Campos Eliminados



3.2.4 ACTUALIZACIÓN DE ÍNDICES

En una base de datos, un índice permite que el programa de la base de datos busque datos en una tabla sin necesidad de examinar toda la tabla. Un índice de una base de datos es una lista de los valores de una tabla con las posiciones de almacenamiento de las filas de la tabla donde se encuentra cada valor. Se pueden crear índices en una sola columna o en una combinación de columnas de una tabla; los índices se implementan en forma de árboles B. Un índice contiene una entrada con una o varias columnas (la clave de búsqueda) de cada fila de una tabla. Un árbol B se ordena con la clave de búsqueda y se puede buscar de forma eficiente en cualquier subconjunto principal de la clave de búsqueda.

Las bases de datos contienen índices para tipos o columnas de datos seleccionados, esto es parecido a un libro con un índice para los nombres de las personas y otro índice para los lugares. Cuando cree una base de datos y la optimice para mejorar el rendimiento, es recomendable que cree índices para las columnas que se utilizan en las consultas con el fin de buscar datos.

SQL Server crea automáticamente índices para determinados tipos de restricciones (por ejemplo, restricciones PRIMARY KEY y UNIQUE). También puede personalizar las definiciones de la tabla mediante la creación de índices independientes de las restricciones.

No obstante, las ventajas que ofrecen los índices por lo que respecta al rendimiento también tienen un costo. Las tablas con índices necesitan más espacio de almacenamiento en la base de datos. Asimismo, es posible que los comandos de inserción, actualización o eliminación de datos sean más lentos y precisen más tiempo de proceso para mantener los índices. Cuando se diseña y crea índices, se deberá asegurar que las ventajas en el rendimiento compensan suficientemente el costo adicional en cuanto a espacio de almacenamiento y recursos de proceso.

Los índices agrupados producen una reordenación de las filas de forma que las filas se almacenan ordenadas según la forma de ordenación del índice. La ordenación de las filas en el almacenamiento mejora el rendimiento cuando se seleccionan las filas mediante el valor así indexado. Sin embargo se puede tener solamente un índice agrupado por tabla, debido a que una tabla puede ser ordenada de una sola forma física. Mediante el uso de un índice agrupado sobre una columna sin significado, tal como FILA_NUMERO se proporciona poca ventaja inherente en el rendimiento. El índice agrupado en una columna que aparece frecuentemente en las cláusulas como apellidos, proporciona una mejora en el rendimiento.



Otro dato importante acerca de los índices es que estos son directamente proporcionales al porcentaje de renglones que son solicitados en una consulta, es decir, pequeños porcentajes son más eficientes.

Estrategia para la reestructuración de Índices

En base a las recomendaciones anteriores se aplicaron las siguientes métricas para definir nuevos índices y reestructurar algunos otros:

- Se detectaron también todos aquellos campos que en todos sus valores fueran NULL y al igual que los anteriores se procedió a analizar si era posible o no su depuración.
- Determinar índices agrupados para campos por los que las búsquedas sean más frecuentes.
- Determinar índices no agrupados para campos por los que se hagan JOINS más frecuentemente, columnas en que se busque rango de valores y columnas por la que se establece el orden.
- Evitar crear índices sobre columnas a las cuales se hace ocasionalmente referencia en las consultas o que contienen pocos valores únicos.
- Definir la llave de un índice con el menor número de columnas posibles en los índices clustered.
- Evitar la indización de tablas pequeñas puede no ser una solución óptima, ya que puede provocar que SQL Server tarde más tiempo en realizar la búsqueda de los datos con ayuda del índice que realizar una simple exploración de la tabla.
- Determinar índices para campos por los que se hagan búsquedas en un intervalo de valores.
- Determinar índices en particular para consultas en las cuales se espera una salida de datos ordenada sin una operación explícita de ordenación.
- Evitar utilizar un gran número de índices en una tabla, sobre todo en aquellas en las cuales hay operaciones de INSERT, UPDATE y DELETE frecuentemente.

Un ejemplo de los índices lo refleja la tabla de “ASPIRANTE”, en la cual se hacen búsquedas muy frecuentes sobre los campos de apellido paterno, materno nombre y RFC. Por otro lado para el manejo interno de sistema se utiliza un identificador del cada registro (ID_DEL_ASPIRANTE) de esta tabla, este último es el campo con el cual se realizan las combinaciones con otras tablas en la mayoría de las consultas. Partiendo de que esta tabla ya tenía creado un índice clustered sobre el campo ID_DEL_ASPIRANTE, se procedió a crear dos índices más de tipo noclustered. Uno se creó sobre el campo RFC_DEL_ASPIRANTE para las búsquedas por RFC y el índice otro sobre los campos APE_PATERNO_ASPIRANTE, APE_MATERNO_ASPIRANTE Y NOM_DEL_ASPIRANTE para búsquedas por apellidos y nombre (ver Figura 3.9).

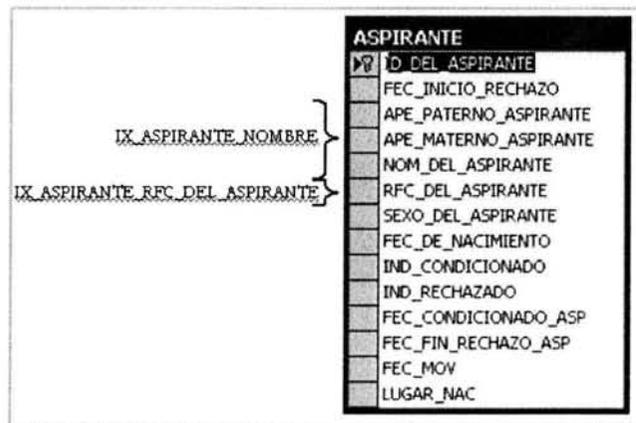


Figura 3.9 Índices de la tabla Aspirante

3.3 ACTUALIZACIÓN DE PROCEDIMIENTOS ALMACENADOS Y DESENCADENADORES

La principal ventaja del empleo de desencadenadores y procedimientos almacenados es que permiten a los usuarios crear y mantener un conjunto de código más manejable para el empleo en varias aplicaciones asociadas con la base de datos existentes y futuras, esto es, permite que todas las aplicaciones que necesitan tener acceso a la base de datos empleen procedimientos como intermediarios en lugar de que manipulen las tablas de manera independiente, lo que nos puede ahorrar miles de horas a largo plazo debido a que se escribe un solo código.



3.3.1 ACTUALIZACIÓN DE PROCEDIMIENTOS ALMACENADOS (STORED PROCEDURED)

Los procedimientos almacenados son instrucciones precompiladas almacenadas en una base de datos, cuyo principal objetivo consiste en automatizar las actualizaciones a la base de datos.

Como los procedimientos almacenados son instrucciones precompiladas, por lo general ofrecen el mejor rendimiento en cualquier tipo de consulta, ya que se ejecutan rápidamente y se guardan en la base de datos para ser posteriormente utilizados

Cuando se ejecuta por primera vez un procedimiento almacenado, se llevan a cabo los pasos que se muestran a continuación; posteriormente se coloca en memoria el



plan de consulta del procedimiento y en las siguientes llamadas a este sólo se ejecuta a partir del paso 5. Es muy factible que esto pueda variar de acuerdo con el manejador que se este utilizando, pero en general esta es la forma en la que se trabajan.

1. Se analizan todas las piezas que componen al procedimiento.
2. Se comprueba la existencia de los componentes a los cuales se hace referencia (tablas, vistas, etc.). Este proceso también se conoce como resolución.
3. El nombre del procedimiento se guarda en la tabla sysobjects, y el código necesario para crear el procedimiento almacenado en syscomments.
4. El proceso de compilación continúa, durante el cual se crea un plano en el que se indica como ejecutar la consulta. Este plano se conoce normalmente como plano normalizado o árbol de consulta. Este árbol se guarda en sysprocedures.
5. Cuando el procedimiento almacenado se ejecuta por primera vez, se lee el árbol de consulta y se optimiza completamente el plan del procedimiento y luego se ejecuta. Esto reduce el tiempo requerido para volver a analizar, resolver y compilar un árbol de consulta cada vez que se ejecuta el procedimiento almacenado.

Otro beneficio de un procedimiento almacenado es que, después de ejecutarse, el plan de procedimiento se guarda en caché de procedimientos. Esto significa que la próxima vez que utilice el procedimiento almacenado en la misma sesión, se leerá directamente de la caché y se ejecutará. Gracias a ello se obtiene una considerable mejora en el rendimiento, en comparación con ejecutar una y otra vez la consulta estándar de SQL.

Los procedimientos almacenados son utilizados para:

- Encapsular reglas de negocio. Una vez encapsuladas, varias aplicaciones pueden utilizar dichas reglas, con lo cual se obtiene una interfaz de datos consistente.
- Administración y mantenimiento de las bases de datos.
- Pasar argumentos y recibir datos.
- Automatizar las actualizaciones a la base de datos.

Algunas de las ventajas que ofrecen los procedimientos almacenados son:

- Los procedimientos almacenados pueden configurarse para ejecutarse automáticamente al iniciar SQL Server.
- Los procedimientos almacenados pueden utilizarse para extraer o modificar.
- Los procedimientos almacenados se invocan explícitamente. A diferencia de los desencadenadores, los procedimientos almacenados deben ser llamados por su aplicación, secuencia de comandos, lote o tarea.

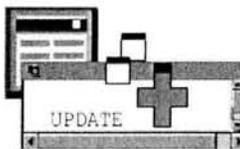


- Los procedimientos almacenados permiten al usuario un mejor control de la transacción que los desencadenadores o las aplicaciones
- Los procedimientos operan con una perspectiva más amplia que los desencadenadores, es decir, sobre varias filas de varias tablas.
- Es posible dar permiso a un usuario para que ejecute el procedimiento almacenado, incluso si el usuario no tiene permisos en las tablas involucradas.

Aunque los procedimientos almacenados son valiosos, tienen algunos factores que los limitan. La sintaxis de estos varía considerablemente de un SGBD a otro. A efectos prácticos intentar escribir un procedimiento almacenado completamente transportable resulta casi imposible. No obstante en el caso de una transferencia de SGBD, es posible transferir las llamadas de la mayor parte de los procedimientos almacenados, por lo que es factible que gran parte de la aplicación del cliente, aunque no todo, puede conservarse si cambia el SGBD.

Los procedimientos almacenados suelen descomponerse en tres partes. La primera identifica todas las declaraciones de las variables que se emplearán en la sección en que se ejecutan los comandos. Posteriormente viene la parte de los comandos ejecutables y la tercera parte es en la cual se hace el tratamiento de las excepciones.

En general, los procedimientos almacenados son componentes poderosos y útiles, en cualquier tarea para la que hayan sido diseñados. Ofrecen además mayores ventajas sobre las vistas y consultas ya que son precompilados, y después de su primera ejecución, se guarda su plan de ejecución en la caché de procedimientos que reside en la RAM (memoria de acceso aleatorio).



3.3.2 ACTUALIZACIÓN DE DESENCADENADORES (TRIGGERS)

Un desencadenador es un fragmento de código que se ejecuta de manera automática en una base de datos antes o después de la modificación de los datos de una tabla.

Los desencadenadores son una alternativa a los procedimientos almacenados, aunque su objetivo es mucho menos ambicioso que estos, dado que están ligados a tablas concretas. No obstante, aunque no son tan globales como los procedimientos, su potencia es bastante valiosa, especialmente para fines de auditoría.

Los desencadenadores son empleados con frecuencia para:

- Asegurar la integridad de sus datos ya que impide que se realicen modificaciones no autorizadas o inconsistentes.



- Implementar reglas de negocios más complejas, las cuales están fuera del alcance de lo que pueden hacer las restricciones.
- Crear auditoría sobre las tablas, debido a que a nivel base de datos se sabe que la auditoría de entrada de datos es crucial, puesto que con demasiada frecuencia los usuarios de la base de datos introducen información errónea que hay que corregir.
- En la identificación de posibles errores, su origen del error, esto mediante una tabla de auditoría.

Es importante destacar que los desencadenadores funcionan en el nivel filas, además de que estos suelen asociarse con las operaciones de INSERT, UPDATE y DELETE en las tablas concretas.

Los desencadenadores no tienen parámetros y solo pueden invocarse en forma explícita, esto es, que para activar un desencadenador se debe intentar una modificación de datos. De manera predeterminada, todos los desencadenadores se ejecutan después de la modificación de los datos.

Otra característica de los desencadenadores es el hecho de que es posible su anidación, es decir, que un desencadenador puede disparar otro. Una tabla puede tener tantos desencadenadores como se deseen para cualquiera de las tres acciones definidas anteriormente.

En lo que se refiere al rendimiento, los desencadenadores tienen una cantidad baja de sobrecarga. La mayor parte del tiempo empleado en la ejecución se ocupa haciendo referencia a otras tablas. Esta referencia puede ser rápida si las demás tablas se encuentran en memoria, o puede ser poco más lenta si se deben leer desde disco. Los desencadenadores siempre se consideran como una parte de la transacción, si el desencadenador o cualquier otra parte de la transacción falla, ésta se deshace.

Estrategia para la optimización de procedimientos almacenados y desencadenadores

Con base en el uso y la funcionalidad de cada uno de los procedimientos y desencadenadores de la base de datos, se considero que era factible llevar a cabo una actualización y optimización aplicando las siguientes medidas:

- No es recomendable incluir sentencias SELECT en los desencadenadores, a menos que sean indispensables para el llenado de parámetros.
- Optimizar consultas mediante las siguientes métricas:
 - Si la consulta utiliza cursores, determinar si se puede escribir la consulta de cursor de forma más eficaz con un tipo de cursor más adecuado, como un cursor de desplazamiento rápido hacia delante, o una consulta única. Las consultas únicas normalmente mejoran las operaciones de cursor, debido a que un conjunto de instrucciones de cursor suele constituir una operación de bucle



externo, en la que cada fila del bucle externo se procesa una vez con una instrucción interna. Se debe considerar la posibilidad de utilizar en su lugar una instrucción GROUP BY, una instrucción CASE o una subconsulta.

- Referenciar únicamente a la información que se requiera.
- Evitar pasar grandes cantidades de información por la red a menos que sea necesario.
- Evitar el uso de subconsultas.
- Evitar vistas que hagan referencia a otras vistas.
- Evitar cursores.
- Haga que las transacciones sean lo más cortas posible. Procese siempre todos los resultados hasta su finalización. No diseñe una aplicación ni utilice una instrucción que detenga el procesamiento de filas de resultados sin cancelar la consulta. Al hacer esto, normalmente se provocan bloqueos y un rendimiento lento.

Un ejemplo que refleja la optimización y actualización que se llevo a cabo en los procedimientos es el de “*SP_PROCESO_CALENDARIZACION*” (ver figura 3.11). El objetivo de este procedimiento almacenado es llevar a cabo el proceso de calendarización masivo de aspirantes en los procesos de evaluación de Promociones, Detecciones y algunos otros procesos especiales. Por otro lado este procedimiento, no diferenciaba entre procesos de evaluación; tampoco tomaba en cuenta las fechas de evaluación. Al ejecutarse, se realizaba la calendarización de todos los aspirantes que estuvieran pendientes por aplicar, esto provocaba algunos conflictos al usuario de administración de información, ya que en varias ocasiones este requería únicamente calendarizar a cierto proceso de evaluación ya que no se tenían máquinas disponibles para aplicar todos los procesos, lo cual le era imposible debido a la falta de flexibilidad de este procedimiento. Como alternativa de solución se agregaron dos parámetros opcionales que le van a permitir llevar a cabo una calendarización específica sin que esta deje de ser masiva (ver figura 3.12). Como los parámetros que se anexaron son opcionales no se afecto la funcionalidad anterior, ya que es posible no especificar estos en la llamada del procedimiento. También se cambio el tipo de cursor a un tipo “FAST_FORWARD”, ya que el objetivo de este cursor en este procedimiento es de solo lectura y la lectura del conjunto de registros es solo hacia adelante.



