



CIB-ESPOL

1  
005.74  
TAP  
F. 2

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad Y Computación

*"Sistema Optimizador de Base de Datos"*

TESIS DE GRADO

Previa la obtención del Título de:

**INGENIERO EN COMPUTACIÓN ESPECIALIZACIÓN**

**SISTEMAS TECNOLÓGICOS**

**INGENIERO EN COMPUTACIÓN ESPECIALIZACIÓN**

**SISTEMAS DE INFORMACIÓN**

Presentado por:

Alex Paúl Tapia Chichande

Luis Miguel Escobar Larenas

Guayaquil – Ecuador

2005



CIB-ESPOL



CIB-ESPOL

# AGRADECIMIENTO

Agradecemos a DIOS quien con su infinita bondad, misericordia y amor nos ha permitido culminar esta tesis de graduación.

A la ESPOL en cuyas aulas se fue forjando nuestro conocimiento diario para formarnos como profesionales de bien.

A nuestro Director de Tesis, Ing Fabricio Echeverría quien con su ayuda incondicional, comprensión, ejemplo de superación y exigencia diaria supo como transmitirnos un gran legado intelectual, cualidades que lo han hecho un guía profesional en nuestras vidas.

A nuestros maestros por los conocimientos otorgados y a nuestros familiares y amigos por el apoyo brindado.

# DEDICATORIA



**CIB-ESPOL**

A Dios por haberme permitido culminar los estudios, con su bendición y protección en todo este tiempo de preparación.

A mi esposa por su incondicional apoyo, y confianza demostrado durante todos estos años, a mi hija por la inspiración y motivación que me entrega cada día, a mis padres, tíos, suegros y familiares que han sido un pilar fundamental en mi vida y por el respaldo brindado durante el transcurso de estos años para la consecución de esta obra.

Alex Paúl Tapia Chichande.

# DEDICATORIA



**CIB-ESPOL**

Ahora puedo comprender las palabras de mi padre “Principio y fin de mi jornada”, haciendo referencia a la razón su vida.

Tomando su incomparable ejemplo dedico esta obra a mis seres amados empezando por la razón de mi existencia mi incondicional esposa y mi inefable hijo, quienes me alientan día a día.

A mis padres por aquella sacrificada abnegación y guía entregada a lo largo de todos los días de mi vida.

A mis hermanos por ese espíritu de superación indomable y esa fe expresada en amor en cada instante de mi ser.

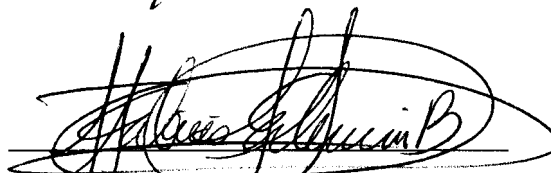
Luis Miguel Escobar Larenas

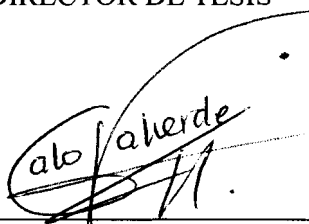
# TRIBUNAL DE GRADUACIÓN

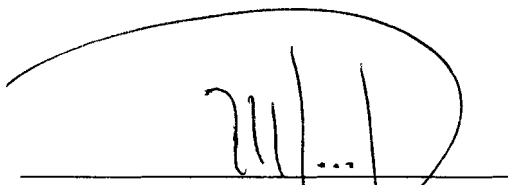
  
MSC. Miguel Yapur Auad  
SUBDECANO FIEC  
PRESIDENTE DEL TRIBUNAL



**CIB-ESPOL**

  
Ing. Fabricio Echeverría Briones  
DIRECTOR DE TESIS

  
Ing. Galo Valverde Landívar  
VOCAL PRINCIPAL

  
Ing. Marcelo Loor Romero  
VOCAL PRINCIPAL

# DECLARACIÓN EXPRESA

“La responsabilidad del contenido de esta Tesis de Grado, nos corresponden exclusivamente y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL”

(Reglamento de Graduación de la ESPOL)



**CIB-ESPOL**

Luis Escobar Larenas

Alex Tapia Chichande

# RESUMEN

Ante el inminente uso de las bases de datos, su constante evolución, integración y el potencial económico que esta representa por el manejo de datos; y considerando el arduo trabajo que realizan los DBA en sus diferentes labores; se implementó un CASE denominado SYOPBASE (Sistema Optimizador de Base de Datos), el cual permitirá realizar un enlace entre el divorcio producido por el modelo lógico y el modelo físico del almacenamiento de los datos, contemplando sólo la optimización basada en la ubicación física de las tablas de acuerdo a las características físicas de los dispositivos de almacenamiento

•

Para la realización de la presente tesis en primera instancia se consideró un estudio la naturaleza de las tablas contenidas en las diferentes bases de datos, para posteriormente relacionarlas con los discos duros a utilizar dependiendo de las características propias de los mismos o de los tipos de RAID usados.

La siguiente fase fue implementar un Algoritmo Genético:

Algoritmo de búsqueda basado en la mecánica de la selección natural y de la genética natural. Combina la supervivencia del más apto entre estructuras de secuencias con un intercambio de información estructurado, aunque aleatorizado, para constituir así un algoritmo de búsqueda que tenga algo de las genialidades de las búsquedas humanas (Goldberg, 1989)

El cual se utilizó para determinar la mejor ubicación de cada una de las tablas en un disco específico, optimizando de esta manera aquel trabajo que hasta la actualidad se realizaba empíricamente, sin ninguna base científica sino basado únicamente en la experiencia propia de la persona encargada de la base de datos.



**CIB-ESPOL**



# ÍNDICE GENERAL

INDICE GENERAL .....	I
INDICE DE ABREVIATURAS .....	II
INDICE DE FIGURAS .....	III
INDICE DE ANEXOS .....	IV
<b>CAPITULO 1 INTRODUCCION .....</b>	<b>1</b>
1.1. DESCRIPCIÓN DEL PROBLEMA .....	1
1.2. AMBIENTES DONDE SE PRODUCE EL PROBLEMA .....	4
1.3. SOLUCIÓN PROPUESTA Y SU FACTIBILIDAD .....	6
1.4. OBJETIVOS DE LA SOLUCIÓN .....	7
<b>CAPITULO 2 FUNDAMENTOS TEORICOS .....</b>	<b>8</b>
2.1. GENERALIDADES DE LOS SISTEMAS OPERATIVOS .....	8
2.2. GENERALIDADES DE LAS BASES DE DATOS .....	16
2.3. CARACTERÍSTICAS DEL MODELO LÓGICO .....	20
2.4. CARACTERÍSTICAS DEL MODELO FÍSICO .....	23
2.5. CARACTERÍSTICAS DE LOS DISPOSITIVOS FÍSICOS DE ALMACENAMIENTO .....	25
2.6. NATURALEZA DE LAS TABLAS DE LAS BASES DE DATOS .....	26
<b>CAPITULO 3 DISEÑO DEL ALGORITMO GENETICO COMO SOLUCION DEL PROBLEMA .....</b>	<b>34</b>
3.1. DISEÑO DEL ALGORITMO GENÉTICO .....	34
3.2. JUSTIFICATIVO PARA EL USO DEL ALGORITMO GENÉTICO .....	43
<b>CAPITULO 4 BENCHMARK.....</b>	<b>45</b>
4.1. JUSTIFICACIÓN .....	45
4.2. SERVIDOR IBM xSERIES 235 .....	46
4.3. EQUIPO PARA ALMACENAMIENTO MASIVO (SAN) STORAGETECH.....	47
4.4. SERVIDOR INTEL SBT2 (1) .....	48
4.5. SERVIDOR INTEL SBT2 (2) .....	49
4.6. SERVIDOR IBM NETFINITY 3500 .....	50
4.7. SERVIDOR HP PROLIANT DL 380 .....	50



**CIB-ESPOL**

4.8.	TIPOS DE PRUEBAS .....	51
4.8.1.	CONFIGURACIÓN 1 .....	53
4.8.2.	CONFIGURACIÓN 2 .....	54
4.8.3.	CONFIGURACIÓN 3 .....	55
4.9.	COMPARACIÓN DE RESULTADOS .....	56
<b>CAPITULO 5 IMPACTO FINANCIERO .....</b>		<b>60</b>
5.1.	BENEFICIOS CUALITATIVOS.....	60
5.2.	BENEFICIOS CUANTITATIVOS.....	61
5.3.	RECURSOS Y COSTOS .....	62
5.3.1	BENEFICIOS CUALITATIVOS.....	62
5.3.2	RECURSOS Y COSTOS.....	63
5.3.3	RECURSOS ADICIONALES REQUERIDOS PARA ADMINISTRAR EL SISTEMA..	64
5.3.4	COSTO DEL SOFTWARE .....	64
5.4.	ROI.....	65
<b>CONCLUSIONES Y RECOMANDACIONES.....</b>		<b>66</b>
<b>BIBLIOGRAFIA.....</b>		<b>68</b>



**CIB-ESPOL**

# ÍNDICE DE ABREVIATURAS

Abreviatura	Ingles	Español
DBA	Database administrator	Administrador de base de datos
DB / BDD	Database	Base de datos
SYOPBASE		Sistema optimizador de base de datos
DBMS / SGBD	Database Management System	Sistema gestor de bases de datos
RAID	Redundant Array of Independent Disks	Arreglo redundante de discos independientes
FMS	Fortran Monitor System	Sistema monitor de Fortran (Sistema Operativo)
IBSYS / SOS	SHARE Operating System	Sistema operativo de IBM
IBM	International Business Machines Corporation	
OS / SO	Operating System	Sistema Operativo
MS DOS	Microsoft Disk Operating System	Sistema operativo de Microsoft perteneciente a la familia DOS.
THE	Technische Hogeschool Eindhoven	Universidad Técnica de Eindhoven
E/R	Entity-Relationship	Entidad relación
DED	Data Element Definition	Diagrama estructura de datos
DDL	Data Definition Language	Lenguaje de definición de datos
I/O	Input / output	Entrada / salida
CASE	Computer Aided Software Engineering	Ingeniería de Software Asistida por Ordenador
GIS	Geographic Information System	Sistemas de información Geográfica
ANSI	American National Standards Institute	Instituto Nacional de Estándares Americanos
SPARC	Scalable Processor ARChitecture	Arquitectura escalable para procesador
SAN	Storage Area Network	Red de área de almacenamiento
DINFOR		Dirección de Informática de la Armada
DIGPER		Dirección General del Personal
DIGPERGYE		Base de datos de personal de la DIGPER Guayaquil



**CIB-ESPOL**

# ÍNDICE DE FIGURAS

Figura 1.1 Niveles de arquitectura de bases de datos .....	5
Figura 2.1 Sistema operativo de estructura monolítica.....	10
Figura 2.2 Sistema operativo de estructura jerárquica.....	11
Figura 2.3 Sistema operativo de estructura anillos concéntricos. ....	12
Figura 2.4 estructura de máquina virtual.....	13
Figura 2.5 Estructura de datos jerárquica .....	18
Figura 3.1 Diseño de la base de datos.....	43



**CIB-ESPOL**

# ÍNDICE DE TABLAS

Tabla 2.1 Características principales a considerar.....	24
Tabla 2.2 Tabla de resumen de análisis de las bases de datos .....	31
Tabla 2.3 Matriz relación Naturaleza de tabla vs. características de dispositivo.....	32
Tabla 3.1 Matriz disco .....	35
Tabla 3.2. Matriz tabla .....	35
Tabla 3.3. Matriz pesos tablas vs. disco .....	36
Tabla 3.4. Matriz tabla vs. disco convertida.....	37
Tabla 3.5. Matriz tabla vs. disco depurada.....	38
Tabla 3.6. Matriz sumatoria tabla vs. disco.....	39
Tabla 3.7. Matriz de tablas pares .....	39
Tabla 4.1 Servidor IBM x Series 235.....	46
Tabla 4.2 Equipo SAN STORAGE TECH .....	47
Tabla 4.3 INTEL SBT2 (1) .....	49
Tabla 4.4 INTEL SBT2 (2) .....	50
Tabla 4.5 Tipos de pruebas DIGPERGYE .....	52
Tabla 4.6 Características Base Sueldos.....	53
Tabla 4.7 Configuración No. 1 .....	53
Tabla 4.8 Configuración No. 2 .....	54
Tabla 4.9 Configuración No. 3 .....	55
Tabla 4.10 Comparación de resultados.....	56
Tabla 4.11 Resumen de resultados .....	57
Tabla 5.1 Recursos y costos .....	62
Tabla 5.2 Recursos utilizados en implementación .....	63
Tabla 5.3 Recursos mínimos necesarios.....	63
Tabla 5.4 Recursos adicionales .....	64
Tabla 5.5 Costo del CASE .....	64

# ÍNDICE DE ANEXOS

<u>ANEXO A</u>	(No. registros y columnas DIGPERGYE)
<u>ANEXO B</u>	(Promedios escritura DIGPERGYE)
<u>ANEXO C</u>	(Promedios lectura DIGPERGYE)
<u>ANEXO D</u>	(No. registros y columnas SUELDOS)
<u>ANEXO E</u>	(Promedios escritura SUELDOS)
<u>ANEXO F</u>	(Promedios lectura SUELDOS)
<u>ANEXO G</u>	(Codigo de case implementado)
<u>ANEXO H</u>	(Matriz posibles cruzamientos de tablas)
<u>ANEXO J</u>	(Cotizacion empresa MAGNANI)
<u>ANEXO K</u>	(Cotizacion Empresa TECNOPLUS)
<u>ANEXO L</u>	(Cotizacion Empresa MC System)
<u>ANEXO M</u>	(Cotizacion Empresa ORCOIN S.A.)
<u>ANEXO N</u>	(Cotizacion Empresa TECMOCOMP)
<u>ANEXO O</u>	(Cotizacion FACTURA MC SYSTEM)
<u>ANEXO P</u>	(Trigger de la tabla PERSONA)
<u>ANEXO Z</u>	(Manual de Usuario)



**CIB-ESPOL**

# CAPITULO 1 INTRODUCCION



**CIB-ESPOL**

## *1.1. Descripción del problema*

Como es bien conocido las bases de datos "... no son simplemente un conjunto de archivos, en vez de ellos, una base de datos es una fuente central de datos que está pensada para que sea compartida por muchos usuarios con una diversidad de aplicaciones."<sup>(1)</sup> Cuyos datos se encuentran organizados y almacenados en diferentes dispositivos, la cual puede ser manipulada por medio de programas que nos permiten un acceso directo a los mismos, para de esta forma obtener: la información requerida en mucho menos tiempo, manejo de mayor cantidad de datos, manejo de varios usuarios simultáneamente (concurrentes), la independencia de su uso, coherencia de resultados y el hecho de evitar la redundancia entre muchas otras virtudes.

---

<sup>(1)</sup> Kendal & Kendal, ANALISIS Y DISEÑO DE SISTEMAS, tercera edición, México, 1997, Pág. 588.

Es así como surgen las bases de datos a mediados de los años sesenta sin embargo no fue sino hasta 1970 en que Codd<sup>(2)</sup> propone el modelo relacional en donde los datos se estructuran lógicamente formando relaciones entre las tablas que lo componen, su objetivo primordial era el de mantener la independencia de su estructura lógica con su modo de almacenamiento. Es el nacimiento de los sistemas de gestión de datos conocidos como SGBD o sus siglas en inglés DBMS (DataBase Management System).

Muchos autores consideran la existencia de dos niveles en las bases de datos para el tratamiento de las entidades o de los objetos los cuales son: nivel lógico y el nivel físico; el primero (conocido muchas veces como conceptual) se refiere a las características de identificación de las entidades junto con su descripción y el lugar que ocupa en la organización; en cambio el nivel físico se refiere a la forma, modo, tipo, lugar de almacenamiento o repositorio físico en donde se almacenan los datos.

De la misma manera existen otros autores que consideran a un tercer nivel conocido como el nombre de Nivel Visión, el cual es parte del conceptual, el cual es el

---

<sup>(2)</sup> Edgar F. "Ted" Codd (Agosto 23, 1923 – abril 18, 2003), científico británico que contribuyó con la teoría relacional de bases de datos.



encargado de dividir a la base de datos en pequeñas parcelas útiles para los usuarios, un ejemplo clásico ha mencionar es el hecho de que un empleado que no debe tener acceso al sueldo de sus compañeros o superiores; este hecho esta relacionado con un Esquema de Visión.

Como se hace mención en los párrafos anteriores generalmente el nivel físico es facilitado por los SGBD, lo que quiere decir que el usuario esta en capacidad de decidir el lugar en donde se realizará la operación de almacenamiento; sin embargo esta operación se realiza sobre la base de la experiencia propia del DBA, lo cual ocasiona una operación poco eficiente con los recursos propios, e ineficaz puesto que se basa únicamente en las observaciones anteriores realizada por el mismo. El problema se ve agravado por el hecho que existe la posibilidad que el DBA no tenga la experiencia suficiente para determinar el lugar mas adecuado para el almacenamiento.

En este punto se considera la existencia de un divorcio entre el conocimiento de los datos a ser almacenados (con las características propias de cada uno de ellos) y el lugar establecido como repositorio de datos en donde serán almacenados. Para hacer esta elección de lugar no se ha tomado en cuenta ninguna de las características propias ni de las bases, ni de los datos, ni del software por una parte y por otra no se



**CIB-ESPOL**

ha considerado características de los discos, tipo de RAID a ser utilizado, velocidades de operación, almacenamiento o algún tipo de degradación.

Es el momento en el cual la interrogante para resolver este problema mediante la creación de un CASE capaz de guiar al DBA, a realizar un trabajo eficaz y eficiente en la administración de las mismas considerando factores que hasta el momento se los había considerado empíricamente o tal vez no se los había considerado.

### *1.2. Ambientes donde se produce el problema*

La reunión de ciertas condiciones tales como excesiva cantidad de tablas, demasiados datos, utilización de diversos tipos de arreglos de discos para el almacenamiento, tipo de operación a realizar con las mencionadas tablas y datos, crecimiento indiscriminado de datos a ser guardados, hacen un ambiente propicio y adecuado para dificultar la toma de decisiones. Por lo cual pensamos que los ambientes más

propensos a este problema son los sistemas bancarios, sistemas de simulación, sistemas creadores de modelos (tales como el GIS<sup>(3)</sup>) entre otros.

Específicamente el problema surge en los dos niveles inferiores, entre el nivel interno y el nivel conceptual, adicionalmente se considera como un problema netamente del DBA, pues este será la persona encargada de tomar decisiones sobre los posibles lugares de almacenamiento de las diferentes tablas de las bases de datos.

Las bases de datos en general tienen una arquitectura de tres niveles (definida por el grupo ANSI<sup>(4)</sup>/SPARC<sup>(5)</sup>), la cual puede apreciarse en la figura 1.1:

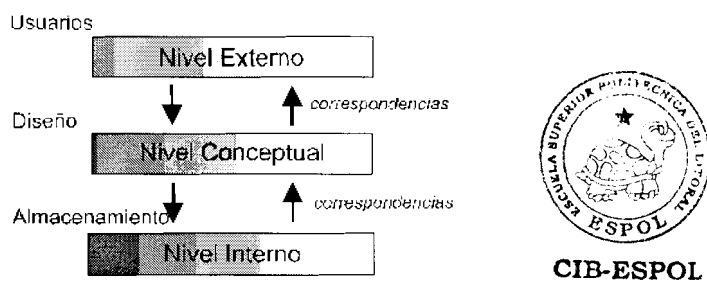


Figura 1.1 Niveles de arquitectura de bases de datos

<sup>(3)</sup> GIS: (Geographical Information System) Sistema de Información Geográfica.- Es una metodología para adquirir, almacenar, gestionar, editar y mostrar datos relacionados con mapas y planos geográficos.

<sup>(4)</sup> ANSI: (American National Standard's Institute) Organización americana para definir estándares.

<sup>(5)</sup> SPARC: Estandar RISC desarrollado por Sun Microsystems.

**CIB-ESPOL**

### *1.3. Solución propuesta y su factibilidad*

Una solución para el problema en mención y basados en la aplicación de un algoritmo genético, el cual toma la naturaleza de las tablas, las propiedades de los arreglos, las características técnicas de los discos, para de esta manera indicar los repositorios adecuados para los datos por medio de una selección natural basada en genética.

Este algoritmo esta en capacidad de crear un conjunto de soluciones óptimas las cuales se almacenan en una base de datos que se considera como el conjunto de soluciones universo, de este universo y aplicando nuevamente el algoritmo se obtiene una población de soluciones las cuales son por selección natural mejores respecto a las anteriores y así sucesivamente, de esta manera va evolucionando cada vez más, se logra conseguir individuos óptimos, los cuales han sido el producto de la selección del universo de las mejores poblaciones y los mejores individuos.

El hecho de poder abstraer todas las características mencionadas y luego plasmarlas en código ha permitido optimizar las acciones realizadas por los DBA en cuanto se refiere a la toma de decisión del lugar más adecuado para su almacenamiento. Adicionalmente se puede mencionar que en el mercado actual existen pocos

**CIB-ESPOL**

programas capaces de realizar las acciones antes mencionadas para automatizar estas funciones tales como el RSS<sup>(6)</sup> (Research Storage System), el RSI<sup>(7)</sup> (Research Storage Interface).

Cabe mencionar que el sistema implementado consideró las velocidades estándares de los discos (no las optimizadas por el RAID), de igual manera no se consideró el almacenamiento de los índices de las bases de datos.

#### *1.4. Objetivos de la solución*

1. Optimizar la capacidad de almacenamiento de los discos utilizados en los arreglos de discos.
2. Optimizar la velocidad de los procesos realizados en las diferentes transacciones ordinarias de las bases de datos.
3. Disponer de varias alternativas óptimas para elección de almacenamiento.
4. Crear modelos basados en una base de datos.

---

<sup>(6)</sup> RSS es el subsistema de almacenamiento de system R, este es el responsable de mantener el almacenamiento físico de las relaciones, de los caminos de acceso a ellas, bloqueos (en un ambiente multiusuario), y de las capacidades de registro de transacciones y recuperación.

<sup>(7)</sup> RSI (Research Storage Interface) es la interfaz orientada a la tupla que system R presenta al usuario final.

# CAPITULO 2 FUNDAMENTOS

## TEORICOS

### *2.1.Generalidades de los Sistemas Operativos*

A partir de los años 50 surge la necesidad imperiosa del uso de la computadora, y sus primeras versiones eran simplemente tableros con capacidad de ser conectados (primera generación), posteriormente apareció la segunda generación implementando el trabajo en lotes (sistemas como FMS<sup>(8)</sup> o el IBSYS<sup>(9)</sup>). Con el surgimiento de las computadoras de la tercera generación aparecen los sistemas operativos OS/360 de IBM<sup>(10)</sup>, es importante destacar que en este período también aparece el concepto de

---

<sup>(8)</sup> FMS (Fortran Monitor System) fue el sistema concebido específicamente para trabajar en el lenguaje de programación FORTRAN y así algunos llevar al máximo su potencial.

<sup>(9)</sup> IBSYS es un sistema operativo creado por IBM en los años 60 para el 7094.

<sup>(10)</sup> IBM (International Bussinnes Machines) es una de las compañías más grandes de computadoras en el mundo. IBM empezó en 1911 como un productor de máquinas para tabular fichas perforadas. En 1953, introdujo su primera computadora, el 701. Durante el 60's y 70's, IBM vino a dominar el campo nuevo de central y las minicomputadoras. En 1981, IBM lanzó su primera computadora personal.

multiprogramación aunque en general se los considera de lote. El avance de la electrónica a gran escala provocó que en la cuarta generación se hable de computadoras personales y estaciones de trabajo con interfases amigables por lo que se hacen populares el MS-DOS<sup>(11)</sup> y UNIX<sup>(12)</sup> en las mencionadas computadoras.

Se da un brinco gigantesco cuando en los años 80 comienza el auge de las redes de computadoras y por ende la necesidad de sistemas operativos capaces de administrar redes y sistemas distribuidos. El crecimiento indiscriminado en todo el mundo de la Internet hace posible el acceso a personas e instituciones a toda la información disponible en sus computadoras personales, claro está que para la época existía una diversidad de sistemas operativos por lo cual se busca una solución para hacer trabajar a estos sistemas operativos diferentes. Sin embargo no fue sino hasta la década de los 90, época en la cual se crean las aplicaciones para ser ejecutadas en una plataforma y los resultados pueden ser observados en otra diferente por lo que se verifica que el nivel de interacción cada vez se hace más profundo.

---

<sup>(11)</sup> MS DOS (MicroSoft-Disk Operating System) es el sistema operativo creado para las Pc de Microsoft, cuyo funcionamiento es muy parecido a la versión del IBM para sus computadoras personales, la peculiaridad de llamarlos por DOS se debe a que anteriormente los sistemas operativos obligatoriamente arrancaban desde el disco.

<sup>(12)</sup> Unix: es uno de los más populares sistemas operativos multiusuario y multitarea desarrollado por Labs Bell en los años 70 con las consideraciones que sea liviano, flexible, barato y de uso exclusivo de los desarrolladores. Fue el primer SO desarrollado en C, lo que significa que puede ser instalado prácticamente en cualquier Pc.



Algunos autores consideran que los sistemas operativos se basan en tres clasificaciones las cuales se mencionan a continuación: sistemas operativos según su estructura, sistemas operativos de acuerdo a los servicios que ofrecen y finalmente sistemas operativos de servicios para los usuarios.

Los sistemas operativos de acuerdo a su estructura cumplen con dos requisitos que son: fácil de usar y de aprender; pero en cuanto a la parte técnica considera: mantenimiento, forma de operación, restricciones de uso, eficiencia, tolerancia a errores y flexibilidad. Tomando en cuenta su estructura podemos mencionar:

- a. Estructura monolítica.- cuentan con un solo programa entrelazados por un conjunto de rutinas, de tal manera que se puedan llamar entre sí. Su construcción final son módulos compilados por separado.

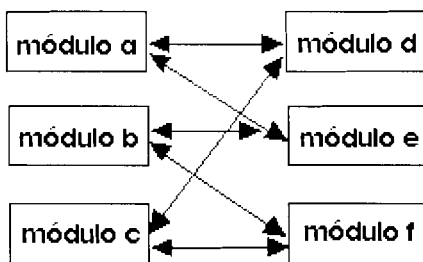


Figura 2.1 Sistema operativo de estructura monolítica.

- b. Estructura jerárquica.- se dividió al sistema operativo en pequeñas partes las cuales tenían una interfase con el resto de elementos, adicionalmente su



organización era por niveles. Esa fue la razón para denominarla estructura jerárquica o de niveles; el primero creado fue denominado THE<sup>(13)</sup> (Technische Hogeschool Eindhoven), perteneciente a Dijkstra<sup>(14)</sup>. Otros autores los consideran de multicapas. Su estructura es la base para la actual estructura de muchos sistemas operativos.

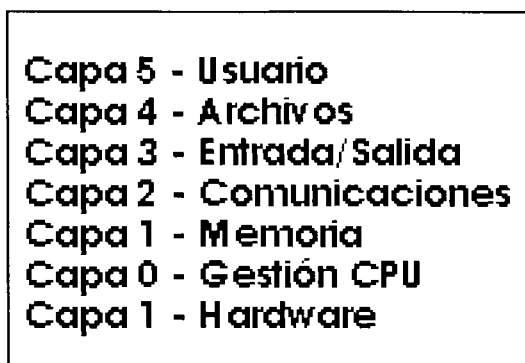


Figura 2.2 Sistema operativo de estructura jerárquica.

Otra forma de denominación es la de estructura de anillos concéntricos o rings.

---

<sup>(13)</sup> THE (Technische Universiteit Eindhoven o TU/E) es una universidad técnica localizada en Eindhoven, (Holanda). Fue fundada el 15 de junio de 1956 por el gobierno holandés. Era la segunda institución de su clase en los países bajos, sólo ha sido precedida por la Universidad Tecnológica de Delft. Originalmente, se llamó el Eindhoven (EL) de Technische Hogeschool.

<sup>(14)</sup> Edsger Wybe Dijkstra fue uno de los miembros más influyentes de la generación fundadora de la ciencia informática. Sus contribuciones científicas más importantes son los lenguajes de programación de diseño del algoritmo, sistemas operativos de diseño distribuidos procesando el diseño formal de la especificación y comprobación de argumentos matemáticos además, Dijkstra se interesó intensamente en la enseñanza, y en las relaciones lo académico, la ciencia y la industria de software. Durante sus cuarenta años como científico, logro posicionarse como uno de los mejores en el campo de la informática, así mismo en la industria y por sus contribuciones le otorgaron muchos premios inclusive el honor más alto de la ciencia, el Premio de ACM Turing.

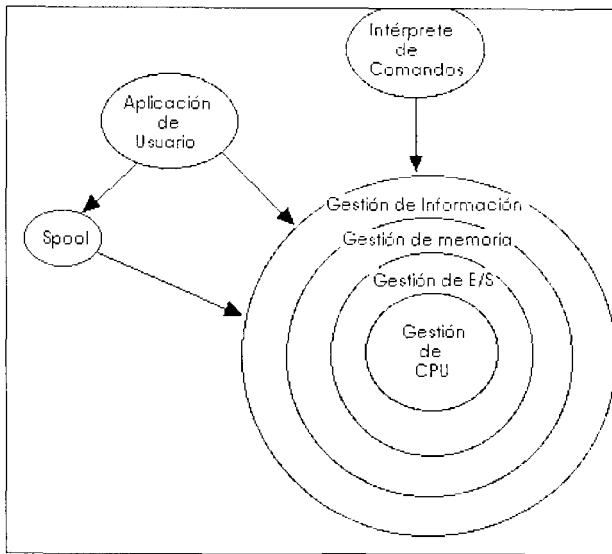


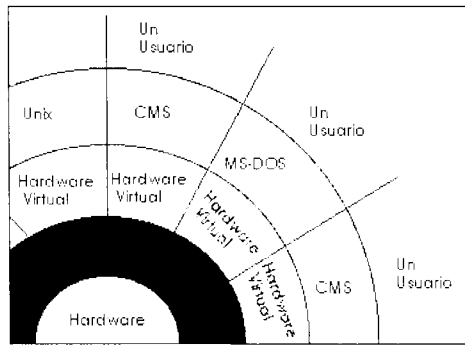
Figura 2.3 Sistema operativo de estructura anillos concéntricos.

En cada uno de los anillos se pueden apreciar puertas de acceso o trampas (traps), que sirven de ingreso para las partes inferiores esta es la manera para proteger las partes internas de accesos no deseados.

c. Estructura de maquina virtual.- en este tipo de sistema operativo cada proceso presenta una interfase la cual es idéntica a la maquina real, en este tipo de estructura se separan la multiprogramación y la máquina extendida, la idea principal es la de dar la sensación de disponer de varias máquinas diferentes las cuales pueden ejecutar sistemas operativos diferentes como se puede apreciar en la figura 2.4.



**CIB-ESPOL**



**CIB-ESPOL**

Figura 2.4 estructura de máquina virtual

d. Estructura cliente servidor.- son de propósito general y sirve para toda clase de aplicaciones; puede ser ejecutado en la mayoría de las computadoras. Lo mas destacado es que los procesos pueden ser tanto en clientes como en servidores.

De la misma manera podemos diferenciar a los sistemas operativos por el tipo de servicios para lo cual los hemos dividido de la siguiente manera:

- División por el número de usuarios
  - a. Monousuarios.- son aquellos SO que soportan únicamente a un usuario sin tomar en cuenta al número de procesadores que disponga la computadora o número de tareas o procesos que tenga que realizar en un instante de tiempo.

b. Multiusuarios.- estos sistemas operativos permiten dar servicio a más de un usuario a la vez, por medio de accesos remotos o la utilización de varias terminales conectadas a la computadora mediante el uso de una red.

- División por el número de tareas

a. Monotarea.- la característica principal de estos SO es que permiten la realización de una tarea a la vez por usuario, es importante mencionar que se permite la combinación de multiusuario monotarea, es decir muchos usuarios realizando una sola tarea.

b. Multitarea.- permite al usuario realizar muchas labores al mismo tiempo, como ejemplo se puede editar el código fuente mientras se compila otro programa y al mismo tiempo recibir un mensaje electrónico, lo cual aumenta notoriamente su productividad.



**CIB-ESPOL**

- División por el número de procesos

a. Uniproceso.- este SO es capaz de manejar únicamente un procesador en la computadora.

b. Multiproceso.- se refiere al trabajo específico de varios procesadores del sistema de tal manera que se puede distribuir la carga del proceso; se los considera de dos formas de trabajo: simétrica si todos los procesos o parte de ellos conocidos como (threads), son ejecutados indistintamente en

cualesquiera de los procesadores disponibles por medio de un algoritmo de mejor distribución y de manera asimétrica en el cual uno de los procesadores será considerado como principal y los demás como esclavos para la ejecución de los mencionados procesos.

La siguiente división de los sistemas operativos es dependiente de los servicios que ofrecen (es decir es una visión desde el punto de vista del usuario) la que se puede apreciar mediante la siguiente división:

- a. SO de red.- son aquellos que nos permiten interactuar con los SO de otras computadoras por algún medio de transmisión con el afán de realizar un sin número de actividades entre sí, tales como transferencia de archivos, accesos remotos, instalar utilitarios, tener acceso a bases de datos etc., es decir lo más importante es disponer del recurso y poderlo compartir.
- b. SO distribuidos.- en realidad no se ha desarrollado un SO distribuido a plenitud pues la idea principal es la de abarcar todos los servicios de red integrando los recursos en una sola máquina virtual a la cual se puede acceder de manera transparente, necesitando conocer únicamente el nombre y de esta manera los recursos como si fuesen locales. El grave problema resulta de la complejidad de distribuir los procesos en varias unidades de procesamiento, reintegrar los resultados al mismo tiempo de resolver

problemas de concurrencia y paralelismo. Las ventajas principales que se puede mencionar son: velocidad, capacidad de crecimiento, confiabilidad y costos. En cambio las desventajas se basan en el aspecto de concurrencia y paralelismo. Puesto que debido a la concurrencia se debe implantar mecanismos para tratar de evitar la competencia entre procesos, evitar una postergación indefinida, evitar condiciones circulares y el hecho de ocupar recursos en espera de otro. En realidad todos estos problemas presentan la mayoría de los sistemas operativos multiusuarios y multitareas; claro está que para los sistemas operativos distribuidos el problema es aún más complejo, razón por la que se necesitarán algoritmos mejor desarrollados. Debido al paralelismo el solo hecho de encontrar las secciones de algún código paralelizable es un trabajo intenso, sin embargo es necesario dividir a un proceso en varios sub-procesos y redirigirlos a diferentes unidades de procesamiento para su distribución.



**CIB-ESPOL**

## *2.2 Generalidades de las bases de datos*

Ampliando aquellos conceptos que se pudieron apreciar en la Descripción del problema perteneciente al Capítulo I, en donde se hace mención brevemente a las

bases de datos, consideramos que es importante mencionar que las bases de datos deben cumplir con algunos objetivos tales como::

1. Todos los usuarios compartan la base.
2. Controlar la veracidad y consistencia de los datos.
3. Disponibilidad de los datos.
4. Considerar la evolución y crecimiento de la base de datos.
5. Todo el proceso sea transparente para el usuario.

Un punto extremadamente importante es como guardar los datos, es así como se considera que existen dos maneras para realizar esta labor, la primera es crear archivos independientes e individuales y la segunda y más desarrollada en la actualidad es la creación de una base de datos que pueda ser compartida por una infinidad de usuarios y una gran cantidad de aplicaciones, lo cual resulta muy útil pues se facilita la captura de datos, almacenamiento y actualización por una sola vez, para lo cual es necesario conocer tres conceptos claramente: el mundo real, los datos y los metadatos.

El mundo real representa todas aquellas entidades, objetos o eventos de los cuales deseamos recolectar y almacenar datos relevantes. sus atributos son las características de estas entidades. En cambio los datos son los valores que pueden ser organizados en

registros los cuales serán accedidos por medio de claves o llaves. Finalmente los metadatos describen a los datos de acuerdo a las restricciones impuestas a estos.

De esta manera consideramos que las bases de datos pueden tener tres tipos de estructuras que son:

1. Estructuras de datos jerárquicos- conocidas también como árboles, implican que una entidad no puede tener más de una entidad que la posea. Es decir se la considera como una estructura compuesta de muchas asociaciones 1:M o 1:1; otros tipos de asociaciones tales como M: 1 o M:N no son permitidas.

Gráficamente se la puede apreciar de la siguiente manera:

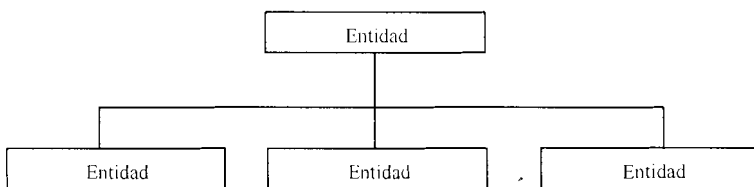


Figura 2.5 Estructura de datos jerárquica

Algunos autores la consideran como ramificaciones donde una entidad puede tener ramificaciones subordinadas semejantes a las ramas de un árbol

2. Estructura de datos en red.- este tipo de estructura permite tener entidades con cualquier cantidad de subordinados o superiores las cuales se encuentran vinculadas por medio de enlaces que representan a conceptos de datos comunes a ambas entidades conectadas. Es una estructura mucho más compleja que su predecesora, su principal ventaja es que el mismo dato



puede ser utilizado por varias entidades, la desventaja por el contrario es que fácilmente podría existir redundancia de datos, en la figura a continuación hemos representado al mundo real como registros lógicos que se relacionan entre sí por medio de flechas hacia los respectivos datos comunes.

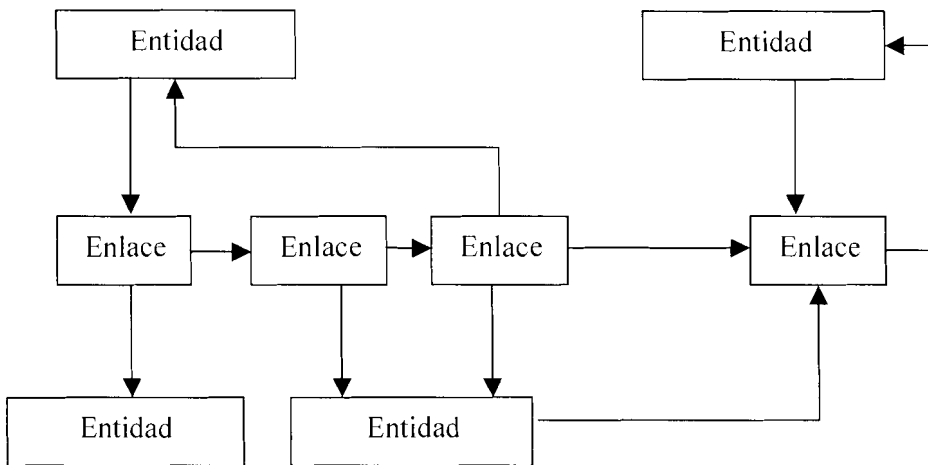


Figura 2.6 Estructura de datos en red

3. Estructura de datos relacional.- representan una o varias tablas bidimensionales (conocidas como relaciones), los renglones de las tablas contienen registros mientras que las columnas representan a sus atributos. Podemos llamar relacional a la base de datos construida por relaciones entre dos tablas o más, las cuales disponen de ventajas como eficiencia en las consultas y factibilidad de crecimiento en la base de datos.

**CIB-ESPOL**

### 2.3. Características del Modelo Lógico

Considerando que cualquier cambio en la estructura física o lógica de los datos afectaba a las aplicaciones directamente, ANSI publicó la definición de una arquitectura de tres niveles la cual se debe seguir para el diseño de la base de datos en donde hace énfasis a la independencia entre las referencias externas a los datos y la forma física de almacenamiento, considera también la organización estos niveles de la siguiente manera:

1. Nivel Externo.- este es únicamente de interacción con el usuario en donde se representa una visión parcial de los datos respecto al usuario, muestra únicamente la parte que le interesa al usuario.
2. Nivel Conceptual.- este en realidad representa el esquema lógico de los datos en donde se visualiza su estructura y sus relaciones, sin embargo no considera especificaciones físicas. En este punto se excluyen datos innecesarios y se define que alternativa de modelo se va a utilizar.
3. Nivel físico.- este debe ser transparente para el usuario y va a depender de la estructura de los datos, hardware, software, así como el modo de almacenamiento de los datos, sistemas operativos utilizados, SGBD etc....

El modelo conceptual es un modelo abstracto que pertenece al nivel externo que permite al lector tener una visión general del asunto a tratar, es decir son todos los procesos del caso en estudio pero con una visión global de la situación que permitirá obtener una imagen amplia de todo el proceso; sin pensar en las limitaciones físicas, ni lógicas, ni estructura alguna. adicionalmente no considera al tipo de base de datos, ni tampoco al SGBD. El modelo entidad relación es un modelo conceptual, (fue propuesto inicialmente por CHEN<sup>(15)</sup> para posteriormente dar lugar al modelo E/R extendido); es decir se puede expresar que el modelo conceptual toma a las relaciones establecidas entre las entidades participantes en el tema a solucionar.

Los componentes del modelo E/R<sup>(16)</sup> son los siguientes: entidades, relación, grados de relación, cardinalidad, atributos. Adicionalmente se pueden mencionar conceptos como: jerarquía, agregación y exclusividad.



**CIB-ESPOL**

Una vez establecido correctamente el modelo conceptual, se considera pertinente continuar con el modelo lógico perteneciente al nivel conceptual, denominado por

---

<sup>(15)</sup> Dr. Peter P. Chen es el científico creador del modelo entidad relación, su labor se la considera como la piedra angular de dotación lógica, su trabajo de investigación y metodologías estructuradas han hecho un impacto significativo en la mayoría de las herramientas tipo CASE.

<sup>(16)</sup> Llamado también un modelo de la entity relationship, es una representación gráfica de las entidades y sus relaciones uno al otro, típicamente utilizado en los sistemas de bases de datos. Una entidad es un grupo de datos, un objeto es el concepto genérico de los datos que se almacenan. Una relación es cómo los datos se comparten entre entidades.

algunos autores como Esquema de Base de Datos (Database Schema), y como se verificará posteriormente será el punto de partida para realizar el modelo físico. En este modelo lógico transformamos las entidades obtenidas y relaciones aceptadas en el modelo anterior a tablas, para lo cual utilizamos las reglas de normalización.

Cada entidad del modelo conceptual será transformada en una tabla y los atributos de dicha entidad serán los atributos de dicha tabla, las relaciones de muchos a muchos se transforman en tablas en donde la clave estará formada por la clave primaria de las entidades relacionadas; en cambio las relaciones de uno a muchos se propaga la clave principal de la entidad cuya cardinalidad es 1 a la de cardinalidad M. Podemos valernos de los Diagramas de Estructura de Datos (DED).

Es importante determinar que este tipo de esquema permite relaciones entre dos entidades; bajo la salvedad de modelar relaciones de dos o más entidades se deberán redefinir todas a relaciones binarias.

Existe una correspondencia entre los diagramas E/R y los DED, permitiendo el paso de uno a otro y viceversa.



**CIB-ESPOL**

Los problemas más importantes que se pueden evidenciar son: redundancia de datos y ambigüedades entre datos; de esta manera surgen las soluciones a los mismos como las reglas de normalización, para lo cual consideramos los conceptos de: superclave, clave primaria, clave candidata; dependencia funcional, transitiva y completa.

Las principales formas de normalización son las siguientes:

1. Primera Forma Normal (1FN).
2. Segunda Forma Normal (2FN).
3. Tercera Forma Normal (3FN).
4. Forma Normal de Boyce Codd<sup>(17)</sup>
5. Cuarta Forma Normal (4FN).
6. Quinta Forma Normal (5FN).

#### *2.4. Características del Modelo Físico*

Una vez que hemos finalizado con los dos pasos anteriores el lector del presente estará en capacidad de diferenciar el modelo de las tablas (más sus derivadas de las

---

<sup>(17)</sup> Edgar F. Codd científico que contribuyó notablemente en el estudio del modelo entidad relación y a quien se le atribuye una de las formas normales de las bases de datos.

relaciones), lo que representa una metáfora de la conversión de esta tabla en un mapeo físico para lo cual se debe considerar el conocimiento de características tales como se presenta en tabla 2.1:

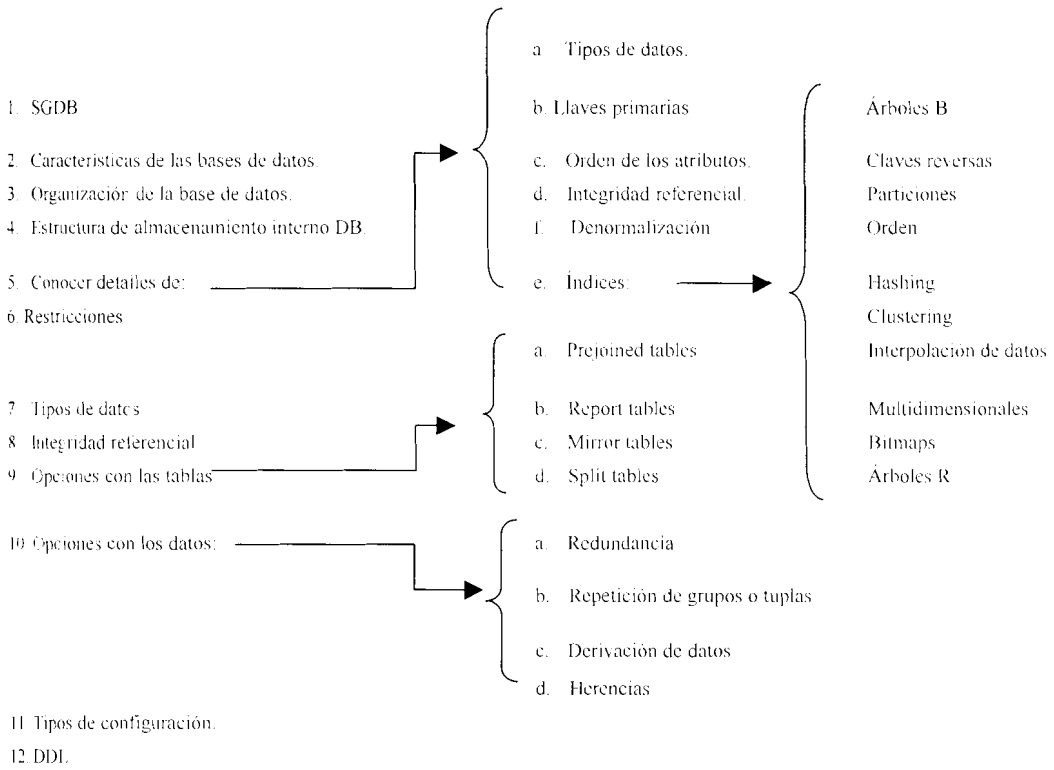


Tabla 2.1 Características principales a considerar

## *2.5. Características de los dispositivos físicos de almacenamiento*

Como es bien conocido los datos contenidos en las tablas será almacenada en discos duros los cuales disponen de ciertas características propias de cada uno de ellos, las principales características se las mencionan a continuación:

1. Capacidad de almacenamiento.- Es una característica propia del disco que limita la cantidad de información que puede ser almacenada.
2. Rendimiento.- Es la suma de factores que implican la capacidad de los discos para efectuar operaciones de I/O, es importante mencionar que los discos dependerán de dos factores importantísimos, factores mecánicos y factores electrónicos.
3. Velocidad.- Es la característica que nos indica las revoluciones por minuto del dispositivo en mención
4. Facilidad de manejo.- Es la habilidad que brinda al operador a fin de realizar las tareas de lectura, escritura, almacenamiento y borrado.
5. Costos.- Es el valor económico que implica la utilización de los mencionados dispositivos.
6. Escalabilidad.- Es la capacidad que tienen los dispositivos que le permiten aumentar en cantidades y seguir trabajando manteniendo o mejorando su rendimiento

7. Tipos de RAID.- Es el conjunto de discos redundantes e independientes usados a fin de preservar la integridad de los datos, mejorar rendimiento y mejorar la tolerancia a errores.

### *2.6 Naturaleza de las tablas de las bases de datos*

Consideramos que las tablas que componen las bases de datos, tienen propiedades entre las cuales mencionamos a:

1. Propiedad de acceso- son todas aquellas tablas a las cuales se las utiliza en las bases de datos.
2. Propiedad de lectura- son todas aquellas tablas las cuales han sido leídas para la obtención de diferentes datos.
3. Propiedad de modificación- son todas aquellas tablas las cuales han sido cambiadas o alteradas de alguna manera.
4. Propiedad de consulta- son aquellas tablas las cuales se han utilizado para realizar las diferentes consultas de los procesos realizados.



Para el trabajo en mención hemos dividido a las tablas de la base de datos de DIGPERGYE en dos cuyas características son:

1. Propiedad concurrente de escritura.- son aquellas que han sido alteradas y nuevamente grabadas y las hemos dividido en:
  - a. Altamente concurridas de escritura.- hemos considerado valores promedio de 144.59 operaciones diarias, aunque el rango va desde 100 hasta 726.13 operaciones diarias. (ver Anexo B)
  - b. Medianamente concurridas de escritura.- son aquellas que tienen interacciones entre 15 y 100 (diariamente), con un promedio de 67.65 operaciones diarias. (ver Anexo B)
  - c. Bajamente concurridas de escritura.- son aquellas que tienen interacciones entre 0 y 15 (diariamente), con un promedio de 2.28 operaciones diarias. (ver Anexo B).
2. Propiedad concurrente de lectura.- son aquellas que no han sido alteradas sino que únicamente han sido accesadas como consulta y las hemos dividido en:
  - a. Altamente concurridas de lectura.- son aquellas tablas que han sido consultadas mas de 50 veces diarias, el promedio referencial tomado es de 258.76 consultas diarias. (ver Anexo C).



CIB-ESPOL

- b. Medianamente concurridas de lectura.- son aquellas que han sido consultadas entre 15 y 50 veces al día, con un promedio de consulta de 38.91 consultas diarias. (ver Anexo C).
- c. Bajamente concurridas de lectura.- son aquellas que tienen menos de 15 consultas al día, con un promedio de 2.39 consultas al día. (ver Anexo C)

Adicionalmente se trabajó en la base de datos de SUELDOS cuyas características se mencionan a continuación:

1. Propiedad concurrente de escritura.-

- a. Altamente concurridas de escritura.- el valor promedio es 80.7 operaciones diarias, aunque el rango va desde 50 hasta 142.4 operaciones diarias. (ver Anexo E)
- b. Medianamente concurridas de escritura.- son aquellas que tienen interacciones entre 15 y 50 (diariamente), con un promedio de 25.3 operaciones diarias. (ver Anexo E)
- c. Bajamente concurridas de escritura.- son aquellas que tienen interacciones entre 0 y 15 (diariamente), con un promedio de 1.6 operaciones diarias. (ver Anexo E).

2. Propiedad concurrente de lectura.-



**CIB-ESPOL**

- a. Altamente concurridas de lectura.- son aquellas tablas que han sido consultadas mas de 15 veces diarias, el promedio referencial tomado es de 28.9 consultas diarias, (ver Anexo F).
- b. Medianamente concurridas de lectura.- son aquellas que han sido consultadas entre 5 y 15 veces al día, con un promedio de consulta de 5.9 consultas diarias. (ver Anexo F).
- c. Bajamente concurridas de lectura.- son aquellas que tienen menos 5 consultas al día, con un promedio de 1.5 consultas diarias. (ver Anexo F).

Los valores mencionados fueron tomados como parámetros gracias al siguiente procedimiento:

1. Se determinó el número de tablas de la base de datos en donde se aplicará el algoritmo genético, (para nuestro caso específico fueron: DIGPERGYE 614 tablas, sueldos 124 tablas)
2. Se determinó el número de registros y el número de columnas en cada una de las tablas mediante la aplicación de las siguientes líneas de código (los resultados se pueden apreciar en el anexo A y anexo D):

```
select TRA_PER_ID, fec_mod
from transaccion
where TRA_TABLA = '-----nombre de la tabla-----'
group by TRA_PER_ID, fec_mod
```

```
order by fec_mod
```

3. Se determinó el número de transacciones realizadas en cada una de las tablas que disponen de un trigger (disparador)<sup>(18)</sup> para lo cual se utilizó las siguientes líneas de código (el resultado se lo puede apreciar en el [anexo B](#) y [anexo E](#)), los diferentes tipos de trigger se los aprecia en el [anexo P](#):

```
select year(fec_mod), TRA_TABLA, count(*)
from transaccion
group by year(fec_mod), TRA_TABLA
order by year(fec_mod), TRA_TABLA
```

4. Posteriormente se determinó el número de consultas realizadas en cada una de las tablas para lo cual se utilizó el siguiente código (el resultado se lo puede apreciar en el [anexo C](#) y [anexo E](#)):

```
select year(FECHA), count(*)
from TRANSACCION_CONSULTA
group by year(FECHA)
order by year(FECHA)
```

---

<sup>(18)</sup> Disparador es un procedimiento especializado, que se almacenan dentro de la base de datos y es manejado por el DBMS. Este no puede ser llamado o ejecutado. El DBMS enciende automáticamente al disparador como resultado de una modificación de los datos a la tabla asociada. Los disparadores son utilizados para mantener la integridad de referencia de datos cambiando los datos en una manera sistemática.

5. La tabla de valores obtenidos se la promedio para el número de días que la base de datos ha sido utilizada arrojando los valores expuestos en el anexo B y anexo C, (para el caso de la base de datos DIGPERGYE), los valores referenciales de días de uso datan desde el 11-ENE-2000 al 04-ABR-2005 (aproximadamente 1365 días). Igual procedimiento se realizo para la base de datos de sueldos con resultados expuestos en el anexo E y anexo F, el periodo referencial es de 60 días.

El resumen de anexos de análisis se los puede apreciar en la siguiente tabla:

Tabla de resumen de Análisis			
Base de datos DIGPERGYE			
Promedios de escritura diaria:	144.5	66.7	2.28
Total tablas:	135	237	42
Promedios de lectura diaria:	258.8	38.9	2.4
Total tablas:	127	29	45
Base de datos SUELDOS			
Promedios de escritura diaria:	80.7	25.3	1.6
Total tablas:	31	17	8
Promedios de lectura diaria:	28.9	5.9	1.5
Total tablas:	24	23	21

Tabla 2.2 Tabla de resumen de análisis de las bases de datos

Tomando las características de los items 2.5 y 2.6 hemos creado la matriz relación naturaleza de tabla vs. características de dispositivo, la cual presentamos en la tabla 2.3:

Naturaleza de tabla	Característica del dispositivo		
	Rápido	Medianamente rápido	Lento
Altamente Accesada	10	5	0
Medianamente accesada	5	10	0
Poco acceso	0	5	10

Tabla 2.3 Matriz relación Naturaleza de tabla vs. características de dispositivo

En la matriz anteriormente indicada hemos calificado como optimas a las relaciones:

- Rápido vs. Altamente accesada (ideal)
- Medianamente rápido vs. medianamente accesada (ideal)
- Lento vs. poco acceso (ideal)

De la misma manera se ha calificado como regulares a las siguientes relaciones:

- Medianamente rápido vs. altamente accesada (con un disco medianamente rápido podría ser aceptable un acceso a datos elevado, aunque este no sea el ideal)
- Rápido vs. medianamente accesada (con un disco rápido podría ser aceptable un acceso mediano, aunque exista desperdicio de recursos)



**CIB-ESPOL**

- Medianamente rápido vs. poco acceso (con un disco de velocidad media podría ser aceptable tener tablas de poco acceso, aunque existe desperdicio de recursos)

De igual forma los autores del presente han calificado como relaciones no deseadas a:

- Lento vs. altamente accesada (al disponer de un disco lento la peor condición que pueda suceder es tener una tabla altamente accesada)
- Lento vs. medianamente accesada (sí el disco es lento una de las peores condiciones es que disponga de tablas medianamente accesadas)
- Rápida vs. poco accesado (si el disco en mención es rápido una de las peores condiciones sería que tenga tablas de muy poco acceso, puesto que estaríamos desperdiciando recursos)

# CAPITULO 3 DISEÑO DEL ALGORITMO GENETICO COMO SOLUCION DEL PROBLEMA

## *3.1.Diseño del algoritmo genético*

El algoritmo genético es un método repetitivo de búsqueda basado en la mecánica de la selección natural y de genética, mediante una secuencia modificada de instrucciones generadas de manera recursiva, de tal manera que “combina la supervivencia del más apto entre estructuras de secuencias con un intercambio de información estructurado, aunque aleatorio”<sup>(19)</sup>

Consideramos la creación de un algoritmo genético basada en una simple parametrización de matrices optimizando el lugar de almacenamiento en una base de datos.

---

<sup>(19)</sup> Goldberg. Glosario de Carlos Von Der Becke.



Utilizando las matrices de la tabla 3.1 y 3.2, posteriormente la combinación de las mismas basados en la tabla 2.2 obtenemos la matriz inicial del algoritmo expuesta en la tabla 3.3:

Matriz Disco	D1	D2	D3	D4	D5
	Rápido	Lento	Rápido	Medianamente Rápido	Medianamente Rápido

Tabla 3.1 Matriz disco

Matriz Tabla		
	T1	Altamente Accesada
	T2	Altamente Accesada
	T3	Poco acceso
	T4	Altamente Accesada
	T5	Poco acceso
	T6	Medianamente accesada
	T7	Medianamente accesada
	T8	Altamente Accesada
	T9	Poco acceso
	T10	Altamente Accesada

Tabla 3.2. Matriz tabla

De la combinación de estas dos matrices se obtiene:

MATRIZ PESOS TABLA VS DISCO					
	D1	D2	D3	D4	D5
T1	10	0	10	5	5
T2	10	0	10	5	5
T3	0	10	0	5	5
T4	10	0	10	5	5
T5	0	10	0	5	5
T6	5	0	5	10	10
T7	5	0	5	10	10
T8	10	0	10	5	5
T9	0	10	0	5	5
T10	10	0	10	5	5

Tabla 3.3. Matriz pesos tablas vs. disco

El algoritmo diseñado llena una matriz randómicamente con 0 y 1, para posteriormente convertirlas en las posibilidades ideales (10) en 1, a las regulares (5) en 1 con la opción de poder cambiarla a 0, o mantenerse como tal y finalmente a las posibilidades no deseadas las convierte en 0, de esta manera la matriz convertida quedaría como se muestra a continuación:



MATRIZ TABLA VS DISCO CONVERTIDA					
	D1	D2	D3	D4	D5
T1	1	0	1	1/0	1/0
T2	1	0	1	1/0	1/0
T3	0	1	0	1/0	1/0
T4	1	0	1	1/0	1/0
T5	0	1	0	1/0	1/0
T6	1/0	0	1/0	1	1
T7	1/0	0	1/0	1	1
T8	1	0	1	1/0	1/0
T9	0	1	0	1/0	1/0
T10	1	0	1	1/0	1/0

Tabla 3.4. Matriz tabla vs. disco convertida

Se clasificó inicialmente la distribución de tablas en los discos, sin embargo observamos que para las tablas: T1, T2, T4, T8 y T10 las posiciones ideales serían que se guarden en los discos: D1 o D3, recomendable en los discos D4 o D5 y no deseado en el disco D2, de esta manera si empezamos a colocar todas las tablas en los primeros discos saturaremos en poco tiempo los mismos mientras que los siguientes estarían poco utilizados siendo esto ineficiente.

El algoritmo genético trabaja con una función conocida como  $fitness^{(20)}$ , (la cual se la puede evidenciar en el [anexo G](#)), la misma que compara el cromosoma actual con uno promedio y de ser mejor lo aprueba, recalculando el nuevo promedio o desecha al

<sup>(20)</sup> Función del algoritmo genético que compara al cromosoma actual con un promedio establecido anteriormente.

inicial, para de esta manera evaluar todas las posibilidades de los cromosomas y determinar cuales serían las mejores posiciones para las diferentes tablas; es así como escoge en segunda instancia a la matriz indicada en la tabla 3.5:

MATRIZ TABLA VS DISCO DEPURADA POR UNA CORRIDA DE ALGORITMO GENETICO					
	D1	D2	D3	D4	D5
T1	1	0	0	0	0
T2	0	0	1	0	0
T3	0	1	0	0	0
T4	1	0	0	0	0
T5	0	1	0	0	0
T6	0	0	0	1	0
T7	0	0	0	1	0
T8	1	0	0	0	0
T9	0	1	0	0	0
T10	0	0	1	0	0



Tabla 3.5. Matriz tabla vs. disco depurada

En la tabla anterior podemos observar claramente como el problema de distribución se ha resuelto puesto que las tablas: T1, T4, T8 han sido colocadas en el disco D1, y que a su vez este disco era el ideal para las mismas. De la misma manera para las tablas T2 y T10 se encuentran ubicadas en el disco D3 y este disco a su vez es uno de

los ideales para las mismas. La misma situación ocurre para el resto de tablas las cuales han sido colocadas en los lugares ideales para su almacenamiento.

Sin embargo el sistema ahora es poco eficaz puesto que no existe ninguna tabla que haya sido guardada en el disco D5, (es decir los recursos han sido mal utilizados), por lo cual el sistema no ha sido explotado en su mayor capacidad, razón por la cual se ha creado la siguiente matriz, la cual nos ha servido para fijarnos en este error:

MATRIZ SUMATORIA TABLA VS DISCO					
	D1	D2	D3	D4	D5
Sumatoria	3	3	2	2	0

Tabla 3.6. Matriz sumatoria tabla vs. disco

Entonces al algoritmo genético inicia un proceso de cruzamiento (cross over), el cual consiste en tomar los 2 predecesores anteriores y de esta manera realizar un proceso de cruzamiento de genes en los mismos, los cuales se verán reflejados en un nuevo cromosoma, de características diferentes respecto a los dos anteriores, para visualizar con un ejemplo disponemos de:

Tomado de Matriz Tabla vs. disco convertida					
T1	1	0	1	1/0	1/0
Tomado de Matriz vs. disco depurada por una corrida de algoritmo genético					
T1	1	0	0	0	0

Tabla 3.7. Matriz de tablas pares

De los cuales se ha obtenido la siguiente tabla de combinaciones en las cuales se aprecia los posibles cromosomas válidos, a los cuales se les aplica nuevamente a cada uno de ellos la función fitness del algoritmo genético, ver anexo II.

Como se puede apreciar muchos de los cromosomas se repiten. esa es la razón para lo cual colocamos literales para agruparlos por las similitudes de sus genes, para adicionalmente calcular la probabilidad de cruzamiento.

MATRIZ POSIBLES GENES REPETIDOS	
TIPO DE CROMOSOMA	No. DE REPETICION
A	8
B	5
C	7
D	4
E	8
F	4
G	8
H	4



**CIB-ESPOL**

Tabla 3.9 Matriz posibles genes repetidos

En caso de existir un cruzamiento no aceptable, es decir, que no cumpla con la función fitness, el algoritmo esta en capacidad de devolver aleatoriamente un descendiente.

Una vez realizado el cruzamiento el algoritmo esta diseñado para realizar un proceso conocido como mutación el cual arbitrariamente altera uno o más genes de un

cromosoma seleccionado; para de esta manera cubrir con posibilidades que no han sido consideradas por el cruzamiento, como lo apreciamos en el siguiente cuadro:

Tomado de Matriz Tabla vs. disco convertida					
T1	I	0	I	I/0	I/0

Tabla 3.10 Matriz antes de mutación

MATRIZ POSIBILIDADES DE MUTACION						
T1	1	0	1	1	1	D
T1	1	0	1	1	0	C
T1	1	0	1	0	1	B
T1	1	0	1	0	0	A
T1	1	0	0	1	1	H
T1	1	0	0	1	0	G
T1	1	0	0	0	1	F
T1	1	0	0	0	0	E
T1	1	1	1	1	1	No considerada
T1	1	1	1	1	0	No considerada
T1	1	1	1	0	1	No considerada
T1	1	1	1	0	0	No considerada
T1	1	1	0	1	1	No considerada
T1	1	1	0	1	0	No considerada
T1	1	1	0	0	1	No considerada
T1	1	1	0	0	0	No considerada
T1	0	0	1	1	1	No considerada
T1	0	0	1	1	0	No considerada
T1	0	0	1	0	1	No considerada
T1	0	0	1	0	0	No considerada
T1	0	0	0	1	1	No considerada
T1	0	0	0	1	0	No considerada
T1	0	0	0	0	1	No considerada
T1	0	0	0	0	0	No considerada
T1	0	1	1	1	1	No considerada
T1	0	1	1	1	0	No considerada
T1	0	1	1	0	1	No considerada
T1	0	1	0	1	0	No considerada
T1	0	1	0	0	1	No considerada
T1	0	1	0	0	0	No considerada

Tabla 3.11 Matriz posibilidades de mutación

De esta manera se puede apreciar como existen posibilidades que no han sido consideradas hasta el momento para su aplicación, de la misma manera al igual que en el caso anterior se calcula la probabilidad de mutación y la creación de nuevos cromosomas a los cuales nuevamente se los vuelve a aplicar la función fitness, para determinar si estos son ideales, regulares o no deseados.