Migración y propuesta de Instalación de las Bases de Datos del Sistema de Reclutamiento y Selección de Personal del Servicio de Administración Tributaria

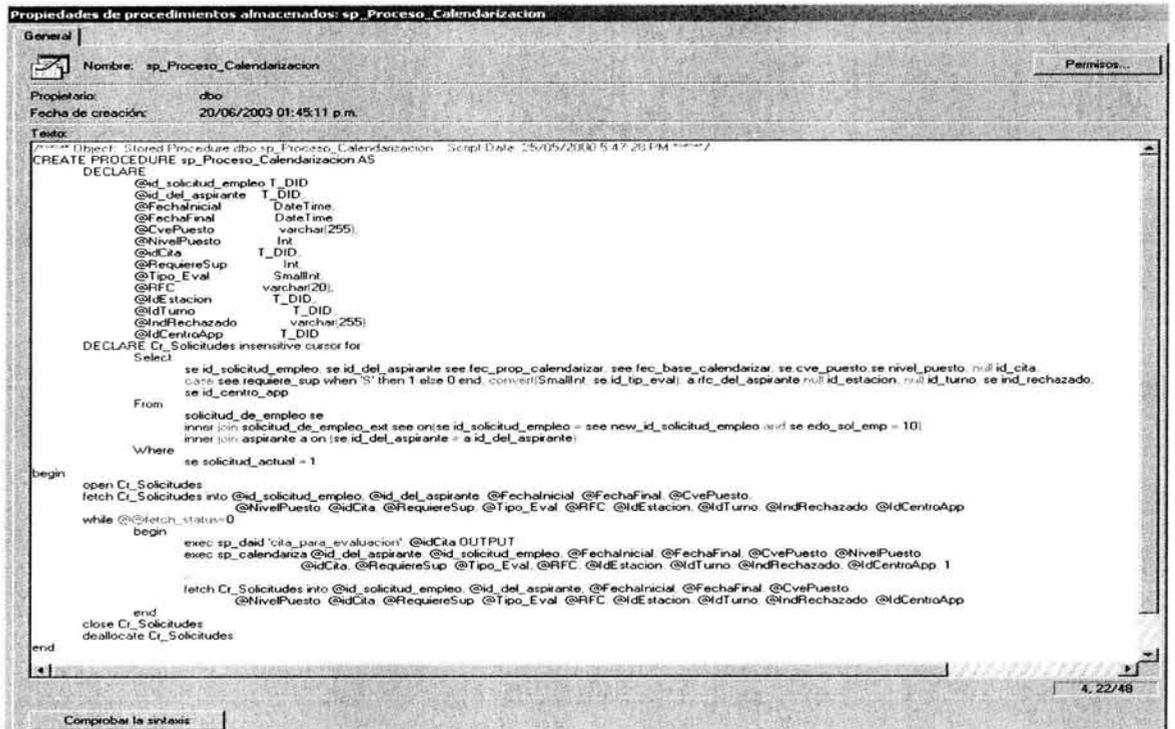


Figura 3.11 Procedimiento Almacenado “SP_PROCESO_CALENDARIZACION” de la BD anterior

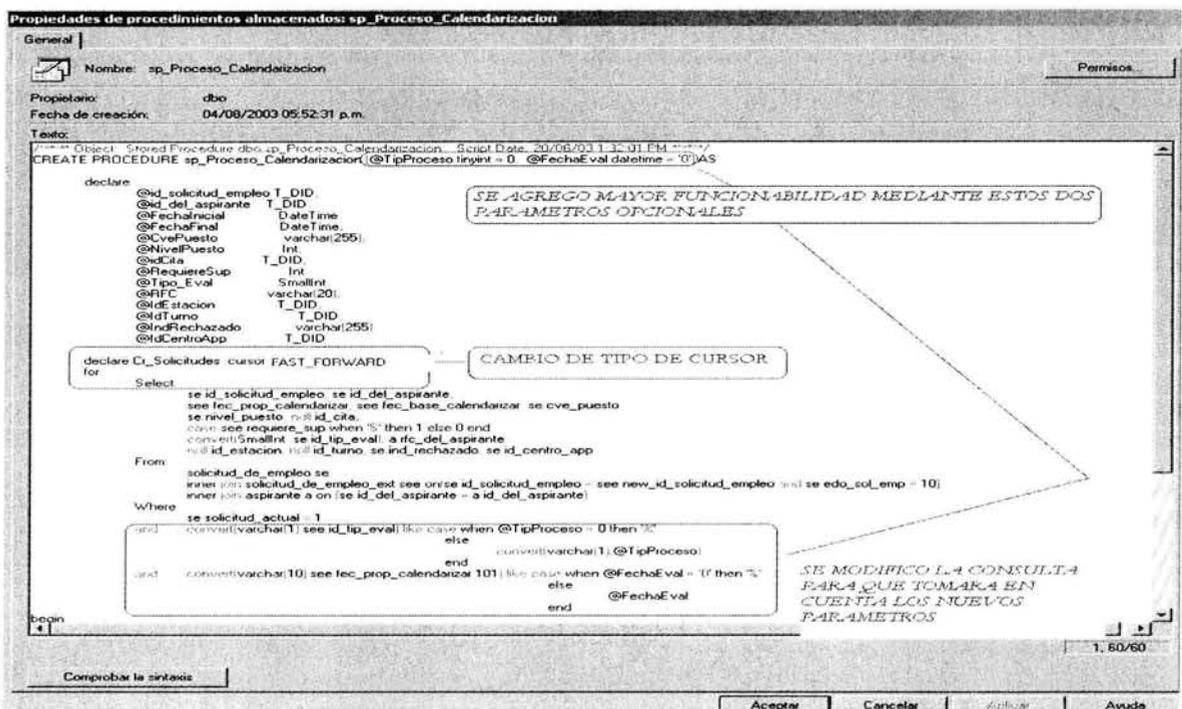


Figura 3.12 Procedimiento Almacenado “SP_PROCESO_CALENDARIZACION” de la BD reestructurada.



Un ejemplo en cuanto a los desencadenadores se refiere, se visualiza en el desencadenador “FIX_SOLICITUD_DE_EMPLEO”, el cual se dispara cuando se hace una inserción en la tabla de “SOLICITUD DE EMPLEO” (ver figura 3.13). El objetivo de este desencadenador es actualizar la última solicitud de empleo como activa y desactivar las anteriores debido a que en los procesos del sistema siempre se trabaja con la solicitud actual activa. El estatus de la solicitud de empleo se determina mediante la actualización del campo SOLICITUD_ACTUAL de la tabla antes mencionada (cuando SOLICITUD_ACTUAL 1: activa, 0: desactiva).

El desencadenador presentaba una inconsistencia al hacer la actualización cuando se llevaban inserciones masivas siendo que solo se puede dar el caso de que hay una solicitud activa, además en algunas ocasiones no llevaba a cabo las actualizaciones de todos los registros y se encontraban en el sistema dos solicitudes activas, lo cual provocaba un desconcierto en el usuario, pues no podía detectar cual era la última. Como una alternativa de solución a estos detalles, se modificaron tanto la condición de la consulta que determinaba si existía otra solicitud como la condición de la modificación para que esta se llevara a cabo únicamente en los registros que fuera necesario (ver figura 3.14).

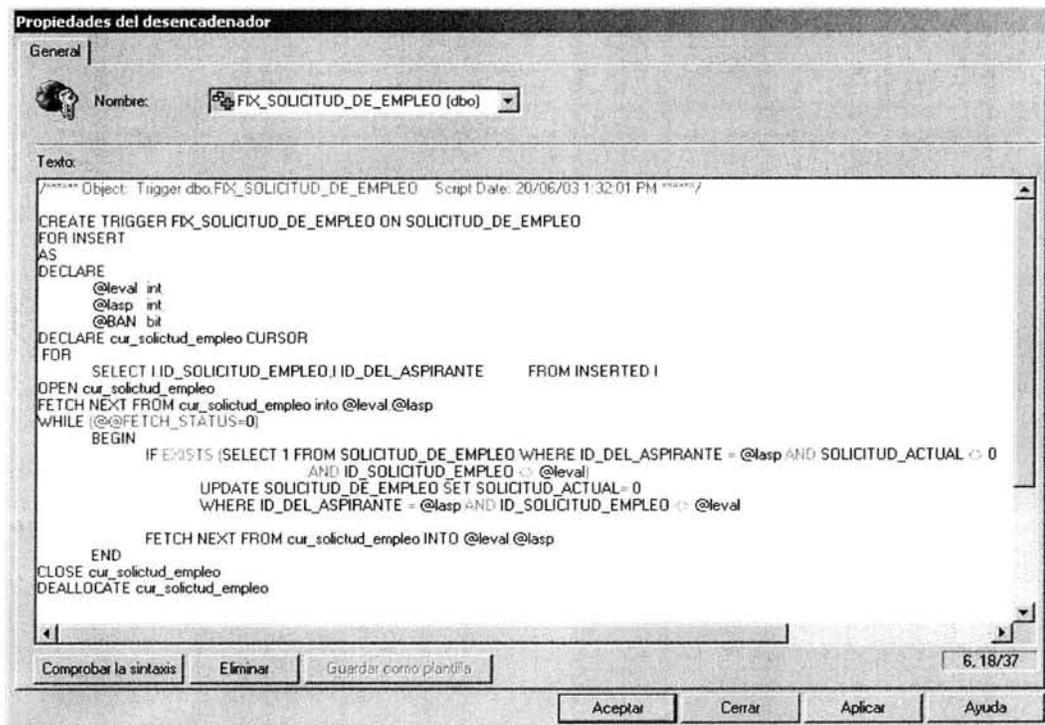


Figura 3.13. Desencadenador “FIX_SOLICITUD_DE_EMPLEO” de la BD actual.

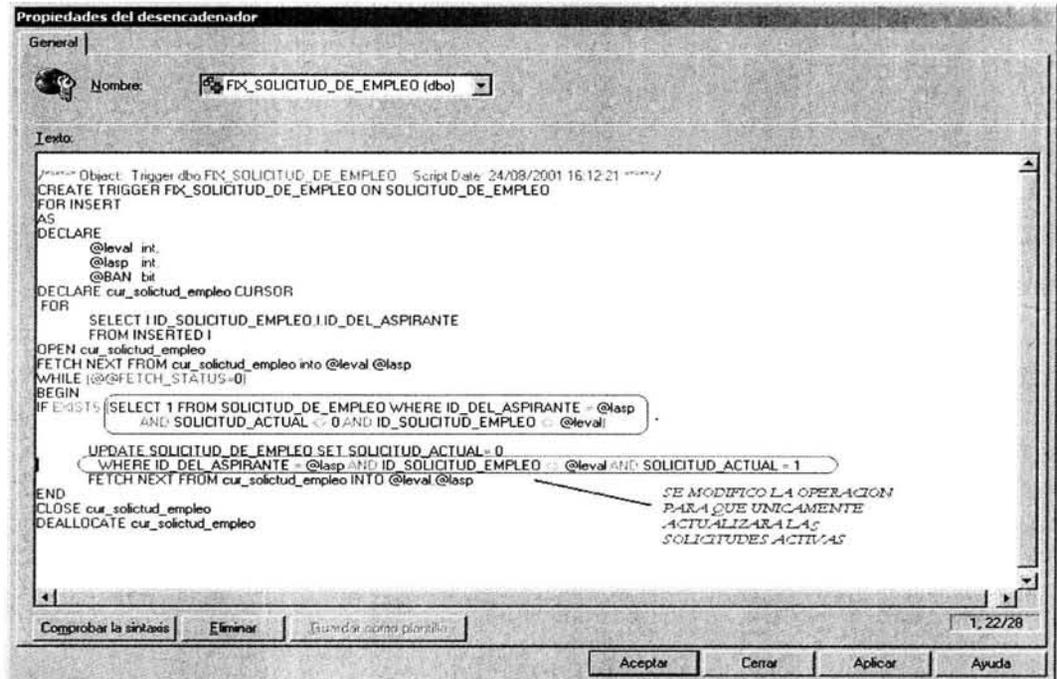


Figura 3.14. Desencadenador “FIX_SOLICITUD_DE_EMPLEO” de la BD reestructurada.

El reporte que muestra (ver figura 3.15) cada uno de los procedimientos almacenados y desencadenadores que fueron actualizados y optimizados se detalla en el ANEXO C.

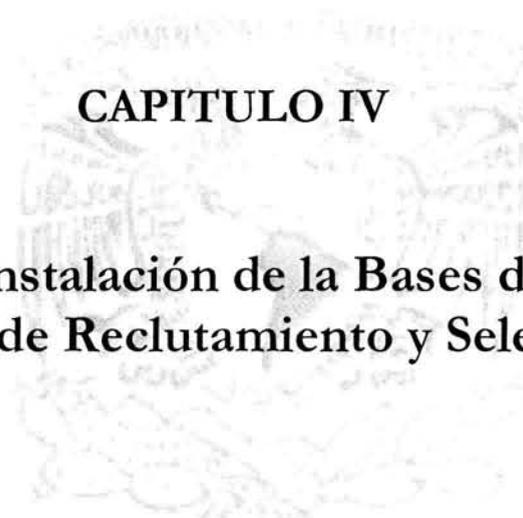
ID OBJETO	TIPO OBJETO	NOMBRE OBJETO
439344192	P	DESCRIPCION_TEMA
455344249	P	NUMCORRESPONDENCIAS
487344363	P	PUNTOSCORRESPONDENCIA
503344420	P	SPCALEND_ASP_EDD
887674210	P	SP_ASP_CONVOCATORIA
519344477	P	SP_BORRA_RECSEL
535344534	P	SP_CANCELACION_CITAS
551344591	P	SP_CLAVE_EVALUACION
407344078	P	SP_DAID
599673184	P	SP_DELRAMASST
615673241	P	SP_DELRAMASST
631673298	P	SP_DELRAMAST
567344648	P	SP_DESC_TEMAS_DEFICITARIOS
663673412	P	SP_ELIMINA_TEMAS
903674267	P	SP_ELIMINA_TEMAS_PUESTOS

Figura 3.15 Reporte de cambios de Procedimientos Almacenados y Desencadenadores.



Con las técnicas propuestas para optimizar los Procedimientos Almacenados y los Desencadenadores se concluye este apartado. Con ello se considera suficiente lo expuesto en este apartado para cumplir con el objetivo de lograr un diseño lógico más efectivo de las Bases de Datos, lo que nos conlleva a obtener un mayor rendimiento de nuestra aplicación.

ESTA TESIS NO SALE
DE LA BIBLIOTECA



CAPÍTULO IV

Migración e Instalación de la Bases de Datos del Sistema de Reclutamiento y Selección

"EL CONOCIMIENTO PERDURARA POR SIEMPRE SOLO SI SE DIFUNDE"

ANONIMO.

CAPITULO IV

PROPUESTA DE MIGRACIÓN E INSTALACIÓN DE LAS BASES DEL SISTEMA DE RECLUTAMIENTO Y SELECCIÓN.

La finalidad de este capítulo es establecer el diseño físico de las bases de datos del Sistema de Reclutamiento y Selección el cual comprende la definición de los almacenes de datos en cuanto a tamaño, la estrategia de distribución física de los archivos de datos de las bases, la estrategia de migración de los datos, la configuración de SQL Server 2000 así como la generación de scripts que nos va a permitir llevar a cabo la instalación semiautomática.

4.1 REESTRUCTURACIÓN DEL DISEÑO FÍSICO DE LAS BASES DE DATOS ACTUALES

Como ya mencionamos es importante diseñar correctamente la base de datos para modelar los requisitos de la compañía y aprovechar las características del hardware y el software desde el primer momento del ciclo de desarrollo de una aplicación de bases de datos, ya que posteriormente es difícil realizar cambios en estos componentes.

Así el diseño físico, es un elemento significativo en el diseño de una base de datos. Este consiste en asignar al diseño lógico medios físicos de tal forma que se aprovechen las características disponibles del hardware y el software, las cuales permiten obtener acceso físico a los datos, mantenerlos de forma que el acceso sea lo más rápido posible e indexarlos.

Partiendo de que en el capítulo III quedo concluido lo referente al diseño lógico de la base de datos aplicando técnicas de diseño que recomiendan los autores, este apartado se pretende definir la mejor forma de almacenar físicamente la nueva estructura de datos en los medios físicos de cada uno de los servidores.

ACTIVIDADES

Mediante las siguientes actividades y aplicando algunas técnicas de diseño recomendadas, se pretende redefinir el diseño físico de las bases de datos actuales así como proponer algunas estrategias de administración, mantenimiento y duplicación de información:



- A. Configuración de SQL Server.
 - 1) Memoria.
 - 2) Actividad de E/S.

- B. Diseño Físico de las Bases de Datos.
 - 1) Sistema de Discos.
 - 2) Definición de los archivos de datos.
 - 3) Definición de las estructuras de datos.
 - 4) Migración de los datos.

- C. Propuesta de la Estrategia de Administración y Mantenimiento de las Bases de Datos.
 - 1) Estrategia de Mantenimiento de Bases de Datos.
 - 2) Estrategia de Respaldo de Información.
 - 3) Estrategia de Recuperación de Información.

- D. Estrategia de Duplicación de Información.
 - 1) Introducción a la duplicación en SQL Server 2000.
 - 2) Metodologías de duplicación.
 - 3) Metodologías de implementación de la duplicación
 - 4) Implementación de la duplicación.

4.2 CONFIGURACIÓN DE SQL SERVER 2000.

En general SQL Server 2000 se puede describir como "autoajutable", ya que ajusta dinámicamente su configuración en función de la actividad y el uso de los recursos. Sin embargo, hay una serie de decisiones relacionadas con la instalación y la configuración que pueden mejorar enormemente el rendimiento de la aplicación.

1) MEMORIA

SQL Server cambia sus requisitos de memoria de manera dinámica basándose en los recursos del sistema disponibles. Al cambiar la carga del equipo y de SQL Server, también cambia la memoria asignada

Cuando SQL Server utiliza la memoria de manera dinámica, realiza una consulta periódica en el sistema para determinar la cantidad de memoria física libre disponible. SQL Server aumenta o reduce la caché del búfer para mantener entre 4 y 10 MB de memoria libre, en función de la actividad del servidor. Si hay menos memoria libre, SQL Server libera memoria para Windows 2000. Si hay más memoria libre, SQL Server volverá a confirmar la memoria para la caché del búfer. SQL Server agrega memoria a la caché del búfer sólo si su carga de trabajo necesita más memoria; la caché del búfer de un servidor inactivo no aumenta.



La configuración recomendada es permitir que SQL Server utilice la memoria de manera dinámica; no obstante, puede establecer manualmente las opciones de memoria y suplantar la capacidad de SQL Server de utilizar la memoria de manera dinámica.

2) ACTIVIDAD DE E/S

Los dos parámetros siguientes pueden influir en el rendimiento de E/S:

- **recovery interval** (min.) (intervalo de recuperación) Número mínimo de minutos que SQL Server necesita para recuperar las bases de datos.
- **Max async io** (nº máx. de operaciones asincrónicas de E/S) Número máximo de solicitudes de E/S pendientes por archivo.

Con el parámetro **recovery interval** puede modificar el intervalo para los puntos de comprobación¹. Después de instalar SQL Server 2000, este parámetro se establece en un minuto, aproximadamente. Es decir, SQL Server calcula el número de puntos de comprobación de manera que el tiempo de recuperación permanezca aproximadamente en un minuto de forma predeterminada. En consecuencia, si desea reducir la actividad de E/S que se produce como resultado del vaciado de la caché, debe aumentar el valor del parámetro **recovery interval**. Un inconveniente de este método es el aumento del tiempo de recuperación.

Además de aumentar el intervalo de recuperación, puede mejorar el rendimiento de E/S si se aprovechan las solicitudes de E/S asincrónicas que SQL Server envía a Windows 2000 y la controladora de disco. De forma predeterminada, SQL Server envía, como máximo, 32 solicitudes de E/S pendientes a un archivo, lo que resulta aceptable para controladoras de disco que no sean sofisticadas. Para reducir el movimiento de las cabezas del disco, las controladoras inteligentes pueden trabajar con más de 32 solicitudes por archivo. Sin embargo, al aumentar el valor del parámetro **max async io**, hay un límite máximo a partir del cual disminuye el rendimiento.

En general, no se recomienda cambiar el parámetro **recovery interval**, ya que aumenta el tiempo de recuperación. No obstante, si la supervisión del sistema indica mucha actividad de E/S causada por los puntos de comprobación (por

¹ Puntos de comprobación. Suceso por el cual el motor de la base de datos escribe en el disco páginas de búfer desfasadas. Las páginas desfasadas son páginas que han sido modificadas, pero las modificaciones aún no se han escrito en el disco. Cada punto de comprobación escribe en el disco todas las páginas que estaban desfasadas en el último punto de comprobación y que aún no se han escrito en el disco. Los puntos de comprobación ocurren periódicamente según el número de grabaciones en el registro que generan las modificaciones de datos o cuando lo solicitan un usuario o el cierre del sistema.



ejemplo, en una aplicación con muchas transacciones pequeñas), el aumento del parámetro `recovery interval` puede tener efectos favorables en el rendimiento.

Otros puntos que puede considerar para reducir la actividad son:

- Distribuir los datos a través de varios discos y varios controladores.
- Reducir la actividad en los casos siguientes:
 - En operaciones de escritura en los registros de transacciones.
 - En operaciones de lectura o escritura en los archivos de datos o de índice que se originan a partir de solicitudes que no se pueden satisfacer en la caché de datos.

4.3 DISEÑO FÍSICO DE LAS BASES DE DATOS

1) SISTEMA DE DISCOS

Generalmente se dice que las bases de datos de SQL Server 2000 de producción no se deben instalar en discos internos, ya que al instalar cada base de datos en su propio subsistema de disco externo, se puede mejorar el rendimiento de las operaciones de lectura y escritura de los datos. Por otro lado a medida que se instalan más discos, aumenta el riesgo de que se produzca un error.

Por lo tanto para evitar la pérdida de datos debido a un error de hardware, se recomienda que todos los entornos se ejecuten en sistemas RAID (Matriz redundante de discos económicos), con ello se proporciona tolerancia a errores y se reducen los riesgos a errores.

RAID

El reflejo y la duplicación, así como la creación de bandas en disco² forman parte de un esquema conocido como RAID (Matriz redundante de discos económicos). Los discos se configuran en estas matrices de RAID para proteger los datos en los discos y ofrecer alta disponibilidad.

Windows 2000 admite los tres niveles de RAID siguientes:

- RAID 0. Esta solución simplemente divide los datos de manera uniforme entre las unidades físicas. Cada volumen distribuido se llama almacenamiento o conjunto de bandas. RAID 0 no protege contra errores de hardware, aunque puede mejorar enormemente el rendimiento de las operaciones de lectura y escritura.

² Se dice que dos discos están *reflejados* si contienen exactamente los mismos datos. La duplicación de discos implica que los discos estén reflejados, pero cada disco es controlado por un dispositivo de controlador separado como las tarjetas (SCSI)



- RAID 1. Denominado comúnmente reflejo de disco (o con duplicación de disco si cada disco tiene una controladora dedicada), RAID 1 escribe datos en dos unidades. Protege contra errores de hardware mediante la creación de dos duplicados exactos de los datos, uno en cada partición del conjunto de espejos. Si la instalación puede elegir en qué espejo se leen los datos, se puede ajustar el uso de los canales de E/S para mejorar el rendimiento. No obstante, el uso de esta solución no es habitual.
- RAID 5. Denominado comúnmente bandas de disco con paridad³, RAID 5 distribuye los datos de manera uniforme entre tres o más discos de la misma forma que RAID 0. Sin embargo, la tolerancia a errores se implementa mediante el cálculo de la información de recuperación acerca de los datos escritos en el disco. Esta información de recuperación (paridad) se distribuye de forma equitativa entre todos los discos participantes, de manera que, si se produce un error en un disco, los datos se pueden reconstruir a partir de la información de paridad contenida en los demás miembros del conjunto de bandas. RAID 5 protege contra errores en una unidad de disco, pero reduce el rendimiento de las operaciones de escritura.

El nivel de RAID más común en entornos de base de datos es una combinación de RAID 0 y 1. Esta combinación se conoce también como RAID 10 o RAID 0/1. Generalmente se considera como la mejor solución para entornos de alto rendimiento, aunque el reflejo o duplicación de discos son muy costosos, ya que es necesario utilizar el doble de la cantidad de espacio en disco para guardar la información dada.

Por otro lado es relevante mencionar las características de hardware de las máquinas-servidores con las que se cuenta recientemente para las Centros de Aplicación (excepto el centro de aplicación del área metropolitana):

Hardware.

COMPAQ EVO

Memoria.

256 Mb mínimo

Componentes.

- Procesador Pentium IV 400 Mhz
- Disco duro de 40 GB
- Monitor
- Unidad de disco flexible de 3.5 pulgadas
- CD-ROM
- Tarjeta de red
- Mouse

³ La información de paridad es el conjunto de valores que pueden utilizarse para recalcularse los datos faltantes.



El Centro de aplicación del área metropolitana tiene una demanda mayor de aplicaciones y el servidor Central en el cual se encuentra concertada la información generada a nivel nacional, se cuenta con servidores con diferentes características debido a las funciones que realizan. Estas se listan a continuación:

Servidor Central

Hardware.

HP 4 XEON

Memoria.

1GB Mb.

Componentes.

- Procesador Pentium II 400 Mhz.
- Disco duro de 20 GB y otro de 15 GB
- Monitor
- Unidad de disco flexible de 3.5 pulgadas
- CD-ROM
- Tarjeta de red
- Controlados de discos SCSI
- Mouse

Servidor del Centro de Aplicación del Área Metropolitana.

Hardware.

HP 4 XEON

Memoria.

1GB Mb.

Componentes.

- Procesador Pentium II 400 Mhz.
- Disco duro de 18 GB
- Monitor
- Unidad de disco flexible de 3.5 pulgadas
- CD-ROM
- Tarjeta de red
- Controlados de discos SCSI
- Mouse

Con base en las recomendaciones anteriores, las características de los equipos, la función que realiza cada equipo y a la cantidad de información de cada centro de aplicación se propone crear los sistemas de discos de la siguiente manera.

Servidor Central

Se propone establecer un esquema en donde se establezca una matriz de disco RAID 5 el cual estará conformado por 5 discos duros (Figura. 4.1) de al menos 10 GB (excepto D el cual tendrá por lo menos 20GB), los cuales se distribuirán de la siguiente forma:



- 1) DISCO(C:). Su función será únicamente la de contener el Sistema Operativo.
- 2) DISCO(D:). Contenedor de archivos de datos.
- 3) DISCO(E:). Contenedor de registro de transacciones.
- 4) DISCO(F:). Contenedor de registro de tempdb.
- 5) DISCO(G:). Contenedor de registro de índices.

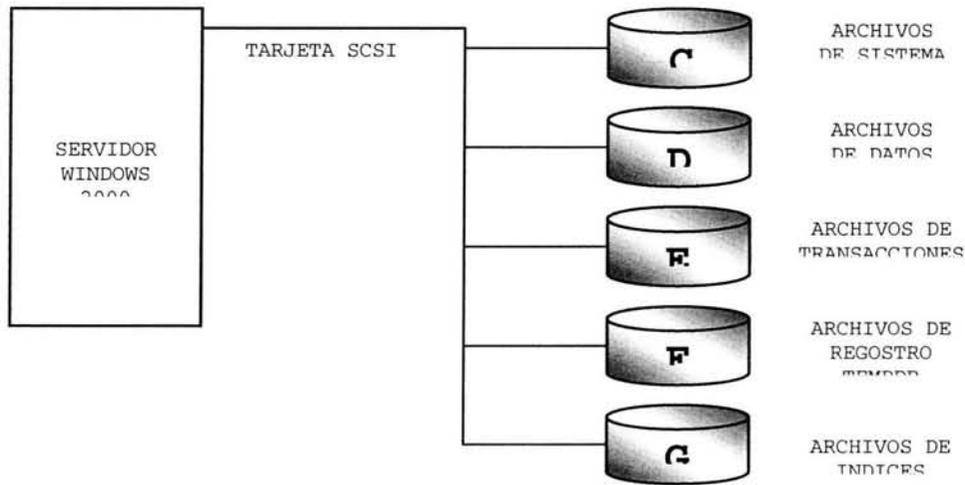


Figura 4.1 Arreglo de discos para el Servidor Central

Servidor del Centro de Aplicación del Área Metropolitana.

Se propone establecer un esquema en donde se establezca una matriz de disco RAID 5 el cual estará conformado por 3 discos duros de al menos 10 GB (Fig. 4.2), los cuales se distribuirán de la siguiente forma:

- 1) DISCO(C:). Su función será únicamente la de contener el Sistema Operativo.
- 2) DISCO(D:). Contenedor de archivos de datos y registro de tempdb.
- 3) DISCO(E:). Contenedor de registro de transacciones y registro de índices.

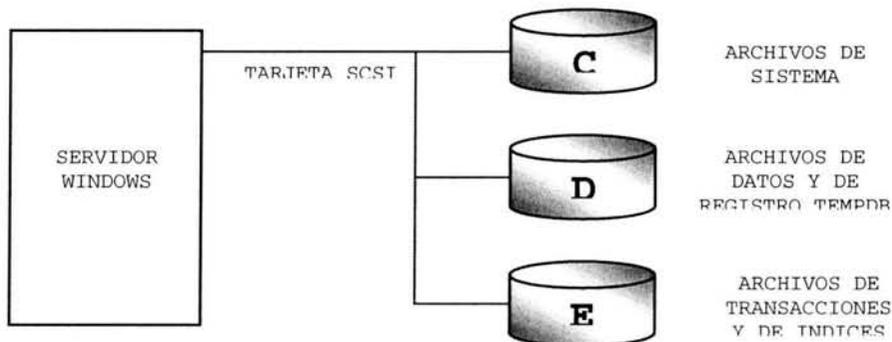


Figura 4.2 Arreglo de discos para el Servidor del Área Metropolitana
Máquinas-Servidores para el resto de los Centros de Aplicación



Para el resto de las máquinas-servidores debido a que se tiene un solo disco, además del costo que representaría realizar un arreglo de disco para cada una de estas máquinas se propone únicamente la creación de dos particiones en el disco (Figura 4.3). La primera partición (C:) en la cual va a residir el Sistema Operativo tendrá un espacio aproximado de 3GB. La segunda partición (D:) contará con alrededor de 37GB y en esta residirán el resto de los programas como SQL Server.

- 1) DISCO(C:). Su función será únicamente la de contener el Sistema Operativo.
- 2) DISCO(D:). Contenedor de archivos de datos, de registro de transacciones, registro de índices y registro de tempdb.

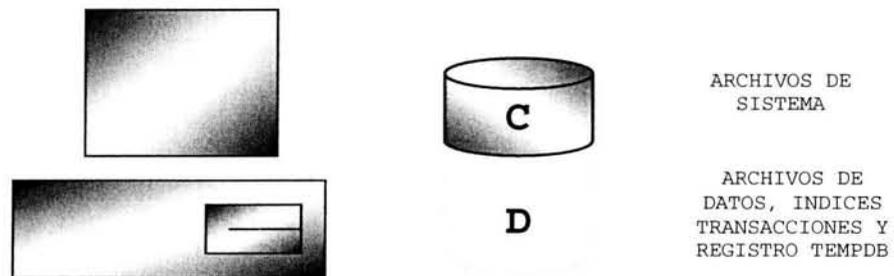


Figura 4.3 Organización del disco de un Centro de Aplicación.

2) DEFINICIÓN DE LOS ARCHIVOS DE DATOS.

SQL Server 2000 almacena una base de datos en un conjunto de archivos del sistema operativo. Las bases de datos de SQL Server 2000 tienen tres tipos de archivos:

- 1) Archivos de datos principales. El archivo de datos principal es el punto de partida de la base de datos y apunta a los otros archivos de la base de datos. La extensión de nombre de archivo recomendada para los archivos de datos principales es *.MDF.
- 2) Archivos de datos secundarios. Los archivos de datos secundarios son todos los archivos de datos menos el archivo de datos principal. La extensión de nombre de archivo recomendada para los archivos de datos secundarios es *.NDF.
- 3) Archivos de registro. Los archivos de registro almacenan toda la información de registro que se utiliza para recuperar la base de datos. Como mínimo, tiene que haber un archivo de registro por cada base de



datos, aunque puede haber varios. La extensión de nombre de archivo recomendada para los archivos de registro es *.LDF.

Todas las bases de datos tienen al menos un archivo primario y un archivo de registro. En cuanto a los archivos de datos secundarios, puede que algunas bases de datos no tengan, mientras que otras pueden tener varios.

Otro aspecto relevante en SQL Server 2000 es el hecho de dividir los archivos de datos en conjuntos, con la finalidad de mejorar el rendimiento mediante su distribución en diferentes dispositivos. Los conjuntos que se pueden considerar son los siguientes:

- Archivos de registro de transacciones.
- Archivos tempdb.
- Archivos de datos.
- Archivos de índice.

Por otro lado es importante mencionar las dimensiones que tienen actualmente las bases de datos con la finalidad de redefinir los nuevos espacios de almacenamiento. El siguiente cuadro muestra esta información:

DIMENSIONES ACTUALES DE LAS BASE DE DATOS		
CENTRAL	DATOS	LOG (REGISTRO)
CENTRAL	10 GB	2 GB
DISTRIBUTION	600 MB	800 MB
TEMPDB	100 MB	100 MB
CENTRO DE APLICACIÓN METROPOLITANO		
METROPOLITANA	3.6 GB	1.25 GB
DISTRIBUTION	600 MB	400 MB
TEMPDB	300 MB	200 MB
RESTO DE LOS CENTROS DE APLICACION		
METROPOLITANA	1.3 GB	500 MB
DISTRIBUTION	300 MB	250 MB
TEMPDB	300 MB	200 MB
BASES EXTERNAS		
ESTOCUP	340MB	150MB
IRH	600MB	200MB
PUESTOS	400MB	100MB

Cuadro 3.1. Dimensiones actuales de las Bases de Datos



En conclusión los almacenes de datos se definieron en base al tipo de archivo, la cantidad de información que contienen actualmente y el número de transacciones que soportan, así como la demanda de evaluación en cada de centro de aplicación.