Al algoritmo genético trabaja con una aproximación igual a 1, y una vez obtenida esta la convierte en posibilidades de almacenamiento binario los cuales se puede apreciar como una Matriz de Tabla vs. Disco, indicando con el número 1 como al ideal y al número 0 como el no deseado. Adicionalmente nos indica el número de transacciones que realizó para llegar a este resultado, el cual después de varias pruebas realizadas lo estimamos en 650000 instancias (se debe considerar que mientras mayores sean las ocasiones de aplicación más preciso será el resultado). Como es obvio suponer no existe una respuesta única ideal y de la misma manera al realizar varias ejecuciones sobre un mismo problema los resultados serán diferentes, los mismos que han sido guardados en una base de datos cuyo esquema se pone de manifiesto a continuación:



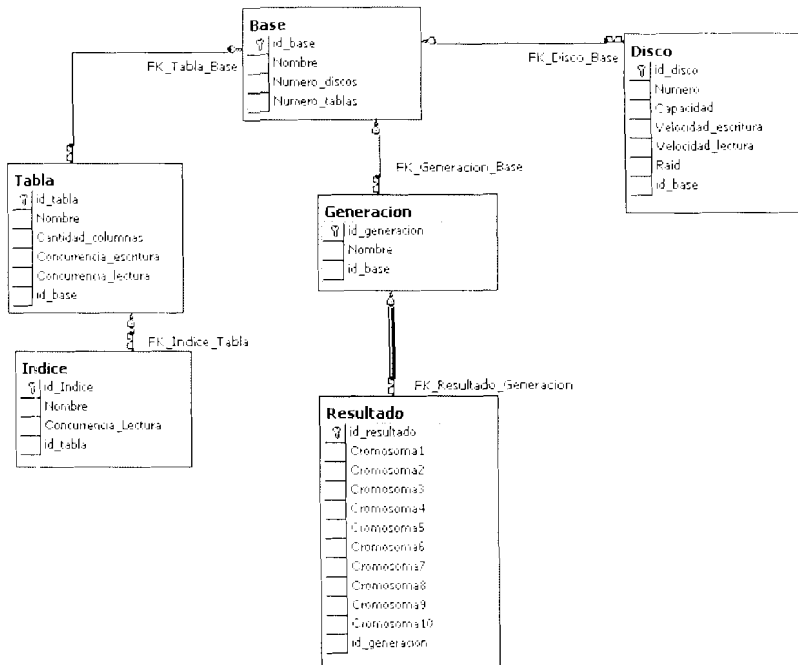


Figura 3.1 Diseño de la base de datos

### 3.2. Justificativo para el uso del algoritmo genético

Sus inicios datan a partir de los años sesenta en donde surgieron muy primitivamente y hasta la actualidad se han ido desarrollando gracias al crecimiento de la tecnología, evolución de la naturaleza de los programas, técnicas de programación entre otros.

Los algoritmos genéticos han sido desarrollados basados en el principio de la evolución es decir, el de mejor estado o condición es el que se impone sobre los demás y sobre la base de él se genera el resto de su especie. Es así como aprovechando técnicas de programación basados en estos mencionados algoritmos los

cuales son aplicados sobre todo en la resolución de problemas de optimización, optimización de funciones lineales y no lineales, problemas del viajero (la ruta mas corta), problemas de itinerarios, partición y control; han sido utilizados cada día mas en diferentes áreas cada día.

Basándonos en las necesidades evidenciadas por parte de los DBA, creemos que existe la necesidad de crear un programa capaz de decidir cual es el lugar más adecuado para los respaldos realizados en las bases de datos razón por la cual era necesario la construcción de un CASE capaz de brindar asesoría para la mencionada labor.

El hecho de ser CASE, ha permitido que sea de fácil instalación y uso, para lo cual se recomienda al lector familiarizarse con el mismo mediante la lectura del manual de usuario en el anexo Z.



**CIB-ESPOL**

# CAPITULO 4 BENCHMARK

## 4.1. Justificación

Los autores del presente hemos trabajado en la Dirección General del Personal de la Armada (DIGPER), en donde nos encontramos con los siguientes equipos y configuraciones:

Los servidores que posee DIGPER son los siguientes:

1 servidor IBM xSeries 235.

1 equipo para almacenamiento masivo (marca STORAGETECH).

1 servidor HP Proliant recientemente adquirido.

2 servidores INTEL SBT2.

1 servidor NETFINITY 3500.



**CIB-ESPOL**

La base de datos que utilizan todos es SQL 7.0, con las licencias de DINFOR (Dirección de Informática de la Armada). Se dispone de estas configuraciones:

#### 4.2. Servidor IBM xSeries 235

Este servidor tiene 6 HD Hot Swap, 5 discos de 73.4 GB de capacidad y 10K rpm, el HD último tiene una capacidad de 36.4 GB.

Numero de discos	Capacidad	velocidad
1	73.4 Gb	10 KRPM
2	73.4 Gb	10 KRPM
3	73.4 Gb	10 KRPM
4	73.4 Gb	10 KRPM
5	73.4 Gb	10 KRPM
6	36.4 Gb	10 KRPM

Tabla 4.1 Servidor IBM x Series 235

Los 5 HD están configurados en RAID 5 con un disco como Hot Spare, por si uno de ellos falla este disco lo reemplaza sin tener problemas de funcionamiento. La capacidad total del arreglo es de 205 GB. El HD de 36.4 GB esta configurado con RAID 0, en el cual se encuentra instalado el sistema operativo. Estos discos se instalaron el mes de Octubre del 2004 y no se han tenido ningún tipo de problemas.



**CIB-ESPOL**

#### 4.3. Equipo para almacenamiento masivo (SAN)<sup>(21)</sup> STORAGETECH

Este equipo tiene instalados 6 HD de 73 GB Hot Swap para fibra óptica de 10K rpm. Los discos están configurados en RAID 5 con un disco como Hot Spare. La capacidad total de los 6 discos es de 273 GB.

Numero de discos	Capacidad	Velocidad	Conectividad
1	73 Gb	10 KRPM	Fibra óptica
2	73 Gb	10 KRPM	Fibra óptica
3	73 Gb	10 KRPM	Fibra óptica
4	73 Gb	10 KRPM	Fibra óptica
5	73 Gb	10 KRPM	Fibra óptica
6	73 Gb	10 KRPM	Fibra óptica

Tabla 4.2 Equipo SAN STORAGETECH

---

<sup>(21)</sup> SAN (Storage Area Network) es una red o subred compartida de media o alta velocidad compuesta por dispositivos de almacenamiento.

Adicional, DIGPER adquirió 2 HD de iguales características de los 6 anteriores, únicamente con diferencia en el firmware. Uno de estos discos está instalado en el SAN debido a una falla que presentó uno de los originales.

Problemas. Este equipo ha presentado 4 fallas desde su instalación que fue realizada el mes de Mayo del 2003.

1. La primera se dio a fines del mes de Octubre del 2003 en la cual falló uno de los discos.
2. Un mes después otro disco falló (sin motivo aparente).
3. La tercera falla se produjo en el mes de Julio del 2004, por problemas de la controladora; esto fue solucionado el mes de Febrero del 2005, es decir el SAN no funcionó alrededor de seis meses.
4. La última falla, se produjo en uno de los discos, a los 2 días de entrar nuevamente en funcionamiento el equipo.

#### *4.4 Servidor INTEL SBT2 (1)*

El servidor que actualmente tiene la BDD y la aplicación de Personal, tiene 4 HD, 3 Discos tienen 16.94 GB de capacidad y un HD tiene 34.18 GB de capacidad. El servidor no tiene una tarjeta RAID para poder crear un arreglo de discos.

Numero de discos	Capacidad	Velocidad
1	16.94 Gb	10 KRPM
2	16.94 Gb	10 KRPM
3	16.94 Gb	10 KRPM
4	34.18 Gb	10 KRPM

Tabla 4.3 INTEL SBT2 (1)

#### 4.5. Servidor INTEL SBT2 (2)

El servidor que actualmente tiene la BDD y la aplicación de Sueldos, tiene 5 HD, 2 Discos tienen 17.10 GB de capacidad y 3 HD tiene 16.94 GB de capacidad. El servidor no tiene una tarjeta RAID para poder crear un arreglo de discos.

Numero de discos	Capacidad	Velocidad
1	17.10 Gb	10 KRPM
2	17.10 Gb	10 KRPM
3	16.94 Gb	10 KRPM
4	16.94 Gb	10 KRPM
5	16.94 Gb	10 KRPM

Tabla 4.4 INTEL SBT2 (2)

#### 4.6. Servidor IBM NETFINITY 3500

Este servidor tiene un disco de 12 GB SCSI, no es Hot Swap. Este servidor actualmente se lo utiliza para el servicio de Internet e ICQ. No se lo utilizó para las pruebas

#### 4.7. Servidor HP Proliant DL 380

Este servidor dispone de 2 HD Hot Swap de 36.4 GB de capacidad y 15K rpm. Desgraciadamente no se obtuvo autorización para la manipulación de servidor para las pruebas.



**CIB-ESPOL**



#### *4.8. Tipos de pruebas*

Es importante mencionar que el CASE desarrollado considera que el DBA tiene un conocimiento previo de la base de datos, en vista de que este deberá conocer características específicas de la misma, es decir, cualquier persona o un DBA que se encuentre pobremente relacionado con la base de datos no podrá conseguir el provecho esperado del CASE. Para las pruebas, utilizamos las bases de datos que se detallan a continuación:

1. DIGPERGYE.- es la base que almacena datos de todo el personal activo y pasivo de la Armada del Ecuador, el equivalente a 10985 personas activas, y es el equivalente a 30 generaciones (promociones), aproximadamente desde el año 1969. El número de usuarios es de 68 internos y 43 desde puntos remotos. La mencionada base de datos dispone de 614 tablas, las cuales después de un análisis evidenciado en los anexos A, B y C muestra los siguientes resultados:

Tipo de tabla	Numero de tabla
Altamente concurridas de escritura	135
Medianamente concurridas de escritura	236
Bajamente concurridas de escritura	42
Altamente concurridas de lectura	127
Medianamente concurridas de lectura	29
Bajamente concurridas de lectura	45
Total de tablas	614
Numero de usuarios internos	68
Numero de usuarios remotos	43
Tamaño	2.636.330 Mb

Tabla 4.5 Tipos de pruebas DIGPERGYE



**CIB-ESPOL**

2. SUELDOS.- es la base de datos que maneja los pagos de salarios de todo el personal de la Armada del Ecuador, su uso es por 22 personas calificadas para ello, y el estudio de las tablas emitió el siguiente resultado:

Tipo de tabla	Numero de tabla
Altamente concurridas de escritura	31
Medianamente concurridas de escritura	17
Bajamente concurridas de escritura	8
Altamente concurridas de lectura	24
Medianamente concurridas de lectura	23
Bajamente concurridas de lectura	21
Total de tablas	124
numero de usuarios	22
Tamaño	4.265.139 Mb

Tabla 4.6 Características Base Sueldos

#### 4.8.1. Configuración 1

Se utilizaron las siguientes configuraciones dando estos resultados:

Base de datos	DIPERGYE	SUELDOS
Servidor	5117Mb	4313 Mb
STORAGETECH	145 minutos	6 fallas
Problemas	Integridad de las tablas	No fue posible de realizar

Tabla 4.7 Configuración No. 1

Durante la realización de las pruebas se presentaron innumerables incidentes tales como: pérdida de datos, pérdida de integridad de las tablas, pérdida de relaciones de las tablas, en conclusión los parámetros ingresados en las tablas que debían ser ordenadas no sirvieron de ayuda, por el contrario el respaldo de los mismos tuvo una duración mayor a la esperada puesto que las mencionadas bases tomaban menos tiempo en sus servidores originales. Al buscar explicación mencionamos que se mantiene un criterio de falla posiblemente en esta configuración por el tipo de servidor y los problemas presentados últimamente.

#### 4.8.2. Configuración 2

Base de datos	DIPERGYE	SUELDOS
Servidor	5117Mb	4313 Mb
Servidor IBM x SERIES 235	96 minutos	52 minutos
Problemas	Nada a mencionar	Nada a mencionar

Tabla 4.8 Configuración No. 2

Durante la realización de estas pruebas en este servidor se evidenció una mejora significativa respecto al anterior y cierta mejora respecto a los servidores originales de las bases de datos.

#### 4.8.3. Configuración 3

Realizamos respaldos en los servidores originales pero con la nueva deposición de orden de las tablas recomendadas por el CASE implementado, obteniéndose los siguientes resultados en promedio. Se debe considerar que los servidores originales no disponen de tarjeta RAID.

Servidor	Base de datos	DIPERGYE	SUELDOS
		5117Mb	4313 Mb
INTEL SBT2 (1)		112 minutos	No aplicable
INTEL SBT2 (2)		No aplicable	52 minutos

Tabla 4.9 Configuración No. 3

## 4.9. Comparación de resultados

Servidor	Base de datos	DIPERGYE	SUELDOS
		5117Mb	
Servidores de prueba			
STORAGETECH		145 minutos	6 intentos fallidos
Servidor IBM x SERIES 235		96 minutos	52 minutos
Servidores originales con nueva disposición de tablas			
Intel SBT2 (1)		112minutos	No aplicable
Intel SBT2 (2)		No aplicable	52minutos
Servidores originales disposición original de tablas			
Intel SBT2 (1)		122minutos	No aplicable
Intel SBT2 (2)		No aplicable	60 minutos

Tabla 4.10 Comparación de resultados

Todos los valores mencionados aquí fueron tomados después de varias pruebas y de esta manera obtuvimos un promedio, tanto en los servidores de prueba como en los originales.



**CIB-ESPOL**

Servidor	Base de datos	DIPERGYE	SUELDOS
		5117Mb	4313 Mb
Servidores de prueba			
STORAGETECH		145	6 intentos fallidos
			Configuración No 1
Servidor IBM x SERIES 235		96	52
			Configuración No 2
Servidores originales con nueva disposición de tablas			
Intel SBT2 (1)		112	No aplicable
			Configuración No 3
Intel SBT2 (2)		No aplicable	55
			No 3
Servidores originales disposición original de tablas			
Intel SBT2 (1)		122	No aplicable
Intel SBT2 (2)		No aplicable	60
desventajas en tiempo de respuesta			
STORAGETECH		118.85%	No aplicable
beneficios en tiempo de respuesta IBM x			
SERIES 235		78.69%	86.67%
beneficios en tiempo de respuesta Intel			
SBT2 (1).		91.80%	No aplicable
beneficios en tiempo de respuesta Intel			
SBT2 (2)		No aplicable	91.66%

Tabla 4.11 Resumen de resultados

Analizando el cuadro anterior concluimos que:



**CIB-ESPOL**

1. Se obtienen muy malos resultados con el servidor STORAGETECH (Configuración No 1), dando una disminución de rendimiento equivalente al 18.85%, siendo la posible causa los problemas mencionados en la descripción del equipo, la degradación del mismo o la incompatibilidad de hardware y software.
2. En el mencionado servidor se produjo excesiva cantidad de errores por lo cual creemos que en versiones posteriores de este CASE se debería considerar la posibilidad de tomar en cuenta el uso y degradación de los discos, para de esta manera tratar de colocar un parámetro más de medición y lograr corregir situaciones como esta.
3. Se tiene buenos resultados con el servidor IBM x SERIES 235 (Configuración No.2), dando una mejora en rendimiento medido en tiempo a un equivalente de 21.31% (para la base de datos DIGPERGYE) y del 13.33% (para la base de datos de SUELDOS). Sin embargo se debe considerar que los servidores disponen de RAID, lo cual no permite verificar la eficiencia auténtica del CASE, en vista de que los servidores originales no disponen de RAID.
4. Con el mencionado servidor no se produjo ningún tipo de conflicto.
5. En los servidores originales, con la nueva disposición de las tablas (Configuración No.3), se obtuvieron mejoras del 8.196% (para la base de datos DIGPERGYE) y del 8.33% (para la base de datos de SUELDOS)



6. Para la utilización del CASE se considera la velocidad standard de los discos, la cual viene como característica propia dada por el fabricante, en ningún momento se considera la velocidad optimizada del RAID.
7. En el CASE elaborado se ha considerado únicamente las tablas en las cuales se guardan datos, motivo por la cual no se ha manipulado ninguna tabla perteneciente al sistema. Esta es la razón por la que los índices se han guardado de acuerdo a la disposición recomendada por el SGBD.

# CAPITULO 5 IMPACTO FINANCIERO

## 5.1. Beneficios cualitativos



Los beneficios cualitativos que se pueden mencionar entre otros son:

1. La utilización de un CASE amigable y sencillo que permita mejorar las tareas del DBA (Ver anexo I, Manual de Usuario).
2. La utilización correcta de los dispositivos de almacenamiento.
3. El mejoramiento de los históricos de las bases de datos.
4. Aprovechamiento de mejor manera de las características tanto de las bases de datos como de los dispositivos utilizados en las operaciones de las mismas.
5. Creación de un control de rendimiento de equipos y su utilización de base de datos.
6. Detección de posibles fallas en dispositivos utilizados por medio de comparación de parámetros guardados en los históricos.
7. Detección de incompatibilidad entre software y hardware.

8. Detección de degradación de equipos por medio de comparaciones de los históricos de grabación.

## *5.2 Beneficios cuantitativos*

Los beneficios cuantitativos más relevantes son:

1. Existe una mejora de los tiempos empleados para las operaciones de mantenimiento de las bases de datos.
2. El algoritmo permite mejorar los procedimientos de respaldo de las bases de datos, logrando de esta manera optimizar este proceso.
3. Al momento disponemos de varias opciones de almacenamiento óptimas, las cuales pueden ser almacenadas en una base de datos, lo que nos permite llevar un histórico de las actividades realizadas en la base de datos.
4. Las disposiciones de almacenamiento recomendadas han sido obtenidas por medio de la aplicación de métodos y técnicas mediante la aplicación de conocimientos establecidos que no son empíricos.
5. Al disminuir el tiempo en una de sus actividades, el DBA, podrá dedicar mayor cantidad de tiempo a otras labores igualmente importantes; como el mantenimiento de la base de datos, la seguridad de la red, la implementación de otros case de optimización, etc.

### 5.3. Recursos y costos

Recursos	Costos	Justificación
Servidores para pruebas		Se ha considerado los valores referenciales que costaron los servidores en los que se realizaron las pruebas. En realidad fueron usados únicamente para la elaboración de las pruebas.
IBM x SERIES 235	\$ 45000,00	
STORAGETECH	\$ 36000,00	
Intel SBT2 (1)	\$ 35000,00	
Intel SBT2 (2)	\$ 42000,00	
Bases de datos		
DIGPERGYE	No aplicable	Puesto que se encuentra el histórico de toda la Armada del Ecuador con datos desde 1969.
Documentos	No aplicable	Tiene valor solo para la Armada del Ecuador
Sueldos	No aplicable	Tiene valor solo para la Armada del Ecuador

Tabla 5.1 Recursos y costos

#### 5.3.1 Beneficios cualitativos



**CIB-ESPOL**

Recursos	Costos	Justificación
Java	No aplicable	La herramienta que los autores han escogido debido a que es un lenguaje universal y libre, que se encuentra al alcance de todos de fácil aprendizaje y orientado a objetos.
PC (para la creación del case)	\$ 400,00	Hemos considerado que una Pc Pentium III de 800 MHz con 128 Mb en RAM, tarjeta de video de 8 Mb, espacio libre en el disco de 100 Mb, con teclado, ratón, monitor de 15", CdRom y tarjeta fax MODEM o de red

Impresora	\$ 80,00	Lexmark
Banda ancha 128 Kbps	\$ 80.00 mensuales	De esta manera evitamos el gasto excesivo en teléfono.
Regulador de voltaje	\$ 20,00	
Cables cruzados		Construcción casera
Software SQL 7.0	\$ 650,00	La licencia para cada cliente.

Tabla 5.2 Recursos utilizados en implementación

### 5.3.2 Recursos y costos

PC (para la utilización del case)	<p>Pc Pentium IV de 2800 MHz</p> <p>512 Mb en RAM</p> <p>Tarjeta de video de 8 Mb, Espacio libre en el disco de 100 Mb</p> <p>Teclado</p> <p>Ratón</p> <p>Monitor de 15''</p> <p>Cd Rom</p>
-----------------------------------	---

Tabla 5.3 Recursos mínimos necesarios

## 5.3.3 Recursos adicionales requeridos para administrar el sistema

Administración del case	Base de datos SQL 7.0 o Access
	Java

Tabla 5.4 Recursos adicionales



## 5.3.4 Costo del software

Recursos	<b>CIB-ESPOL</b> Costos
Desarrollo 1500 por autor	\$ 3000
Java	No aplicable
PC (para la creación del case), una por autor	\$ 800,00 <u>ver anexo J,</u> <u>ver anexo K</u>
Impresora <u>anexo J, anexo K</u>	\$ 80,00
Banda ancha 128 KPSS (\$80 mensuales por 12 meses)	\$ 960,00
Regulador de voltaje	\$ 20,00
Cables cruzados	Construcción casera
Software SQL 7.0, <u>anexo L, anexo M, anexo N,</u> <u>anexo O</u>	\$ 650,00
Total	\$ 6160

Tabla 5.5 Costo del CASE

#### *5.4.ROI*

Lastimosamente los autores del presente no tienen valores económicos para compararlos con los beneficios obtenidos sobre la inversión puesto que las pruebas realizadas fueron hechas en entidades estatales que no producen dinero sino por el contrario el ente DIGPER, es un organismo netamente consumidor de recursos del estado. Los beneficios obtenidos han sido parametrizados en cuestión de tiempo y ventajas respecto a los usuarios del mismo, DBA.

## CONCLUSIONES

1. Se optimizó la capacidad de almacenamiento en los discos utilizados, tanto en los sistemas originales como en los que disponen de RAID.
2. Se mejoró la velocidad en los procesos de respaldo realizado en las diferentes transacciones ordinarias en las bases de datos.



**CIB-ESPOL**

3. Actualmente se dispone de varias alternativas óptimas para la elección del sitio de almacenamiento de las diferentes tablas de las bases de datos, y no simplemente estar a la disposición de lo que nos indica la base de datos.
4. Se puede crear un modelo de comportamiento, Histórico del CASE, basados en las acciones realizadas sobre la misma; una vez que se obtenga la suficiente cantidad de datos.



## RECOMENDACIONES

1. Para versiones futuras se podrá considerar el uso y la degradación de los discos duros y dispositivos, para de esta manera obtener un parámetro más a ser medido y mejorar los valores obtenidos para equipos con problemas.
2. Creación de históricos de bases de datos ya utilizadas.
3. Creación de modelos aplicables a situaciones parecidas de diferentes bases de datos.
4. Creación de índices con la ubicación y manipulación física de estos

## BIBLIOGRAFIA

1. ZBIGNIEW, Michalewicz, "GENETIC ALGORITHMS + DATA STRUCTURES = EVOLUTION PROGRAMS", third extended revised edition, Springer Verlag heidenberg New York, 1999.
2. KENDAL & KENDAL, "ANALISIS Y DISEÑO DE SISTEMAS", tercera edición, México, 1997.
3. GILL & RAO, DATA WARE HOUSING, edición en español, Prentice Hall, Hispanoamericana, Naucaplan de Juárez, Edo. México, México, 1996.
4. FLANAGAN, David, "JAVA EN POCAS PALABRAS", Mc Graw Hill, segunda Edición, 1998.
5. NAUGHTON y SCHILDT "JAVA MANUAL DE REFERENCIAS" Mc Graw Hill, 1999.
6. LATHROP, WEBSTER, TEMPLE, "ARIADNE: PATTERN DIRECTED INFERENCE AND HIERARCHICAL ABSTRACTION IN STRUCTURE RECOGNITION", paper, Journal ACM, Massachusetts, USA, 1997.
7. GRIMSON, Eric, "THE COMBINATORICS OF LOCAL CONSTRAINTS IN MODEL BASED", paper, Journal ACM, California, USA, 2003.



**CIB-ESPOL**

8. LIBERATORE, Paolo, "COMPABILITY AND COMPACT REPRESENTATION OF REVISION OF HORN KNOWLEDGE BASES", paper, Journal New York, USA, 2000.
9. AGUILAR, Luis, "FUNDAMENTOS DE PROGRAMACION: ALGORITMOS Y ESTRUCTURAS DE DATOS", segunda edición, Mc Graw Hill, Madrid, España, 1996.
10. <http://ccp.servidores.net/genetico.html> COSTA, Carlos, "Algoritmos Genéticos", 2003
11. <http://geocities.com/CapeCanaveral/9802/3d5ca000.htm> Larraña, "Algoritmos Genéticos", Departamento de Ciencias de Computación e inteligencia artificial, Universidad del país vasco, 2003.
12. <http://www.ianet.com/users/jcontre/genetic> CONTRERAS, J, "Tutorial sobre algoritmos genéticos", Maracaibo, 2003.
13. <http://redcientifica.com/doc/doc199904260011.html> COELLO, Carlos, "Introducción a los algoritmos genéticos", 2003.
14. <http://geocities.com/ohcop/busqueda.html>
15. <http://geocities.com/ohcop/indeex.html>
16. <http://www.programacion.com/tutorial/moddatos/4/>

17. <http://tramullas.com/documatica/indice.html>
18. <http://www.programacion.com/tutorial/moddatos/4/>
19. <http://www.tau.org.ar/base/lara.pue.udlap.mx/sistoper/index.html>
20. <http://www.cindoc.csic.es/isis/01-3-1.htm>
21. <http://ict.udlap.mx/people.html>
22. <http://www.inei.gob.pe/>
23. [http://exa.unne.edu.ar/depar/areas/informatica/anasistem2/public\\_html/apuntes/maf/anexos/estructura.htm](http://exa.unne.edu.ar/depar/areas/informatica/anasistem2/public_html/apuntes/maf/anexos/estructura.htm)
24. <http://perseo.dif.um.es/~fernan/clasifica.ppt>
25. [http://docentes.usaca.edu.co/wildiaz/BDII\\_02.html](http://docentes.usaca.edu.co/wildiaz/BDII_02.html)
26. <http://www.algorithmia.net/articles.php?id=12>

27. [http://www.lafacu.com/apuntes/informatica/algorit\\_ordena/default.htm](http://www.lafacu.com/apuntes/informatica/algorit_ordena/default.htm)

28. [http://decsai.ugr.es/docencia/ii/ii\\_mod\\_av\\_bd.html](http://decsai.ugr.es/docencia/ii/ii_mod_av_bd.html)

29. [http://www.itlp.edu.mx/publica/tutoriales/admonbasedat/tema2\\_1.htm](http://www.itlp.edu.mx/publica/tutoriales/admonbasedat/tema2_1.htm)

## ANEXO A

**NUMERO DE REGISTROS Y NUMERO DE COLUMNAS  
OBTENIDOS EN LA BASE DE DATOS PERSONAL (DIGPERGYE)**

No	NOMBRE TABLA	# REGISTROS	# DE COLUMNAS
1	A DISPOSICION	91	23
2	ACCION TOMADA	19	7
3	anios_rep_especifico	0	4
4	AÑO	43	5
5	ASCENSO CIVIL	11060	24
6	ASCENSOS NAVALES	53928	37
7	ASCENSOS SALIDAS ANUALES	634	10
8	AUTORIDAD REPORTES	396	12
9	AUTORIZACION MATRIMONIO	3517	13
10	aux_cal	8986	3
11	aux_pase prest	107283	10
12	auxiliar1	1487	11
13	auxiliar2	107376	7
14	back_conde_ofi	17	45
15	BACKUP COND HIST OFI 032003	29	45
16	BACKUP CONDEC ABRIL	63	45
17	BAJAS CI	0	12
18	BANCO	5	7
19	BANCO PERSONA	10282	14
20	BASE CODES	6064	28
21	BENEFICIO	7	12
22	BONIFICACION CURSOS EMCI	84	8
23	BONO ASCENSO	0	13
24	CAB PRESTAMO LVN	0	8
25	Calif Ayu RO GO EE	2314	11
26	CALIFICACION ANUAL	271649	28
27	CALIFICACION AYUDANTE	970	15
28	CALIFICACION DIREDU	415	11
29	CALIFICACION SEGURIDAD	7	8
30	CALIFICACION TRIPULANTES	33	7
31	CALIFICACIONtemp	60762	20
32	CALIFICACIONtemp CIVILES	1839	15
33	calificaciontemp_cond2	4387	20
34	CALIFICACIONTEMP_UNIDADES	2840	14
35	calificaciontempciviles	0	20
36	CANDIDATOS CURSOS ESP	0	7
37	CANDIDATOS EXP LAB	204	8
38	CANDIDATOS PREP ACAD	0	8
39	CANDIDATOS TRASBORDOS	49	13
40	CAPREMC1	1732	10
41	CARGO	10	<b>CIB-ESPOL</b> 6
42	CATEGORIA	16	8
43	CAUSA A DISPOSICION	4	7
44	CAUSA ADICIONAL	36	7
45	CAUSA ASCENSO	40	11
46	CAUSA CESANTIA	0	6
47	CAUSA CONCESION PERMISO	13	7
48	CAUSA CONFIDENCIA	15	7
49	CAUSA CUOTA	9	7
50	CAUSA DISPONIBILIDAD	80	9
51	CAUSA EGRESO	8	7
52	CAUSA EXAMEN	8	7
53	CAUSA FINALIZACION	73	7
54	CAUSA INGRESO	9	7



55	CAUSA INTERRUPCION	5	7
56	CAUSA JUICIO	66	8
57	CAUSA JUICIO ANTERIOR	20	8
58	CAUSA RECONOCIMIENTO	10	7
59	CAUSA RETIRO	1	7
60	CAUSA SALIDA EXTERIOR	13	7
61	CAUSA SERMAN	13	7
62	CAUSA SUBSIDIO	1	6
63	CAUSA SUBSISTE	6	7
64	CEDULA	1222	1
65	CEDULAS	469	4
66	CEDULAS ORDENADO	439	10
67	CENSURA AYUDANTE	7	12
68	CENSURAS ASCENSO	146	16
69	CENTRO SALUD	20	7
70	ci prueba	2639	1
71	CIRCULO	24	2
72	CLASE CURSO	18	1
73	CLASE GRADO	10	1
74	CLASE SANCION	26	16
75	CLASIFICACION ESPECIALIDAD	24	1
76	CLIEN	2	1
77	CODIGO EMBARQUE	0	1
78	COLOR CABELLO	7	1
79	COLOR OJO	5	1
80	COLOR PIEL	5	1
81	COMANDO OPERATIVO	133	1
82	Compensacion Transitoria	221	1
83	COND RESP DIC	52	18
84	cond trip marzo 2002	75	45
85	CONDEC	9	45
86	condec emci	21	1
87	condec enero trip2003	196	1
88	CONDEC INDIV	1	20
89	condec indiv grupo	2344	20
90	CONDEC INDIV GRUPO2	312	20
91	CONDEC INDIV GRUPO3	290	20
92	condec marz 2003	56	45
93	CONDEC ORDEN GENERAL	3171	19
94	CONDEC PROY	1006	45
95	CONDEC PROY WAY2	180	45
96	condec prueba	133	45
97	CONDEC REC TIE SERVICIO	1387	4
98	CONDEC SERV ANT	728	4
99	condecoraciones	269	45
100	condecoraciones emcis	154	20
101	CONDECORACIONES EMCIS2	643	23
102	condecoraciones historico	131	45
103	condecoraciones proyeccion	0	45
104	condecoraciones respaldio	62	45
105	CONDICION TRABAJO	7	7
106	cons	6	13
107	CONSEJO	4	7
108	CONY SIN CODIGO ISSFA	90	8
109	CONYUGUE	9797	24
110	copia plazapp enero9	46825	22
111	copia ppp enero9	409	22
112	COPIA SUBJUDICE 23_02_2001	171	24
113	COPIA UNI ADM	3722	33
114	ep_23jun03	94	1
115	CSTO	216	3
116	cuadro condecoraciones	0	13

118	CUADRO CONDECORACIONES3	539	26
119	CUMPLE MINIMO EMBARQUE	69	2
120	CUOTA ELIMINACION	36	17
121	CUR_PROF	4110	16
122	CURSO	5982	18
123	CURSO ASCENSO AGREGADOS	40	13
124	CURSO CPNV	15	16
125	curso espec ayudante	0	10
126	CURSO MANDO SUBM	2279	14
127	CURSO PERSONA	73810	50
128	CURSOS AGREGADOS	444	14
129	CURSOS ESPECIALES	4	11
130	CURSOS FALLIDOS PROFORMA	0	4
131	CURSOS REALIZADOS	1432	20
132	D OPERACION FORMA_SYS	32	5
133	D PERFIL OPE_SYS	73	5
134	DATOS FECHA CAMBIADOS	456	2
135	DECLARACION	8115	29
136	DECLARACION REPORTE	0	29
137	DEMERITO GRADO AYUDANTE	196	12
138	DEMERITO PRISION	4	13
139	demerito temp ayudante	797	19
140	demeritos_subm_temp	2449	20
141	DEMERITOS_TEMP	18643	20
142	DEMERITOS_TEMP_AGUENA	124	16
143	desapare_1	2	1
144	desapare_2	56	5
145	desapare_28022002	0	98
146	desapare_real	287	1
147	DESAPARECIDO	3	18
148	DESIGNA AYUDANTE	0	5
149	DET PRESTAMO LVN	0	3
150	DETALLE RECON PROCESO	380	6
151	DEVENGAR ESTUDIO	4	7
152	DIPLOMADO	5	7
153	DIRECC	7383	19
154	DISCAPACIDAD	0	15
155	DISPONIBILIDAD	23083	27
156	DIV_POL_ECUA	1073	9
157	DOCUMENTOS_VIDA NAVAL	10	11
158	dtproperties	77	7
159	EMBAR EFECTIVO	30	6
160	embarque	17	8
161	embarque_2	254	10
162	embarque_bonif	156	7
163	embarque_candi_ofi	17	8
164	embarque_cond	5892	9
165	embarque_cond_trip	0	9
166	embarque_cond666	9	16
167	embarque_cond666_tripu	12582	16
168	embarque_detalle	0	10
169	embarque_detalle_tripu	1526	10
170	EMBARQUE_IM	0	8
171	embarque_moran	10	7
172	embarque1	37	11
173	embarque2	1061	10
174	embarque2_copia	6087	10
175	embarque3	5876	10
176	EMBARQUE3 COPIA	0	10
177	EMCIS CALIFICACION BAJA	435	1
178	EMCIS_PROM_ASCENSOS	507	13