IMPLEMENTACIÓN DE ALMACENES DE DATOS

Mediante el siguiente script (*SC_CREA_BASES_DE_DATOS.SQL*) se pretende implementar de forma automática los almacenes de datos de cada tipo de servidor:

```
USE master
GO

DECLARE @TIP_SERVER TINYINT,
        @CAE VARCHAR(20)

IF @TIP_SERVER = 1 -- SERVER CENTRAL
BEGIN
    CREATE DATABASE CENTRAL
    ON
    (NAME = CENTRAL_dat,
     FILENAME = 'D:\DEVICES\CENTRAL_dat.mdf',
     SIZE = 100MB, MAXSIZE = 18GB, FILEGROWTH = 500MB),
    FILEGROUP CENTRALINDICES
    (NAME = INDICES_dat,
     FILENAME = 'D:\DEVICES\INDICES_dat.ndf',
     SIZE = 100MB, MAXSIZE = 8GB, FILEGROWTH = 100MB)
    LOG ON(NAME = CENTRAL_log,
           FILENAME = 'D:\DEVICES\CENTRAL_log.ndf',
           SIZE = 50MB, MAXSIZE = 5GB, FILEGROWTH = 500MB)

    CREATE DATABASE DISTRIBUTION
    ON
    (NAME = DISTRIBUTION_dat,
     FILENAME = 'D:\DEVICES\DISTRIBUTION_dat.mdf',
     SIZE = 100MB, MAXSIZE = 1000MB, FILEGROWTH = 50MB)
    LOG ON(NAME = DISTRIBUTION_log,
           FILENAME = 'D:\DEVICES\DISTRIBUTION_log.ndf',
           SIZE = 50MB, MAXSIZE = 800MB, FILEGROWTH = 50MB)

END

IF @TIP_SERVER = 2 -- SERVER METROPOLITANA
BEGIN
    CREATE DATABASE METROPOLITANA
    ON
    (NAME = METROPOLITANA_dat,
     FILENAME = 'D:\DEVICES\METROPOLITANA_dat.mdf',
     SIZE = 4GB, MAXSIZE = 10GB, FILEGROWTH = 500MB),
    FILEGROUP METROINDICES
    (NAME = INDICES_dat,
     FILENAME = 'E:\DEVICES\INDICES_dat.ndf',
     SIZE = 2GB, MAXSIZE = 5GB, FILEGROWTH = 100MB)
```



```
LOG ON(NAME = METROPOLITANA_log,  
FILENAME = 'D:\DEVICES\METROPOLITANA_log.ndf',  
SIZE= 500MB,MAXSIZE=1000MB,FILEGROWTH = 100MB )
```

```
CREATE DATABASE DISTRIBUTION  
ON  
(NAME = CENTRAL_dat,  
FILENAME = 'E:\DEVICES\DISTRIBUTION_dat.mdf',  
SIZE = 600MB, MAXSIZE = 1000MB,  
FILEGROWTH = 50MB )  
LOG ON(NAME = DISTRIBUTION_log,  
FILENAME = 'E:\DEVICES\DISTRIBUTION_log.ndf',  
SIZE= 400MB, MAXSIZE= 800MB,FILEGROWTH = 50MB )  
END
```

IF @TIP_SERVER = 3 --SERVER REGIONAL

```
BEGIN  
CREATE DATABASE @CAE  
ON  
(NAME = REGIONAL_dat,  
FILENAME = 'D:\DEVICES\REGIONAL_dat.mdf',  
SIZE = 1500MB, MAXSIZE = 5000MB,  
FILEGROWTH = 500MB )  
LOG ON(NAME = REGIONAL_log,  
FILENAME = 'D:\DEVICES\REGIONAL_log.ndf',  
SIZE = 500MB, MAXSIZE = 1000MB,  
FILEGROWTH = 100MB )  
  
CREATE DATABASE DISTRIBUTION  
ON  
(NAME = DISTRIBUTION_dat,  
FILENAME = 'D:\DEVICES\DISTRIBUTION_dat.mdf',  
SIZE = 300MB, MAXSIZE = 800MB,  
FILEGROWTH = 50MB )  
LOG ON(NAME = DISTRIBUTION_log,  
FILENAME = 'D:\DEVICES\DISTRIBUTION_log.ndf',  
SIZE = 200MB, MAXSIZE = 500MB,  
FILEGROWTH = 50MB )  
END
```

--BASES DE DATOS COMPLEMENTARIAS

```
CREATE DATABASE PUESTOS  
ON  
(NAME = PUESTOS_dat,  
FILENAME = 'D:\DEVICES\PUESTOS_dat.mdf',  
SIZE = 600MB, MAXSIZE = 700MB, FILEGROWTH = 20MB)  
LOG ON(NAME = PUESTOS_log,  
FILENAME = 'D:\DEVICES\PUESTOS_log.ndf',  
SIZE = 200MB, MAXSIZE = 300MB, FILEGROWTH = 10MB)  
  
CREATE DATABASE ESTOCUP  
ON  
(NAME = ESTOCUP_dat,  
FILENAME = 'D:\DEVICES\ESTOCUP_dat.mdf',  
SIZE = 340MB, MAXSIZE = 400MB, FILEGROWTH = 20MB)
```



```
LOG ON(NAME = ESTOCUP_log,  
FILENAME = 'D:\DEVICES\ESTOCUP_log.ndf',  
SIZE = 150MB, MAXSIZE = 250MB, FILEGROWTH = 10MB)  
  
CREATE DATABASE IRH  
ON  
(NAME = IRH_dat,  
FILENAME = 'D:\DEVICES\IRH_dat.mdf',  
SIZE = 400MB, MAXSIZE = 600MB, FILEGROWTH = 20MB)  
LOG ON(NAME = IRH_log,  
FILENAME = 'D:\DEVICES\IRH_log.ldf',  
SIZE = 100MB, MAXSIZE = 200MB, FILEGROWTH = 10MB)
```

3) CREACIÓN DE LA ESTRUCTURA DE DATOS

La estructura de datos estará basada en la reestructuración del diseño lógico de las base de datos que se realizó en el capítulo anterior.

Ahora bien, SQL Server permite generar secuencia de comandos⁴ de esquemas ya existentes, por lo tanto los siguientes scripts se generaron a partir de las bases de datos ya modificadas, a las cuales como vimos en el capítulo anterior, se les aplicaron algunas técnicas de diseño recomendadas con la finalidad de mejorar su diseño. Las secuencias de instrucciones de SQL Server son:

- Secuencia de instrucciones que genera los tipos de datos definidos por el usuario.
 - *SC_TIPOS_DATOS.SQL*
- Secuencia de instrucciones que crea las tablas de datos en la base de datos.
 - *SC_CREA_TABLAS.SQL*
- Secuencia de instrucciones que crea los índices en la base de datos.
 - *SC_CREA_INDICES.SQL*
- Secuencia de instrucciones que crea las llaves foráneas (relaciones entre las tablas) en la base de datos.
 - *SC_CREA_FK.SQL*
- Secuencia de instrucciones que crea los stored procedures (procedimientos almacenados) en la base de datos.
 - *SC_CREA_STORED_PROCEDURES.SQL*

⁴ Las secuencias de comandos SQL contienen descripciones de las instrucciones utilizadas para crear una base de datos y sus objetos. Puede generar secuencias de comandos a partir de los objetos de una base de datos existente y agregar esos objetos a otra base de datos mediante la ejecución de las secuencias de comandos en esa base de datos. De ese modo, se vuelve a crear la estructura de la base de datos y todos sus objetos



- Secuencia de instrucciones que crea los triggers (desencadenadores) en la base de datos.
 - *SC_CREA_TRIGGERS.SQL*
- Secuencia de instrucciones que crea las estructuras de las bases de datos complementarias.
 - *SC_CREA_ESTRUC_PUESTOS.SQL*
 - *SC_CREA_ESTRUC_IRH.SQL*
 - *SC_CREA_ESTRUC_ESTOCUP.SQL*

IMPLEMENTACIÓN DE LA ESTRUCTURA DE DATOS

Mediante la ejecución secuencial de los scripts antes descritos se pretende implementar automáticamente la estructura de cada una de las bases de datos del sistema de Reclutamiento y Selección, esto nos permitirá asegurar que las estructuras de cada una de las bases de datos sean idénticas en todos los centros de aplicación.⁵

4) MIGRACIÓN DE LOS DATOS

Antes de describir como se realizará la migración de la información, se describirán brevemente los tres métodos con los cuales cuenta SQL Server2000 para migrar los datos de un servidor a otro.

1) Transferencia de datos con una copia de seguridad y restauración.

La copia de seguridad y restauración permite mover los datos de un servidor a otro ya que al hacer una copia de seguridad se realiza también una copia de toda la estructura y datos de una base de datos, lo cual nos permite restaurar esta en otro servidor.

Cabe mencionar que no es muy recomendable el uso de la copia de seguridad y restauración para mover los datos de un servidor a otro cuando se va a realizar un movimiento diario de sus datos, ya que podría toparse con problemas relacionados con la seguridad, esto, porque muy probablemente los usuarios del servidor de origen no estarán contemplados en el servidor de destino. Esta técnica es recomendada en casos en los cuales se va a realizar una copia del servidor en otro equipo, que cumpla la función de servidor de reserva. Además es importante mencionar que mediante esta técnica no es posible realizar modificaciones en sus datos.

⁵ NOTA: El contenido de los scripts anteriores y a los que se hace mención a lo largo de este trabajo, se podrán visualizar en el CD que se anexa en la parte final de este trabajo.



2) Uso de BCP para cargar datos.

BCP es un programa que se utiliza para cargar datos en forma masiva, ya sea de SQL Server a una archivo o viceversa. Puede utilizar esta opción cuando tenga que transferir datos de SQL Server 2000 a versiones anteriores o hacia otros manejadores de bases de datos como Oracle.

Esta herramienta solo transfiere datos desde o hacia un archivo plano de texto, no transfiere el esquema de la base de datos. Además aunque esta herramienta no realiza modificaciones en sus datos, pueda utilizarla para recopilar información solo de ciertas filas y columnas de una o varias tablas.

Por lo general un BCP transfiere de manera más rápida un conjunto de datos a una tabla , comparado con cualquier otro programa que pueda a escribir e insertar datos en una tabla.

Un BCP permite transferir datos de dos maneras: modo nativo y modo de carácter. El modo carácter especifica que los datos se van a transferir en forma legible, es decir, de tal forma que se puede comprender su valor. Una ventaja adicional del modo carácter es que si se produce una error en un archivo, es más fácil detectarlo y corregirlo.

A diferencia del modo anterior, el modo nativo guarda los datos en formato interno, es decir, en código hexadecimal. Por lo tanto el nativo solo puede transferir datos entre servidores con la misma plataforma, es decir *si la copia es entre servidores de SQL Server es óptimo realizarla con un formato nativo.*

Al cargar datos en SQL Server por medio de un BCP, todos los datos se copian de manera predeterminada en un lote y una transacción individual. Esto significa que la carga tiene éxito o falla en su totalidad. Si embargo es posible ejecutar una operación de “commit” en forma periódica para mejorar la concurrencia e iniciar un nuevo lote.

BCP puede operar en dos modos: BCP normal y BCP rápido. En su nivel básico, BCP consiste en ejecutar inserciones. Esto significa que cada fila se procesa como una operación de inserción, se escribe en el registro de transacciones, se modifican los datos y luego se agregan los índices. El BCP rápido se logra quitando los índices y configurando la opción SELECT INTO o copia masiva en una base de datos. Por consiguiente como no hay índices que mantener ni registro de transacciones esta operación se realiza más rápido.

Los siguientes pasos permite que ocurra una trasferencia rápida de datos:

- 1) Quite los índices de la tablas afectadas
- 2) Configure la opción SELECT INTO o copia masiva.
- 3) Cargue sus datos por medio de BCP.
- 4) Vuelva a crear sus índices



- 5) Desactive la opción de SELECT INTO o copia masiva.
- 6) Realice una copia de seguridad de su base de datos

Una razón atractiva de la copia de seguridad masiva y los archivos de formato es que se trata de una herramienta de línea de comandos que permite ejecutar copias masivas en secuencias de comandos .BAT y .CMD por lotes de procesamiento automatizado.

3) Comando BULK INSERT.

Otro método para transferir datos de un servidor a otro es el comando BULK INSERT. Este permite leer un archivo de datos como si fuera un conjunto de registro OLE DB⁶, de esta manera SQL Server puede transferir todo el conjunto de registros hacia una tabla en uno o más pasos (si utiliza un tamaño de lote predeterminado).

Es recomendable utilizar la instrucción BULK INSERT en lugar del programa **BCP** para realizar la copia masiva de datos en SQL Server, debido a que la instrucción BULK INSERT es más rápida que el programa **BCP**. La instrucción BULK INSERT se ejecuta en el servidor en el contexto del servicio MSSQLServer, no en el cliente. Además si el archivo que se copia de forma masiva está también en el servidor, los datos no se mueven por la red. Esto hace que ejecutar la instrucción BULK INSERT desde un archivo del servidor sea la opción más rápida para realizar una copia masiva. No obstante, la instrucción BULK INSERT no puede realizar copias masivas desde una instancia de SQL Server a un archivo de datos.

BULK INSERT permite realizar copias masivas de datos en una instancia de SQL Server mediante la funcionalidad de la herramienta BCP con una instrucción Transact-SQL y no desde el símbolo del sistema.

Recomendaciones para optimizar la copia masiva.

Algunas recomendaciones adicionales para realizar una copia masiva son las siguientes:

- Utilice el bloqueo a nivel de tabla durante la operación de copia masiva.
- Se recomienda utilizar un solo lote que represente el tamaño del archivo completo.
- No se deben crear copias de seguridad del registro de transacciones después de realizar una operación no registrada.
- Si la tabla de origen contiene un índice agrupado o realizar una copia masiva de datos en una tabla con un índice agrupado:

⁶ Interfaz de programación de aplicaciones (API) basada en COM (Modelo de Objeto de Componentes) para tener acceso a datos. OLE DB acepta el acceso a datos almacenados en cualquier formato (bases de datos, hojas de cálculo, archivos de texto, etc.) para los que haya un proveedor OLE DB disponible.



- Realice la copia masiva de datos de la tabla de origen con una instrucción SELECT y una cláusula ORDER BY apropiada para crear un archivo de datos ordenado.
 - Utilice la sugerencia **ORDER** cuando realice copias masivas de datos en SQL Server.
-
- Cuide que la tabla de destino no este siendo duplicada.
 - Cuide que la tabla de destino no contenga desencadenadores

Analizando cada uno de los métodos anteriores nos encontramos con el hecho de que lo más factible para la migración de los datos es utilizar dos de ellos, esto debido a las ventajas y desventajas que tiene cada uno de ellos. Por lo tanto se propone hacer la migración de los datos tomando en cuenta los siguientes puntos:

- Las tablas se deberá catalogar en dos grupos basándose en la función que desempeñan. Un grupo corresponderá a las tablas catalogo, a las cuales no se les da mantenimiento (es decir son estáticas), y el segundo grupo corresponderá a tablas de datos, es decir, tabla que normalmente son actualizadas a nivel regional o central.
- El formato de las copias de seguridad se realizara en formato nativo.
- Se realizarán una copia de los datos de SQL Server 6.5 a archivos mediante el método de BCP. Las copias de los datos de las tablas catalogo se obtendrán de la base de datos CENTRAL y las copias de datos de las tablas de datos se obtendrán de cada uno de los servidores de los Centros de Aplicación.
- El proceso de copias de seguridad se hará mediante una secuencia automática de comandos en línea (*.cmd).
- La recuperación de la información se hará mediante el comando BULK INSERT con las opciones de: bloqueo a nivel tabla (TABLALOCK) y transferencia por lote (ROWS_PER_BATCH).
- El proceso de recuperación se hará mediante un script con instrucciones en el lenguaje TRANSACT SQL de SQL Server(BULK INSERT).