179	emcis prom ascensos antes	602	13
180	emcis prom ascensos2	582	13
181	emcis temporal	473	13
182	ENFERMEDAD	17	7
183	error migra pz	321	6
184	error migra pz copia	1117	6
185	ESPECIALIDAD	27860	20
186	ESPECIALIDAD EQUIV	0	4
187	ESPERA CI SANIA	0	7
188	Estadis Tiem Servicio	5906	11
189	ESTADO CIVIL	7	8
190	ETAPA	0	2
191	EVA CIVIL RA	0	18
192	EVAL CUA ATRIBUTOS	0	25
193	EVALUACION	7964	37
194	EVALUACION AGREGADOS	80	32
195	EVALUACION AGUENA	98	32
196	EVALUACION AYUDANTE	258	24
197	EVALUACION CPNV	11	23
198	EVALUACION CURSO AGREGADOS	410	10
199	EVALUACION FALTAS AGREGADOS	80	17
200	EVALUACION NOTAS AGREGADOS	469	12
201	EVALUACION RES 01 05 00	1364	25
202	EVALUACION SALIDAS EXTERIOR	231	5
203	EVALUACION SUBMI	181	20
204	EXAMEN MEDICO	19615	20
205	EXCEPCION REQUISITO	80	13
206	EXPERIENCIA LABORAL	87	11
207	FALTA SANCION	257686	33
208	FALTA SANCION NOVIDADES	2172	5
209	FALTAS	42	2
210	faltas grado	0	6
211	FICHA INDIVIDUAL	0	30
212	fonifa	2640	3
213	formulario issta	403	13
214	FORMULARIO ISSEA I	82	12
215	FOTOGRAFIA	7021	10
216	fuera organico temporal	98	7
217	GORRA	36	9
218	GORRAL	25	9
219	GRADO	34	18
220	GRADO CURSO	86	15
221	GRADO DISCAPACIDAD	11	7
222	GRADO POR PASI	0	3
223	grados persona sueldos 04	5924	4
224	grados persona sueldos 05	910	4
225	grados persona sueldos 06	1503	4
226	GRUPO ESPECIALIDAD	10	11
227	GRUPO MEDICO	4	7
228	GRUPO PERSONA	9	8
229	Grupos Directiva	6850	7
230	HI	0	2
231	HIJO	27766	33
232	HIJOS DIFERENTES MADRES	13	6
233	hijos prueba	4	2
234	HIJOS SIN CODIGO ISSEA	901	8
235	HIJOS SIN CODIGO ISSEA OFI	85	11
236	HIJOS SIN CODIGO ISSEA TRIP	818	11
237	HIS CONTRATO	2	20
238	hist condec ofi ene 2003	12	45
239	HISTORICO CAL TRI	1532	14
240	HISTORICO CONDECORACIONES	4323	45

242	HISTORICO ESPECIALIDAD	16384	17
243	HISTORICO FORMACION	23579	17
244	HISTORICO PERF	8517	17
245	HISTORICO PERSONAL	29368	19
246	HISTORICO SUBJUDICE	0	6
247	HISTORICO TRANSACCION	324136	12
248	HOSPITALIZACION	11866	21
249	IDS	1455	1
250	INDIV_COND_EMCI	1	11
251	INDIV2	2	20
252	INDIV3	2	20
253	INFORMACION_ADICIONAL	340	21
254	INFORMACION_RESERVADA	28	21
255	INFORMACION_SUMARIA	88	23
256	INGENIO_PERSONA	0	21
257	INSTITUTOS_MILITARES	300	13
258	INSTRUCCION_CIVIL	15004	33
259	interes_curso	3	6
260	ISSFA_CONYUGUE	0	8
261	ISSFA_ESPOSA	44	3
262	ISSFA_HIJOS	11135	8
263	ISSFA_PADRES	7743	8
264	ISSFA_TIPO_HIJO	38	4
265	ISSFA_TITULAR	6932	7
266	ito	908	1
267	JUZGADO	7	7
268	JUZGADO_ANTERIOR	1	6
269	LICENCIA_PERSONAL	325321	30
270	LICENCIA_PROFORMA	4352	20
271	LICENCIA_PROFORMA_NOV	201	22
272	LICENCIA_PROFORMA_TEMP	0	20
273	LICENCIA_TEMPORAL	0	14
274	LISTA_CALIFICACION	8	8
275	MATERIA_AYUDANTIA	13	9
276	MAXFEC	16292	2
277	MEDIDAS	13488	23
278	MERITO_AYUDANTE	7	7
279	MERITO_TEMP_AYUDANTE	785	18
280	MERITOS_ASCENSO	12	8
281	MERITOS_SUBM_TEMP	744	16
282	MERITOS_TEMP	10032	1
283	MERITOS_TEMP_AGUENA	100	3
284	MESES	12	5
285	MODIF_PERSONA_TIPO_DOC	7812	2
286	moran_2002	452	4
287	MORANI_2002	389	5
288	moran2_2002	405	6
289	MOT_FIN_CON	2	2
290	movimientotemp	10677	3
291	MULTIPLES_EMBARQUE	978	4
292	NEW_GRADE	23	2
293	NIVEL	14	6
294	NIVEL_CONESUP	7	2
295	NIVEL_INST_CIVIL	42	3
296	NIVEL_REQUI	0	2
297	no_encuentra_pz	0	4
298	NOMBRE	56	2
299	NOVEDAD_ASCENSO	3191	2
300	novedad_ascenso_3_mayo	192	9
301	NOVEDAD_ASCENSO_RESP0501	192	2
302	NOVEDAD_AYUDANTE	95	5

304	NUMERO AFILIACION ISSFA_TEMP	181	2
305	OFICIALES PASE	0	2
306	OPCION SISTEMA	1131	6
307	OPCION TIPO DOC	261	2
308	OPCIONES INTEGRACION	4	19
309	OPERACION FORMA	458	6
310	ORGANICO	7	5
311	organicoyefectivo	97	7
312	ORGANICOYEFFECTIVO_TRIPU	409	15
313	OrganicoyEfectivoMaugi	174	5
314	otros	121	7
315	otrosids	129	5
316	PADRE	24631	1
317	PADRES ISSFA_2001	12981	1
318	PADRES ISSFA_2002	306	19
319	PADRES ISSFA_2003	33	2
320	PADRES SIN_CODIGO	1715	2
321	padres1	729	2
322	padres2	1485	7
323	PAGO CONDEC	1594	7
324	PAIS	225	13
325	PARAMETRO EVAL AYUDANTE	7	23
326	PARAMETROS EVALUACION	81	7
327	PARAMETROS EVALUACION AYUDAN	0	8
328	PARAMETROS PROCESO PASES	21	13
329	PARTIDA PRESUPUESTO	19	13
330	Pasaporte Visas	1838	6
331	PASE	151265	8
332	pase prest temp	109289	26
333	PASE TEMPORAL	2108	32
334	PASE ULTIMO	9284	15
335	PASES FALLIDOS PROFORMA	27	6
336	pases_in	154	2
337	pases_in otros	340	4
338	PASES OFICIALES	0	2
339	pases provisionales	938	2
340	PER CONT_NO_PROF	55	13
341	per id	21	18
342	PER ID EFEC_CAP	338	1
343	per id salen_Cap	217	6
344	per id2	873	1
345	PER ID3	5817	1
346	per maugi	49	1
347	per no sale pz	0	1
348	per pase especifico	2756	1
349	PER PRO_CUR COPIA	1396	2
350	per prof_pas copia	5714	3
351	per sale pz gye	70	18
352	per sin fec efe	166	22
353	PERCON	0	1
354	PERFECCIONAMIENTO	7	1
355	PERFIL AUTORIZARPT_SYS	8	9
356	PERFIL CONSULTA DOCU_SYS	18	9
357	PERFIL CONSULTA GRADO_SYS	75	6
358	PERFIL CONSULTA_SYS	534	7
359	PERFIL GRUPO_SYS	6183	8
360	PERFIL OPERACION_SYS	3867	6
361	PERFIL PLAZA	38	9
362	PERIODO ASCENSO	21	6
363	PERIODO AYUDANTE	20	24
364	PERMANENCIA_GRADO	33	12

365	PERMANENCIA OPERATIVA	10618	5
366	PERMANENCIA OPERATIVA TEMP	0	13
367	PERMISO OPCION OBJETO SYS	5026	24
368	PERSONA	26413	16
369	PERSONA VACANTE_ASC	0	12
370	PERSONACI	2	100
371	personal inteligencia	144	4
372	PERSONAL PROFORMA CURSOS	1650	23
373	PERSONAL PROFORMA PASES	28609	3
374	pla iden	92	18
375	pla_pro_ofi_35	1460	22
376	PLAZA ORG TEMPORAL	15786	1
377	PLAZA ORG TEMPORAL AUX	6319	22
378	PLAZA ORG TEMPORAL_me	16220	28
379	PLAZA ORGANICA	90211	28
380	PLAZA ORGANICA 12 01	16216	28
381	plaza organica 22 ENE	14172	28
382	plaza organica 27 DIC	14172	28
383	plaza organica prueba	18165	28
384	PLAZA PROFORMA	15896	28
385	PLAZA PROFORMA copia_me	26056	28
386	PLAZA PROFORMA PASES	1049355	28
387	plaza_proforma_pases_14ene04	151091	28
388	plaza_proforma_pases_67	11013	22
389	plaza_proforma_pases_68	1756	22
390	PLAZA PROFORMA QUITO	15786	22
391	plaza temporal	390	22
392	plaza temporal_me	390	28
393	plazas proyeccion	9	7
394	po temp_ofi_actual	119	7
395	po temp_ofi_nuevo	120	5
396	PREPARACION ACADEMICA	0	1
397	PRESTACION SERVICIOS	10733	1
398	PROCESO CANDIDATOS	7	12
399	PROFES	4086	22
400	proforma	1	7
401	PROFORMA CURSOS	26	24
402	proforma disponibles emci	0	16
403	proforma disponibles ofi	3	16
404	proforma disponibles trip	1	1
405	PROFORMA PASES	108	1
406	proforma vacantes ofi	22	1
407	PROM GRADUACION	1	14
408	promedio cal	6	1
409	PROMEDIO CAL ADJUNTOS	14	10
410	PROMEDIO CAL AGUENA	373	3
411	PROMEDIO CAL AYUDANTE	795	5
412	PROMEDIO CAL CPNV	48	4
413	promedio cal emcis	4	5
414	PROMEDIO CAL EVALUACION	3309	5
415	PROMEDIO CAL EVALUACION TRIPU	24641	3
416	promedio cal2	1411	5
417	PROMEDIO CALIFICACION TRIPU	53	5
418	PROMOCION ASCENSO	389	3
419	PROYECCION_ASC	3535	2
420	prueba embarque	580	13
421	PRUEBA FISICA	0	12
422	prueba seguridad1	33	7
423	PRY VACANTES	110	21
424	PUESTO	9446	21
425	Puesto_05_01_2000	8592	23
426	PUESTO OCUPADO	11081	17



**CIB-ESPOL**

428	RAMA	7	15
429	REC TIEM SERVICIO	16939	17
430	REC TIEM SERVICIO TEMP	0	7
431	RECOMPENSA	36045	21
432	REFERENCIA LEGAL	35	12
433	REGION PLAZA	16	26
434	REGLAMENTO REPARTO OPERATIVO	3	18
435	rep_anterior	0	6
436	reparto_operativo	53	6
437	REPARTO OPERATIVO FECHA VAL	317	1
438	REPARTO PADRE	188	7
439	REPARTO PADRE NUEVO ORG	189	4
440	repartos_persona sueldos 04	5901	3
441	repartos_persona sueldos 05	910	3
442	repartos_persona sueldos 06	1501	4
443	REPORTE PASES	348	4
444	reportes	0	4
445	REQUISITO PERSONA	0	7
446	REQUISITOS	0	9
447	REQUISITOS ASCENSO	544	11
448	REQUISITOS CURSOS PUESTO	0	6
449	REQUISITOS PUESTOS PUESTO	0	17
450	RESOLUCION CONSEJO	2185	8
451	RESOLUCION PERSONA	5509	8
452	RESOLUCION PERSONA TEMP	1895	19
453	resp_pla_org_06_06_2000	42346	11
454	resp_pla_org_21_06_2000	42346	5
455	resp_puesto_21_06_2000	8559	23
456	resp_uni_adm_21_06_2000	10509	23
457	respa_condec	162	12
458	respaldo_condec_20012003	149	29
459	respaldo_condec_trip_dic_2002	51	45
460	RESPALDO CONDECORACIONES	44	45
461	respaldo_condecoraciones2	101	45
462	Resultados	2015	45
463	resumen efectivos plazas	60	45
464	RETMIL	0	2
465	RPT EVALUACION	57	11
466	RPT VACANTES	598	8
467	SALIDA EXTERIOR	10734	9
468	SALTOS DIGPER	0	16
469	SALTOS ESDEIM	0	22
470	SANCION JUICIO	4	5
471	SECTOR ASCENSO CEREMONIA	138	6
472	SELECCION AGREGADOS	80	2
473	SELECCION AYUDANTE	254	5
474	SELECCION EMBARQUE	16655	19
475	SELECCION EMBARQUE OFI	2664	29
476	seleccion_embarque_ofi_temp	6	17
477	SELECCION LLAMADA	7198	16
478	SELECCION LLAMADA CIVILES	1762	12
479	SELECCION LLAMADA MORAN	909	30
480	SELECCION LLAMADA RESP0501	427	12
481	SELECCION PERSONAL PASES	1023	27
482	SELECCION PLAZAS PASES	0	26
483	SERVICIO ANTERIOR	3747	17
484	SEXO	3	13
485	SIN PLAZA	9215	20
486	SISTEMA SYS	16	6
487	SOCIO	0	5
488	SUBJ PER	0	3

489	SUBJUDICE	368	6
490	SUBSISTE	4927	2
491	t persona	10287	36
492	t reparto	260	15
493	t01ago	49	11
494	t01ene	53	2
495	t01feb	31	3
496	t01jul	39	3
497	t01mar	22	3
498	t01nov	23	3
499	t01oct	32	3
500	t01sep	56	3
501	t16dic	708	3
502	t16ene	29	3
503	t16feb	28	3
504	t16jun	183	3
505	t16mar	32	3
506	tab_temp_pase	12	3
507	TABLA_EMCI	0	3
508	TABLA_INGRESO	1	3
509	TABLA_NAVAL	6655	4
510	TABLA1	3	2
511	TABLA2	4	4
512	tablaz1	9	2
513	TALLA	6	3
514	TALLAS MEDIDAS	24	1
515	TB_NOTRANSAC	5	8
516	TB_TRANSAC	247398	8
517	temp_asc_civ_cal_cur	2692	1
518	TEMP_ASC_CIV_PROM_1999	536	9
519	TEMP_ASC_CIV_PROM_2002	70	22
520	TEMP_ASCENSOS_EMCI	556	23
521	TEMP_CEDULA	1	23
522	Temp_numero	456985	24
523	temp_per_id_coguar_calif	115	1
524	temp_reparto_nuevo_org	189	2
525	templ	357	1
526	TEMPO	21	1
527	TEMPORAL	4121	5
528	TEMPORAL	340	3
529	TEMPORAL_OFICIALES	959	24
530	TIE_REP_OPE	59	24
531	TIEM_PER	0	4
532	tiem_rep	6864	4
533	TIEMPO_EMBARQUE_TOTAL	83	9
534	TIEMPO_REPARTO	0	9
535	TIEMPO_REPARTO_TEMP	43	10
536	tiempo_tripu	510	9
537	TIEMPO1	252	16
538	Tiempos Sectores	7863	6
539	TIPO_AUDITORIA	8	11
540	TIPO_BREVET	8	25
541	TIPO_CARGO	5	3
542	TIPO_CAUSA_DECLARACION	5	8
543	TIPO_CONDECORACION	536	6
544	TIPO_CONTRATO	4	2
545	TIPO_CUENTA	3	20
546	TIPO_CURSO	4	2
547	TIPO_CURSO_EMCI	4	6
548	TIPO_DIV_POL	1	6
549	TIPO_DOC	45	6
550	TIPO_DOCUMENTO_DIGITAL	6	6



**CIB-ESPOL**

552	TIPO FALTA	606	8
553	TIPO INFORMACION	6	6
554	TIPO INGENIO	4	18
555	TIPO INSTRUCCION	959	3
556	TIPO JUICIO	3	3
557	TIPO JUICIO_anterior	1	14
558	TIPO RANCHO	8	6
559	TIPO REF LEGAL	6	6
560	TIPO RESPONSABILIDAD	4	8
561	TIPO SANGRE	9	6
562	TIPO UNI ADM	11	2
563	tipoins	1844	8
564	TITULO	165	6
565	tmp_ract_sin_historico	11	2
566	TOTAL_EMBARQUE	0	10
567	total_marosg	186	1
568	TRABAJO_PROFES_AYUDANTE	429	10
569	TRANSACCION	1729032	1
570	TRANSACCION_ASCENSOS	21269	15
571	TRANSACCION_CONSULTA	30946	12
572	TRANSACCION_DIARIA	14493	20
573	transaccion_diaria_historico	25895	6
574	transaccion_reporte	61898	7
575	TRANSACCION_ROLES	297	7
576	TRANSACCION_SEGURIDAD_SYS	99607	9
577	TRANSACCION_USUARIO	3447	8
578	trip_detalle_embarque	0	4
579	trip_detalle_embarque_22Nov04	0	7
580	trip_detalle_embarque2	0	14
581	trip_detalle_embarqueMRAMIREZ	0	14
582	trip_detalle_pasesHPEREZ	0	12
583	trip_detalle_pasesMRAMIREZ	0	14
584	trip_no_esta_rep	0	14
585	trip_no_esta_rep_2	0	14
586	trip_no_esta_rep_22nov04	0	11
587	trip_no_esta_rep_copia	0	3
588	trip_no_esta_repMRAMIREZ	0	11
589	TRIP_SOLTERO	147	11
590	trip_tiempo_repartoHPEREZ	0	11
591	trip_tiempo_repartoMRAMIREZ	0	2
592	TRIP_TIEMPO_SEGUIDO_GYE	0	21
593	TRIP_TIEMPO_SEGUIDO_UIO	0	21
594	UBICACION_LVN	0	11
595	UNI_ADM_ESPRAN_JARNAV	0	11
596	UNI_ADM_PROFORMA	3892	11
597	UNI_ADM_PROFORMA_copia_me	0	13
598	UNI_ADM_PROFORMA_QUITO	0	30
599	UNI_ADM_SUBREP_SUBDES	0	30
600	UNI_ADM_SUELDOS	0	30
601	UNI_ADM_SUINLI	0	33
602	UNI_ADM_TEMPORAL	0	10
603	UNI_ADM_TEMPORAL_AUX	0	33
604	UNI_ADM_TEMPORAL_me	0	33
605	unidad_adm_prueba	0	30
606	UNIdad_ADMimistrativa_TEMPORAL	0	30
607	UNIdad_ADMimistrativa_TEMPORAL_me	0	33
608	UNIDAD_ADMINISTRATIVA	21978	33
609	UNIDAD_ADMINISTRATIVA_12_01	0	33
610	UNIDAD_ADMINISTRATIVA2	0	34
611	unidades_flote_escuadron	0	33
612	verifica_pase	0	33



**CIB-ESPOL**

613	xPLAZA_PROFORMA	0	8
614	xUNI_ADM_PROFORMA	0	8



# ANEXO B

## NUMERO DE TRANSACCIONES DE ESCRITURA Y PROMEDIOS OBTENIDOS EN LA BASE DE DATOS PERSONAL (DIGPERGYE)

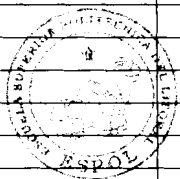
Numero de días laborados: 1365

No	NOMBRE TABLA	# TRANSACCIONES	PROMEDIO DE ESCRITURA		
			ALTAS	MEDIAS	BAJAS
1	A DISPOSICION	134	0.0	0.0	0.1
2	ACCION TOMADA	153248	112.3	0.0	0.0
3	anios rep especifico	120563	0.0	88.3	0.0
4	AÑO	115874	0.0	84.9	0.0
5	ASCENSO CIVIL	8969	0.0	0.0	6.6
6	ASCENSOS NAVALES	14077	0.0	0.0	10.3
7	ASCENSOS SALIDAS ANUALES	168547	123.5	0.0	0.0
8	AUTORIDAD REPORTES	95874	0.0	70.2	0.0
9	AUTORIZACION MATRIMONIO	65873	0.0	48.3	0.0
10	aux cal		0.0	0.0	0.0
11	aux pase prest	125874	0.0	92.2	0.0
12	auxiliar1		0.0	0.0	0.0
13	auxiliar2	115870	0.0	84.9	0.0
14	back conde ofi		0.0	0.0	0.0
15	BACKUP COND HIST OFI 032003	205845	150.8	0.0	0.0
16	BACKUP CONDEC ABRIL	120205	0.0	88.1	0.0
17	BAJAS CI	159632	116.9	0.0	0.0
18	BANCO	158745	116.3	0.0	0.0
19	BANCO PERSONA	679	0.0	0.0	0.5
20	BASE CODES	102587	0.0	75.2	0.0
21	BENEFICIO	158412	116.1	0.0	0.0
22	BONIFICACION CURSOS EMCI	187596	137.4	0.0	0.0
23	BONO ASCENSO	199856	146.4	0.0	0.0
24	CAB PRESTAMO LVN	189632	138.9	0.0	0.0
25	Calif Ayu RO GO EE	84596	0.0	62.0	0.0
26	CALIFICACION ANUAL	10344	0.0	0.0	7.6
27	CALIFICACION AYUDANTE		0.0	0.0	0.0
28	CALIFICACION DIREDU		0.0	0.0	0.0
29	CALIFICACION SEGURIDAD	197865	145.0	0.0	0.0
30	CALIFICACION TRIPULANTES	268593	196.8	0.0	0.0
31	CALIFICACIONtemp	98520	0.0	72.2	0.0
32	CALIFICACIONtemp CIVILES	96458	0.0	70.7	0.0
33	calificaciontemp cond2	58746	0.0	43.0	0.0
34	CALIFICACIONTEMP UNIDADES	65879	0.0	48.3	0.0
35	calificaciontempciviles	87540	0.0	64.1	0.0
36	CANDIDATOS CURSOS ESP	169536	124.2	0.0	0.0
37	CANDIDATOS EXP LAB	156323	114.5	0.0	0.0
38	CANDIDATOS PREP ACAD	169842	124.4	0.0	0.0
39	CANDIDATOS TRASBORDOS	140029	102.6	0.0	0.0
40	CAPREMCI	87566	0.0	64.2	0.0
41	CARGO	189654	138.9	0.0	0.0
42	CATEGORIA	198632	145.5	0.0	0.0
43	CAUSA A DISPOSICION	187542	137.4	0.0	0.0
44	CAUSA ADICIONAL	109683	0.0	80.4	0.0
45	CAUSA ASCENSO	185969	136.2	0.0	0.0
46	CAUSA CESANTIA	206873	151.6	0.0	0.0
47	CAUSA CONCESION PERMISO	263535	193.1	0.0	0.0
48	CAUSA CONFIDENCIA	65325	0.0	47.9	0.0
49	CAUSA CUOTA	58759	0.0	43.0	0.0
50	CAUSA DISPONIBILIDAD	156960	115.0	0.0	0.0
51	CAUSA EGRESO	157889	115.7	0.0	0.0
52	CAUSA EXAMEN	154036	112.8	0.0	0.0
53	CAUSA FINALIZACION	159630	116.9	0.0	0.0
54	CAUSA INGRESO	158723	116.3	0.0	0.0

56	CAUSA JUICIO	153662	112.6	0.0	0.0
57	CAUSA JUICIO ANTERIOR	158203	115.9	0.0	0.0
58	CAUSA RECONOCIMIENTO	138027	101.1	0.0	0.0
59	CAUSA RETIRO	98320	0.0	72.0	0.0
60	CAUSA SALIDA EXTERIOR		0.0	0.0	0.0
61	CAUSA SERMAN	56986	0.0	41.7	0.0
62	CAUSA SUBSIDIO	142563	104.4	0.0	0.0
63	CAUSA SUBSISTE	145252	106.4	0.0	0.0
64	CEDULA	152303	111.6	0.0	0.0
65	CEDULAS	87596	0.0	64.2	0.0
66	CEDULAS ORDENADO	63963	0.0	46.9	0.0
67	CENSURA AYUDANTE	156201	114.4	0.0	0.0
68	CENSURAS ASCENSO	158745	116.3	0.0	0.0
69	CENTRO SALUD	164520	120.5	0.0	0.0
70	ci prueba	152031	111.4	0.0	0.0
71	CIRCULO	143002	104.8	0.0	0.0
72	CLASE CURSO	187014	137.0	0.0	0.0
73	CLASE GRADO	87569	0.0	64.2	0.0
74	CLASE SANCION		0.0	0.0	0.0
75	CLASIFICACION ESPECIALIDAD	65879	0.0	48.3	0.0
76	CLIEN	146750	107.5	0.0	0.0
77	CODIGO EMBARQUE	162487	119.0	0.0	0.0
78	COLOR CABELLO	142896	104.7	0.0	0.0
79	COLOR OJO	154022	112.8	0.0	0.0
80	COLOR PIEL	56894	0.0	41.7	0.0
81	COMANDO OPERATIVO	98653	0.0	72.3	0.0
82	Compensacion Transitoria	66983	0.0	49.1	0.0
83	COND RESP DIC		0.0	0.0	0.0
84	cond trip marzo 2002		0.0	0.0	0.0
85	CONDEC	87563	0.0	64.1	0.0
86	condec emci	54873	0.0	40.2	0.0
87	condec enero trip2003		0.0	0.0	0.0
88	CONDEC INDIV	146875	107.6	0.0	0.0
89	condec indiv grupo	138750	101.6	0.0	0.0
90	CONDEC INDIV GRUPO2	152320	111.6	0.0	0.0
91	CONDEC INDIV GRUPO3		0.0	0.0	0.0
92	condec marz 2003	144701	106.0	0.0	0.0
93	CONDEC ORDEN GENERAL	58787	0.0	43.1	0.0
94	CONDEC PROY	95875	0.0	70.2	0.0
95	CONDEC PROY WAY2	56852	0.0	41.6	0.0
96	condec prueba		0.0	0.0	0.0
97	CONDEC REC TIE SERVICIO	187563	137.4	0.0	0.0
98	CONDEC SERV ANT	209983	153.8	0.0	0.0
99	condecoraciones	206845	151.5	0.0	0.0
100	condecoraciones emcis		0.0	0.0	0.0
101	CONDECORACIONES EMCIS2	87562	0.0	64.1	0.0
102	condecoraciones historico	98547	0.0	72.2	0.0
103	condecoraciones proyeccion	65321	0.0	47.9	0.0
104	condecoraciones respaldño		0.0	0.0	0.0
105	CONDICION TRABAJO	196328	143.8	0.0	0.0
106	cons		0.0	0.0	0.0
107	CONSEJO	205874	150.8	0.0	0.0
108	CONY SIN CODIGO ISSFA		0.0	0.0	0.0
109	CONYUGUE	11290	0.0	0.0	8.3
110	copia plazapp enero9	126902	0.0	93.0	0.0
111	copia ppp enero9		0.0	0.0	0.0
112	COPIA SUBJUDICE 23 02 2001	78563	0.0	57.6	0.0
113	COPIA UNI ADM	58749	0.0	43.0	0.0
114	cp 23jun03		0.0	0.0	0.0
115	CSTO	86524	0.0	63.4	0.0
116	cuadro condecoraciones		0.0	0.0	0.0

8	CUADRO CONDECORACIONES3	162058	118.7	0.0	0.0
9	CUMPLE MINIMO EMBARQUE	208632	152.8	0.0	0.0
10	CUOTA ELIMINACION	29	0.0	0.0	0.0
11	CUR_PROF	3448	0.0	0.0	2.5
12	CURSO	655	0.0	0.0	0.5
13	CURSO ASCENSO AGREGADOS	115876	0.0	84.9	0.0
14	CURSO CPNV	128753	0.0	94.3	0.0
15	curso espec ayudante		0.0	0.0	0.0
16	CURSO MANDO SUBM	206987	151.6	0.0	0.0
17	CURSO PERSONA		0.0	0.0	0.0
18	CURSOS AGREGADOS	203654	149.2	0.0	0.0
19	CURSOS ESPECIALES	215860	158.1	0.0	0.0
20	CURSOS FALLIDOS PROFORMA	98210	0.0	71.9	0.0
21	CURSOS REALIZADOS	85742	0.0	62.8	0.0
22	D OPERACION FORMA SYS		0.0	0.0	0.0
23	D PERFIL OPE SYS	98650	0.0	72.3	0.0
24	DATOS FECHA CAMBIADOS	58745	0.0	43.0	0.0
25	DECLARACION	528	0.0	0.0	0.4
26	DECLARACION REPORTE	206741	151.5	0.0	0.0
27	DEMERITO GRADO AYUDANTE	65873	0.0	48.3	0.0
28	DEMERITO PRISION	59860	0.0	43.9	0.0
29	demerito temp ayudante	115690	0.0	84.8	0.0
30	demeritos subm temp	126857	0.0	92.9	0.0
31	DEMERITOS TEMP	105412	0.0	77.2	0.0
32	DEMERITOS TEMP AGUENA	158320	116.0	0.0	0.0
33	desapare_1	65872	0.0	48.3	0.0
34	desapare_2	56841	0.0	41.6	0.0
35	desapare_28022002	58965	0.0	43.2	0.0
36	desapare real	49850	0.0	36.5	0.0
37	DESAPARECIDO	1	0.0	0.0	0.0
38	DESIGNA AYUDANTE	156987	115.0	0.0	0.0
39	DET PRESTAMO LVN	169874	124.4	0.0	0.0
40	DETALLE RECON PROCESO		0.0	0.0	0.0
41	DEVENGAR ESTUDIO	153986	112.8	0.0	0.0
42	DIPLOMADO	78325	0.0	57.4	0.0
43	DIRECC	59874	0.0	43.9	0.0
44	DISCAPACIDAD		0.0	0.0	0.0
45	DISPONIBILIDAD	3855	0.0	0.0	2.8
46	DIV_POL_ECUA	95820	0.0	70.2	0.0
47	DOCUMENTOS VIDA NAVAL	153103	112.2	0.0	0.0
48	dtproperties		0.0	0.0	0.0
49	EMBAR_EFECTIVO	95873	0.0	70.2	0.0
50	embarque		0.0	0.0	0.0
51	embarque_2		0.0	0.0	0.0
52	embarque bonif	85624	0.0	62.7	0.0
53	embarque candi ofi	84751	0.0	62.1	0.0
54	embarque cond	58426	0.0	42.8	0.0
55	embarque cond trip	64875	0.0	47.5	0.0
56	embarque cond666		0.0	0.0	0.0
57	embarque cond666 tripu		0.0	0.0	0.0
58	embarque detalle		0.0	0.0	0.0
59	embarque detalle tripu		0.0	0.0	0.0
60	EMBARQUE_IM	84597	0.0	62.0	0.0
61	embarque moran		0.0	0.0	0.0
62	embarque1		0.0	0.0	0.0
63	embarque2	122587	0.0	89.8	0.0
64	embarque2_copia	98562	0.0	72.2	0.0
65	embarque3	102587	0.0	75.2	0.0
66	EMBARQUE3 COPIA	56987	0.0	41.7	0.0
67	EMCIS CALIFICACION BAJA	98720	0.0	72.3	0.0
68	EMCIS PROM ASCENSOS	105832	0.0	77.5	0.0

emcis_prom ascensos2			0.0	0.0	0.0
emcis temporal			0.0	0.0	0.0
1 ENFERMEDAD			0.0	0.0	0.0
2 error migra pz	105873		0.0	77.6	0.0
3 error migra pz copia	130258		0.0	95.4	0.0
4 ESPECIALIDAD	16		0.0	0.0	0.0
5 ESPECIALIDAD EQUIV			0.0	0.0	0.0
6 ESPERA CESANTIA	86542		0.0	63.4	0.0
7 Estadis Tiem Servicio	108452		0.0	79.5	0.0
8 ESTADO CIVIL	65874		0.0	48.3	0.0
9 ETAPA	102687		0.0	75.2	0.0
10 EVA CIVIL RA			0.0	0.0	0.0
11 EVAL_CUA ATRIBUTOS			0.0	0.0	0.0
12 EVALUACION			0.0	0.0	0.0
13 EVALUACION AGREGADOS	187569	137.4	0.0	0.0	0.0
14 EVALUACION AGUENA	209235	153.3	0.0	0.0	0.0
15 EVALUACION AYUDANTE	268756	196.9	0.0	0.0	0.0
16 EVALUACION CPNV	105876	0.0	0.0	77.6	0.0
17 EVALUACION CURSO AGREGADOS	210586	154.3	0.0	0.0	0.0
18 EVALUACION FALTAS AGREGADOS	126987	0.0	0.0	93.0	0.0
19 EVALUACION NOTAS AGREGADOS	129872	0.0	0.0	95.1	0.0
20 EVALUACION RES 01 05 00	105741	0.0	0.0	77.5	0.0
21 EVALUACION SALIDAS EXTERIOR			0.0	0.0	0.0
22 EVALUACION SUBM			0.0	0.0	0.0
23 EXAMEN MEDICO	84		0.0	0.0	0.1
24 EXCEPCION REQUISITO			0.0	0.0	0.0
25 EXPERIENCIA LABORAL	3		0.0	0.0	0.0
26 FALTA SANCION	13064		0.0	0.0	9.6
27 FALTA SANCION NOVEDADES	69875		0.0	51.2	0.0
28 FALTAS			0.0	0.0	0.0
29 faltas grado			0.0	0.0	0.0
30 FICHA INDIVIDUAL	658725	482.6	0.0	0.0	0.0
31 fonifa	78524	0.0	0.0	57.5	0.0
32 formulario issfa			0.0	0.0	0.0
33 FORMULARIO ISSFA T	578489	423.8	0.0	0.0	0.0
34 FOTOGRAFIA	658742	482.6	0.0	0.0	0.0
35 fuera organico temporal			0.0	0.0	0.0
36 GORRA	65874		0.0	48.3	0.0
37 GORRA I	106845		0.0	78.3	0.0
38 GRADO			0.0	0.0	0.0
39 GRADO CURSO			0.0	0.0	0.0
40 GRADO DISCAPACIDAD			0.0	0.0	0.0
41 GRADO POR PASE			0.0	0.0	0.0
42 grados persona sueldos 04			0.0	0.0	0.0
43 grados persona sueldos 05			0.0	0.0	0.0
44 grados persona sueldos 06	65872		0.0	48.3	0.0
45 GRUPO ESPECIALIDAD			0.0	48.3	0.0
46 GRUPO MEDICO	69825		0.0	51.2	0.0
47 GRUPO PERSONA			0.0	0.0	0.0
48 Grupos Directiva	210654	154.3	0.0	0.0	0.0
49 HI	108745		0.0	79.7	0.0
50 HIJO	5928		0.0	0.0	4.3
51 HIJOS DIFERENTES MADRES	65987		0.0	48.3	0.0
52 hijos prueba	187563	137.4	0.0	0.0	0.0
53 HIJOS SIN CODIGO ISSFA	78598		0.0	57.6	0.0
54 HIJOS SIN CODIGO ISSFA OFI	65873		0.0	48.3	0.0
55 HIJOS SIN CODIGO ISSFA TRIP	95874		0.0	70.2	0.0
56 HIS CONTRATO	74589		0.0	54.6	0.0
57 hist condec ofi ene 2003	228367	167.3	0.0	0.0	0.0
58 HISTORICO CAL TRI	6		0.0	0.0	0.0
59 HISTORICO CONDECORACIONES	142587	104.5	0.0	0.0	0.0