IMPLEMENTACIÓN DE MIGRACIÓN DE DATOS

La implantación de la migración de los datos se realizara mediante los siguientes comandos en línea y scripts, lo cual nos permitirá implementar de forma automática la estrategia anterior:

EL proceso de respaldo de seguridad de las bases de datos en SQL SERVER 6.5 se realizara mediante comando en línea **bcp**, por ejemplo para la tabla "UNIDADES_ADM" el comando queda de la siguiente manera:



```
bcp CENTRAL.dbo.Unidades_Adm OUT  
C:\BCP_TABLES_CENTRAL\Unidades_Adm.TXT -m0 -e  
C:\BCP_TABLES_CENTRAL\Unidades_Adm.LOG -n -V65 -E -  
SSATCPNCENICNP -UREPLICA -PREPLICA
```

Ahora bien la secuencia de comandos “BCP_TABLES_CENTRAL_OUT.CMD” implementa de forma automática el respaldo de todas las tablas de la base de datos central (ver Fig. 4.4) y la secuencia de comandos “BCP_TABLES_REGIONAL_OUT.CMD” implementa de forma automática el respaldo de todas las tablas de la base de datos regional (ver Fig. 4.5).

```
BCP_TABLES_CENTRAL_OUT - Bloc de notas  
Archivo Edición Formato Ayuda  
@ECHO OFF  
rem *****  
rem **  
rem ** Sistema de Reclutamiento y seleccion de Personal **  
rem ** (RecSel 2.0) **  
rem ** **  
rem ** **  
rem ** Proceso: Migración de Información **  
rem ** Proceso: Programa de Recuperación Masiva **  
rem ** Nombre: BCP_TABLES_CENTRAL_OUT.CMD **  
rem ** Fecha: OCTUBRE 2003 **  
rem ** **  
rem *****  
rem *****  
cls  
if "%x1%" == "/?" goto AYUDA  
if "%x1%" == "/d" goto BORRA_DIRECTORIO  
if "%x2%" == "/d" goto BORRA_DIRECTORIO  
if "%x1%" == "/s" goto INICIO  
if "%x2%" == "/s" goto INICIO  
if exist C:\BCP_TABLES_CENTRAL goto ERROR_CARPETA  
goto DIRECTORIO  
:BORRA_DIRECTORIO  
DEL C:\BCP_TABLES_CENTRAL\*. *  
RD C:\BCP_TABLES_CENTRAL  
:DIRECTORIO  
MD C:\BCP_TABLES_CENTRAL  
:INICIO  
if "%x1%" == "/DPP" goto PUESTOS  
if "%x1%" == "/EO" goto ESTOCUP  
if "%x1%" == "/IRH" goto IRH  
if "%x1%" == "/RS" goto RECSEL  
:RECSEL  
echo .  
echo Inicio del Proceso de Extracción de datos de RecSel  
echo Inicio del Proceso de Extracción de datos de RecSel... >> BCP_TABLES_CENTRAL_OUT.log  
echo .  
echo .  
echo Inicio del Proceso de Extracción de Información de Definición de Exámenes Psicológicos  
echo Inicio del Proceso de Extracción de Información de Definición de Exámenes Psicológicos ... >> BCP_TABLES_CENTRAL_OUT.log  
echo .  
bcp CENTRAL.dbo.COMBINACIONES out C:\BCP_TABLES_CENTRAL\RS_COMBINACIONES.TXT /m0 /e C:\BCP_TABLES_CENTRAL\RS_COMBINACIONES.L
```

Figura 4.4 Secuencia de comandos “BCP_TABLES_CENTRAL_OUT.CMD”



```
BCP_TABLES_REGIONAL_OUT - Bloc de notas
Archivo  Edición  Formato  Ayuda
ECHO OFF
rem *****
rem *****
rem "" Sistema de Reclutamiento y Selección de Personal ""
rem "" (RecSel 2.0) ""
rem "" ""
rem "" Proceso: Migración de Información ""
rem "" Proceso: Programa de Recuperación Masiva Regional ""
rem "" ""
rem "" Fecha: OCTUBRE 2003 ""
rem "" ""
rem *****
rem *****

cls
if "%2" == "/" goto AYUDA

if "%2" == "/d" goto BORRA_DIRECTORIO
if "%3" == "/d" goto BORRA_DIRECTORIO

if "%2" == "/s" goto INICIO
if "%3" == "/s" goto INICIO

if exist C:\BCP_TABLES_REGIONAL goto ERROR_CARPETA
goto DIRECTORIO

:BORRA_DIRECTORIO
DEL C:\BCP_TABLES_REGIONAL\*. *
RD C:\BCP_TABLES_REGIONAL

:DIRECTORIO
MD C:\BCP_TABLES_REGIONAL

:INICIO
:RECSEL
echo .
echo Inicio del Proceso de Extracción de datos de RecSel
echo Inicio del Proceso de Extracción de datos de RecSel... >> BCP_TABLES_REGIONAL_OUT.log
echo .

echo .
echo Inicio del Proceso de Extracción de Información de solicitudes
echo Inicio del Proceso de Extracción de Información de solicitudes... >> BCP_TABLES_REGIONAL_OUT.log
echo .

bcdl.dbo.ASPIRANTE out C:\BCP_TABLES_REGIONAL\RS_ASPIRANTE.TXT /m0 /e C:\BCP_TABLES_REGIONAL\RS_ASPIRANTE.LOG /n /v65 /e /
bcdl.dbo.ESTUDIO_SOCIOECONOMICO C:\BCP_TABLES_REGIONAL\RS_ESTUDIO_SOCIOECONOMICO.TXT /m0 /e C:\BCP_TABLES_REGIONAL\RS_ESTU
bcdl.dbo.EXAMEN_MEDICO C:\BCP_TABLES_REGIONAL\RS_EXAMEN_MEDICO.TXT /m0 /e C:\BCP_TABLES_REGIONAL\RS_EXAMEN_MEDICO.LOG /n /
bcdl.dbo.RESULTADO_ENTREVISTA C:\BCP_TABLES_REGIONAL\RS_RESULTADO_ENTREVISTA.TXT /m0 /e C:\BCP_TABLES_REGIONAL\RS_RESULTAD
```

Figura 4.5 Secuencia de comandos “BCP_TABLES_REGIONAL.CMD”

Proceso de recuperación de información En cuanto al proceso de recuperación de cada tabla se realizaran mediante instrucciones en el lenguaje TRANSACT, por ejemplo para la tabla “UNIDADES_ADM”:

```
BULK INSERT paso.dbo.Unidades_Adm
FROM 'C:\BCP_TABLES_CENTRAL\Unidades_Adm.TXT'
WITH (CODEPAGE = 'RAW',DATAFILETYPE = 'native',
KEEPNULLS, MAXERRORS = 0, TABLOCK)
```

El conjunto de todas estas instrucciones en lenguaje de SQL Server se agrupara en el script “BULKINSERT_CENTRAL.SQL” para las tablas de la base de datos central (ver Fig. 4.6) y en “BULKINSERT_REGIONAL.SQL” para las tablas de la base de datos regional (ver Fig. 4.7).



```
BULKINSERT_TABLES_CENTRAL - Bloc de notas
Archivo Edición Formato Ayuda
ECHO OFF
rem *****
rem *****
rem ** Sistema de Reclutamiento y Selección de Personal
rem ** (RecSel 2.0)
rem **
rem **
rem ** Proceso: Migración de Información
rem ** Programa de Inserción Masiva
rem ** Nombre: BULKINSERT_TABLES_CENTRAL.CMD
rem ** Fecha: OCTUBRE 2003
rem **
rem *****
rem *****
cls
if "%xl%" == "?" goto AYUDA
if NOT exist C:\BCP_TABLES_CENTRAL goto ERROR_CARPETA
:DIRECTORIO
MD C:\BCP_TABLES_CENTRAL
:INICIO
if "%xl%" == "/DPP" goto PUESTOS
if "%xl%" == "/EO" goto ESTOCUP
if "%xl%" == "/IRH" goto IRH
if "%xl%" == "/RS" goto RECSEL
if "%xl%" == "/ALL" goto RECSEL
:RECSEL
echo .
echo Inicio del Proceso de Extracción de datos de RecSel
echo Inicio del Proceso de Extracción de datos de RecSel... >> BULKINSERT_TABLES_CENTRAL.Log
echo .
echo .
echo Inicio del Proceso de Extracción de Información de Definición de Exámenes Psicológicos
echo Inicio del Proceso de Extracción de Información de Definición de Exámenes Psicológicos ... >> BULKINSERT_TABLES_CENTRAL.1
echo .
lsq /Sx2 /U sa /P%4 /d CENTRAL /n /Q "BULK INSERT CENTRAL.dbo.COMBINACIONES FROM 'C:\BCP_TABLES_CENTRAL\RS_COMBINACIONES.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx2 /U sa /P%4 /d CENTRAL /n /Q "BULK INSERT CENTRAL.dbo.EXAMEN FROM 'C:\BCP_TABLES_CENTRAL\RS_EXAMEN.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx2 /U sa /P%4 /d CENTRAL /n /Q "BULK INSERT CENTRAL.dbo.GRAFICAS FROM 'C:\BCP_TABLES_CENTRAL\RS_GRAFICAS.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx2 /U sa /P%4 /d CENTRAL /n /Q "BULK INSERT CENTRAL.dbo.PALABRAS FROM 'C:\BCP_TABLES_CENTRAL\RS_PALABRAS.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx2 /U sa /P%4 /d CENTRAL /n /Q "BULK INSERT CENTRAL.dbo.REACTIVOS_COMBINADOS FROM 'C:\BCP_TABLES_CENTRAL\RS_REACTIVOS_COMBINADOS.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx2 /U sa /P%4 /d CENTRAL /n /Q "BULK INSERT CENTRAL.dbo.TEMA FROM 'C:\BCP_TABLES_CENTRAL\RS_TEMA.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx2 /U sa /P%4 /d CENTRAL /n /Q "BULK INSERT CENTRAL.dbo.TEMAS_X_SUBTEST FROM 'C:\BCP_TABLES_CENTRAL\RS_TEMAS_X_SUBTEST.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx2 /U sa /P%4 /d CENTRAL /n /Q "BULK INSERT CENTRAL.dbo.REACTIVOS_LARGOS FROM 'C:\BCP_TABLES_CENTRAL\RS_REACTIVOS_LARGOS.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx2 /U sa /P%4 /d CENTRAL /n /Q "BULK INSERT CENTRAL.dbo.REACTIVOS_PRACTICA FROM 'C:\BCP_TABLES_CENTRAL\RS_REACTIVOS_PRACTICA.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx2 /U sa /P%4 /d CENTRAL /n /Q "BULK INSERT CENTRAL.dbo.TEXTOS_EJERCICIO FROM 'C:\BCP_TABLES_CENTRAL\RS_TEXTOS_EJERCICIO.TXT' WITH (CODEPAGE='RAW', TABLOCK)
```

Figura 4.6 Secuencia de comandos "BULKINSERT_TABLES_CENTRAL.CMD"

```
BULKINSERT_TABLES_REGIONAL - Bloc de notas
Archivo Edición Formato Ayuda
ECHO OFF
rem *****
rem *****
rem ** Sistema de Reclutamiento y Selección de Personal
rem ** (RecSel 2.0)
rem **
rem **
rem ** Proceso: Migración de Información
rem ** Programa de Inserción Masiva Regional
rem ** Nombre: BULKINSERT_TABLES_REGIONAL.CMD
rem ** Fecha: OCTUBRE 2003
rem **
rem *****
rem *****
cls
if "%xl%" == "?" goto AYUDA
if NOT exist C:\BCP_TABLES_REGIONAL goto ERROR_CARPETA
:DIRECTORIO
MD C:\BCP_TABLES_REGIONAL
echo .
echo Inicio del Proceso de Extracción de Información de Solicitudes
echo Inicio del Proceso de Extracción de Información de Solicitudes... >> BULKINSERT_TABLES_REGIONAL.LOG
echo .
echo .
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.ASPIRANTE FROM 'C:\BCP_TABLES_REGIONAL\RS_ASPIRANTE.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.ESTUDIO_SOCIOECONOMICO FROM 'C:\BCP_TABLES_REGIONAL\RS_ESTUDIO_SOCIOECONOMICO.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.EXAMEN_MEDICO FROM 'C:\BCP_TABLES_REGIONAL\RS_EXAMEN_MEDICO.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.RESULTADO_ENTREVISTA FROM 'C:\BCP_TABLES_REGIONAL\RS_RESULTADO_ENTREVISTA.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.SOLICITUD_DE_EMPLEO FROM 'C:\BCP_TABLES_REGIONAL\RS_SOLICITUD_DE_EMPLEO.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.CARRERAS_ASPIRANTE FROM 'C:\BCP_TABLES_REGIONAL\RS_CARRERAS_ASPIRANTE.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.IDIOMAS_ASPIRANTE FROM 'C:\BCP_TABLES_REGIONAL\RS_IDIOMAS_ASPIRANTE.TXT' WITH (CODEPAGE='RAW', TABLOCK)
echo .
echo Inicio del Proceso de Extracción de Información de Seguridad y Auditoría
echo Inicio del Proceso de Extracción de Información de Seguridad y Auditoría... >> BULKINSERT_TABLES_REGIONAL.LOG
echo .
echo .
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.DETALLE_SEGURIDAD_USUARIO FROM 'C:\BCP_TABLES_REGIONAL\RS_ASPIRANTE.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.GRUPOS FROM 'C:\BCP_TABLES_REGIONAL\RS_GRUPOS.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.PISTAS_AUDITORIA FROM 'C:\BCP_TABLES_REGIONAL\RS_PISTAS_AUDITORIA.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.USUARIO_POR_GRUPO FROM 'C:\BCP_TABLES_REGIONAL\RS_USUARIO_POR_GRUPO.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.USUARIOS FROM 'C:\BCP_TABLES_REGIONAL\RS_USUARIOS.TXT' WITH (CODEPAGE='RAW', TABLOCK)
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.PERMISOS FROM 'C:\BCP_TABLES_REGIONAL\RS_PERMISOS.TXT' WITH (CODEPAGE='RAW', TABLOCK)
echo .
echo Inicio del Proceso de Extracción de Información de Programación de Solicitudes
echo Inicio del Proceso de Extracción de Información de Programación de Solicitudes... >> BULKINSERT_TABLES_REGIONAL.LOG
echo .
lsq /Sx1 /U sa /P%3 /d%2 /n /Q "BULK INSERT dbo.CITA_PARA_EVALUACION FROM 'C:\BCP_TABLES_REGIONAL\RS_CITA_PARA_EVALUACION.TXT' WITH (CODEPAGE='RAW', TABLOCK)
```

Figura 4.7 Secuencia de comandos "BULKINSERT_TABLES_REGIONAL.CMD"



4.4 PROPUESTA DE MANTENIMIENTO DE LA BASE DE DATOS

La realización de copias de seguridad y la restauración son probablemente la parte más importante del mantenimiento de una base de datos. Tanto la realización de la copia de seguridad como la restauración requieren de una planeación cuidadosa. El tipo de copias de seguridad depende en un mayor grado del tamaño de sus bases de datos y de las necesidades de restauración que se establezcan.

1) RESTAURACIÓN

Existen dos principales motivos por lo cuales se tienen que restaurar una base de datos:

- 1) Si un disco deja de funcionar, o un archivo de datos se daña.
- 2) Si se requiere deshacer una serie de cambios a la base de datos.

SQL Server tiene dos tipos distintos de recuperación: automática y manual.

Recuperación Automática.

La recuperación automática es el proceso por el cual SQL Server pasa cuando se reinician los servicios del manejador. Este tipo de recuperación no es posible deshabilitarla y no es un proceso que pueda ser controlado, por lo que no se necesita hacer nada especial para que ocurra.

El proceso de recuperación automática garantiza que sin importar el motivo por el cual se detengan los servicios de SQL Server, esté iniciará en un estado consistente lógicamente, esto es, todas la paginas del registro de transacciones que hayan sido confirmadas se escriben en disco (escritura síncrona).

La recuperación automática recupera cada una de las base de datos en el siguiente orden, esto es de acuerdo a la funcionalidad:

- 1) MASTER
- 2) MODEL
- 3) TEMPDB(vuelve a crearse cada vez que se inicia el manejador)
- 4) MSDB
- 5) Las bases de datos NORWIND y PUBS
- 6) Las bases del usuario

Recuperación Manual

La recuperación manual consiste en recobrar una base de datos específica. Este proceso podría incluir desde una copia de base de datos completa, una copia de seguridad diferencial, o la restauración de una o más copias de seguridad del registro de transacciones.

La restauración de una base datos de una copia de seguridad tiene las siguientes restricciones:

- La bases de datos no puede estar en uso.



- Solo se permite restaurar a los usuarios que tengan los permisos apropiados para esta operación.
- Se debe restaurar las copias de seguridad del registro de transacciones en el orden en el que fueron creados.

ESTRATEGIA DE RESTAURACIÓN

En caso de algún *problema* con las bases de datos del servidor, se deberá tener disponible los siguientes elementos:

- 1) Windows 2000 Advance Server.
- 2) Service Pack más reciente de Windows 2000 Advance Server.
- 3) SQL Server 2000, junto con el Service Pack más reciente.
- 4) Copias de seguridad.