CTB-ESPCL

242	HISTORICO ESPECIALIDAD	223	0.0	0.0	0.2
243	HISTORICO FORMACION	483	0.0	0.0	0.4
244	HISTORICO PERF	113	0.0	0.0	0.1
245	HISTORICO PERSONAL	109658	0.0	80.3	0.0
246	HISTORICO SUBJUDICE	109256	0.0	80.0	0.0
247	HISTORICO TRANSACCION	115873	0.0	84.9	0.0
248	HOSPITALIZACION	106985	0.0	78.4	0.0
249	IDS		0.0	0.0	0.0
250	INDIV_COND_EMCI	121983	0.0	89.4	0.0
251	INDIV2	132698	0.0	97.2	0.0
252	INDIV3	135874	0.0	99.5	0.0
253	INFORMACION ADICIONAL		0.0	0.0	0.0
254	INFORMACION RESERVADA		0.0	0.0	0.0
255	INFORMACION SUMARIA	1402	0.0	0.0	1.0
256	INGENIO PERSONA		0.0	0.0	0.0
257	INSTITUTOS MILITARES		0.0	0.0	0.0
258	INSTRUCCION CIVIL	3905	0.0	0.0	2.9
259	interes curso		0.0	0.0	0.0
260	ISSFA CONYUGUE	168747	123.6	0.0	0.0
261	ISSFA ESPOSA	165874	121.5	0.0	0.0
262	ISSFA HIJOS	162753	119.2	0.0	0.0
263	ISSFA PADRES	156630	114.7	0.0	0.0
264	ISSFA TIPO HIJO		0.0	0.0	0.0
265	ISSFA TITULAR	139621	102.3	0.0	0.0
266	ito		0.0	0.0	0.0
267	JUZGADO	59863	0.0	43.9	0.0
268	JUZGADO ANTERIOR	59876	0.0	43.9	0.0
269	LICENCIA PERSONAL	2826	0.0	0.0	2.1
270	LICENCIA PROFORMA		0.0	0.0	0.0
271	LICENCIA PROFORMA NOV	98325	0.0	72.0	0.0
272	LICENCIA PROFORMA TEMP	104852	0.0	76.8	0.0
273	LICENCIA TEMPORAL	125960	0.0	92.3	0.0
274	LISTA CALIFICACION		0.0	0.0	0.0
275	MATERIA AYUDANTIA		0.0	0.0	0.0
276	MAXFEC		0.0	0.0	0.0
277	MEDIDAS	48211	0.0	35.3	0.0
278	MERITO AYUDANTE		0.0	0.0	0.0
279	MERITO TEMP AYUDANTE		0.0	0.0	0.0
280	MERITOS ASCENSO	126875	0.0	92.9	0.0
281	MERITOS SUBM TEMP		0.0	0.0	0.0
282	MERITOS TEMP		0.0	0.0	0.0
283	MERITOS TEMP AGUENA		0.0	0.0	0.0
284	MESES		0.0	0.0	0.0
285	MODIF PERSONA TIPO DOC	99502	0.0	72.9	0.0
286	moran 2002		0.0	0.0	0.0
287	MORANI 2002		0.0	0.0	0.0
288	moran2 2002		0.0	0.0	0.0
289	MOT FIN CON	65821	0.0	48.2	0.0
290	movimientotemp		0.0	0.0	0.0
291	MULTIPLES EMBARQUE	65921	0.0	48.3	0.0
292	NEW GRADE		0.0	0.0	0.0
293	NIVEL	74521	0.0	54.6	0.0
294	NIVEL CONESUP	69250	0.0	50.7	0.0
295	NIVEL INST CIVIL	137895	101.0	0.0	0.0
296	NIVEL REQUI	74852	0.0	54.8	0.0
297	no encuentra pz		0.0	0.0	0.0
298	NOMBRE		0.0	0.0	0.0
299	NOVEDAD ASCENSO	136784	100.2	0.0	0.0
300	novedad ascenso 3 mayo		0.0	0.0	0.0
301	NOVEDAD ASCENSO RESP0501	63529	0.0	46.5	0.0
302	NOVEDAD AYUDANTE	165785	121.5	0.0	0.0

304	NUMERO AFILIACION ISSFA TEMP	96521	0.0	70.7	0.0
305	OFICIALES PASE		0.0	0.0	0.0
306	OPCION SISTEMA	85632	0.0	62.7	0.0
307	OPCION TIPO DOC		0.0	0.0	0.0
308	OPCIONES INTEGRACION	142587	104.5	0.0	0.0
309	OPERACION FORMA	85413	0.0	62.6	0.0
310	ORGANICO	146987	107.7	0.0	0.0
311	organicoyefectivo		0.0	0.0	0.0
312	ORGANICOYEFECTIVO TRIPU	136587	100.1	0.0	0.0
313	OrganicoyEfectivoMaugi		0.0	0.0	0.0
314	otros		0.0	0.0	0.0
315	otrosids	108652	0.0	79.6	0.0
316	PADRE	2153	0.0	0.0	1.6
317	PADRES ISSFA 2001	59873	0.0	43.9	0.0
318	PADRES ISSFA 2002	65842	0.0	48.2	0.0
319	PADRES ISSFA 2003	84521	0.0	61.9	0.0
320	PADRES SIN CODIGO	62958	0.0	46.1	0.0
321	padres1		0.0	0.0	0.0
322	padres2		0.0	0.0	0.0
323	PAGO CONDEC		0.0	0.0	0.0
324	PAIS		0.0	0.0	0.0
325	PARAMETRO EVAL AYUDANTE	95624	0.0	70.1	0.0
326	PARAMETROS EVALUACION	84745	0.0	62.1	0.0
327	PARAMETROS EVALUACION AYUDANTE	95846	0.0	70.2	0.0
328	PARAMETROS PROCESO PASES	62518	0.0	45.8	0.0
329	PARTIDA PRESUPUESTO	95847	0.0	70.2	0.0
330	Pasaporte Visas		0.0	0.0	0.0
331	PASE	135334	0.0	99.1	0.0
332	pase prest temp		0.0	0.0	0.0
333	PASE TEMPORAL		0.0	0.0	0.0
334	PASE ULTIMO	51203	0.0	37.5	0.0
335	PASES FALLIDOS PROFORMA	95989	0.0	70.3	0.0
336	pases im	30625	0.0	22.4	0.0
337	pases im otros		0.0	0.0	0.0
338	PASES OFICIALES		0.0	0.0	0.0
339	pases provisionales		0.0	0.0	0.0
340	PER CONT NO PROF	95623	0.0	70.1	0.0
341	per id		0.0	0.0	0.0
342	PER ID EFEC CAP	126845	0.0	92.9	0.0
343	per id salen Cap	102685	0.0	75.2	0.0
344	per id2	92584	0.0	67.8	0.0
345	PER ID3	54879	0.0	40.2	0.0
346	per maugi		0.0	0.0	0.0
347	per no sale pz		0.0	0.0	0.0
348	per pase especifico		0.0	0.0	0.0
349	PER PRO CUR COPIA	136875	100.3	0.0	0.0
350	per prof pas copia	106852	0.0	78.3	0.0
351	per sale pz gye		0.0	0.0	0.0
352	per sin fec efe		0.0	0.0	0.0
353	PERCON		0.0	0.0	0.0
354	PERFECCIONAMIENTO	95624	0.0	70.1	0.0
355	PERFIL AUTORIZARPT SYS	65321	0.0	47.9	0.0
356	PERFIL CONSULTA DOCU SYS	98654	0.0	72.3	0.0
357	PERFIL CONSULTA GRADO SYS	54872	0.0	40.2	0.0
358	PERFIL CONSULTA SYS	65214	0.0	47.8	0.0
359	PERFIL GRUPO SYS	65874	0.0	48.3	0.0
360	PERFIL OPERACION SYS	62548	0.0	45.8	0.0
361	PERFIL PLAZA	266	0.0	0.0	0.2
362	PERIODO ASCENSO		0.0	0.0	0.0
363	PERIODO AYUDANTE	148987	109.1	0.0	0.0
364	PERMANENCIA GRADO		0.0	0.0	0.0



**CIB-ESPOL**

366	PERMANENCIA OPERATIVA TEMP	148753	109.0	0.0	0.0
367	PERMISO OPCION OBJETO SYS	56847	0.0	41.6	0.0
368	PERSONA	991176	726.1	0.0	0.0
369	PERSONA VACANTE ASC		0.0	0.0	0.0
370	PERSONACI	154784	113.4	0.0	0.0
371	personal inteligencia	65983	0.0	48.3	0.0
372	PERSONAL PROFORMA CURSOS	68625	0.0	50.3	0.0
373	PERSONAL PROFORMA PASES	68674	0.0	50.3	0.0
374	pla iden	67621	0.0	49.5	0.0
375	pla pro ofi 35		0.0	0.0	0.0
376	PLAZA ORG TEMPORAL	165478	121.2	0.0	0.0
377	PLAZA ORG TEMPORAL AUX	158423	116.1	0.0	0.0
378	PLAZA ORG TEMPORAL me	169875	124.5	0.0	0.0
379	PLAZA ORGANICA	471128	345.1	0.0	0.0
380	PLAZA ORGANICA 12 01		0.0	0.0	0.0
381	plaza organica 22 ENE		0.0	0.0	0.0
382	plaza organica 27 DIC		0.0	0.0	0.0
383	plaza organica prueba		0.0	0.0	0.0
384	PLAZA PROFORMA	168759	123.6	0.0	0.0
385	PLAZA PROFORMA copia me	68654	0.0	50.3	0.0
386	PLAZA PROFORMA PASES	98687	0.0	72.3	0.0
387	plaza proforma pasas 14ene04	69584	0.0	51.0	0.0
388	plaza proforma pasas 67	78459	0.0	57.5	0.0
389	plaza proforma pasas 68	56897	0.0	41.7	0.0
390	PLAZA PROFORMA QUITO	158759	116.3	0.0	0.0
391	plaza temporal		0.0	0.0	0.0
392	plaza temporal me	95685	0.0	70.1	0.0
393	plazas proyeccion		0.0	0.0	0.0
394	po temp ofi actual		0.0	0.0	0.0
395	po temp ofi nuevo		0.0	0.0	0.0
396	PREPARACION ACADEMICA	84789	0.0	62.1	0.0
397	PRESTACION SERVICIOS	4404	0.0	0.0	3.2
398	PROCESO CANDIDATOS		0.0	0.0	0.0
399	PROFES	1070	0.0	0.0	0.8
400	proforma		0.0	0.0	0.0
401	PROFORMA CURSOS	145896	106.9	0.0	0.0
402	proforma disponibles emci	95874	0.0	70.2	0.0
403	proforma disponibles ofi	98569	0.0	72.2	0.0
404	proforma disponibles trip	65987	0.0	48.3	0.0
405	PROFORMA PASES	149698	109.7	0.0	0.0
406	proforma vacantes ofi	98604	0.0	72.2	0.0
407	PROM GRADUACION	75841	0.0	55.6	0.0
408	promedio cal	95857	0.0	70.2	0.0
409	PROMEDIO CAL ADJUNTOS	165874	121.5	0.0	0.0
410	PROMEDIO CAL AGUENA	148563	108.8	0.0	0.0
411	PROMEDIO CAL AYUDANTE	139521	102.2	0.0	0.0
412	PROMEDIO CAL CPNV	158743	116.3	0.0	0.0
413	promedio cal emcis		0.0	0.0	0.0
414	PROMEDIO CAL EVALUACION	62950	0.0	46.1	0.0
415	PROMEDIO CAL EVALUACION TRIPU	95623	0.0	70.1	0.0
416	promedio cal2		0.0	0.0	0.0
417	PROMEDIO CALIFICACION TRIPU		0.0	0.0	0.0
418	PROMOCION ASCENSO	168536	123.5	0.0	0.0
419	PROYECCION ASC	154786	113.4	0.0	0.0
420	prueba embarque		0.0	0.0	0.0
421	PRUEBA FISICA	156987	115.0	0.0	0.0
422	prueba seguridad I		0.0	0.0	0.0
423	PRY VACANTES	95628	0.0	70.1	0.0
424	PUESTO	343	0.0	0.0	0.3
425	Puesto 05 01 2000	54875	0.0	40.2	0.0
426	PUESTO OCUPADO	65892	0.0	48.3	0.0

490	SUBSISTE	5634	0.0	0.0	4.1
491	t persona	99563	0.0	72.9	0.0
492	t reparto	102587	0.0	75.2	0.0
493	t01ago		0.0	0.0	0.0
494	t01ene		0.0	0.0	0.0
495	t01feb		0.0	0.0	0.0
496	t01jul		0.0	0.0	0.0
497	t01mar		0.0	0.0	0.0
498	t01nov		0.0	0.0	0.0
499	t01oct		0.0	0.0	0.0
500	t01sep		0.0	0.0	0.0
501	t16dic		0.0	0.0	0.0
502	t16ene		0.0	0.0	0.0
503	t16feb		0.0	0.0	0.0
504	t16jun		0.0	0.0	0.0
505	t16mar		0.0	0.0	0.0
506	tab temp pase		0.0	0.0	0.0
507	TABLA EMCI	136562	100.0	0.0	0.0
508	TABLA INGRESO	95638	0.0	70.1	0.0
509	TABLA NAVAL	65874	0.0	48.3	0.0
510	TABLA1	105875	0.0	77.6	0.0
511	TABLA2	125832	0.0	92.2	0.0
512	tablaz1		0.0	0.0	0.0
513	TALLA	169547	124.2	0.0	0.0
514	TALLAS MEDIDAS	109851	0.0	80.5	0.0
515	TB NOTRANSAC	68523	0.0	50.2	0.0
516	TB TRANSAC	65874	0.0	48.3	0.0
517	temp asc civ cal cur		0.0	0.0	0.0
518	TEMP ASC CIV PROM 1999	125873	0.0	92.2	0.0
519	TEMP ASC CIV PROM 2002	106985	0.0	78.4	0.0
520	TEMP ASCENSOS EMCI	112563	0.0	82.5	0.0
521	TEMP CEDULA	116850	0.0	85.6	0.0
522	Temp numero		0.0	0.0	0.0
523	temp per id coguar calif		0.0	0.0	0.0
524	temp reparto nuevo org		0.0	0.0	0.0
525	temp1		0.0	0.0	0.0
526	TEMPO		0.0	0.0	0.0
527	TEMPORAL		0.0	0.0	0.0
528	TEMPORAL		0.0	0.0	0.0
529	TEMPORAL OFICIALES		0.0	0.0	0.0
530	TIE REP OPE	110568	0.0	81.0	0.0
531	TIEM PER		0.0	0.0	0.0
532	tiem rep		0.0	0.0	0.0
533	TIEMPO EMBARQUE TOTAL	116875	0.0	85.6	0.0
534	TIEMPO REPARTO	152623	111.8	0.0	0.0
535	TIEMPO REPARTO TEMP		0.0	0.0	0.0
536	tiempo tripu	115269	0.0	84.4	0.0
537	TIEMPO1	98210	0.0	71.9	0.0
538	Tiempos Sectores		0.0	0.0	0.0
539	TIPO AUDITORIA	118526	0.0	86.8	0.0
540	TIPO BREVET	105875	0.0	77.6	0.0
541	TIPO CARGO	105698	0.0	77.4	0.0
542	TIPO CAUSA DECLARACION	105876	0.0	77.6	0.0
543	TIPO CONDECORACION	106325	0.0	77.9	0.0
544	TIPO CONTRATO		0.0	0.0	0.0
545	TIPO CUENTA	118265	0.0	86.6	0.0
546	TIPO CURSO	114753	0.0	84.1	0.0
547	TIPO CURSO EMCI	119854	0.0	87.8	0.0
548	TIPO DIV POL	96853	0.0	71.0	0.0
549	TIPO DOC	112587	0.0	82.5	0.0
550	TIPO DOCUMENTO DIGITAL	162653	119.2	0.0	0.0



TIPO EMPLEADO	125963	0.0	92.3	0.0
TIPO FALTA	290	0.0	0.0	0.2
TIPO INFORMACION	103681	0.0	76.0	0.0
TIPO INGENIO	258961	189.7	0.0	0.0
TIPO INSTRUCCION		0.0	0.0	0.0
TIPO JUICIO	105871	0.0	77.6	0.0
TIPO JUICIO anterior	119026	0.0	87.2	0.0
TIPO RANCHO	258745	189.6	0.0	0.0
TIPO REF LEGAL		0.0	0.0	0.0
TIPO RESPONSABILIDAD	253658	185.8	0.0	0.0
TIPO SANGRE	116845	0.0	85.6	0.0
TIPO UNI ADM	254154	186.2	0.0	0.0
tipoins		0.0	0.0	0.0
TITULO	116874	0.0	85.6	0.0
tmp ract sin historico		0.0	0.0	0.0
TOTAL EMBARQUE		0.0	0.0	0.0
total marosg		0.0	0.0	0.0
TRABAJO PROFES AYUDANTE	165878	121.5	0.0	0.0
TRANSACCION		0.0	0.0	0.0
TRANSACCION ASCENSOS	254713	186.6	0.0	0.0
TRANSACCION CONSULTA	165742	121.4	0.0	0.0
TRANSACCION DIARIA	103982	0.0	76.2	0.0
transaccion diaria historico		0.0	0.0	0.0
transaccion reporte		0.0	0.0	0.0
TRANSACCION ROLES	258456	189.3	0.0	0.0
TRANSACCION SEGURIDAD SYS	265485	194.5	0.0	0.0
TRANSACCION USUARIO	268426	196.6	0.0	0.0
trip detalle embarque		0.0	0.0	0.0
trip detalle embarque 22Nov04		0.0	0.0	0.0
trip detalle embarque2		0.0	0.0	0.0
trip detalle embarqueMRAMIREZ		0.0	0.0	0.0
trip detalle pasesHPEREZ		0.0	0.0	0.0
trip detalle pasesMRAMIREZ		0.0	0.0	0.0
trip no esta rep		0.0	0.0	0.0
trip no esta rep 2		0.0	0.0	0.0
trip no esta rep 22nov04		0.0	0.0	0.0
trip no esta rep copia		0.0	0.0	0.0
trip no esta repMRAMIREZ		0.0	0.0	0.0
TRIP SOLTERO	128956	0.0	94.5	0.0
trip tiempo repartoHPEREZ		0.0	0.0	0.0
trip tiempo repartoMRAMIREZ		0.0	0.0	0.0
TRIP TIEMPO SEGUIDO GYE	220454	161.5	0.0	0.0
TRIP TIEMPO SEGUIDO UIO		0.0	0.0	0.0
UBICACION LVN	256305	187.8	0.0	0.0
UNI ADM ESPRAN JARNAV	118759	0.0	87.0	0.0
UNI ADM PROFORMA	268743	196.9	0.0	0.0
UNI ADM PROFORMA copia me		0.0	0.0	0.0
UNI ADM PROFORMA QUITO	119625	0.0	87.6	0.0
UNI ADM SUBREP SUBDES	112587	0.0	82.5	0.0
UNI ADM SUELDOS	241368	176.8	0.0	0.0
UNI ADM SUINLI	254125	186.2	0.0	0.0
UNI ADM TEMPORAL		0.0	0.0	0.0
UNI ADM TEMPORAL AUX		0.0	0.0	0.0
UNI ADM TEMPORAL me		0.0	0.0	0.0
unidad adm prueba	135874	0.0	99.5	0.0
UNIDAD ADMimistrativa TEMPORAL		0.0	0.0	0.0
UNIDAD ADMimistrativa TEMPORAL me		0.0	0.0	0.0
UNIDAD ADMINISTRATIVA	245732	180.0	0.0	0.0
UNIDAD ADMINISTRATIVA 12 01	106874	0.0	78.3	0.0
UNIDAD ADMINISTRATIVA2	254158	186.2	0.0	0.0
unidades flote escuadron	102584	0.0	75.2	0.0
verifica pase	104753	0.0	76.7	0.0



**CIB-ESPOL**

613	xPLAZA_PROFORMA	265745	194.7	0.0	0.0
614	xUNI_ADM_PROFORMA	215874	158.1	0.0	0.0
<b>Promedios de escritura diaria</b>			<b>144.6</b>	<b>66.8</b>	<b>2.3</b>
<b>Total tablas:</b>			<b>135</b>	<b>237</b>	<b>42</b>

## ANEXOC

**NUMERO DE TRANSACCIONES DE LECTURA Y PROMEDIOS  
OBTENIDOS EN LA BASE DE DATOS PERSONAL (DIGPERGYE)**

Numero de dias laborados: 1365

No	NOMBRE TABLA	# TRANSACCIONES	PROMEDIO DE LECTURA		
			ALTAS	MEDIAS	BAJAS
1	A DISPOSICION		0.0	0.0	0.0
2	ACCION TOMADA		0.0	0.0	0.0
3	anos rep especifico		0.0	0.0	0.0
4	AÑO		0.0	0.0	0.0
5	ASCENSO CIVIL		0.0	0.0	0.0
6	ASCENSOS NAVALES		0.0	0.0	0.0
7	ASCENSOS SALIDAS ANUALES		0.0	0.0	0.0
8	AUTORIDAD REPORTES		0.0	0.0	0.0
9	AUTORIZACION MATRIMONIO		0.0	0.0	0.0
10	aux_cal	158750	116.3	0.0	0.0
11	aux_pase prest		0.0	0.0	0.0
12	auxiliar1	159872	117.1	0.0	0.0
13	auxiliar2		0.0	0.0	0.0
14	back_conde_ofi	156873	114.9	0.0	0.0
15	BACKUP COND HIST OFI 032003		0.0	0.0	0.0
16	BACKUP CONDEC ABRIL		0.0	0.0	0.0
17	BAJAS CI		0.0	0.0	0.0
18	BANCO		0.0	0.0	0.0
19	BANCO PERSONA		0.0	0.0	0.0
20	BASE CODES		0.0	0.0	0.0
21	BENEFICIO		0.0	0.0	0.0
22	BONIFICACION CURSOS EMCI		0.0	0.0	0.0
23	BONO ASCENSO		0.0	0.0	0.0
24	CAB PRESTAMO LVN		0.0	0.0	0.0
25	Calif Ayu RO GO EE		0.0	0.0	0.0
26	CALIFICACION ANUAL		0.0	0.0	0.0
27	CALIFICACION AYUDANTE	152684	111.9	0.0	0.0
28	CALIFICACION DIREDU	136203	99.8	0.0	0.0
29	CALIFICACION SEGURIDAD		0.0	0.0	0.0
30	CALIFICACION TRIPULANTES		0.0	0.0	0.0
31	CALIFICACIONtemp		0.0	0.0	0.0
32	CALIFICACIONtemp CIVILES		0.0	0.0	0.0
33	calificaciontemp cond2		0.0	0.0	0.0
34	CALIFICACIONTEMP UNIDADES		0.0	0.0	0.0
35	calificaciontempciviles		0.0	0.0	0.0
36	CANDIDATOS CURSOS ESP		0.0	0.0	0.0
37	CANDIDATOS EXP LAB		0.0	0.0	0.0
38	CANDIDATOS PREP ACAD		0.0	0.0	0.0
39	CANDIDATOS TRASBORDOS		0.0	0.0	0.0
40	CAPREMICI		0.0	0.0	0.0
41	CARGO		0.0	0.0	0.0
42	CATEGORIA		0.0	0.0	0.0
43	CAUSA A DISPOSICION		0.0	0.0	0.0
44	CAUSA ADICIONAL		0.0	0.0	0.0
45	CAUSA ASCENSO		0.0	0.0	0.0
46	CAUSA CESANTIA		0.0	0.0	0.0
47	CAUSA CONCESION PERMISO		0.0	0.0	0.0
48	CAUSA CONFIDENCIA		0.0	0.0	0.0
49	CAUSA CUOTA		0.0	0.0	0.0
50	CAUSA DISPONIBILIDAD		0.0	0.0	0.0
51	CAUSA EGRESO		0.0	0.0	0.0
52	CAUSA EXAMEN		0.0	0.0	0.0
53	CAUSA FINALIZACION		0.0	0.0	0.0
54	CAUSA INGRESO		0.0	0.0	0.0

56	CAUSA JUICIO		0.0	0.0	0.0
57	CAUSA JUICIO ANTERIOR		0.0	0.0	0.0
58	CAUSA RECONOCIMIENTO		0.0	0.0	0.0
59	CAUSA RETIRO		0.0	0.0	0.0
60	CAUSA SALIDA EXTERIOR	99256	72.7	0.0	0.0
61	CAUSA SERMAN		0.0	0.0	0.0
62	CAUSA SUBSIDIO		0.0	0.0	0.0
63	CAUSA SUBSISTE		0.0	0.0	0.0
64	CEDULA		0.0	0.0	0.0
65	CEDULAS		0.0	0.0	0.0
66	CEDULAS ORDENADO		0.0	0.0	0.0
67	CENSURA AYUDANTE		0.0	0.0	0.0
68	CENSURAS ASCENSO		0.0	0.0	0.0
69	CENTRO SALUD		0.0	0.0	0.0
70	ci prueba		0.0	0.0	0.0
71	CIRCULO		0.0	0.0	0.0
72	CLASE CURSO		0.0	0.0	0.0
73	CLASE GRADO		0.0	0.0	0.0
74	CLASE SANCION	458256	335.7	0.0	0.0
75	CLASIFICACION ESPECIALIDAD		0.0	0.0	0.0
76	CLIEN		0.0	0.0	0.0
77	CODIGO EMBARQUE		0.0	0.0	0.0
78	COLOR CABELLO		0.0	0.0	0.0
79	COLOR OJO		0.0	0.0	0.0
80	COLOR PIEL		0.0	0.0	0.0
81	COMANDO OPERATIVO		0.0	0.0	0.0
82	Compensacion Transitoria		0.0	0.0	0.0
83	COND RESP DIC	6958	0.0	0.0	5.1
84	cond trip marzo 2002	5875	0.0	0.0	4.3
85	CONDEC		0.0	0.0	0.0
86	condec emci		0.0	0.0	0.0
87	condec enero trip2003	56124	0.0	41.1	0.0
88	CONDEC INDIV		0.0	0.0	0.0
89	condec indiv grupo		0.0	0.0	0.0
90	CONDEC INDIV GRUPO2		0.0	0.0	0.0
91	CONDEC INDIV GRUPO3	65248	0.0	47.8	0.0
92	condec marz 2003		0.0	0.0	0.0
93	CONDEC ORDEN GENERAL		0.0	0.0	0.0
94	CONDEC PROY		0.0	0.0	0.0
95	CONDEC PROY WAY2		0.0	0.0	0.0
96	condec prueba	95847	70.2	0.0	0.0
97	CONDEC REC TIE SERVICIO		0.0	0.0	0.0
98	CONDEC SERV ANT		0.0	0.0	0.0
99	condecoraciones		0.0	0.0	0.0
100	condecoraciones emcis	92586	67.8	0.0	0.0
101	CONDECORACIONES EMCIS2		0.0	0.0	0.0
102	condecoraciones historico		0.0	0.0	0.0
103	condecoraciones proyeccion		0.0	0.0	0.0
104	condecoraciones respaldño	1256	0.0	0.0	0.9
105	CONDICION TRABAJO		0.0	0.0	0.0
106	cons	152854	112.0	0.0	0.0
107	CONSEJO		0.0	0.0	0.0
108	CONY SIN CODIGO ISSFA	96850	71.0	0.0	0.0
109	CONYUGUE		0.0	0.0	0.0
110	copia plazapp enero9		0.0	0.0	0.0
111	copia ppp enero9	96584	70.8	0.0	0.0
112	COPIA SUBJUDICE 23_02_2001		0.0	0.0	0.0
113	COPIA UNI ADM		0.0	0.0	0.0
114	cp 23jun03	66368	0.0	48.6	0.0
115	CSTO		0.0	0.0	0.0
116	cuadro condecoraciones	75824	55.5	0.0	0.0

1	ADRO CONDECORACIONES3		0.0	0.0	0.0
2	EMPLE MINIMO EMBARQUE		0.0	0.0	0.0
3	FOJA ELIMINACION		0.0	0.0	0.0
4	GR PROF		0.0	0.0	0.0
5	GRSO		0.0	0.0	0.0
6	GRSO ASCENSO AGREGADOS		0.0	0.0	0.0
7	GRSO CPNA		0.0	0.0	0.0
8	GRSO espec ayudante	105875	77.6	0.0	0.0
9	GRSO MANDO SUBM		0.0	0.0	0.0
10	GRSO PERSONA		0.0	0.0	0.0
11	CURSOS AGREGADOS		0.0	0.0	0.0
12	CURSOS ESPECIALES		0.0	0.0	0.0
13	CURSOS FALLIDOS PROFORMA		0.0	0.0	0.0
14	CURSOS REALIZADOS		0.0	0.0	0.0
15	D OPERACION FORMASYS	589632	432.0	0.0	0.0
16	D PERFIL OPE SYS		0.0	0.0	0.0
17	DATOS FECHA CAMBIADOS		0.0	0.0	0.0
18	DECLARACION		0.0	0.0	0.0
19	DECLARACION REPORTE		0.0	0.0	0.0
20	DEMERITO GRADO AYUDANTE		0.0	0.0	0.0
21	DEMERITO PRISION		0.0	0.0	0.0
22	demerito temp ayudante		0.0	0.0	0.0
23	demeritos subm temp		0.0	0.0	0.0
24	DEMERITOS TEMP		0.0	0.0	0.0
25	DEMERITOS TEMP AGUENA		0.0	0.0	0.0
26	desapare 1		0.0	0.0	0.0
27	desapare 2		0.0	0.0	0.0
28	desapare 28022002		0.0	0.0	0.0
29	desapare real		0.0	0.0	0.0
30	DESAPARECIDO		0.0	0.0	0.0
31	DESIGNA AYUDANTE		0.0	0.0	0.0
32	DET PRESTAMIO TVN		0.0	0.0	0.0
33	DETALLE RECÓN PROCESO	587562	430.4	0.0	0.0
34	DEVENGAR ESTUDIO		0.0	0.0	0.0
35	DIPLOMADO		0.0	0.0	0.0
36	DIRECC		0.0	0.0	0.0
37	DISCAPACIDAD	875251	641.2	0.0	0.0
38	DISPONIBILIDAD		0.0	0.0	0.0
39	DIV POL ECUA		0.0	0.0	0.0
40	DOCUMENTOS VIDA NAVAL		0.0	0.0	0.0
41	diproperties	598688	438.6	0.0	0.0
42	EMBAR EFECTIVO		0.0	0.0	0.0
43	embarque	487598	357.2	0.0	0.0
44	embarque 2	598524	438.5	0.0	0.0
45	embarque bonif		0.0	0.0	0.0
46	embarque candi ofi		0.0	0.0	0.0
47	embarque cond		0.0	0.0	0.0
48	embarque cond trip		0.0	0.0	0.0
49	embarque cond666	568256	416.3	0.0	0.0
50	embarque cond666 tripu	589548	431.9	0.0	0.0
51	embarque detalle	524514	384.3	0.0	0.0
52	embarque detalle tripu	589647	432.0	0.0	0.0
53	EMBARQUE IMI		0.0	0.0	0.0
54	embarque moran	587520	430.4	0.0	0.0
55	embarque1	599820	439.4	0.0	0.0
56	embarque2		0.0	0.0	0.0
57	embarque2 copia		0.0	0.0	0.0
58	embarque3		0.0	0.0	0.0
59	EMBARQUE3 COPIA		0.0	0.0	0.0
60	EMCIS CALIFICACION BAJA		0.0	0.0	0.0
61	EMCIS PROM ASCENSOS		0.0	0.0	0.0

179	emcis_prom ascensos antes		0.0	0.0	0.0
180	emcis_prom ascensos2	156987	115.0	0.0	0.0
181	emcis temporal	568254	416.3	0.0	0.0
182	ENFERMEDAD	195875	143.5	0.0	0.0
183	error migra pz		0.0	0.0	0.0
184	error migra pz copia		0.0	0.0	0.0
185	ESPECIALIDAD		0.0	0.0	0.0
186	ESPECIALIDAD EQUIV	168532	123.5	0.0	0.0
187	ESPERA CESANTIA		0.0	0.0	0.0
188	Estadis Tiem Servicio		0.0	0.0	0.0
189	ESTADO CIVIL		0.0	0.0	0.0
190	ETAPA		0.0	0.0	0.0
191	EVA CIVIL RA	698526	511.7	0.0	0.0
192	EVAL CUA ATRIBUTOS	625874	458.5	0.0	0.0
193	EVALUACION	691478	506.6	0.0	0.0
194	EVALUACION AGREGADOS		0.0	0.0	0.0
195	EVALUACION AGUENA		0.0	0.0	0.0
196	EVALUACION AYUDANTE		0.0	0.0	0.0
197	EVALUACION CPNV		0.0	0.0	0.0
198	EVALUACION CURSO AGREGADOS		0.0	0.0	0.0
199	EVALUACION FALTAS AGREGADOS		0.0	0.0	0.0
200	EVALUACION NOTAS AGREGADOS		0.0	0.0	0.0
201	EVALUACION RES 01 05 00		0.0	0.0	0.0
202	EVALUACION SALIDAS EXTERIOR	125698	92.1	0.0	0.0
203	EVALUACION SUBM	548753	402.0	0.0	0.0
204	EXAMEN MEDICO		0.0	0.0	0.0
205	EXCEPCION REQUISITO	105874	77.6	0.0	0.0
206	EXPERIENCIA LABORAL		0.0	0.0	0.0
207	FALTA SANCION		0.0	0.0	0.0
208	FALTA SANCION NOVEDADES		0.0	0.0	0.0
209	FALTAS	568720	416.6	0.0	0.0
210	faltas grado	658322	482.3	0.0	0.0
211	FICHA INDIVIDUAL		0.0	0.0	0.0
212	fonifa		0.0	0.0	0.0
213	formulario issfa	125873	92.2	0.0	0.0
214	FORMULARIO ISSFA T		0.0	0.0	0.0
215	FOTOGRAFIA		0.0	0.0	0.0
216	fuera organico temporal	569320	417.1	0.0	0.0
217	GORRA		0.0	0.0	0.0
218	GORRA I		0.0	0.0	0.0
219	GRADO	356875	261.4	0.0	0.0
220	GRADO CURSO	156987	115.0	0.0	0.0
221	GRADO DISCAPACIDAD	95687	70.1	0.0	0.0
222	GRADO POR PASE	365255	267.6	0.0	0.0
223	grados persona sueldos 04	156852	114.9	0.0	0.0
224	grados persona sueldos 05	70256	51.5	0.0	0.0
225	grados persona sueldos 06		0.0	0.0	0.0
226	GRUPO ESPECIALIDAD	135201	99.0	0.0	0.0
227	GRUPO MEDICO		0.0	0.0	0.0
228	GRUPO PERSONA	154235	113.0	0.0	0.0
229	Grupos Directiva		0.0	0.0	0.0
230	HI		0.0	0.0	0.0
231	HIJO		0.0	0.0	0.0
232	HIJOS DIFERENTES MADRES		0.0	0.0	0.0
233	hijos prueba		0.0	0.0	0.0
234	HIJOS SIN CODIGO ISSFA		0.0	0.0	0.0
235	HIJOS SIN CODIGO ISSFA OFI		0.0	0.0	0.0
236	HIJOS SIN CODIGO ISSFA TRIP		0.0	0.0	0.0
237	HIS CONTRATO		0.0	0.0	0.0
238	hist condec ofi ene 2003		0.0	0.0	0.0
239	HISTORICO CAL TRI		0.0	0.0	0.0
240	HISTORICO CONDECORACIONES		0.0	0.0	0.0

241	HISTORICO DIPLOMADO		0.0	0.0	0.0
242	HISTORICO ESPECIALIDAD		0.0	0.0	0.0
243	HISTORICO FORMACION		0.0	0.0	0.0
244	HISTORICO PERF		0.0	0.0	0.0
245	HISTORICO PERSONAL		0.0	0.0	0.0
246	HISTORICO SUBJUDICE		0.0	0.0	0.0
247	HISTORICO TRANSACCION		0.0	0.0	0.0
248	HOSPITALIZACION		0.0	0.0	0.0
249	IDS	159253	116.7	0.0	0.0
250	INDIV COND EMCI		0.0	0.0	0.0
251	INDIV2		0.0	0.0	0.0
252	INDIV3		0.0	0.0	0.0
253	INFORMACION ADICIONAL	568250	416.3	0.0	0.0
254	INFORMACION RESERVADA	958785	702.4	0.0	0.0
255	INFORMACION SUMARIA		0.0	0.0	0.0
256	INGENIO PERSONA	458695	336.0	0.0	0.0
257	INSTITUTOS MILITARES	103265	75.7	0.0	0.0
258	INSTRUCCION CIVIL		0.0	0.0	0.0
259	interes curso	152304	111.6	0.0	0.0
260	ISSFA CONYUGUE		0.0	0.0	0.0
261	ISSFA ESPOSA		0.0	0.0	0.0
262	ISSFA HIJOS		0.0	0.0	0.0
263	ISSFA PADRES		0.0	0.0	0.0
264	ISSFA TIPO HIJO	958258	702.0	0.0	0.0
265	ISSFA TITULAR		0.0	0.0	0.0
266	ito	168523	123.5	0.0	0.0
267	JUZGADO		0.0	0.0	0.0
268	JUZGADO ANTERIOR		0.0	0.0	0.0
269	LICENCIA PERSONAL		0.0	0.0	0.0
270	LICENCIA PROFORMA	895623	656.1	0.0	0.0
271	LICENCIA PROFORMA NOV		0.0	0.0	0.0
272	LICENCIA PROFORMA TEMP		0.0	0.0	0.0
273	LICENCIA TEMPORAL		0.0	0.0	0.0
274	LISTA CALIFICACION	658250	482.2	0.0	0.0
275	MATERIA AYUDANTIA	458785	336.1	0.0	0.0
276	MAXFEC	365856	268.0	0.0	0.0
277	MEDIDAS		0.0	0.0	0.0
278	MERITO AYUDANTE	568425	416.4	0.0	0.0
279	MERITO TEMP AYUDANTE	569220	417.0	0.0	0.0
280	MERITOS ASCENSO		0.0	0.0	0.0
281	MERITOS SUBM TEMP	158485	116.1	0.0	0.0
282	MERITOS TEMP	168523	123.5	0.0	0.0
283	MERITOS TEMP AGUENA	96520	70.7	0.0	0.0
284	MESES	149536	109.6	0.0	0.0
285	MODIF PERSONA TIPO DOC		0.0	0.0	0.0
286	moran 2002	56985	0.0	41.7	0.0
287	MORANI 2002	52658	0.0	38.6	0.0
288	moran2 2002	59874	0.0	43.9	0.0
289	MOT FIN CON		0.0	0.0	0.0
290	movimientotemp	69852	51.2	0.0	0.0
291	MULTIPLES EMBARQUE		0.0	0.0	0.0
292	NEW GRADE	95863	70.2	0.0	0.0
293	NIVEL		0.0	0.0	0.0
294	NIVEL CONESUP		0.0	0.0	0.0
295	NIVEL INST CIVIL		0.0	0.0	0.0
296	NIVEL REQUI		0.0	0.0	0.0
297	no encuentra_pz	96854	71.0	0.0	0.0
298	NOMBRE	948753	695.1	0.0	0.0
299	NOVEDAD ASCENSO		0.0	0.0	0.0
300	novedad ascenso 3 mayo	58632	0.0	43.0	0.0
301	NOVEDAD ASCENSO RESP0501		0.0	0.0	0.0
302	NOVEDAD AYUDANTE		0.0	0.0	0.0

303	NOVEDAD CURSO REPLICADO		0.0	0.0	0.0
304	NUMERO AFILIACION ISSFA TEMP		0.0	0.0	0.0
305	OFICIALES PASE	985320	721.8	0.0	0.0
306	OPCION SISTEMA		0.0	0.0	0.0
307	OPCION TIPO DOC	52685	0.0	38.6	0.0
308	OPCIONES INTEGRACION		0.0	0.0	0.0
309	OPERACION FORMA		0.0	0.0	0.0
310	ORGANICO		0.0	0.0	0.0
311	organicoyefectivo	568525	416.5	0.0	0.0
312	ORGANICOYEFFECTIVO_TRIPU		0.0	0.0	0.0
313	OrganicoyEfectivoMaugi	658230	482.2	0.0	0.0
314	otros	1252	0.0	0.0	0.9
315	otrosids		0.0	0.0	0.0
316	PADRE		0.0	0.0	0.0
317	PADRES ISSFA 2001		0.0	0.0	0.0
318	PADRES ISSFA 2002		0.0	0.0	0.0
319	PADRES ISSFA 2003		0.0	0.0	0.0
320	PADRES SIN CODIGO		0.0	0.0	0.0
321	padres1	26856	0.0	19.7	0.0
322	padres2	25698	0.0	18.8	0.0
323	PAGO CONDEC	569528	417.2	0.0	0.0
324	PAIS	65856	0.0	48.2	0.0
325	PARAMETRO EVAL AYUDANTE		0.0	0.0	0.0
326	PARAMETROS EVALUACION		0.0	0.0	0.0
327	PARAMETROS EVALUACION AYUDANTE		0.0	0.0	0.0
328	PARAMETROS PROCESO PASES		0.0	0.0	0.0
329	PARTIDA PRESUPUESTO		0.0	0.0	0.0
330	Pasaporte Visas	96854	71.0	0.0	0.0
331	PASE		0.0	0.0	0.0
332	pase prest temp	45630	0.0	33.4	0.0
333	PASE TEMPORAL	56874	0.0	41.7	0.0
334	PASE ULTIMO		0.0	0.0	0.0
335	PASES FALLIDOS PROFORMA		0.0	0.0	0.0
336	pases_im		0.0	0.0	0.0
337	pases_im otros	42587	0.0	31.2	0.0
338	PASES OFICIALES	48751	0.0	35.7	0.0
339	pases provisionales	56874	0.0	41.7	0.0
340	PER CONT NO PROF		0.0	0.0	0.0
341	per_id	985633	722.1	0.0	0.0
342	PER ID EFEC CAP		0.0	0.0	0.0
343	per_id salen Cap		0.0	0.0	0.0
344	per_id2		0.0	0.0	0.0
345	PER ID3		0.0	0.0	0.0
346	per maugi	56856	0.0	41.7	0.0
347	per no sale_pz	45875	0.0	33.6	0.0
348	per pase especifico	569856	417.5	0.0	0.0
349	PER PRO CUR COPIA		0.0	0.0	0.0
350	per prof pas copia		0.0	0.0	0.0
351	per sale_pz_gye		0.0	0.0	0.0
352	per sin fec efe	598630	438.6	0.0	0.0
353	PERCON	521785	382.3	0.0	0.0
354	PERFECCIONAMIENTO	958203	702.0	0.0	0.0
355	PERFIL AUTORIZARPT_SYS		0.0	0.0	0.0
356	PERFIL CONSULTA DOCU_SYS		0.0	0.0	0.0
357	PERFIL CONSULTA GRADO_SYS		0.0	0.0	0.0
358	PERFIL CONSULTA_SYS		0.0	0.0	0.0
359	PERFIL GRUPO_SYS		0.0	0.0	0.0
360	PERFIL OPERACION_SYS		0.0	0.0	0.0
361	PERFIL PLAZA		0.0	0.0	0.0
362	PERIODO ASCENSO	15875	0.0	0.0	11.6
363	PERIODO AYUDANTE		0.0	0.0	0.0
364	PERMANENCIA GRADO	658253	482.2	0.0	0.0



365	PERMANENCIA OPERATIVA		0.0	0.0	0.0
366	PERMANENCIA OPERATIVA TEMP		0.0	0.0	0.0
367	PERMISO OPCION OBJETO SYS		0.0	0.0	0.0
368	PERSONA	1589784	1164.7	0.0	0.0
369	PERSONA VACANTE ASC	125874	92.2	0.0	0.0
370	PERSONACI		0.0	0.0	0.0
371	personal inteligencia		0.0	0.0	0.0
372	PERSONAL PROFORMA CURSOS		0.0	0.0	0.0
373	PERSONAL PROFORMA PASES		0.0	0.0	0.0
374	pla iden		0.0	0.0	0.0
375	pla_pro_ofi_35	458745	336.1	0.0	0.0
376	PLAZA ORG TEMPORAL		0.0	0.0	0.0
377	PLAZA ORG TEMPORAL AUX		0.0	0.0	0.0
378	PLAZA ORG TEMPORAL_me		0.0	0.0	0.0
379	PLAZA ORGANICA		0.0	0.0	0.0
380	PLAZA ORGANICA 12 01	56985	0.0	41.7	0.0
381	plaza organica 22 ENE	69856	51.2	0.0	0.0
382	plaza organica 27 DIC	62587	0.0	45.9	0.0
383	plaza organica prueba	62856	0.0	46.0	0.0
384	PLAZA PROFORMA		0.0	0.0	0.0
385	PLAZA PROFORMA copia me		0.0	0.0	0.0
386	PLAZA PROFORMA PASES		0.0	0.0	0.0
387	plaza_proforma_pases_14ene04		0.0	0.0	0.0
388	plaza_proforma_pases_67		0.0	0.0	0.0
389	plaza_proforma_pases_68		0.0	0.0	0.0
390	PLAZA PROFORMA QUITO		0.0	0.0	0.0
391	plaza temporal	78526	57.5	0.0	0.0
392	plaza temporal me		0.0	0.0	0.0
393	plazas proyeccion	87524	64.1	0.0	0.0
394	po temp ofi actual	89568	65.6	0.0	0.0
395	po temp ofi nuevo	82574	60.5	0.0	0.0
396	PREPARACION ACADEMICA		0.0	0.0	0.0
397	PRESTACION SERVICIOS		0.0	0.0	0.0
398	PROCESO CANDIDATOS	45875	0.0	33.6	0.0
399	PROFES		0.0	0.0	0.0
400	proforma	85968	63.0	0.0	0.0
401	PROFORMA CURSOS		0.0	0.0	0.0
402	proforma disponibles emci		0.0	0.0	0.0
403	proforma disponibles ofi		0.0	0.0	0.0
404	proforma disponibles trip		0.0	0.0	0.0
405	PROFORMA PASES		0.0	0.0	0.0
406	proforma vacantes ofi		0.0	0.0	0.0
407	PROM GRADUACION		0.0	0.0	0.0
408	promedio cal		0.0	0.0	0.0
409	PROMEDIO CAL ADJUNTOS		0.0	0.0	0.0
410	PROMEDIO CAL AGUENA		0.0	0.0	0.0
411	PROMEDIO CAL AYUDANTE		0.0	0.0	0.0
412	PROMEDIO CAL CPNV		0.0	0.0	0.0
413	promedio cal emcis	85746	62.8	0.0	0.0
414	PROMEDIO CAL EVALUACION		0.0	0.0	0.0
415	PROMEDIO CAL EVALUACION TRIPU		0.0	0.0	0.0
416	promedio cal2	89784	65.8	0.0	0.0
417	PROMEDIO CALIFICACION TRIPU	785896	575.7	0.0	0.0
418	PROMOCION ASCENSO		0.0	0.0	0.0
419	PROYECCION ASC		0.0	0.0	0.0
420	prueba embarque	78589	57.6	0.0	0.0
421	PRUEBA FISICA		0.0	0.0	0.0
422	prueba seguridad1	1278	0.0	0.0	0.9
423	PRY VACANTES		0.0	0.0	0.0
424	PUESTO		0.0	0.0	0.0
425	Puesto 05 01 2000		0.0	0.0	0.0
426	PUESTO OCUPADO		0.0	0.0	0.0



**CIB-ESPOL**

427	PUESTO QUITO		0.0	0.0	0.0
428	RAMA		0.0	0.0	0.0
429	REC TIEM SERVICIO		0.0	0.0	0.0
430	REC TIEM SERVICIO TEMP	153628	112.5	0.0	0.0
431	RECOMPENSA		0.0	0.0	0.0
432	REFERENCIA LEGAL		0.0	0.0	0.0
433	REGION PLAZA		0.0	0.0	0.0
434	REGLAMENTO REPARTO OPERATIVO		0.0	0.0	0.0
435	rep anterior	198235	145.2	0.0	0.0
436	reparto operativo	204582	149.9	0.0	0.0
437	REPARTO OPERATIVO FECHA VAL		0.0	0.0	0.0
438	REPARTO PADRE		0.0	0.0	0.0
439	REPARTO PADRE NUEVO ORG		0.0	0.0	0.0
440	repartos persona sueldos 04		0.0	0.0	0.0
441	repartos persona sueldos 05	65859	0.0	48.2	0.0
442	repartos persona sueldos 06	98587	72.2	0.0	0.0
443	REPORTE PASES		0.0	0.0	0.0
444	reportes	652875	478.3	0.0	0.0
445	REQUISITO PERSONA		0.0	0.0	0.0
446	REQUISITOS		0.0	0.0	0.0
447	REQUISITOS ASCENSO		0.0	0.0	0.0
448	REQUISITOS CURSOS PUESTO	85748	62.8	0.0	0.0
449	REQUISITOS PUESTOS PUESTO	95874	70.2	0.0	0.0
450	RESOLUCION CONSEJO		0.0	0.0	0.0
451	RESOLUCION PERSONA		0.0	0.0	0.0
452	RESOLUCION PERSONA TEMP		0.0	0.0	0.0
453	resp pla org 06 06 2000		0.0	0.0	0.0
454	resp pla org 21 06 2000		0.0	0.0	0.0
455	resp puesto 21 06 2000		0.0	0.0	0.0
456	resp uni adm 21 06 2000		0.0	0.0	0.0
457	respa condec		0.0	0.0	0.0
458	respaldo condec 20012003	159875	117.1	0.0	0.0
459	respaldo condec trip dic 2002	152698	111.9	0.0	0.0
460	RESPALDO CONDECORACIONES		0.0	0.0	0.0
461	respaldo condecoraciones2	136985	100.4	0.0	0.0
462	Resultados	152875	112.0	0.0	0.0
463	resumen efectivos plazas	195685	143.4	0.0	0.0
464	RETMIL		0.0	0.0	0.0
465	RPT EVALUACION	150258	110.1	0.0	0.0
466	RPT VACANTES	156236	114.5	0.0	0.0
467	SALIDA EXTERIOR		0.0	0.0	0.0
468	SALTOS DIGPER	56892	0.0	41.7	0.0
469	SALTOS ESDEIM	85985	63.0	0.0	0.0
470	SANCION JUICIO		0.0	0.0	0.0
471	SECTOR ASCENSO CEREMONIA		0.0	0.0	0.0
472	SELECCION AGREGADOS		0.0	0.0	0.0
473	SELECCION AYUDANTE		0.0	0.0	0.0
474	SELECCION EMBARQUE		0.0	0.0	0.0
475	SELECCION EMBARQUE OFI		0.0	0.0	0.0
476	seleccion embarque ofi temp	658562	482.5	0.0	0.0
477	SELECCION LLAMADA		0.0	0.0	0.0
478	SELECCION LLAMADA CIVILES		0.0	0.0	0.0
479	SELECCION LLAMADA MORAN		0.0	0.0	0.0
480	SELECCION LLAMADA RESP0501		0.0	0.0	0.0
481	SELECCION PERSONAL PASES		0.0	0.0	0.0
482	SELECCION PLAZAS PASES		0.0	0.0	0.0
483	SERVICIO ANTERIOR		0.0	0.0	0.0
484	SEXO	856854	627.7	0.0	0.0
485	SIN PLAZA	458256	335.7	0.0	0.0
486	SISTEMA SYS		0.0	0.0	0.0
487	SOCIO		0.0	0.0	0.0
488	SUBJ PER		0.0	0.0	0.0

489	SUBJUDICE		0.0	0.0	0.0
490	SUBSISTE		0.0	0.0	0.0
491	t_persona		0.0	0.0	0.0
492	t_reparto		0.0	0.0	0.0
493	t01ago	1254	0.0	0.0	0.9
494	t01ene	1548	0.0	0.0	1.1
495	t01feb	1857	0.0	0.0	1.4
496	t01jul	1875	0.0	0.0	1.4
497	t01mar	1485	0.0	0.0	1.1
498	t01nov	1358	0.0	0.0	1.0
499	t01oct	1587	0.0	0.0	1.2
500	t01sep	1025	0.0	0.0	0.8
501	t16dic	1596	0.0	0.0	1.2
502	t16ene	1526	0.0	0.0	1.1
503	t16feb	1524	0.0	0.0	1.1
504	t16jun	2653	0.0	0.0	1.9
505	t16mar	1859	0.0	0.0	1.4
506	tab_temp_pase	1698	0.0	0.0	1.2
507	TABLA EMCI		0.0	0.0	0.0
508	TABLA_INGRESO		0.0	0.0	0.0
509	TABLA_NAVAL		0.0	0.0	0.0
510	TABLA1		0.0	0.0	0.0
511	TABLA2		0.0	0.0	0.0
512	tablaz1	158752	116.3	0.0	0.0
513	TALLA		0.0	0.0	0.0
514	TALLAS MEDIDAS		0.0	0.0	0.0
515	TB NOTRANSAC		0.0	0.0	0.0
516	TB TRANSAC		0.0	0.0	0.0
517	temp_asc_civ_cal_cur	96258	70.5	0.0	0.0
518	TEMP_ASC_CIV_PROM_1999		0.0	0.0	0.0
519	TEMP_ASC_CIV_PROM_2002		0.0	0.0	0.0
520	TEMP_ASCENSOS EMCI		0.0	0.0	0.0
521	TEMP CEDULA		0.0	0.0	0.0
522	Temp_numero	6857	0.0	0.0	5.0
523	temp_per_id_coguar_calif	1586	0.0	0.0	1.2
524	temp_reparto_nuevo_org	1695	0.0	0.0	1.2
525	templ	1785	0.0	0.0	1.3
526	TEMPO	1203	0.0	0.0	0.9
527	TEMPORAL	1502	0.0	0.0	1.1
528	TEMPORAL	1475	0.0	0.0	1.1
529	TEMPORAL OFICIALES	1830	0.0	0.0	1.3
530	TIE REP OPE		0.0	0.0	0.0
531	TIEM PER	1502	0.0	0.0	1.1
532	tiem_rep	1428	0.0	0.0	1.0
533	TIEMPO EMBARQUE TOTAL		0.0	0.0	0.0
534	TIEMPO REPARTO		0.0	0.0	0.0
535	TIEMPO REPARTO TEMP	189563	138.9	0.0	0.0
536	tiempo_tripu		0.0	0.0	0.0
537	TIEMPO1		0.0	0.0	0.0
538	Tiempos Sectores	152623	111.8	0.0	0.0
539	TIPO AUDITORIA		0.0	0.0	0.0
540	TIPO BREVET		0.0	0.0	0.0
541	TIPO CARGO		0.0	0.0	0.0
542	TIPO CAUSA DECLARACION		0.0	0.0	0.0
543	TIPO CONDECORACION		0.0	0.0	0.0
544	TIPO CONTRATO	195874	143.5	0.0	0.0
545	TIPO CUENTA		0.0	0.0	0.0
546	TIPO CURSO		0.0	0.0	0.0
547	TIPO CURSO EMCI		0.0	0.0	0.0
548	TIPO DIV POL		0.0	0.0	0.0
549	TIPO DOC		0.0	0.0	0.0
550	TIPO DOCUMENTO DIGITAL		0.0	0.0	0.0



**CIB-ESPOL**

551	TIPO EMPLEADO		0.0	0.0	0.0
552	TIPO FALTA		0.0	0.0	0.0
553	TIPO INFORMACION		0.0	0.0	0.0
554	TIPO INGENIO		0.0	0.0	0.0
555	TIPO INSTRUCCION	168759	123.6	0.0	0.0
556	TIPO JUICIO		0.0	0.0	0.0
557	TIPO JUICIO anterior		0.0	0.0	0.0
558	TIPO RANCHO		0.0	0.0	0.0
559	TIPO REF LEGAL	145254	106.4	0.0	0.0
560	TIPO RESPONSABILIDAD		0.0	0.0	0.0
561	TIPO SANGRE		0.0	0.0	0.0
562	TIPO UNI ADM		0.0	0.0	0.0
563	tipoins	56845	0.0	41.6	0.0
564	TITULO		0.0	0.0	0.0
565	tmp ract sin historico	56874	0.0	41.7	0.0
566	TOTAL EMBARQUE	857485	628.2	0.0	0.0
567	total marosg	478223	350.3	0.0	0.0
568	TRABAJO PROFES AYUDANTE		0.0	0.0	0.0
569	TRANSACCION	872325	639.1	0.0	0.0
570	TRANSACCION ASCENSOS		0.0	0.0	0.0
571	TRANSACCION CONSULTA		0.0	0.0	0.0
572	TRANSACCION DIARIA		0.0	0.0	0.0
573	transaccion diaria historico	45872	0.0	33.6	0.0
574	transaccion reporte	87526	64.1	0.0	0.0
575	TRANSACCION ROLES		0.0	0.0	0.0
576	TRANSACCION SEGURIDAD SYS		0.0	0.0	0.0
577	TRANSACCION USUARIO		0.0	0.0	0.0
578	trip detalle embarque	165852	121.5	0.0	0.0
579	trip detalle embarque 22Nov04	152022	111.4	0.0	0.0
580	trip detalle embarque2	145852	106.9	0.0	0.0
581	trip detalle embarqueMRAMIREZ	5620	0.0	0.0	4.1
582	trip detalle pasesHPEREZ	5874	0.0	0.0	4.3
583	trip detalle pasesMRAMIREZ	8562	0.0	0.0	6.3
584	trip no esta rep	1254	0.0	0.0	0.9
585	trip no esta rep 2	3568	0.0	0.0	2.6
586	trip no esta rep 22nov04	32541	0.0	23.8	0.0
587	trip no esta rep copia	1542	0.0	0.0	1.1
588	trip no esta repMRAMIREZ	1524	0.0	0.0	1.1
589	TRIP SOLTERO		0.0	0.0	0.0
590	trip tiempo repartoHPEREZ	5874	0.0	0.0	4.3
591	trip tiempo repartoMRAMIREZ	4582	0.0	0.0	3.4
592	TRIP TIEMPO SEGUIDO GYE		0.0	0.0	0.0
593	TRIP TIEMPO SEGUIDO UIO	8547	0.0	0.0	6.3
594	UBICACION LVN		0.0	0.0	0.0
595	UNI ADM ESPRAN JARNAV		0.0	0.0	0.0
596	UNI ADM PROFORMA		0.0	0.0	0.0
597	UNI ADM PROFORMA copia me	56874	0.0	41.7	0.0
598	UNI ADM PROFORMA QUITO		0.0	0.0	0.0
599	UNI ADM SUBREP SUBDES		0.0	0.0	0.0
600	UNI ADM SUELDOS		0.0	0.0	0.0
601	UNI ADM SUINLI		0.0	0.0	0.0
602	UNI ADM TEMPORAL	1523	0.0	0.0	1.1
603	UNI ADM TEMPORAL AUX	3568	0.0	0.0	2.6
604	UNI ADM TEMPORAL me	4587	0.0	0.0	3.4
605	unidad adm prueba		0.0	0.0	0.0
606	UNidad ADMministrativa TEMPORAL	9587	0.0	0.0	7.0
607	UNidad ADMministrativa TEMPORAL me	4857	0.0	0.0	3.6
608	UNIDAD ADMINISTRATIVA		0.0	0.0	0.0
609	UNIDAD ADMINISTRATIVA 12 01		0.0	0.0	0.0
610	UNIDAD ADMINISTRATIVA2		0.0	0.0	0.0
611	unidades flote escuadron		0.0	0.0	0.0
612	verifica pase		0.0	0.0	0.0

3	xPLAZA_PROFORMA		0.0	0.0	0.0
4	xUNI_ADM_PROFORMA		0.0	0.0	0.0
	<b>Promedios de lectura diaria</b>		<b>258.8</b>	<b>38.9</b>	<b>2.4</b>
	<b>Total tablas</b>		<b>127</b>	<b>29</b>	<b>45</b>



**CIB-ESPOL**

# ANEXO D

## NUMERO DE REGISTROS Y NUMERO DE COLUMNAS OBTENIDOS EN LA BASE DE DATOS SUELDOS

No	NOMBRE TABLA	# REGISTROS	# DE COLUMNAS
1	CONTADOR_TABLAS	152	3
2	dtproperties	452	7
3	rol_periodo	369	11
4	alcance_historico	140278	4
5	BENEFICIARIO	6150	4
6	BONO_TRIP	507	13
7	CARGA	1567303	1
8	cedulas	5576	1
9	cedulas_tot	877	11
10	Compensacion_Transitoria	163	8
11	CONDECORACION	253	17
12	CUADRO_DIFERENCIA	77	7
13	DETALLE	13758	7
14	detalle_retencion	11075	18
15	documento	18	18
16	EMPRESA	2	18
17	ESTADO_CIVIL	3	18
18	ESTADO_PERIODO	5	18
19	form_SRI	9051	18
20	form_SRI_TMP	9051	19
21	form_SRI_TMP_2004	1	18
22	form_SRI_TMP_2004_2	1	19
23	form_SRI_TMP_2004_3	1	18
24	form_SRI_TMP_2004_4	1	18
25	form_SRI_TMP_2004_OFC	879	23
26	form_SRI_TMP_2004_TMP	1	6
27	form_SRI_TMP_2004_TRIP	8314	16
28	form_SRI_TMP10	1	1
29	form_SRI_TMP2	1	8
30	FORMA_PAGO	6	8
31	formulario	1	8
32	GRADO	36	8
33	GRUPO_USUARIO	7	9
34	HISTORICO_CONTRATOS	712	8
35	HISTORICO_GASTOREP	1	8
36	IDS	256	8
37	imp_renta_2001	11338	8
38	imp_renta_2001_Anterior	11338	8
39	imp_renta_2002	11234	8
40	imp_renta_2002_Anterior	11234	8
41	imp_renta_2002_Audqui	152129	8
42	imp_renta_2002_New	11234	8