Además se debe tener a la mano el manual de instalación del Servidor, en el cual se incluye los siguientes parámetros:

- 5) Número de particiones de los discos.
- 6) Ubicación física en la que se instala SQL Server 2000.
- 7) Configuración de SQL Server.
- 8) Cuentas de red para iniciar los servicios de SQL Server.
- 9) CD de instalación y configuración automática de los elementos de SQL Sever como son la configuración de SQL Server, la creación de bases de datos y la implementación de la duplicación de información.

Por otro lado se contara con dos estrategias de recuperación, el administrador deberá determinar cual seguir de acuerdo a las necesidades:

Recuperación después de perder un disco.

Uno de los escenarios de recuperación más comunes es restaurar bases de datos después de perder un disco en el servidor.

Ante esta situación como la anterior se proponen los siguientes pasos para recuperar la o las bases de datos:

- 1) Debe asegurarse de contar con toda la información necesaria para restaurar los dispositivos y las bases de datos (es decir, los elementos que se listaron en el apartado anterior).
- 2) Debe capturar el registro de transacciones, si es que todavía esta disponible. Si los archivos de registro están disponibles o no fueron afectados, deberán ser respaldados de cada uno de ellos mediante el siguiente script:

```
BACKUP LOG @DATABASE WITH NO_TRUNCATE TO  
BK_TAPE_BASESRECSEL
```



- 3) Quite la o las bases de datos afectadas. Deberá quitar la base o bases de datos afectadas mediante el siguiente script:

```
DROP DATABASE @DATABASE
```

- 4) Reemplace el disco que fallo. Aunque este paso es obvio, es necesario mencionarlo. En el caso del servidor CENTRAL y METROPOLITANA es necesario volver a configurar la matriz RAID 5 desde el Administrador de discos de Windows 2000 Advance Server, lo cual no se describirá ya que esta fuera del objetivo de este trabajo.
- 5) Restaure las bases de datos y los registros de transacciones según sea el caso. Recupere la copia de seguridad de la bases de datos completa y a continuación.

Recuperación después de perder la base de datos master

Hay dos probables escenarios. El primer escenario es aquel en el que el archivo de base de datos o del registro esta dañado o tiene problemas lógicos, pero aún SQL Server funciona. Para restaurar la base es necesario seguir los siguientes pasos:

- 1) Obtener la copia completa de las base de datos *master* mas reciente.
- 2) Inicie SQL Server en modo de un solo usuario mediante los siguientes secuencia de comandos en el símbolo de sistema:

```
> NET STOP MSSQLSERVER  
> SQLSERVREXEC -m
```

- 3) Posteriormente restaure la base de datos master como lo haría con cualquier otra copia de seguridad.
- 4) Al completarse la restauración SQL Server se detiene a sí mismo. Por lo que se requiere únicamente iniciar nuevamente el servicio para regresar al funcionamiento normal.

```
> NET START MSSQLSERVER
```

El otro escenario es que su base de datos master no está disponible ni funcionando y básicamente el servidor se encuentra bloqueado. Para reconstruir la base de datos master siga los siguientes pasos:

- 1) Ejecutar la herramienta de SQL Server para reconstruir la base de datos master(rebuildm.exe).
- 2) Restaurar las bases de datos del sistema: model (en caso de que se haya modificado), la base de datos MSDB y la base de datos de distribución.
- 3) Restaurar las bases de datos de usuario en caso de que alguna de estas haya sido afectada.
- 4) Por último deberá ejecutar el comando *DBCC CHECKDB* en todas las bases de datos con la finalidad de verificar la consistencia de estas.



2) COPIAS DE SEGURIDAD

En lo que se refiere a copias de seguridad el manejador de SQL Sever 2000 es capaz de realizar estas en forma dinámica, es decir, que no se tiene que dejar de usar la base de datos para realizar su copia de seguridad, no obstante es conveniente realizar esta cuando haya menos actividad, ya que esta operación hace uso de los recursos aunque en menor grado. Una característica adicional de esta versión del manejador es que las copias de seguridad son consistentes hasta el punto en que se completa la copia de seguridad.

Se manejan tres tipos de copias de seguridad:

- **Copias de seguridad de bases de datos completas y diferenciales.** Las copias de seguridad completas incluyen el respaldo de tablas, índices, tablas de sistema, objetos de la base de datos y el registro de las transacciones de la base de datos. En lo referente a las copias diferenciales, estas copian todas las páginas de datos modificadas desde la última copia de seguridad completa, por lo que es más fácil restaurarlas a diferencia de una copia de seguridad del registro de transacciones.
- **Copias de seguridad de archivos-grupos de archivos.** Consiste en llevar a cabo el respaldo de un archivo o grupo de archivos con la finalidad de restaurar únicamente el o los archivos que hayan sido afectados.
- **Copias de seguridad de registro de transacciones.** Consiste en respaldar el registro de cada una de las transacciones realizadas en la base de datos con la finalidad de restaurar una base de datos mediante la reproducción de toda la actividad (transacciones) en una base durante un período de tiempo.

Las copias de seguridad pueden realizarse sobre varios dispositivos como son discos, cinta una unidad de red o canalización con nombre. La cinta es el medio preferido, pero tiene algunas limitaciones considerables, siendo una de ellas la velocidad. Las copias de seguridad en disco son por lo general las más veloces.

ESTRATEGIA DE COPIA DE SEGURIDAD

Algunos administradores creen que una alternativa de realizar copias de seguridad, es detener el servicio de SQL Server y luego realizar las copias de seguridad de todos los archivos asociados con SQL Server (en especial la carpeta de /mssql/data). Sin embargo, este método no es tan funcional, ya que tal vez no se permita recuperar las bases de manera individual, y no proporciona la capacidad de realizar una recuperación de fecha u hora.

Considerando lo anterior se propone respaldar las bases de datos de la siguiente manera:

- Se realizaran copias de seguridad directamente de SQL Server y no directamente de archivos individuales.



- *Bases a respaldar.* Se realizarán copias de seguridad de las siguientes bases de datos:
 - Bases de datos del usuario
 - CENTRAL O REGIONAL según sea el caso.
 - IRH
 - ESTOCUP
 - PUESTOS
 - Bases de datos de sistema.
 - MASTER
 - MSDM
 - DISTRIBUTION
- *Frecuencia.* La frecuencia con la que se realizará la copia de seguridad es:
 - Bases de datos del usuario.
 - CENTRAL o REGIONAL según sea el caso. Se realizará una copia completa de base de datos por día y dos copias del registro de transacciones durante el día.
 - IRH, ESTOCUP y PUESTOS Se hará una copia completa de base de datos por semana y una copia del registro de transacciones durante el día.
 - Bases de datos de sistema.
 - MASTER. Se realizará diariamente una copia completa de base de datos. Además de hacer algunas copias especiales cuando se lleven a cabo cambios considerables en el servidor, como cambios de configuración; crear, cambiar o quitar bases de datos; agregar servidores remotos o vinculados, o habilitar la duplicación.
 - MSDM. Esta base de datos contiene soporte para el servicio Agente SQL Server, así como para las operaciones de duplicación, por lo tanto se llevará a cabo una copia de seguridad completa semanalmente así como una copia del registro de transacciones diariamente.
 - DISTRIBUTION Se realizará una copia completa de base de datos por día y dos copias del registro de transacciones durante el día.
- *Dispositivo.* Las copias de seguridad se realizarán en el dispositivo de cinta para los servidores principales (CENTRAL, METROPOLITANA), debido a que son los únicos que cuentan con este dispositivo. Para el resto de las máquinas-servidores estos se realizarán en dispositivos de disco.
- *Caducidad.* Los respaldos diarios caducarán cada semana, los semanales cada mes. Se tendrá que guardar una copia de seguridad completa como un histórico de cada mes, el cual caducará cada 6 meses. También se



resguardará un histórico anual de todas las bases de datos el cual es permanente.

- *Resguardo.* Las copias de seguridad de las bases de datos deberán ser resguardadas en la cintoteca de cada una de las coordinaciones. Además se debe reservar un espacio físico seguro cerca de cada una de las máquinas-servidores para resguardar algunas copias de seguridad y tenerlas al alcance para cualquier percance.
- *Verificación de Integridad.* Aunque SQL Server 2000 incluye la capacidad de verificar la integridad de la copia de seguridad después de realizarla. Se deberá verificar periódicamente que las copias de seguridad no sean corruptas mediante la restauración de estas en un servidor de prueba y asegurando el correcto funcionamiento de las base de datos.

IMPLEMENTACIÓN DE COPIAS DE SEGURIDAD

El primer paso en la preparación de las copias de seguridad es crear un dispositivo de copia de seguridad. Los dispositivos de copias de seguridad se crearan mediante los siguientes scripts:

- *Servidor Central y servidor del área Metropolitana.* Se creara solo un dispositivo para la unidad de cinta, ya que esto es más recomendable debido al costo y la velocidad.

```
EXEC sp_addumpdevice 'tape', 'BK_TAPE_BASESRECSEL', '\\.\tape0'
```

- *Máquinas-Servidores del resto de los centros de aplicación.* Se creara un dispositivo por cada base de datos, ya que esto es más recomendable debido a la seguridad que proporciona tener por separado los respaldos, de forma que si se llega a dañar no se perderán todas las copias de seguridad.

```
EXEC sp_addumpdevice 'disk', 'BK_MASTER',
                    'd:\BK_RECSEL\BK_MASTER.bak'
EXEC sp_addumpdevice 'disk', 'BK_MSDB',
                    'd:\BK_RECSEL\BK_MSDB.bak'
EXEC sp_addumpdevice 'disk', 'BK_REGIONAL',
                    'd:\BK_RECSEL\BK_REGIONAL.bak'
EXEC sp_addumpdevice 'disk', 'BK_DISTRIBUTION',
                    'd:\BK_RECSEL\BK_DISTRIBUTION.bak'
EXEC sp_addumpdevice 'disk', 'BK_IRH',
                    'd:\BK_RECSEL\BK_IRH.bak'
EXEC sp_addumpdevice 'disk', 'BK_PUESTOS',
                    'd:\BK_RECSEL\BK_PUESTOS.bak'
EXEC sp_addumpdevice 'disk', 'BK_ESTOCUP',
                    'd:\BK_RECSEL\BK_ESTOCUP.bak'
```

Es importante que antes de que se realice una copia de seguridad se verifique la integridad de esta, esto porque en el caso de que la base de datos este corrupta, no habrá manera de recuperar una copia de seguridad. Por lo tanto siempre se deberá confirmar la integridad de cada una de las bases de datos antes de



realizar cualquier copia. El siguiente comando nos permitirá comprobar lo anterior:

```
DBCC CHECKDB (@DATABASE)WITH NO_INFOMSGS, TABLERESULTS
```

Por último el siguiente comando nos permitirá realizar las copias de seguridad:

- *Servidor Central y servidor del área Metropolitana.*

- Copias de Bases de Datos completas.

```
BACKUP DATABASE @DATABASE TO BK_TAPE_BASESRECSEL  
WITH FORMAT, RETAINT_DAYS = @NUMDAYS, BLOCKSIZE=8192
```

- Copias del Registro de transacciones.

```
BACKUP LOG @DATABASE WITH NO_TRUNCATE TO  
BK_TAPE_BASESRECSEL WITH FORMAT, RETAINT_DAYS =  
@NUMDAYS, BLOCKSIZE=8192
```

- *Máquinas-Servidores del resto de los centros de aplicación.*

```
BACKUP DATABASE @DATABASE TO @DISPOSITIVO WITH INIT
```

Resumiendo lo anterior el escenario propuesto se establecería más o menos de la siguiente manera:

3:00 AM	Realizar copias de seguridad completas formateando el dispositivo (WITH FORMAT).
7:00 AM	Realizar copias de seguridad del registro de transacciones con la opción de formateo (FORMAT) para las bases de datos.
12:00 PM	Realizar una nueva copia de seguridad del registro de transacciones sin inicializar el dispositivo.

3) SEGURIDAD EN LAS BASES DE DATOS

Una base de datos debe tener un sistema de seguridad sólido para controlar qué actividades se pueden realizar y qué información se puede ver y modificar. Un sistema de seguridad sólido garantiza la protección de los datos, independientemente de la manera en que los usuarios tengan acceso a la base de datos. Por ello se considera importante describir de forma breve la arquitectura de seguridad de SQL Server 2000 con la finalidad de proponer un plan de seguridad que se ajuste al sistema.

ARQUITECTURA DE SEGURIDAD DE SQL SERVER

La arquitectura de seguridad de SQL Server se basa en usuarios y grupos de usuarios como la mayoría de los manejadores de base de datos. Estos usuarios o grupos de usuarios tienen que atravesar por dos etapas de seguridad: *autenticación y autorización* (validación de los permisos). La etapa de autenticación consiste en identificar al usuario con una cuenta de inicio de sesión, si la autenticación tiene



éxito, el usuario se conecta a una instancia de SQL Server. El usuario precisa entonces permisos para el acceso a las bases de datos del servidor. La etapa de validación de permisos (autorización) consiste en controlar las actividades a las que el usuario tiene permiso en la base de datos.

Autenticación de SQL Server

El modo de autenticación determina quién es el responsable de validar las solicitudes de conexión a SQL Server. Existen dos modos de autenticación que son: el Modo de autenticación de Windows, el cual permite a un usuario conectarse mediante una cuenta de usuario de Windows NT 4.0 o Windows 2000 y el Modo de autenticación mixto el cual permite a un usuario conectarse a través de una cuenta de usuario de Windows NT 4.0 o Windows 2000 o una cuenta de SQL Server.

La autenticación de Windows tiene algunas ventajas con respecto a la autenticación de SQL Server, principalmente debido a su integración en el sistema de seguridad de Windows NT 4.0 o Windows 2000. La seguridad de Windows NT 4.0 y Windows 2000 ofrece más posibilidades, como, la validación segura y el cifrado de las contraseñas, las auditorías, la caducidad de contraseñas, la longitud mínima de contraseñas y el bloqueo de la cuenta tras varios intentos de inicio de sesión no válidos. Por otro lado también se disminuyen las tareas de administración.

Autorización (Validación de permisos)

Una vez que se ha autenticado un usuario y se le ha permitido iniciar una sesión en una instancia de SQL Server, es necesario que tenga una cuenta en cada base de datos a la que deba tener acceso. El hecho de requerir una cuenta de usuario en cada base de datos evita que los usuarios se conecten a una instancia de SQL Server y tengan acceso a todas las bases de datos de un servidor.

PLAN DE SEGURIDAD

Un plan de seguridad debe identificar qué usuarios pueden ver qué datos y qué actividades pueden realizar en la base de datos. Para desarrollar un plan de seguridad se debe :

1. Enumerar todos los elementos y actividades de la base de datos que deban controlarse con un plan de seguridad.
2. Identificar los individuos y grupos de la organización.
3. Establecer las referencias cruzadas entre las dos listas para identificar qué conjuntos de datos puede ver cada usuario y las actividades que puede realizar en la base de datos.



4.5 DUPLICACIÓN DE INFORMACIÓN

4.5.1 INTRODUCCIÓN A LA DUPLICACIÓN

La duplicación en SQL Server 2000 es una tecnología que permite copiar, distribuir y posiblemente modificar datos y objetos entre bases de datos, con la finalidad de mantener sincronizadas las bases de datos de una organización y así en consecuencia garantizar la consistencia entre estas.

La duplicación de información tiene las siguientes ventajas:

- Permite que los datos estén más cerca de los usuarios, es decir, los usuarios acceden localmente a los datos evitando retrasos por la inaccesibilidad o la lentitud de la red.
- Permite mantener a usuarios móviles ya que estos pueden estar desconectados de la red y propagar sus cambios en cuanto se conecten.
- Reduce los conflictos entre varios usuarios que quieran acceder a los mismos datos.
- Distribuye el tiempo de procesamiento en múltiples servidores.
- Aumenta el rendimiento acumulado de lectura.
- Particiona los datos de acuerdo a las necesidades de las diferentes unidades de negocio o unidades de usuarios.
- Permite una mayor autonomía debido a que los usuarios remotos pueden operar en forma autónoma con los datos duplicados.
- Permite utilizar la duplicación como parte de una estrategia personalizada de servidor de reserva.

La duplicación se implementa principalmente cuando se tienen los siguientes casos:

- Múltiples usuarios o localidades necesitan los mismos datos.
- Construir aplicaciones de datos que tienen que utilizarse en entornos en línea y sin conexión.
- Construir aplicaciones web con las que los usuarios puedan examinar grandes cantidades de datos.
- Cuando es necesario proveer de un alto rendimiento local.

4.5.2 MODELO DE DUPLICACIÓN

El modelo está compuesto de los siguientes elementos: un publicador, un distribuidor, suscriptores, publicaciones, artículos y suscripciones. Además existen varios procesos de duplicación responsables de copiar y mover los datos entre el publicador y el suscriptor, estos son: el Agente de instantáneas, el Agente de distribución, el Agente del lector del registro, el Agente del lector de cola y el Agente de mezcla.