43	imp_renta_2005	12394	17
44	imp_renta_2004	12846	16
45	imp_renta_2004_anterior	12747	16
46	imp_renta_2004_Bonos	95	20
47	imp_renta_2004_Condecoracion	1	9
48	imp_renta_2004_General	1	3
49	imp_renta_2005_Condecoracion	3	13
50	imp_renta_2005_General	1642	7
51	impuesto_renta	11338	20
52	impuesto_renta_2000	2411	9
53	impuesto_rentaI_2000	8	10
54	INGRESO_VARIO	579	15
55	INSTITUCION_BENEFICIARIA	1	9
56	LOTE	2183	1
57	nombres	841	6
58	OPCION_SISTEMA	128	6
59	OPCION_SISTEMA1	102	9
60	OPERACION_FORMA	1	6
61	OTRO_RUBRO	9	7
62	PAGADURIA	8	12
63	PAGO_PROVISION	5	33
64	PARAMETRO_HISTORICO	132	33
65	PARAMETRO_RUBRO	4744	2
66	PARAMETRO_SISTEMA	1	27
67	parametros_sep	46	10
68	PER_ID3	5817	11
69	PERFIL_AUTORIZARPT_SYS	1	14
70	PERFIL_CONSULTA_SYS	1	16
71	PERFIL_GRUPO	111	11
72	PERFIL_GRUPO_SYS	186	11
73	PERFIL_OPERACION_SYS	1	20
74	PERIODO_ROL	219	20
75	PERIODOS_DIFERENCIA	1	20
76	PERMISO_OPCION_OBJETO_SY	405	16
77	PERSONA	33016	16
78	persona_bk	21491	16
79	PERSONA_BK0319	22253	20
80	persona_fechas	9139	16
81	PERSONA_HISTORICO	871691	16
82	PERSONAL_ASCENDIDO	1	16
83	PLAZA	102	3
84	PROVISION	1	1
85	RANCHO	4	1
86	REPARTO	18471	7
87	Results	2000000	5
88	rol_diferencia	0	4
89	ROL_HISTORICO	28662483	2
90	rol_periodo	214379	2
91	rol_prueba	8733	2
92	RUBRO	364	2

93	RUBRO_decimo	903	9
94	RUBRO_DIFERENCIA	2114	16
95	RUBRO_GASTOREP	1228	13
96	rubro_imprensa	2298	15
97	rubro_impresal	1975	7
98	rubro_imprensa99	0	7
99	RUBRO_RELACIONADO	22	10
100	RUBRO_VARIO	98913	8
101	RUBRO_VARIO_BK0319	67114	7
102	rubro_vario_ene	85924	9
103	rubro_vario_feb	61630	8
104	SISTEMA_SYS	1	10
105	TABLE1	0	9
106	TEMPO	471	14
107	TIPO_CONTRATO	6	33
108	TIPO_CUENTA	4	10
109	TIPO_RUBRO	6	7
110	TOTALES	349	11
111	TRANSACCION	227639	20
112	TRANSACCION_SEGURIDAD_S	3571	7
113	USUARIO	11	7
114	valor_decimo	12045	8
115	valor_diciembre	142	8
116	valor_enero	142	10
117	VALOR_SUMATORIA	15	7
118	VALORES_SEP	9397	2
119	VARIO_HISTORICO	10767632	16
120	vario_octubre	67483	16
121	vehiculos	884	25
122	vehiculos_sep	884	25
123	VEHICULOS1	884	25
124	vivfis	89	2



**CIB-ESPOL**



# ANEXO E

## NUMERO DE TRANSACCIONES DE ESCRITURA Y PROMEDIOS OBTENIDOS EN LA BASE DE SUELDOS

Numero de dias laborados: 60

No	NOMBRE TABLA	# TRANSACCIONES	PROMEDIO DE ESCRITURA		
			ALTAS	MEDIAS	BAJAS
1	CONTADOR_TABLAS	236.0	0.0	0.0	3.9
2	dtproperties	124.0	0.0	0.0	2.1
3	rol_periodo	98.0	0.0	0.0	1.6
4	alcance_historico	3568.0	59.5	0.0	0.0
5	BENEFICIARIO	3125.0	52.1	0.0	0.0
6	BONO_TRIP	0.0	0.0	0.0	0.0
7	CARGA	5874.0	97.9	0.0	0.0
8	cedulas	1587.0	0.0	26.5	0.0
9	cedulas_tot	953.0	0.0	15.9	0.0
10	Compensacion_Transitoria	0.0	0.0	0.0	0.0
11	CONDECORACION	0.0	0.0	0.0	0.0
12	CUADRO_DIFERENCIA	1254.0	0.0	20.9	0.0
13	DETALLE	4725.0	78.8	0.0	0.0
14	detalle_retencion	6875.0	114.6	0.0	0.0
15	documento	0.0	0.0	0.0	0.0
16	EMPRESA	0.0	0.0	0.0	0.0
17	ESTADO_CIVIL	0.0	0.0	0.0	0.0
18	ESTADO_PERIODO	0.0	0.0	0.0	0.0
19	form_SRI	4587.0	76.5	0.0	0.0
20	form_SRI_TMP	5024.0	83.7	0.0	0.0
21	form_SRI_TMP_2004	0.0	0.0	0.0	0.0
22	form_SRI_TMP_2004_2	0.0	0.0	0.0	0.0
23	form_SRI_TMP_2004_3	0.0	0.0	0.0	0.0
24	form_SRI_TMP_2004_4	0.0	0.0	0.0	0.0
25	form_SRI_TMP_2004_OFC	3524.0	58.7	0.0	0.0
26	form_SRI_TMP_2004_TMP	0.0	0.0	0.0	0.0
27	form_SRI_TMP_2004_TRIP	3421.0	57.0	0.0	0.0
28	form_SRI_TMP10	0.0	0.0	0.0	0.0
29	form_SRI_TMP2	0.0	0.0	0.0	0.0
30	FORMA_PAGO	0.0	0.0	0.0	0.0
31	formulario	0.0	0.0	0.0	0.0
32	GRADO	0.0	0.0	0.0	0.0
33	GRUPO_USUARIO	0.0	0.0	0.0	0.0
34	HISTORICO_CONTRATOS	1452.0	0.0	24.2	0.0
35	HISTORICO_GASTOREP	0.0	0.0	0.0	0.0
36	IDS	56.0	0.0	0.0	0.9
37	imp_renta_2001	0.0	0.0	0.0	0.0
38	imp_renta_2001_Anterior	0.0	0.0	0.0	0.0
39	imp_renta_2002	0.0	0.0	0.0	0.0
40	imp_renta_2002_Anterior	0.0	0.0	0.0	0.0
41	imp_renta_2002_Audqui	0.0	0.0	0.0	0.0
42	imp_renta_2002_New	0.0	0.0	0.0	0.0
43	imp_renta_2003	0.0	0.0	0.0	0.0

44	imp_renta_2004	0.0	0.0	0.0	0.0
45	imp_renta_2004_anterior	0.0	0.0	0.0	0.0
46	imp_renta_2004_Bonos	0.0	0.0	0.0	0.0
47	imp_renta_2004_Condecoracion	0.0	0.0	0.0	0.0
48	imp_renta_2004_General	0.0	0.0	0.0	0.0
49	imp_renta_2005_Condecoracion	0.0	0.0	0.0	0.0
50	imp_renta_2005_General	3589.0	59.8	0.0	0.0
51	impuesto_renta	6587.0	109.8	0.0	0.0
52	impuesto_renta_2000	0.0	0.0	0.0	0.0
53	impuesto_renta1_2000	0.0	0.0	0.0	0.0
54	INGRESO_VARIO	129.0	0.0	0.0	2.2
55	INSTITUCION_BENEFICIARIA	0.0	0.0	0.0	0.0
56	LOTE	4583.0	76.4	0.0	0.0
57	nombres	0.0	0.0	0.0	0.0
58	OPCION_SISTEMA	1548.0	0.0	25.8	0.0
59	OPCION_SISTEMA1	0.0	0.0	0.0	0.0
60	OPERACION_FORMA	0.0	0.0	0.0	0.0
61	OTRO_RUBRO	0.0	0.0	0.0	0.0
62	PAGADURIA	0.0	0.0	0.0	0.0
63	PAGO_PROVISION	0.0	0.0	0.0	0.0
64	PARAMETRO_HISTORICO	0.0	0.0	0.0	0.0
65	PARAMETRO_RUBRO	2563.0	0.0	42.7	0.0
66	PARAMETRO_SISTEMA	0.0	0.0	0.0	0.0
67	parametros_sep	0.0	0.0	0.0	0.0
68	PER_ID3	0.0	0.0	0.0	0.0
69	PERFIL_AUTORIZARPT_SYS	0.0	0.0	0.0	0.0
70	PERFIL_CONSULTA_SYS	0.0	0.0	0.0	0.0
71	PERFIL_GRUPO	1258.0	0.0	21.0	0.0
72	PERFIL_GRUPO_SYS	1489.0	0.0	24.8	0.0
73	PERFIL_OPERACION_SYS	0.0	0.0	0.0	0.0
74	PERIODO_ROL	1485.0	0.0	24.8	0.0
75	PERIODOS_DIFERENCIA	0.0	0.0	0.0	0.0
76	PERMISO_OPCION_OBJETO_SY	1895.0	0.0	31.6	0.0
77	PERSONA	4589.0	76.5	0.0	0.0
78	persona_bk	3586.0	59.8	0.0	0.0
79	PERSONA_BK0319	3258.0	54.3	0.0	0.0
80	persona_fechas	3654.0	60.9	0.0	0.0
81	PERSONA_HISTORICO	0.0	0.0	0.0	0.0
82	PERSONAL_ASCENDIDO	0.0	0.0	0.0	0.0
83	PLAZA	0.0	0.0	0.0	0.0
84	PROVISION	0.0	0.0	0.0	0.0
85	RANCHO	0.0	0.0	0.0	0.0
86	REPARTO	3568.0	59.5	0.0	0.0
87	Results	6254.0	104.2	0.0	0.0
88	rol_diferencia	0.0	0.0	0.0	0.0
89	ROL_HISTORICO	3052.0	50.9	0.0	0.0
90	rol_periodo	1220.0	0.0	20.3	0.0
91	rol_prueba	5487.0	91.5	0.0	0.0
92	RUBRO	5812.0	96.9	0.0	0.0
93	RUBRO_decimo	5902.0	98.4	0.0	0.0



CIB-ESPOL

94	RUBRO_DIFERENCIA	1248.0	0.0	20.8	0.0
95	RUBRO_GASTOREP	1058.0	0.0	17.6	0.0
96	rubro_imprensa	5630.0	93.8	0.0	0.0
97	rubro_impresnal	4589.0	76.5	0.0	0.0
98	rubro_imprensa99	0.0	0.0	0.0	0.0
99	RUBRO_RELACIONADO	0.0	0.0	0.0	0.0
100	RUBRO_VARIO	4259.0	71.0	0.0	0.0
101	RUBRO_VARIO_BK0319	5681.0	94.7	0.0	0.0
102	rubro_vario_ene	0.0	0.0	0.0	0.0
103	rubro_vario_feb	0.0	0.0	0.0	0.0
104	SISTEMA_SYS	0.0	0.0	0.0	0.0
105	TABLE1	0.0	0.0	0.0	0.0
106	TEMPO	0.0	0.0	0.0	0.0
107	TIPO_CONTRATO	6.0	0.0	0.0	0.1
108	TIPO_CUENTA	0.0	0.0	0.0	0.0
109	TIPO_RUBRO	4586.0	76.4	0.0	0.0
110	TOTALES	5897.0	98.3	0.0	0.0
111	TRANSACCION	8542.0	142.4	0.0	0.0
112	TRANSACCION_SEGURIDAD_S	0.0	0.0	0.0	0.0
113	USUARIO	86.0	0.0	0.0	1.4
114	valor_decimo	4850.0	80.8	0.0	0.0
115	valor_diciembre	0.0	0.0	0.0	0.0
116	valor_enero	0.0	0.0	0.0	0.0
117	VALOR_SUMATORIA	15.0	0.0	0.0	0.3
118	VALORES_SEP	5472.0	91.2	0.0	0.0
119	VARIO_HISTORICO	0.0	0.0	0.0	0.0
120	vario_octubre	0.0	0.0	0.0	0.0
121	vehiculos	1587.0	0.0	26.5	0.0
122	vehiculos_sep	1682.0	0.0	28.0	0.0
123	VEHICULOS1	1695.0	0.0	28.3	0.0
124	vivfis	1854.0	0.0	30.9	0.0

**Promedios de escritura diaria:      80.7      25.3      1.6**

**Total tablas:                      31                      17                      8**

# ANEXO F

## NUMERO DE TRANSACCIONES DE LECTURA Y PROMEDIOS OBTENIDOS EN LA BASE DE SUELDOS

Numero de días laborados: 60

No	NOMBRE TABLA	# TRANSACCIONES	PROMEDIO DE LECTURA		
			ALTAS	MEDIAS	BAJAS
1	CONTADOR_TABLAS	0.0	0.0	0.0	0.0
2	dtproperties	0.0	0.0	0.0	0.0
3	rol_periodo	0.0	0.0	0.0	0.0
4	alcance historico	0.0	0.0	0.0	0.0
5	BENEFICIARIO	0.0	0.0	0.0	0.0
6	BONO_TRIP	1583.0	26.4	0.0	0.0
7	CARGA	0.0	0.0	0.0	0.0
8	cedulas	0.0	0.0	0.0	0.0
9	cedulas_tot	0.0	0.0	0.0	0.0
10	Compensacion_Transitoria	2014.0	33.6	0.0	0.0
11	CONDECORACION	564.0	0.0	9.4	0.0
12	CUADRO_DIFERENCIA	0.0	0.0	0.0	0.0
13	DETALLE	0.0	0.0	0.0	0.0
14	detalle_retencion	0.0	0.0	0.0	0.0
15	documento	235.0	0.0	0.0	3.9
16	EMPRESA	1652.0	27.5	0.0	0.0
17	ESTADO_CIVIL	1625.0	27.1	0.0	0.0
18	ESTADO_PERIODO	1498.0	25.0	0.0	0.0
19	form_SRI	0.0	0.0	0.0	0.0
20	form_SRI_TMP	0.0	0.0	0.0	0.0
21	form_SRI_TMP_2004	1254.0	20.9	0.0	0.0
22	form_SRI_TMP_2004_2	1568.0	26.1	0.0	0.0
23	form_SRI_TMP_2004_3	1526.0	25.4	0.0	0.0
24	form_SRI_TMP_2004_4	1258.0	21.0	0.0	0.0
25	form_SRI_TMP_2004_OFC	0.0	0.0	0.0	0.0
26	form_SRI_TMP_2004_TMP	1520.0	25.3	0.0	0.0
27	form_SRI_TMP_2004_TRIP	0.0	0.0	0.0	0.0
28	form_SRI_TMP10	1268.0	21.1	0.0	0.0
29	form_SRI_TMP2	1458.0	24.3	0.0	0.0
30	FORMA_PAGO	1253.0	20.9	0.0	0.0
31	formulario	1463.0	24.4	0.0	0.0
32	GRADO	2685.0	44.8	0.0	0.0
33	GRUPO_USUARIO	1552.0	25.9	0.0	0.0
34	HISTORICO_CONTRATOS	0.0	0.0	0.0	0.0
35	HISTORICO_GASTOREP	845.0	0.0	14.1	0.0
36	IDS	0.0	0.0	0.0	0.0
37	imp_renta_2001	352.0	0.0	5.9	0.0
38	imp_renta_2001_Anterior	345.0	0.0	5.8	0.0
39	imp_renta_2002	296.0	0.0	0.0	4.9
40	imp_renta_2002_Anterior	145.0	0.0	0.0	2.4
41	imp_renta_2002_Audqui	113.0	0.0	0.0	1.9
42	imp_renta_2002_New	98.0	0.0	0.0	1.6



43	imp_renta_2003	256.0	0.0	0.0	4.3
44	imp_renta_2004	455.0	0.0	7.6	0.0
45	imp_renta_2004_anterior	356.0	0.0	5.9	0.0
46	imp_renta_2004_Bonos	351.0	0.0	5.9	0.0
47	imp_renta_2004_Condecoracion	369.0	0.0	6.2	0.0
48	imp_renta_2004_General	358.0	0.0	6.0	0.0
49	imp_renta_2005_Condecoracion	1875.0	31.3	0.0	0.0
50	imp_renta_2005_General	0.0	0.0	0.0	0.0
51	impuesto_renta	0.0	0.0	0.0	0.0
52	impuesto_renta_2000	26.0	0.0	0.0	0.4
53	impuesto_rental_2000	12.0	0.0	0.0	0.2
54	INGRESO_VARIO	0.0	0.0	0.0	0.0
55	INSTITUCION_BENEFICIARIA	568.0	0.0	9.5	0.0
56	LOTE	0.0	0.0	0.0	0.0
57	nombres	356.0	0.0	5.9	0.0
58	OPCION_SISTEMA	0.0	0.0	0.0	0.0
59	OPCION_SISTEMA1	452.0	0.0	7.5	0.0
60	OPERACION_FORMA	485.0	0.0	8.1	0.0
61	OTRO_RUBRO	354.0	0.0	5.9	0.0
62	PAGADURIA	459.0	0.0	7.7	0.0
63	PAGO_PROVISION	457.0	0.0	7.6	0.0
64	PARAMETRO_HISTORICO	2015.0	33.6	0.0	0.0
65	PARAMETRO_RUBRO	0.0	0.0	0.0	0.0
66	PARAMETRO_SISTEMA	1872.0	31.2	0.0	0.0
67	parametros_sep	358.0	0.0	6.0	0.0
68	PER_ID3	5873.0	97.9	0.0	0.0
69	PERFIL_AUTORIZARPT_SYS	568.0	0.0	9.5	0.0
70	PERFIL_CONSULTA_SYS	459.0	0.0	7.7	0.0
71	PERFIL_GRUPO	0.0	0.0	0.0	0.0
72	PERFIL_GRUPO_SYS	0.0	0.0	0.0	0.0
73	PERFIL_OPERACION_SYS	458.0	0.0	7.6	0.0
74	PERIODO_ROL	0.0	0.0	0.0	0.0
75	PERIODOS_DIFERENCIA	1589.0	26.5	0.0	0.0
76	PERMISO_OPCION_OBJETO_SY	0.0	0.0	0.0	0.0
77	PERSONA	0.0	0.0	0.0	0.0
78	persona_bk	0.0	0.0	0.0	0.0
79	PERSONA_BK0319	0.0	0.0	0.0	0.0
80	persona_fechas	0.0	0.0	0.0	0.0
81	PERSONA_HISTORICO	10256.0	170.9	0.0	0.0
82	PERSONAL_ASCENDIDO	568.0	0.0	9.5	0.0
83	PLAZA	652.0	0.0	10.9	0.0
84	PROVISION	1548.0	25.8	0.0	0.0
85	RANCHO	265.0	0.0	0.0	4.4
86	REPARTO	0.0	0.0	0.0	0.0
87	Results	0.0	0.0	0.0	0.0
88	rol_diferencia	129.0	0.0	0.0	2.2
89	ROL_HISTORICO	0.0	0.0	0.0	0.0
90	rol_periodo	0.0	0.0	0.0	0.0
91	rol_prueba	0.0	0.0	0.0	0.0
92	RUBRO	0.0	0.0	0.0	0.0

CIB-EST

93	RUBRO_decimo	0.0	0.0	0.0	0.0
94	RUBRO_DIFERENCIA	0.0	0.0	0.0	0.0
95	RUBRO_GASTOREP	0.0	0.0	0.0	0.0
96	rubro_imprensa	0.0	0.0	0.0	0.0
97	rubro_imprental	0.0	0.0	0.0	0.0
98	rubro_imprensa99	193.0	0.0	0.0	3.2
99	RUBRO_RELACIONADO	92.0	0.0	0.0	1.5
100	RUBRO_VARIO	0.0	0.0	0.0	0.0
101	RUBRO_VARIO_BK0319	0.0	0.0	0.0	0.0
102	rubro_vario_ene	84.0	0.0	0.0	1.4
103	rubro_vario_feb	62.0	0.0	0.0	1.0
104	SISTEMA_SYS	753.0	0.0	12.6	0.0
105	TABLE1	23.0	0.0	0.0	0.4
106	TEMPO	259.0	0.0	0.0	4.3
107	TIPO_CONTRATO	0.0	0.0	0.0	0.0
108	TIPO_CUENTA	3548.0	59.1	0.0	0.0
109	TIPO_RUBRO	0.0	0.0	0.0	0.0
110	TOTALES	0.0	0.0	0.0	0.0
111	TRANSACCION	0.0	0.0	0.0	0.0
112	TRANSACCION_SEGURIDAD_S	174.0	0.0	0.0	2.9
113	USUARIO	0.0	0.0	0.0	0.0
114	valor_decimo	0.0	0.0	0.0	0.0
115	valor_diciembre	44.0	0.0	0.0	0.7
116	valor_enero	35.0	0.0	0.0	0.6
117	VALOR_SUMATORIA	0.0	0.0	0.0	0.0
118	VALORES_SEP	0.0	0.0	0.0	0.0
119	VARIO_HISTORICO	42.0	0.0	0.0	0.7
120	vario_octubre	125.0	0.0	0.0	2.1
121	vehiculos	0.0	0.0	0.0	0.0
122	vehiculos_sep	0.0	0.0	0.0	0.0
123	VEHICULOS1	0.0	0.0	0.0	0.0
124	vivfis	0.0	0.0	0.0	0.0

**Promedios de lectura diaria:            28.9            5.9            1.5**

**Total tablas:                                24                        23                        21**



**CIB-ESPOL**

# ANEXO G

## “CODIGO DEL CASE IMPLEMENTADO”

### DECLARACION DE VARIABLES

```
* Syopbase.java
* Created on 19 de febrero de 2004, 01:55 PM
package SYOPBASE;
public class Syopbase extends javax.swing.JFrame {
    /** Creates new form Syopbase*/
    public Syopbase() {
        initComponents();
        // setBounds(250,80,700,650);
        int inset = 200;
        Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();
        setBounds(inset + 100, inset, screenSize.width - inset*3, screenSize.height-
inset*2);
    }
    private void initComponents() {
        //Ingreso de todos los componentes
        jDesktopPane1 = new javax.swing.JDesktopPane();
        iconoNew = new
javax.swing.ImageIcon("C:\\forte4j\\bin\\forte4j\\sampledir\\SYOPBASE\\imagenes\\
New.gif");
        iconoOpen = new
javax.swing.ImageIcon("C:\\forte4j\\bin\\forte4j\\sampledir\\SYOPBASE\\imagenes\\
Open.gif");
        iconoSave = new
javax.swing.ImageIcon("C:\\forte4j\\bin\\forte4j\\sampledir\\SYOPBASE\\imagenes\\
Save.gif");
        barraMenu = new javax.swing.JMenuBar();
        consulta = new javax.swing.JMenu();
        nuevaItem = new javax.swing.JMenuItem("Nueva",iconoNew);
        separador2 = new javax.swing.JSeparator();
        abrirItem = new javax.swing.JMenuItem("Abrir", iconoOpen);
        guardarItem = new javax.swing.JMenuItem("Guardar", iconoSave);
        separador1 = new javax.swing.JSeparator();
        salirItem = new javax.swing.JMenuItem();
        editar = new javax.swing.JMenu();
        ayuda = new javax.swing.JMenu();
        temasItem = new javax.swing.JMenuItem();
        acercaltem = new javax.swing.JMenuItem();
        // Se Inicializa el panel
```

```

setTitle("SYOPBASE");
setResizable(false);
getContentPane().setLayout(null);
addWindowListener(new java.awt.event.WindowAdapter() {
    public void windowClosing(java.awt.event.WindowEvent evt) {
        exitForm(evt);
    }
});
//Crear Panel
getContentPane().add(jDesktopPane1);
jDesktopPane1.setBounds(0, 0, 700, 600);
// Menu Principal
consulta.setText("Consulta");
nuevoItem.setFont(new java.awt.Font("Dialog", 1, 10));
nuevoItem.setText("Nueva");
consulta.add(nuevoItem);
consulta.add(separador2);
abrirItem.setFont(new java.awt.Font("Dialog", 1, 10));
abrirItem.setText("Abrir");
consulta.add(abrirItem);
guardarItem.setFont(new java.awt.Font("Dialog", 1, 10));
guardarItem.setText("Guardar");
consulta.add(guardarItem);
consulta.add(separador1);
salirItem.setFont(new java.awt.Font("Dialog", 1, 10));
salirItem.setText("Salir");
consulta.add(salirItem);
barraMenu.add(consulta);
editar.setText("Edicion");
barraMenu.add(editar);
ayuda.setText("Ayuda");
temasItem.setFont(new java.awt.Font("Dialog", 1, 10));
temasItem.setText("Temas de ayuda");
ayuda.add(temasItem);
acercaltem.setFont(new java.awt.Font("Dialog", 1, 10));
acercaltem.setText("Acerca");
ayuda.add(acercaltem);
barraMenu.add(ayuda);
nuevoItem.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        nuevoItemActionPerformed(evt);
    }
});
abrirItem.addActionListener(new java.awt.event.ActionListener() {

```



**CIB-ESPOL**



```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            abrirItemActionPerformed(evt);
        }
    });
    guardarItem.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            guardarItemActionPerformed(evt);
        }
    });
    salirItem.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            salirItemActionPerformed(evt);
        }
    });
    acercaItem.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            acercaItemActionPerformed(evt);
        }
    });
    setJMenuBar(barraMenu);
    pack();
}
private void nuevaItemActionPerformed(java.awt.event.ActionEvent evt) { //GEN-
FIRST:event_nuevaItemActionPerformed
    new Base().show();
} //GEN-LAST:event_nuevaItemActionPerformed
private void abrirItemActionPerformed(java.awt.event.ActionEvent evt) { //GEN-
FIRST:event_abrirItemActionPerformed
    Paneles aPaneles = new Paneles();
    jDesktopPanel.add(aPaneles);
    aPaneles.setClosable(true);
    aPaneles.setVisible(true);
    aPaneles.setBounds(80, 50, 500, 400);
} //GEN-LAST:event_abrirItemActionPerformed
private void guardarItemActionPerformed(java.awt.event.ActionEvent evt) {
}
private void salirItemActionPerformed(java.awt.event.ActionEvent evt) {
    System.exit(0);
}
private void acercaItemActionPerformed(java.awt.event.ActionEvent evt) { //GEN-
FIRST:event_acercaItemActionPerformed
    new Acerca().show();
} //GEN-LAST:event_acercaItemActionPerformed
/** Exit the Application */

```

```

private void exitForm(java.awt.event.WindowEvent evt) { //GEN-FIRST:event_exitForm
    System.exit(0);
} //GEN-LAST:event_exitForm
public static void main(String args[]) {
    new Syopbase().show();
}
// Variables declaration - do not modify
private javax.swing.JDesktopPane jDesktopPane1;
private Paneles jInternalPanel;
private javax.swing.JMenuBar barraMenu;
private javax.swing.JSeparator separador2;
private javax.swing.JSeparator separador1;
private javax.swing.JMenu consulta;
private javax.swing.JMenu editar;
private javax.swing.JMenu ayuda;
private javax.swing.JMenuItem salirItem;
private javax.swing.JMenuItem acercaltem;
private javax.swing.JMenuItem nuevaltem;
private javax.swing.JMenuItem abrirltem;
private javax.swing.JMenuItem guardarItem;
private javax.swing.JMenuItem temasItem;
private javax.swing.ImageIcon iconoNew;
private javax.swing.ImageIcon iconoOpen;
private javax.swing.ImageIcon iconoSave;
// End of variables declaration
}

```

**INDIVIDUAL**

```

package SYOPBASE;
import java.io.* ;
public class Individual implements Serializable
{ private Chromosoma chrom;
private int L;
private double fitness;
public Individual(int L)
{ chrom = new Chromosoma(L);
fitness = 0.0;
this.L = L;
}
public void print()
{ chrom.print();
System.out.print(" ");
System.out.println(fitness);
}
}

```



**CIB-ESPOL**

```

public int get_length()
{
    return L;
}
public void set_fitness(double fit)
{
    fitness = fit;
}
public double get_fitness()
{
    return fitness;
}
public void set_allele(int index, byte value)
{
    chrom.set_allele(index,value);
}
public byte get_allele(int index)
{
    return chrom.get_allele(index);
}
private void copy(Chromosoma source, Chromosoma destination)
{
    for (int i=0; i<L; i++)
    {
        destination.set_allele(i,source.get_allele(i));
    }
}

```

## PANELES

```

* Paneles.java
* Created on 7 de marzo de 2004, 09:48 AM
package SYOPBASE;
//import javax.swing.table.*;
import javax.swing.ListSelectionModel;
import javax.swing.event.ListSelectionEvent;
import javax.swing.event.ListSelectionListener;
import javax.swing.event.*;
import javax.swing.*;
public class Paneles extends javax.swing.JInternalFrame {
    /** Creates new form Paneles */
    public Paneles() {
        initComponents();
        setBounds(50, 50, 500, 400);
    }
    private void initComponents() { //EN-BEGIN:inicomponents
        frame = new javax.swing.JFrame();
        jTabbedPane = new javax.swing.JTabbedPane();

```



**CIB-ESPOL**

```

        iconoAdd = new
javax.swing.ImageIcon("C:\\forte4j\\bin\\forte4j\\sampledir\\SYOPBASE\\imagenes\\
add.gif");
        iconoDel = new
javax.swing.ImageIcon("C:\\forte4j\\bin\\forte4j\\sampledir\\SYOPBASE\\imagenes\\
del.gif");
        iconoGen = new
javax.swing.ImageIcon("C:\\forte4j\\bin\\forte4j\\sampledir\\SYOPBASE\\imagenes\\
gen.gif");
        iconoSave = new
javax.swing.ImageIcon("C:\\forte4j\\bin\\forte4j\\sampledir\\SYOPBASE\\imagenes\\
Save.gif");
        basePanel = new javax.swing.JPanel();
        baseScroll = new javax.swing.JScrollPane();
        baseTabla = new javax.swing.JTable();
        baseBotonDel = new javax.swing.JButton("Eliminar",iconoDel);
        baseBotonGen = new javax.swing.JButton("Generar",iconoGen);
        discoPanel = new javax.swing.JPanel();
        discoScroll = new javax.swing.JScrollPane();
        discoTabla = new javax.swing.JTable();
        discoBotonAdd = new javax.swing.JButton("Adicionar",iconoAdd);
        discoBotonDel = new javax.swing.JButton("Eliminar",iconoDel);
        discoBotonSave = new javax.swing.JButton("Grabar",iconoSave);
        tablaPanel = new javax.swing.JPanel();
        tablaScroll = new javax.swing.JScrollPane();
        tablaTabla = new javax.swing.JTable();
        tablaBotonAdd = new javax.swing.JButton("Adicionar",iconoAdd);
        tablaBotonDel = new javax.swing.JButton("Eliminar",iconoDel);
        tablaBotonSave = new javax.swing.JButton("Grabar",iconoSave);
        basedata = new BaseTablaDatos("Select * from Base");
        discodata = new DiscoTablaDatos("Select Numero, Raid, Velocidad_escritura,
Velocidad_lectura, Capacidad from Disco");
        tabladata = new TablaTablaDatos("Select Nombre,Concurrencia_escritura,
Concurrencia_lectura, Cantidad_columnas from Tabla");
        // resuldata = new ResulTablaDatos("Select Cromosomal from Resultados"
        getContentPane().setLayout(null);
        jTabbedPane.setEnabled(false);
        basePanel.setName("Bases");
        basePanel.setLayout(null);
        baseTabla.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(0,
0, 0), 1, true));
        baseTabla.setForeground(new java.awt.Color(0, 0, 0));
        baseTabla.setBounds(50,50,350,160);
        baseTabla.setModel(basedata);

```

```

baseScroll.setViewportView(baseTabla);
basePanel.add(baseScroll);
baseScroll.setBounds(50,50,350,160);
basePanel.add(baseBotonGen);
basePanel.add(baseBotonDel);
baseBotonGen.setBounds(90,250, 120, 30);
baseBotonGen.setEnabled(false);
baseBotonGen.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        baseBotonGenActionPerformed(evt);
    }
});
baseBotonDel.setBounds(250, 250, 120, 30);
baseBotonDel.setEnabled(false);
baseBotonDel.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        baseBotonDelActionPerformed(evt);
    }
});
//Seleccion de Base
baseTabla.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
baseLista = baseTabla.getSelectionModel();
baseLista.addListSelectionListener(new ListSelectionListener() {
    public void valueChanged(ListSelectionEvent e) {
        baseFilaSel(e);
    }
});
jTabbedPane.addTab("Bases", basePanel);
discoPanel.setName("Discos");
discoPanel.setLayout(null);
discoTabla.setBorder(new javax.swing.border.LineBorder(new
java.awt.Color(0, 0, 0), 1, true));
discoTabla.setForeground(new java.awt.Color(0, 0, 0));
discoTabla.setBounds(50,50,350,160);
discoTabla.setModel(discodata);
discoScroll.setViewportView(discoTabla);
discoPanel.add(discoScroll);
discoScroll.setBounds(50,50,350,160);
discoPanel.add(discoBotonAdd);
discoPanel.add(discoBotonDel);
discoPanel.add(discoBotonSave);
discoBotonAdd.setBounds(50, 250, 100, 30);
discoBotonAdd.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        discoBotonAddActionPerformed(evt);
    }
});
discoBotonDel.setBounds(180, 250, 100, 30);
discoBotonDel.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        discoBotonDelActionPerformed(evt);
    }
});
discoBotonSave.setBounds(310, 250, 100, 30);
discoBotonSave.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        discoBotonSaveActionPerformed(evt);
    }
});

//Seleccion de disco
discoTabla.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
discoLista = discoTabla.getSelectionModel();
discoLista.addListSelectionListener(new ListSelectionListener() {
    public void valueChanged(ListSelectionEvent e) {
        discoFilaSel(e);
    }
});
jTabbedPane.addTab("Discos", discoPanel);
tablaPanel.setName("Tablas");
tablaPanel.setLayout(null);
tablaTabla.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(0,
0), 1, true));
tablaTabla.setForeground(new java.awt.Color(0, 0, 0));
tablaTabla.setBounds(50, 50, 350, 160);
tablaTabla.setModel(tabladata);
tablaScroll.setViewportView(tablaTabla);
tablaPanel.add(tablaScroll);
tablaScroll.setBounds(50,50,350,160);
tablaPanel.add(tablaBotonAdd);
tablaPanel.add(tablaBotonDel);
tablaPanel.add(tablaBotonSave);
tablaBotonAdd.setBounds(50, 250, 100, 30);
tablaBotonAdd.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        tablaBotonAddActionPerformed(evt);
    }
});