A continuación se da una breve explicación de los elementos de la duplicación con la finalidad de comprender de forma más amplia el funcionamiento del modelo de duplicación y las diferentes metodologías que están disponibles para aplicar.

El **publicador (Publisher)** es un servidor que pone los datos a disposición de otros servidores para poder duplicarlos. El publicador puede disponer de una o más publicaciones, cada una representa un conjunto de datos relacionados lógicamente. Además se trata del servidor donde se especifican los datos que se van a duplicar, el publicador detecta qué datos han cambiado durante la duplicación transaccional y mantiene información acerca de todas las publicaciones del sitio.

El **distribuidor (Distribuidor)** es un servidor que aloja la base de datos de distribución y almacena los datos históricos, transacciones y metadatos. La función del distribuidor varía según el tipo de duplicación implementada.

Un **distribuidor remoto** es un servidor en un lugar diferente del publicador, y que se configura como distribuidor de duplicación. Un **distribuidor local** es un servidor configurado como publicador y distribuidor de duplicación.

Los **suscriptores** son servidores que reciben los datos duplicados. Los suscriptores se suscriben a publicaciones, no a artículos individuales de una publicación, y solamente se suscriben a publicaciones que necesitan, no a todas las publicaciones disponibles de un publicador. Según el tipo y las opciones de duplicación elegidas, el suscriptor también puede devolver los datos modificados al publicador, o volver a publicar los datos en otros suscriptores.

Un **artículo** es una tabla de datos, una partición de datos o un objeto de base de datos especificado para su duplicación. Un artículo puede ser una tabla entera, algunas columnas (con un filtrado vertical), algunas filas (con un filtrado horizontal), un procedimiento almacenado o una definición de vista, la ejecución de un procedimiento almacenado, una vista, una vista indexada o una función definida por el usuario.

Una **publicación** es un conjunto de uno o más artículos de una base de datos. Esta agrupación de varios artículos facilita especificar un conjunto de datos relacionados lógicamente y los objetos de bases de datos que desea duplicar conjuntamente.

Una **suscripción** es una petición de copia de datos o de objetos de base de datos para duplicar. Una suscripción define qué publicación se recibirá, dónde y cuándo. Un Publicador (**suscripción de inserción**) o un Suscriptor (**suscripción de extracción**) pueden solicitar la sincronización o distribución de datos de una suscripción. Una publicación puede admitir una combinación de suscripciones de inserción y extracción.



El **Agente de instantáneas** se utiliza con todos los tipos de duplicación. Prepara esquemas y archivos de datos iniciales de procedimientos almacenados y tablas publicadas, almacena los archivos de instantáneas y registra la información acerca del estado de sincronización en la base de datos de distribución.

El **Agente de lector del registro** se utiliza en la duplicación transaccional. Mueve las transacciones marcadas para duplicación desde el registro de transacciones del publicador a la base de datos de distribución. Cada base de datos publicada tiene su propio Agente de lector del registro, que se ejecuta en el distribuidor y se conecta con el publicador.

El **Agente de distribución** se utiliza en la duplicación de instantáneas y transaccional. Transfiere a los suscriptores los trabajos de instantáneas y las transacciones mantenidas en la base de datos de distribución. Este suele ejecutarse en el distribuidor, para suscripciones de inserción, o en el suscriptor, para suscripciones de extracción.

El **Agente de mezcla** se utiliza con la duplicación de mezcla. Aplica la instantánea inicial en el suscriptor, y transfiere y reconcilia los cambios incrementales de datos que suceden. Cada suscripción de mezcla tiene su propio Agente de mezcla, que se conecta con el publicador y con el suscriptor, y los actualiza. Este suele ejecutarse en el distribuidor, para suscripciones de inserción, o en el suscriptor, para suscripciones de extracción. El Agente de mezcla suele enviar los cambios desde el suscriptor al publicador, y después descarga los cambios desde el publicador hasta el suscriptor, durante una mezcla bidireccional normal. Los cambios también pueden transferirse en una sola dirección si configura el tipo de intercambio del agente.

En general el modelo de duplicación trabaja de la siguiente manera (ver Figura 4.8):

1. El publicador es el propietario de la información de la base de datos origen, por tanto este hace que los datos estén disponibles para la publicación y envía al distribuidor los cambios en los datos publicados.
2. El distribuidor recibe todas las modificaciones realizadas a los datos publicados. Luego guarda los datos y los reenvía a los suscriptores en el momento apropiado.
3. La base de datos de suscripción recibe copias de los datos o de transacciones sostenidas en la base de datos de distribución.

Consideraciones Adicionales de la duplicación.

- Un suscriptor no puede suscribirse a un solo artículo de una publicación
- En caso de mantener la integridad referencial de las tablas en los suscriptores deberá sincronizar todos los artículos involucrados al mismo tiempo.



- En caso de que un artículo de una publicación haga referencia a otro objeto de la base de datos, este deberá ser también publicado.
- Un servidor de distribución puede soportar más de un publicador o suscriptores.

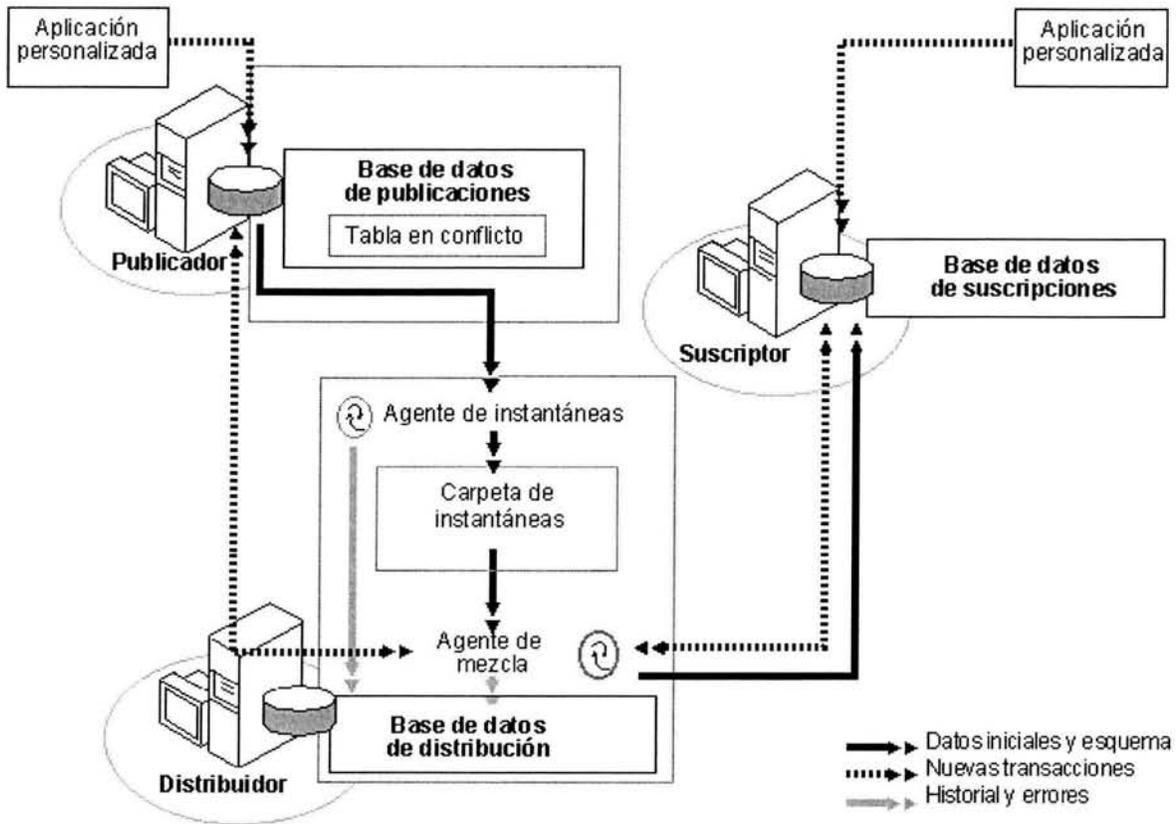


Figura 4.8 Modelo de Duplicación de Mezcla

4.5.3 METODOLOGÍAS DE DUPLICACIÓN

Existen tres tipos básicos de duplicación que son: la duplicación transaccional, la duplicación de instantáneas y la duplicación de mezcla. Al combinarse en varios grados se crean distintas metodologías, las cuales tienen sus propias características de autonomía⁷ y lactancia⁸ (ver figura 4.9). Entre estas metodologías encontramos las siguientes:

⁷ Autonomía. Independencia que un sitio tiene respecto a otros sitios cuando se realizan modificaciones a los datos.

⁸ Lactancia. Intervalo de tiempo transcurrido entre el momento en que se completa un cambio de datos en un servidor y el cambio aparece en otro (por ejemplo, intervalo transcurrido entre que el cambio se realiza en un publicador y aparece en el suscriptor).



- Duplicación de mezcla.
- Duplicación de instantáneas.
- Duplicación transaccional.
- Duplicación de instantáneas con actualización de suscriptores.
- Duplicación transaccional con actualización de suscriptores.
- Transacciones distribuidas.

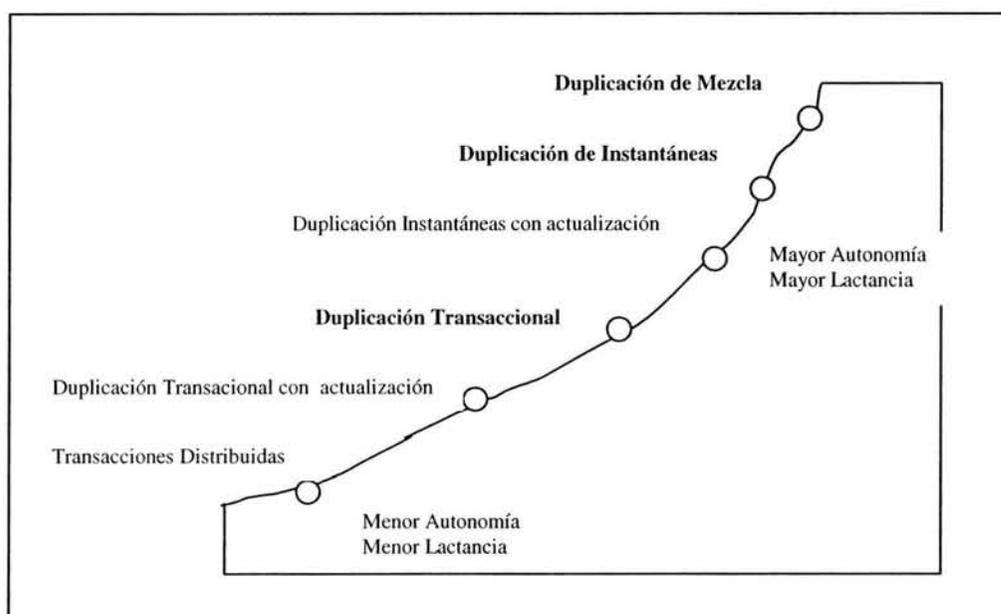


Figura 4.9. Metodologías de duplicación

Duplicación de Mezcla

La duplicación de mezcla permite que cada sitio realice cambios en su copia local de los datos duplicados. En cierto punto, las modificaciones se envían a la base de datos de publicación, en donde se mezclan con las modificaciones de otros sitios, de tal manera que todos los sitios reciben las actualizaciones de los demás sitios.

La duplicación de mezcla ofrece la mayor cantidad de autonomía del sitio y puede aceptar mayor lactancia. No obstante tiene la menor cantidad de consistencia transaccional.

La consistencia transaccional se pierde aquí, ya que varios sitios podrían estar actualizando datos en distintos momentos, es decir, un sitio en específico no espera a que se envíen sus actualizaciones a todos los sitios antes de continuar trabajando.



La duplicación de mezcla es más adecuada en los entornos en los cuales se minimiza la probabilidad de que varios sitios estén trabajando, en especial modificando, sobre los mismos datos. Por ejemplo los sitios que tienden a modificar sus registros solamente, pero que hacen uso de la información del resto de las ubicaciones son candidatos para la duplicación de mezcla.

Duplicación de Instantáneas

La duplicación de instantáneas consiste en copiar y distribuir todos los elementos del servidor publicador hacia la base de datos de suscripción. Este tipo de duplicación es el más fácil de configurar y mantener.

La duplicación de instantáneas tiene un alto nivel de autonomía, además de garantizar la consistencia transaccional, ya que todas las transacciones se aplican solo en el servidor de publicación. La autonomía puede ser útil en lugares en los que los datos se necesitan de solo lectura y en donde no afecte una mayor cantidad de lactancia.

Duplicación de Transaccional

La duplicación transaccional consiste en enviar las transacciones del publicador al suscriptor en un solo sentido. La única manera en que el suscriptor puede hacer modificaciones es hacerlo directamente en el servidor de publicación, ya que el suscriptor debe tratar a los datos como de solo lectura.

Este tipo de duplicación permite un nivel medio de autonomía y lactancia, ya que el suscriptor no debe estar en contacto en todo momento con el publicador, pero las sincronizaciones regulares son útiles y la cantidad de datos que se mueve es relativamente pequeña.

Duplicación de Instantáneas con suscripciones actualizables

La duplicación de instantáneas con suscriptores actualizables es la duplicación de instantáneas pero con la opción de que a los suscriptores se les permite modificar los datos en el publicador.

Duplicación de Transaccional con suscripciones actualizables

La duplicación de transaccional con suscriptores actualizables es la duplicación de transaccional con la opción de que los suscriptores modifiquen datos en el publicador.

Transacciones distribuidas

La duplicación de transacciones distribuidas consiste en replicar las modificaciones a la base de datos del publicador en forma simultánea.

Como cada suscriptor tiene que estar en contacto con el publicador y con los demás en todo momento, hay muy poca autonomía y no hay lactancia. Este tipo



de duplicación es útil en escenarios en los que todos deben tener versiones en tiempo real de los datos.

4.5.4 MÉTODOS DE IMPLEMENTACIÓN

Existen dos métodos para implementar la duplicación:

- 1) Asistentes de duplicación y propiedades.
- 2) Interfaces de programación

1) Asistentes de duplicación y propiedades.

Los asistentes de duplicación y propiedades son herramientas gráficas que se utilizan para facilitar la instalación y el mantenimiento de la duplicación.

2) Interfaces de programación de la duplicación.

Las interfaces de programación son una alternativa al uso del Administrador corporativo de SQL Server (Asistentes de duplicación) para implementar la duplicación. A continuación se describen brevemente las interfaces de programación existentes para implementar la duplicación:

- Los controles de Microsoft ActiveX, utilizados en aplicaciones personalizadas mediante Microsoft Visual Basic o Microsoft Visual C++, permiten programarlos para administrar y controlar el Agente de instantáneas, el Agente de distribución y el Agente de mezcla. Estos controles pueden utilizarse para programar la actividad necesaria para que funcione la duplicación. Por ejemplo, es posible que desee mostrar un botón Sincronizar en una aplicación que pueda utilizarse con y sin conexión. Este botón puede asociarse con el control ActiveX de mezcla y, cuando los usuarios hagan clic en él, se conectarán al publicador y al Agente de mezcla para que la publicación especificada mezcle y sincronice los datos.
- SQL-DMO (SQL Distributed Management Objects) permite crear aplicaciones personalizadas mediante Visual Basic o C++, que permiten configurar, implementar o mantener la topología de duplicación. SQL-MO puede utilizarse para programar la administración de la duplicación, como configurar la distribución, crear suscripciones, etc.
- La Interfaz de distribuidor de duplicación permite duplicar datos de orígenes de datos heterogéneos como Microsoft Access u Oracle. La interfaz del Distribuidor de duplicación la utilizan principalmente los proveedores de servicios independientes u otros usuarios que necesitan desarrollar una aplicación de duplicación personalizada basada en los orígenes de datos de que disponen.
- La creación de secuencias de comandos de duplicación mediante procedimientos almacenados del sistema Transact-SQL permite



automatizar algunas tareas de duplicación, configurar la duplicación e implementar suscripciones en múltiples servidores. Los procedimientos almacenados se suelen utilizar en secuencias de comandos que pueden ejecutarse al configurar la duplicación en múltiples servidores (por ejemplo, al crear suscripciones a una publicación en múltiples suscriptores).

4.2.5 IMPLEMENTACIÓN DE LA DUPLICACIÓN

La implementación de la duplicación de información en el Sistema de Reclutamiento y Selección de Personal se llevará a cabo mediante interfaces de programación de secuencias de comandos utilizando procedimientos almacenados del sistema Transact-SQL, debido a que la ejecución de una secuencia de comandos puede ser más rápida y eficiente que la ejecución manual de los mismos pasos repetidamente, mediante el Administrador corporativo de SQL Server. Además este método de implementación constituye una herramienta para reducir el número de errores en el proceso de instalación y configuración de la duplicación.

La implementación de la duplicación comprende las siguientes etapas:

1. Configuración de la duplicación.
2. Publicación de los datos y objetos de bases de datos.
3. Suscripción a publicaciones.
4. Generación de la instantánea inicial.
5. Aplicación de la instantánea inicial.
6. Sincronización de datos.

A continuación se pretende llevar a cabo la implementación de la duplicación mediante la describe breve de cada una de las etapas cada una de ellas.

Configuración de la duplicación

La configuración de duplicación es el proceso de identificar a los publicadores, distribuidores y suscriptores. La configuración de la duplicación comprende los siguientes pasos:

- 1) El primer paso en la configuración de la duplicación es la identificación del servidor de distribución. En nuestro caso los servidores de distribución corresponden a estos mismos, es decir, cada servidor es el encargado de distribuir su propia información. Existen dos razones por la que los servidores fungen como sus propios distribuidores:
 - La disminución de recursos económicos lo cual no permite comprar más equipos de cómputo que desempeñen esta actividad.
 - La distribución en 63 Centros de aplicación y en consecuencia en 63 equipos, permite distribuir las cargas de trabajo de forma que cada una



de las máquinas-servidores tenga la capacidad de realizar el trabajo de la distribución de la información.

En el caso del servidor Central sería recomendable utilizar otro equipo que realice esta función, debido a la carga de trabajo que representa el replicar a 63 centros de aplicación.

El siguiente procedimiento almacenado nos permite identificar el distribuidor de una base de datos:

```
sp_adddistributor 'SATCPNCENICNP'
```

- 2) Creación de una base de datos de distribución. Es necesario reservar un espacio de almacenamiento, que nos permita controlar los procedimientos, esquema y metadatos utilizados en la duplicación. La base de datos deberá crearse en cada uno de los servidores, mediante el procedimiento almacenado *sp_adddistributiondb*. Ejemplo:

```
sp_adddistributiondb 'distribution', 'D:\DEVICES\', 'DISTRIBUTION_dat.mdf', 100,  
'D:\DEVICES\','DISTRIBUTION_log.log', 50, 0, 2, 8, 'sa', '7becver7', 0
```

- 3) Especificar el publicador. Se deberá habilitar cada uno de los servidores como publicadores. En nuestro caso todos los servidores tienen la necesidad de duplicar su información. Mediante el procedimiento almacenado *sp_adddistpublisher* se habilitarán los publicadores. Ejemplo:

```
sp_adddistpublisher 'SATCPNCENICNP', 'DISTRIBUTION', 0, 'sa', '7becver7',  
'D:\Microsoft SQL Server\MSSQL\DATA\','NULL',1,0
```

- 4) Especificar las bases de datos de publicación. La base de datos de publicaciones es la base de datos del publicador que es el origen de los datos y objetos de base de datos que va a duplicar. Cada base de datos utilizada en la duplicación debe habilitarse como base de datos de publicaciones mediante el procedimiento almacenado del sistema *sp_replicationdboption*. Ejemplo:

```
sp_replicationdboption 'CENTRAL','merge publish','TRUE',0
```

- 5) Especificar el suscriptor o los suscriptores. Se deberá habilitar cada uno de los servidores que funcionan como suscriptores. En nuestro caso todos los servidores tienen la necesidad de recibir información, específicamente del servidor CENTRAL.

Publicación de datos y objetos de las bases de datos.



La publicación de datos requiere que se definan la información a duplicarse, es decir, las tablas (artículos), sus particiones⁹ horizontales (de fila) y verticales (de columna) y con ello las publicaciones.

Como ya mencionamos en el capítulo número dos, en el Sistema de Reclutamiento y Selección existen dos diferentes esquemas de replicación de tipo SQL Server (Central a Regional y Regional a Central). Con base a ello están definidas actualmente las publicaciones en SQL Server 6.5, por ejemplo se tienen los siguientes diferentes tipos de publicación:

de Central a Regional

Publicación incondicional: Para este tipo de publicación solo se tiene una sola publicación a la cual se suscriben todos los Centros de aplicación debido a que la información que reside en cada una de las tablas se envía completamente, sin distinción, a cada centro de aplicación por ser información referente a las baterías de exámenes.

BATERÍA DE EXÁMENES

TABLAS O ARTICULOS	
1	BATERIA DE EXAMENES
2	EXAMENES POR BATERIA
3	BATERIA PROC NIVEL REGIONAL
4	TIPO DE EVALUACIÓN

Publicación condicional (con filtro): Para este tipo de publicación se tiene una publicación por cada centro de aplicación debido a que la información es filtrada de acuerdo al centro de aplicación por ser información referente a funcionarios que van a ser evaluados dependiendo de la unidad administrativa a la que pertenezcan.

Cabe mencionar que este tipo de publicaciones tienen un mantenimiento muy complejo debido a el número de publicaciones que se genera, es decir, se tienen 63 publicaciones iguales de “PROCESO DE CALENDARIZACION”, solo con diferencias en el filtro. Ahora bien si se tienen 3 publicaciones de este tipo multiplicadas por 63 CAES que hay, se generan un total de 189.

Es muy importante mencionar que SQL Server 2000, existe un solución para este tipo de esquemas, filtros dinámicos.

⁹ Las particiones horizontales, verticales y dinámicas permiten filtrar los datos de tal forma que se duplique únicamente la información necesaria con la finalidad de reducir espacio en el suscriptor y conseguir mayor seguridad.



Los filtros dinámicos le permiten crear una publicación de mezcla y, a continuación, filtrar datos de la tabla de publicación proporcionando distintas particiones de datos a diferentes suscriptores. Las ventajas que presenta utilizar filtros dinámicos son:

- Se almacenarán menos publicaciones en el publicador. Esto reduce la sobrecarga que supone administrar múltiples publicaciones.
- Utilizar funciones definidas por el usuario en el filtro dinámico le permite filtrar criterios.
- El suscriptor sólo recibe la información que necesita ya que los datos están filtrados basándose en las propiedades de conexión del Agente de mezcla de la suscripción.

PROCESO_DE_CALENDARIZACION

TABLAS O ARTICULOS		
1	ASPIRANTE_EXT	ID_CENTRO_APP>=80000000 AND ID_CENTRO_APP<=89999999
2	CARRERAS_ASPIRANTE_EXT	ID_CENTRO_APP>=80000000 AND ID_CENTRO_APP<=89999999
3	IDIOMAS_ASPIRANTE_EXT	ID_CENTRO_APP>=80000000 AND ID_CENTRO_APP<=89999999
4	SOLICITUD_EMPLEO_EXT	ID_CENTRO_APP>=80000000 AND ID_CENTRO_APP<=89999999

de Regional a Central

Publicación incondicional: Para este tipo de publicación se tiene una publicación en cada servidor correspondiente a un centro de aplicación, esto debido a que se debe concentrar en el servidor Central todas las programaciones de exámenes generadas en los CAES.

PROGRAMACION_DE_SOLICITUDES

TABLAS O ARTICULOS	
1	CITA PARA EVALUACIÓN
2	EVALUACIONES
3	EXÁMENES_APLICAR
4	EXPEDIENTES
5	PROGRAMACIONES
6	SOLICITUDES

Por otro lado la creación de las publicaciones comprende los siguientes pasos:



- 1) Crear la publicación. Deberá dar de alta cada una de las publicaciones que existe actualmente en SQL Server 6.5¹⁰, mediante el procedimiento almacenado *sp_addmergepublication*.

Con base en los diferentes tipos de publicaciones y la funcionalidad de cada una de estas se determino los parámetros de implementación. Por ejemplo la publicación de “CENTRO DE APLICACION”.

```
sp_addmergepublication
@publication = CENTRO_DE_APLICACION',
@description = 'INFORMACION REFERENTE AL CENTRO', @retention = 14,
@sync_mode = 'native', @allow_push = 'TRUE', @allow_pull = 'TRUE',
@allow_anonymous = 'FALSE', @enabled_for_internet = 'FALSE',
@centralized_conflicts = 'FALSE', @dynamic_filters = 'TRUE',
@snapshot_in_defaultfolder = 'FALSE', @alt_snapshot_folder = 'D:\DEVICES',
@compress_snapshot = 'FALSE', @conflict_retention = 14,
@keep_partition_changes = 'FALSE', @allow_subscription_copy = 'FALSE',
@validate_subscriber_info = 'SUSER_SNAME()', @add_to_active_directory = 'FALSE',
@max_concurrent_merge = 0, @max_concurrent_dynamic_snapshots = 0
```

- 2) Crear una instantánea inicial¹¹. Es necesario para cualquier de los tres tipos de duplicación, crear una instantánea inicial por cada publicación con la finalidad de situar el esquema y los datos en el directorio de trabajo de duplicación. El procedimiento almacenado nos va a permitir realizar esta actividad. Un ejemplo:

```
sp_addpublication_snapshot @publication = 'CENTRO_DE_APLICACION'
```

- 3) Agregar los artículos a las publicaciones. Ya definidas cada una de las publicaciones se procederá a agregar cada uno de los artículos que le corresponden (tablas), mediante el procedimiento almacenado *sp_addmergearticle*. El siguiente ejemplo muestra el como

```
sp_addmergearticle
@publication = 'CENTRO_RECSEL',
@article = 'UNIDAD_ADM_RECSEL', @source_owner = 'dbo',
@source_object = 'UNIDAD_ADM_RECSEL', @type = 'table', @description = null,
@column_tracking = 'true', @pre_creation_cmd = 'none', @creation_script = null, @schema_option =
0x000000000000CFF1, @article_resolver = null,
@subset_filterclause = 'ID_CENTRO_APP = CASE HOST_NAME() WHEN
"REGV77AM" THEN 2 ELSE 3 END',
@vertical_partition = 'false', @destination_owner = 'dbo',
@auto_identity_range = 'false', @verify_resolver_signature = 0, @allow_interactive_resolver = 'false',
@fast_multicol_updateproc = 'true', @check_permissions = 0
```

¹⁰ El Anexo D muestra la definición de todas las publicaciones a crear

¹¹ Archivos que incluyen el esquema y los datos, las restricciones, las propiedades extendidas, los índices, los desencadenadores y las tablas del sistema necesarios para la duplicación. La instantánea inicial se transfiere a los suscriptores cuando se implementa la duplicación.



Por otro lado, el proceso de duplicación requiere de cumplir de los siguientes requisitos básicos para llevar a cabo su implementación:

Cuestiones de seguridad

- Cada servidor involucrado debe estar registrado en el administrador Corporativo(Consola de SQL Server 2000).
- Se deben establecer relaciones de confianza entre servidores que se ubiquen en distintos dominios.
- La cuenta del servicio SQLServerAgent de SQL debe tener privilegios administrativos, y debe ser miembro del grupo local de administradores.
- La cuenta debe tener el derecho de usuario avanzado de inicio de sesión como un servicio.
- Se sugiere crear una cuenta individual de Windows que sea compartida por todos los Agentes de SQL Server de forma que pueda evitar problemas en referencia a la conectividad.
- En casos de utilizar cuentas alternativas, estas deben tener la capacidad de acceder a la carpeta de trabajo del servidor de distribución. Además de agregar estas cuentas al grupo local de Administradores del servidor de distribución.

Cuestiones relacionadas acerca del servidor de distribución

- Se debe instalar el servidor de distribución antes de poder instalar cualquier servidor de publicación.
- Se debe asegurar que el servidor de distribución tenga suficiente espacio en memoria. Debe contar con al menos 32 MB en el servidor y con 16MB asignados a SQL Server. En caso de dar soporte a muchos publicadores o suscriptores es conveniente incrementar la memoria.
- Se debe tener suficiente espacio en disco para la carpeta de trabajo de distribución y para la base de datos de distribución.

El resto de los pasos de la implementación: suscripción a publicaciones, generación de la instantánea inicial, aplicación de la instantánea inicial y la sincronización de datos de la duplicación deberá llevarse a cabo manualmente de acuerdo como cada Centro de aplicación vaya finalizando las fases anteriores y para mayor control y seguridad de la Coordinación de informática.

Con la implementación de la duplicación damos por concluido este trabajo esperando haya sido de su agrado e interés.



CONCLUSIONES

CONCLUSIONES

La planeación del diseño de las bases de datos relacionales en el desarrollo de un Sistema de Información, así como en nuestro caso lo es para el Sistema de Reclutamiento y Selección de Personal del Servicio de Administración Tributaria, es fundamental para el desempeño y en consecuencia el éxito de una aplicación.

El uso de técnicas de diseño lógico en bases de datos relacionales como lo son: la normalización que en términos sencillos trata de simplificar una base de datos a través de la búsqueda de la mejor estructuración; la integridad de los datos tiene finalidad es la calidad y consistencia de los datos; las técnicas de indexación que pretenden que el acceso a los datos se haga con mayor rapidez; y el uso de directrices para codificar instrucciones en Transact-SQL (la extensión del lenguaje definido en los estándares de SQL de Microsoft) en procedimientos almacenados y desencadenadores nos permiten garantizar un diseño lógico adecuado de las bases de datos relacionales lo cual trae como consecuencia el incremento de la consistencia, el acceso seguro de la información, la recuperación a tiempo de la información, así como la nula pérdida que información.

Por otro lado el uso de técnicas de diseño físico como son: los sistemas de discos RAID nos permiten distribuir la actividad de operaciones de lectura y de escritura en varios discos; el uso adecuado de los recursos que nos proporciona SQL Server como las herramientas de migración de información (BCP y BULK INSERT), las herramientas de Respaldo (BACKUP) y Recuperación (RESTORE) de información proveen un incremento en la seguridad de la información, un mejor desempeño en el rendimiento de nuestras bases de datos, así como una alta disponibilidad de la información.

Es relevante mencionar que la distribución de información (bases de datos) a nivel nacional es un factor de éxito del Sistema de Reclutamiento y Selección debido a que es posible hacer más autónomo los centros de aplicación evitando retrasos por la inaccesibilidad o la lentitud de la red.

En la mayoría de las organizaciones de la administración pública los recursos materiales se han venido decrementando en una porción considerable, lo cual trae como consecuencia la disminución en la capacitación del personal, el uso de equipo obsoleto, la carencia de herramientas y por tanto el bajo desempeño de los aplicativos. Sin embargo esta disminución en los recursos materiales este factor difícil de atacar debido a que obedece a políticas internas que por ende está fuera de nuestro alcance, no obstante fue posible contrarrestarlas haciendo un buen diseño de la base de datos y óptimo uso de las herramientas con las que se cuenta actualmente.



Por último podemos afirmar que no sirve de nada un diseño lógico y físico adecuado, si se pierde la calidad cuando se realizan nuevos requerimientos sin un orden establecido.



BIBLIOGRAFIA



REFERENCIAS BIBLIOGRÁFICAS

CAPITULO I

Sistema de Reclutamiento y Selección, Propuesta de Desarrollo
Servicio de Administración Tributaria
Servicio Fiscal de Carrera.
Secretariado Técnico de la Comisión del Servicio Fiscal de Carrera, 1999.

Anteproyecto del Servicio Profesional de Carrera en la UAEM
Contreras Camarena, José A.
Universidad Autónoma del Estado de México, 2000.

CAPITULO II

Sistema de Reclutamiento y Selección, Propuesta de Desarrollo
Servicio de Administración Tributaria
Servicio Fiscal de Carrera.
Secretariado Técnico de la Comisión del Servicio Fiscal de Carrera, 1999.

Análisis del Sistema, Sistema de Reclutamiento y Selección
Informática Anahuac, 2000
Versión 2.0

CAPITULO III

Books on Line Microsoft SQL Sever 2000
Microsoft Corporation
1996-2000

Resolución de Problemas en SQL
Houlette, Forrest
Ed. McGraw-Hall

SQL Server 2000 en 21 días
Waymire, Richard
Ed. Prentice-Hall
1ra. Edición 2001

Sistema de Reclutamiento y Selección, documentación de la Base de Datos
Diccionario de datos
Informática Anahuac, 2000
Versión 2.0



CAPITULO IV

Books on Line Microsoft SQL Sever 2000

Microsoft Corporation
1996-2000

SQL Server 2000 en 21 días

Waymire, Richard
Ed. Prentice-Hall
1ra. Edición 2001

Implementing Replication Using Microsoft SQL Server 2000

Microsoft Corporation Training & Certification
2002

REFERENCIAS DE DIRECCIONES ELECTRÓNICAS EN INTERNET

www.itlp.edu.mx/publica/tutoriales/basedat2/hcuatro4_1.htm

www.itlp.edu.mx/publica/tutoriales/basedat1/temas4.htm

http://www.microsoft.com/latam/technet/articulos/sql/operation_guide/art03/default.asp