```



**CIB-ESPOL**

```

tablaBotonDel.setBounds(180, 250, 100, 30);
tablaBotonDel.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        tablaBotonDelActionPerformed(evt);
    }
});
tablaBotonSave.setBounds(310, 250, 100, 30);
tablaBotonSave.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        tablaBotonSaveActionPerformed(evt);
    }
});
// Seleccion de tabla
tablaTabla.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
tablaLista = tablaTabla.getSelectionModel();
tablaLista.addListSelectionListener(new ListSelectionListener() {
    public void valueChanged(ListSelectionEvent e) {
        tablaFilaSel(e);
    }
});
jTabbedPane.addTab("Tablas", tablaPanel);
getContentPane().add(jTabbedPane);
jTabbedPane.setBounds(20, 20, 450, 320);
pack();
}
//Metodo de Botones de Base
private void baseBotonGenActionPerformed(java.awt.event.ActionEvent evt) {
    int row = baseLista.getMinSelectionIndex();
    String a1 = (String)baseTabla.getValueAt(row, 2);
    String a2 = (String)baseTabla.getValueAt(row, 3);
    int a3 = Integer.parseInt(a1);
    int a4 = Integer.parseInt(a2);
    algoritmo = new Exe( a3,a4 );
}
private void baseBotonDelActionPerformed(java.awt.event.ActionEvent evt) {
    int row = baseLista.getMinSelectionIndex();
    b = (String)baseTabla.getValueAt(row, 1);
    System.out.println( b );
    new Query ("DELETE FROM Base WHERE Nombre =" + b + "");
    if (basedata.delete(row)) {
        baseTabla.tableChanged(new TableModelEvent(basedata, row, row,
        TableModelEvent.ALL_COLUMNS,TableModelEvent.INSERT));
        baseTabla.repaint();
    }
}

```

```

}
//Metodos de Boton Disco
private void discoBotonDelActionPerformed(java.awt.event.ActionEvent evt) {
    int row = discoLista.getMinSelectionIndex();
    String d = (String)discoTabla.getValueAt(row, 0);
    System.out.println( d );
    new Query("Delete from Disco where numero = " + d + " AND id_base = (Select
id_base from Base where Nombre =" + b + ")");
    if (discodata1.delete(row)) {
        discoTabla.tableChanged(new TableModelEvent(discodata1, row, row,
TableModelEvent.ALL_COLUMNS,TableModelEvent.INSERT));
        discoTabla.repaint();
    }
}

private void discoBotonAddActionPerformed(java.awt.event.ActionEvent evt) {
    int row = discoLista.getMinSelectionIndex();
    discodata1.insert(row+1);
    discoTabla.tableChanged(new TableModelEvent(discodata1, row, row,
TableModelEvent.ALL_COLUMNS, TableModelEvent.INSERT));
    discoTabla.repaint();
}

private void discoBotonSaveActionPerformed(java.awt.event.ActionEvent evt) {
    int row = discoLista.getMinSelectionIndex();
    String n = (String)discoTabla.getValueAt(row, 0);
    String r = (String)discoTabla.getValueAt(row, 1);
    String e = (String)discoTabla.getValueAt(row, 2);
    String l = (String)discoTabla.getValueAt(row, 3);
    String c = (String)discoTabla.getValueAt(row, 4);
    new Query ("Insert into Disco (Numero, Raid, Velocidad_escritura,
Velocidad_lectura, Capacidad, id_base) values ("
        + n + "," + r + "," + e + "," + l + "," + c + "," + id + ")");
    JOptionPane.showMessageDialog(frame, "Disco Ingresado");
}

// Metodo de Boton Tabla
private void tablaBotonDelActionPerformed(java.awt.event.ActionEvent evt) {
    int row = tablaLista.getMinSelectionIndex();
    String t = (String)tablaTabla.getValueAt(row, 0);
    System.out.println( t );
    new Query("Delete from Tabla where Nombre = " + t + " AND id_base =
(Select id_base from Base where Nombre =" + b + ")");
    if (tabladata1.delete(row)) {
        tablaTabla.tableChanged(new TableModelEvent(tabladata1, row, row,
TableModelEvent.ALL_COLUMNS,TableModelEvent.INSERT));
        tablaTabla.repaint();
    }
}

```



```

    }
}
private void tablaBotonAddActionPerformed(java.awt.event.ActionEvent evt) {
    int row = tablaLista.getMinSelectionIndex();
    tabldata1.insert(row+1);
    tablaTabla.tableChanged(new TableModelEvent(discodata1, row, row,
TableModelEvent.ALL_COLUMNS, TableModelEvent.INSERT));
    tablaTabla.repaint();
}
private void tablaBotonSaveActionPerformed(java.awt.event.ActionEvent
evt) {
    int row = tablaLista.getMinSelectionIndex();
    String n = (String)tablaTabla.getValueAt(row, 0);
    String r = (String)tablaTabla.getValueAt(row, 1);
    String e = (String)tablaTabla.getValueAt(row, 2);
    String l = (String)tablaTabla.getValueAt(row, 3);
    System.out.println( n + r + e + l + id );
    new Query ("Insert into Tabla (Nombre, Concurrencia_escritura,
Concurrencia_lectura, Cantidad_columnas,id_base) values ( "
        + n + " , " + r + " , " + e + " , " + l + " , " + id + ")");
    JOptionPane.showMessageDialog(frame, "Tabla Ingresada");
}
// Metodos de seleccion
public void baseFilaSel(ListSelectionEvent e) {
    if (e.getValueIsAdjusting()) return;
    baseLista =(ListSelectionModel)e.getSource();
    if (baseLista.isSelectionEmpty()) {
        System.out.println("No rows are selected.");
    } else {
        baseBotonGen.setEnabled(true);
        baseBotonDel.setEnabled(true);
        jTabbedPane.setEnabled(true);
        int selectedRow = baseLista.getMinSelectionIndex();
        id = (String)baseTabla.getValueAt(selectedRow, 0);
        b = (String)baseTabla.getValueAt(selectedRow, 1);
        System.out.println( b + id );
        discodata1 = new DiscoTablaDatos("Select Numero, Raid, Velocidad_escritura,
Velocidad_lectura, Capacidad from Disco where id_base = (Select id_base from Base
where Nombre = " + b + ") order by Numero");
        tabldata1 = new TablaTablaDatos("Select Nombre,Concurrencia_escritura,
Concurrencia_lectura, Cantidad_columnas from Tabla where id_base = (Select
id_base from Base where Nombre = " + b + ") order by Nombre");
        discoTabla.setModel(discodata1);
        tablaTabla.setModel(tabldata1);
    }
}

```

```

}
}
private void discoFilaSel(ListSelectionEvent e) {
if (e.getValueIsAdjusting()) return;
discoLista = (ListSelectionModel)e.getSource();
if (discoLista.isEmpty()) {
System.out.println("No rows are selected.");
} else {
int selectedRow = discoLista.getMinSelectionIndex();
String d = (String)discoTabla.getValueAt(selectedRow, 0);
System.out.println(b + d);
}
}
}

```

```

private void tablaFilaSel(ListSelectionEvent e) {{
if (e.getValueIsAdjusting()) return;
tablaLista = (ListSelectionModel)e.getSource();
if (tablaLista.isEmpty()) {
System.out.println("No rows are selected.");
} else {
int selectedRow = tablaLista.getMinSelectionIndex();
String t = (String)tablaTabla.getValueAt(selectedRow, 0);
System.out.println(t);
}
}}

```

// Variables declaration - do not modify//EN-BEGIN:variables

```

private javax.swing.JFrame frame;
private javax.swing.JTabbedPane jTabbedPane;
private javax.swing.JTable baseTabla;
private javax.swing.JScrollPane baseScroll;
private javax.swing.JPanel basePanel;
private javax.swing.ListSelectionModel baseLista;
private javax.swing.JButton baseBotonDel;
private javax.swing.JButton baseBotonGen;
private javax.swing.JTable discoTabla;
private javax.swing.JScrollPane discoScroll;
private javax.swing.JPanel discoPanel;
private javax.swing.ListSelectionModel discoLista;
private javax.swing.JButton discoBotonAdd;
private javax.swing.JButton discoBotonDel;
private javax.swing.JButton discoBotonSave;
private javax.swing.JTable tablaTabla;
private javax.swing.JScrollPane tablaScroll;
private javax.swing.JPanel tablaPanel;
private javax.swing.ListSelectionModel tablaLista;

```



**CIB-ESPOL**

```
private javax.swing.JButton tablaBotonAdd;
private javax.swing.JButton tablaBotonDel;
private javax.swing.JButton tablaBotonSave;
private javax.swing.ImageIcon iconoAdd;
private javax.swing.ImageIcon iconoDel;
private javax.swing.ImageIcon iconoGen;
private javax.swing.ImageIcon iconoSave;
public BaseTablaDatos basedata;
public BaseTablaDatos basedata1;
private DiscoTablaDatos discodata;
private TablaTablaDatos tabladata;
private DiscoTablaDatos discodata1;
private TablaTablaDatos tabladata1;
private String b;
private String id;
private Exe algoritmo;
// End of variables declaration
}
```

## PROBLEMAS BASE

```
package SYOPBASE;
public class ProblemaBase extends Problema {
    public double Evaluate(Individual Indiv)
    {
        return BASE(Indiv);
    }
    private double MMM4(Individual indiv)
    {
        int i, j, index, len;
        int unitation;
        double v;
        v = 0.0;
        index = 0;
        len = CL;
        for(i=0; i<len; i++)
        { unitation = 0;
          for(j=0; j<3; j++)
            if(indiv.get_allele(index++)==1)
              unitation++;
            switch (unitation)
            { case 0 : v += 1.000000; break;
              case 1 : v += 0.000000; break;
              case 2 : v += 0.000000; break;
              case 3 : v += 0.000000; break;
            }
          }
        indiv.set_fitness(v);
        return v;
    }
    private static int w[]={0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001,
                          1010, 1011, 1100, 1101, 1110, 1111};
    private static int C = 300500;
    private double SUBSET_SUM(Individual indiv)
    {
        int i;
        double fitness = 0.0;
        if(CL!=001)
            System.out.println("Length mismatch error in Subset sum function.");
        for( i=0; i<CL; i++)
            fitness += indiv.get_allele(i)*w[i];
        if (fitness>C)
        {
            fitness = C - fitness*0.1;
        }
    }
}
```

```

    if(fitness<0.0) fitness=0.0;
  }
  indiv.set_fitness(fitness);
  return fitness;
}
private static int N; // longitud del cromosoma
private static int P = 4;
private static short peak[][];
private double BASE(Individual indiv)
{
  double fitness = 0.0;
  int nearest_peak;
  int j, p, hd;
  if(fitness_counter==1)
  {
    N = indiv.get_length();
    peak = new short[P][N];
    for(p=0;p<P;p++)
    {
      for(j=0;j<N;j++)
        if(r.nextDouble()<0.5)
          peak[p][j] = 1;
        else peak[p][j] = 0;
    }
  }
  nearest_peak = 0;
  for(p=0; p<P; p++)
  {
    // Compute Hamming distance
    for(hd=0,j=0;j<N;j++) if(peak[p][j]!=indiv.get_allele(j)) hd++;
    if((N-hd)>nearest_peak)
      nearest_peak = N-hd;
  }
  fitness = (double)nearest_peak / (double)N;
  indiv.set_fitness(fitness);
  return fitness;
}
}

```

## ALGORITMO

```
package SYOPBASE;
import java.util.Random;
public class Algoritmo
{
    private int    chrom_length;
    private int    gene_number; // Numero de tablas en cada chromosome
    private int    gene_length; // Numero de discos por gene
    private int    popsize;    // Number of individuals in the population
    private double pc, pm;     // Probability of applying crossover and mutation
    private Problema problema; // The problem being solved
    private Poblacion pop;    // The population
    private static Random r;  // Source for random values in this class
    private Individual aux_indiv; // Internal auxiliar individual being computed
    // CONSTRUCTOR
    public Algoritmo(Problema p, int popsize, int gn, int gl, double pc, double pm)
    /*throws Exception*/
    {
        this.gene_number = gn;
        this.gene_length = gl;
        this.chrom_length = gn*gl;
        this.popsize = popsize;
        this.pc = pc;
        this.pm = pm;
        this.problema = p;
        this.pop = new Poblacion(popsize,chrom_length);// Create initial population
        this.r = new Random();
        this.aux_indiv = new Individual(chrom_length);
        for(int i=0;i<popsize;i++)
        try {
            pop.set_fitness(i,problema.evaluateStep(pop.get_ith(i)));
        } catch (Exception e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        pop.compute_stats();
    }
    // BINARY TOURNAMENT
    public Individual select_tournament() throws Exception
    {
        int p1, p2;
        Individual aIndividual = null;
        p1 = (int)(r.nextDouble()* (double)popsize + 0.5); // Round and then trunc to int
```



**CIB-ESPOL**

```

if(p1>popsiz-1) p1=popsiz-1;
do
{ p2 = (int)(r.nextDouble()*(double)popsiz + 0.5); // Round and then trunc to int
  if(p2>popsiz-1) p2=popsiz-1;
} while (p1 == p2);
try {
    if (pop.get_ith(p1).get_fitness()>pop.get_ith(p2).get_fitness())
        aIndividual = pop.get_ith(p1);
    else
        aIndividual = pop.get_ith(p2);
} catch (Exception e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
return aIndividual;
}
// punto de crossover - solamente es creado un hijo (randomicamente descartamos
los otros)
public Individual SPX (Individual p1, Individual p2)
{
    int rand;
    rand = (int)(r.nextDouble()*(double)chrom_length - 1 + 0.5); // From 0 to L-1
rounded
    if(rand>chrom_length-1)
        rand = chrom_length-1;
    if(r.nextDouble()>pc) // If no crossover then randomly returns one parent
return r.nextDouble()>0.5?p1:p2;
// Copy CHROMOSOME 1
for (int i=0; i<rand; i++)
{
    aux_indiv.set_allele(i,p1.get_allele(i));
}
// Copy CHROMOSOME 2
for (int i=rand; i<chrom_length; i++)
{
    aux_indiv.set_allele(i,p2.get_allele(i));
}
return aux_indiv;
}

// MUTACION BINARIA DEL CROMOSOMA
public Individual mutate(Individual p1)
{
    byte allele=0;

```

```

Random r = new Random();

aux_indiv.assign(p1);
for(int i=0; i<chrom_length; i++)
if (r.nextDouble()<=pm) // Check mutation bit by bit...
{
    if(aux_indiv.get_allele(i)==1)
    aux_indiv.set_allele(i,(byte)0);
    else
    aux_indiv.set_allele(i,(byte)1);
}
return aux_indiv;
}
// REPLAZO DEL PEOR INDIVIDUO
public void replace(Individual new_indiv) //throws Exception
{
    try {
        pop.set_ith(pop.get_worstp(),new_indiv);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    pop.compute_stats(); // Recompute avg, best, worst, etc.
}
// EVALUACION DE LA FUNCION FITNESS A UN INDIVIDUO
private double evaluateStep(Individual indiv)
{
    return problema.evaluateStep(indiv);
}
public void go_one_step()// throws Exception
{
    try {
        aux_indiv.assign( SPX(select_tournament(),select_tournament()) );
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    aux_indiv.set_fitness(problema.evaluateStep(mutate(aux_indiv)));
    replace(aux_indiv);
}
public Individual get_solution()// throws Exception
{
    Individual aIndividual = null;
    try {

```



```

        aIndividual = pop.get_ith(pop.get_bestp());
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    } // The better individual is the solution
    return aIndividual;
}
public int  get_worstp() { return pop.get_worstp(); }
public int  get_bestp() { return pop.get_bestp(); }
public double get_worstf() { return pop.get_worstf(); }
public double get_avgf() { return pop.get_avgf(); }
public double get_bestf() { return pop.get_bestf(); }
public double get_BESTF() { return pop.get_BESTF(); }
public Individual get_ith(int index)// throws Exception
{
    Individual aIndividual = null;
    try {
        aIndividual = pop.get_ith(index);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return aIndividual;
}
public void set_ith(int index, Individual indiv)// throws Exception
{
    try {
        pop.set_ith(index,indiv);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
}
// END OF CLASS: Algorithm

```



**CIB-ESPOL**

## POBLACION

```
package SYOPBASE;
public class Poblacion
{
    // PRIVATE MEMORY
    private int    popsize; // numero de individuos
    private Individual pop[]; // vector de individuos
    private int chrom_length; // longitud del cromosoma
    // STATISTICS
    private int    bestp; // Tposiciona el mejor individuo
    private int    worstp; // posiciona el peor individuo
    private double bestf; // presenta la mejor poblacion
    private double avgf; // el promedio de la mejor poblacion
    private double worstf; // el peor de la presente poblacion
    private double BESTF; // Tel mejor de la presente poblacion
    public Poblacion(int ps, int chroml)
    {
        popsize    = ps;
        pop        = new Individual[popsize];
        chrom_length = chroml;
        for(int i=0;i<popsize;i++) pop[i] = new Individual(chroml);

        // Initialize statistics
        bestp = 0;    worstp = 0;
        bestf = 0.0; avgf  = 0.0; worstf = 999999999.0; BESTF = 0.0;
    }
    public int get_popsize()
    {
        return popsize;
    }
    public int worst_pos()
    {
        return worstp;
    }
    public Individual get_ith (int index) throws Exception
    {
        if ((index<popsize) && (index>=0))
            return pop[index];
        else
            throw new Exception("Index out of range when getting a copy of an individual");
    }
    public void set_ith(int index, Individual indiv) throws Exception
    {
        if ((index<popsize) && (index>=0))
```



**CIB-ESPOL**

```

pop[index].assign(indiv);
else
throw new Exception("Index out of range when inserting and individual");
compute_stats();
}
public void set_fitness( int index, double fitness ) //throws Exception
{
pop[index].set_fitness(fitness);
}
public void compute_stats()
{
double f, total;
// Initialize values (always needed!!!)
total = 0.0;
worstf = pop[0].get_fitness();   worstp = 0;
bestf = pop[0].get_fitness();   bestp = 0;
for(int i=0;i<popsize;i++)
{ f = pop[i].get_fitness();
if(f<=worstf) {worstf = f; worstp = i;}
if(f>=bestf) {bestf = f; bestp = i;}
if(f>=BESTF) {BESTF = f;          }
total += f;
}
avgf = total/(double)popsize;
}
public int  get_worstp() { return worstp; }
public int  get_bestp()  { return bestp;  }
public double get_worstf() { return worstf; }
public double get_avgf()  { return avgf;  }
public double get_bestf() { return bestf;  }
public double get_BESTF() { return BESTF;  }
/** public void print()
{
for(int i=0;i<popsize;i++)
{ System.out.print(i);
System.out.print(" ");
for(int j=0;j<chrom_length;j++)
System.out.print(pop[i].get_allele(j));
System.out.print(" ");
System.out.println(pop[i].get_fitness());
}
}
}

```

## BASE

```
package SYOPBASE;
import java.sql.*;
import javax.swing.*;
public class Base extends javax.swing.JDialog {
private static final String DRIVER =
"com.microsoft.jdbc.sqlserver.SQLServerDriver";
private static final String URL =
"jdbc:microsoft:sqlserver://localhost:1433;DatabaseName=Syopbase";
public Base() {
    initComponents();
    setBounds( 500, 300, 300, 250 );
}
/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */
private void initComponents() { //EN-BEGIN:initComponents
    frame = new javax.swing.JFrame();
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jTextField1 = new javax.swing.JTextField();
    jTextField2 = new javax.swing.JTextField();
    jTextField3 = new javax.swing.JTextField();
    jButton1 = new javax.swing.JButton();
    getContentPane().setLayout(null);
    setTitle("Base de Datos");
    addWindowListener(new java.awt.event.WindowAdapter() {
        public void windowClosing(java.awt.event.WindowEvent evt) {
            closeDialog(evt);
        }
    });
    jLabel1.setText("Nombre de la Base");
    getContentPane().add(jLabel1);
    jLabel1.setBounds(30, 30, 110, 16);
    jLabel2.setText("Numero de Discos");
    getContentPane().add(jLabel2);
    jLabel2.setBounds(30, 60, 110, 16);
    jLabel3.setText("Numero de Tablas");
    getContentPane().add(jLabel3);
    jLabel3.setBounds(30, 90, 110, 16);
    getContentPane().add(jTextField1);
```



```

jTextField1.requestFocus();
jTextField1.setBounds(160, 30, 100, 20);
getContentPane().add(jTextField2);
jTextField2.setBounds(160, 60, 100, 20);
getContentPane().add(jTextField3);
jTextField3.setBounds(160, 90, 100, 20);
jButton1.setBackground(new java.awt.Color(153, 153, 153));
jButton1.setText("Aceptar");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
getContentPane().add(jButton1);
jButton1.setBounds(100, 170, 79, 26);
pack();
}
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-
FIRST:event_jButton1ActionPerformed
    insertaBase();
    JOptionPane.showMessageDialog(frame, "Base Ingresada");
    setVisible(false);
    dispose();
    // NUEVO Disco(new javax.swing.JFrame(), true).show();
} //GEN-LAST:event_jButton1ActionPerformed
public void insertaBase () {
String m_query = "Insert INTO Base (Nombre, Numero_discos, Numero_tablas)
values(" + jTextField1.getText() + "," + jTextField2.getText() + "," +
jTextField3.getText() + ")";
    try {
//      Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
//      Connection conn = DriverManager.getConnection("jdbc:odbc:local");
        Class.forName(DRIVER);
Connection conn = DriverManager.getConnection(URL, "atapia", "atapia");
conn.setTransactionIsolation(Connection.TRANSACTION_READ_UNCOMMITTE
D);
        Statement stmt = conn.createStatement();
        stmt.executeUpdate(m_query);
        stmt.close();
        conn.close();
    }
    catch (Exception e) {
        e.printStackTrace();
        System.err.println("Load data error: "+e.toString());
    }
}

```

```
    }  
  }  
private void closeDialog(java.awt.event.WindowEvent evt) {//GEN-  
FIRST:event_closeDialog  
    setVisible(false);  
    dispose();  
} //GEN-LAST:event_closeDialog  
// Variables declaration - do not modify //GEN-BEGIN:variables  
private javax.swing.JButton jButton1;  
private javax.swing.JTextField jTextField3;  
private javax.swing.JTextField jTextField2;  
private javax.swing.JTextField jTextField1;  
private javax.swing.JLabel jLabel3;  
private javax.swing.JLabel jLabel2;  
private javax.swing.JLabel jLabel1;  
// End of variables declaration //GEN-END:variables  
private javax.swing.JFrame frame;  
}
```



**CIB-ESPOL**

## **CROMOSOMA**

```
package SYOPBASE;
import java.util.Random;
import java.io.* ;
public class Chromosoma implements Serializable
{
    private byte alleles[];           // Allele vector
    private int L;                    // Length of the allele vector
    private static Random r = new Random(); // Only the first time it is initialized
    // CONSTRUCTOR - FILL UP THE CONTENTS
    public Chromosoma(int length)
    {
        alleles = new byte[length];
        L = length;
        for (int i=0; i<length;i++)
            if(r.nextDouble()>0.5) // Returns values in [0..1]
                alleles[i] = 1;
            else
                alleles[i] = 0;
    }
    public void set_allele(int index, byte value)
    {
        alleles[index] = value;
    }
    public byte get_allele(int index)
    {
        return alleles[index];
    }
    public void print()
    {
        for(int i=0; i<L; i++)
            System.out.print(alleles[i]);
    }
}
// END OF CLASS: Chromosome
```

## PACKAGE SYOPBASE

```
package SYOPBASE;
public class Exe
{
public Exe(int d, int t) /*throws Exception*/ {
    int numtabla = t;
    int numdisco = d;
// int numtabla = 80; // numero genes
// int numdisco = 5; // longitud genes
    int popsize = 512; // Population size
    double pc = 0.8; // Probabilidad de cruzamiento
    double pm = 1.0/((double)((double)numtabla*(double)numdisco); // probabilidad de
mutacion
    double tf = (double)1 ;
    long MAX_ISTEPS = 100000;
    byte[] arrayOfByte = new byte[1000];
    Problema problema;
    problema = new ProblemaBase();
    problema.set_geneN(numtabla);
    problema.set_geneL(numdisco);
    problema.set_target_fitness(tf);
    Algoritmo ga;
    ga = new Algoritmo(problema, popsize, numtabla, numdisco, pc, pm);
    for (int step=0; step<MAX_ISTEPS; step++)
    {
        ga.go_one_step();
        System.out.print(step);
        System.out.print(" ");
        System.out.println(ga.get_bestf());
        if((problema.tf_known() &&
(ga.get_solution()).get_fitness()>=problema.get_target_fitness()))
        { System.out.print("Solucion encontrada despues de ");
            System.out.print(problema.get_fitness_counter());
            System.out.println(" evaluaciones");
            break;
        }
    }
}
// Imprime la solucion
for(int i=0;i<numtabla*numdisco;i++)
{ arrayOfByte [i] = (ga.get_solution()).get_allele(i);
// System.out.print( (ga.get_solution()).get_allele(i) + " ");
for( int j=1;j<numtabla*numdisco;j++)
if (i == (j*numdisco)-1)
System.out.print(" ");
```



**CIB-ESPOL**



```
}  
System.out.print("\n");  
  
for (int x=0;x < numtabla;x++)  
{ System.out.print("Tabla # " + x + ": ");  
  for (int y=0;y < numdisco;y++)  
  {  
    System.out.print( arrayOfByte [y + (x*numdisco)] + " ");  
  }  
  System.out.print("\n");  
}  
}  
}  
// END OF CLASS: Exe
```

## PROBLEMA

```
package SYOPBASE;
import java.util.Random;
public abstract class Problema // Optimizacion
{
    protected int GL=11; // longitud del den en binario
    protected int GN=1000; // numero de gen en string
    protected int CL=GN*GL; // longitud del cromosoma
    protected long fitness_counter; // numero de evaluaciones
    protected double target_fitness;
    protected boolean tf_known;
    protected static Random r = new Random(); // Randomizacion del gen
    public Problema() {
        CL = GN*GL;
        fitness_counter = 0;
        tf_known = false;
        target_fitness = -999999.9;
    }
    public int get_geneL() { return GL; }
    public int get_geneN() { return GN; }
    public void set_geneL(int gl) { GL = gl; CL=GN*GL; }
    public void set_geneN(int gn) { GN = gn; CL=GN*GL; }
    public long get_fitness_counter() { return fitness_counter; }
    public double get_target_fitness() { return target_fitness; }
    public boolean tf_known() { return tf_known; }
    public void set_target_fitness(double tf) {
        target_fitness = tf;
        tf_known = true;
    }
    public double evaluateStep(Individual Indiv) {
        fitness_counter++;
        return Evaluate(Indiv);
    }
    public abstract double Evaluate(Individual Indiv);
}
// END OF CLASS: Problem
```



**CIB-ESPOL**

# ANEXO H

MATRIZ POSIBLE CRUZAMIENTO DE TABLAS						
T1	1	0	1	0	0	A
T1	1	0	1	0	1	B
T1	1	0	1	0	0	A
T1	1	0	1	1	0	C
T1	1	0	1	1	1	D
T1	1	0	1	1	0	C
T1	1	0	0	0	0	E
T1	1	0	0	0	1	F
T1	1	0	0	0	0	E
T1	1	0	0	1	0	G
T1	1	0	0	1	1	H
T1	1	0	0	1	0	G
T1	1	0	1	0	0	A
T1	1	0	1	0	1	B
T1	1	0	1	0	0	A
T1	1	0	1	1	0	B
T1	1	0	1	1	1	D
T1	1	0	1	1	0	C
T1	1	0	0	0	0	E
T1	1	0	0	0	1	F
T1	1	0	0	0	0	E
T1	1	0	0	1	0	G
T1	1	0	0	1	1	H
T1	1	0	0	1	0	G
T1	1	0	1	0	0	A
T1	1	0	1	0	1	B
T1	1	0	1	0	0	A
T1	1	0	1	1	0	C
T1	1	0	1	1	1	D
T1	1	0	1	1	0	C
T1	1	0	0	0	0	E
T1	1	0	0	0	1	F
T1	1	0	0	0	0	E
T1	1	0	0	1	0	G
T1	1	0	0	1	1	H
T1	1	0	0	1	0	G
T1	1	0	1	0	0	A
T1	1	0	1	0	1	B
T1	1	0	1	0	0	A
T1	1	0	1	1	0	C
T1	1	0	1	1	1	D
T1	1	0	1	1	0	C
T1	1	0	0	0	0	E
T1	1	0	0	0	1	F
T1	1	0	0	0	0	E
T1	1	0	0	1	0	G
T1	1	0	0	1	1	H
T1	1	0	0	1	0	G

Tabla 3.8 Matriz posibles cruzamientos de tablas

# ANEXO J

Cotizacion empresa MAGNANI



ANEXO "D"  
TABLA DE CANTIDADES Y PRECIOS

ITEM	CANTIDAD	UNIDAD	DETALLE	MARCA Y MODELO	PRECIO	
					UNITARIO	TOTAL
01	7	Unid	ESTACION DE TRABAJO DIGITALIZACION	Computador Hurricane, ensamblado con la Plataforma INTEL D915CAYL	\$845,00	\$5.915,00
02	2	Unid	ESTACION DE TRABAJO REEMPLAZO	Computador Hurricane, ensamblado con la Plataforma INTEL D915GAVL	\$845,00	\$1.690,00
03	5	Unid	IMPRESORAS MATRICIALES 80 COLUMNAS	Lexmark, 2480, Part Number: 1215055	\$365,00	\$1.825,00
04	1	Unid	SCANNER DE ALTA VELOCIDAD DIGITALIZACION	Hewlett Packard, C8033A HP Scanjet 8290 Digital Flatbed Scanner	\$2.130,00	\$2.130,00
05	1	Unid	IMPRESORA DE INYECCION	Hewlett Packard, C8154A HP Inkjet 1200JD	\$295,00	\$295,00
06	2	Unid	IMPRESORA LASER	Hewlett Packard, Q55271 HP LaserJet 1320	\$560,00	\$1.120,00
07	7	Unid	UPS	APC, Back UPS BE725BB, 725VA	\$100,00	\$700,00
08	1	Unid	SOFTWARE DE APLICACION PARA DESARROLLO OPEN VISUAL STUDIO NET CD S ORIGINALES - DISPER	650-01293 Visual Studio NET Pro 2003 Win32 Spanish (Licencia OPEN)	\$520,00	\$520,00
09	1	Unid	SOFTWARE Y LICENCIA MICROSOFT OFFICE PROFESIONAL CD S ORIGINAL	Office Profesional 2003 Español (Paquete OEM para uso en equipos nuevos)	\$315,00	\$315,00
10	6	Unid	SOFTWARE Y LICENCIA MICROSOFT OFFICE STANDARD EDITION CD S ORIGINAL	Office Small Business 2003 Español (Paquete OEM para uso en equipos nuevos)	\$240,00	\$1.440,00
11	1	Unid	SEAGATE CRYSTAL REPORT VERSION 3	WRP2010 Business Object Version 10 (Standard Full Products)	\$200,00	\$200,00
12	1	Unid	UREAMWEAVER MAX 2004	QVAD370L020 Dreamweaver MX 2004 COMM Spanish Latin AM (Cajal)	\$390,00	\$390,00
13	3	Unid	SOFTWARE DE APLICACION PARA DESARROLLO OPEN VISUAL STUDIO NET CD S ORIGINALES - DISPER	650-01293 Visual Studio NET Pro 2003 Win32 Spanish (Licencia OPEN)	\$520,00	\$1.560,00
<b>SUBTOTAL</b>					<b>\$18.100,00</b>	<b>\$18.100,00</b>
IVA 12%					\$2.172,00	\$2.172,00
<b>TOTAL</b>					<b>\$20.272,00</b>	<b>\$20.272,00</b>

SON: VEINTE MIL DOSCIENTOS, SESENTA Y DOS, 997,09 U.S. DOLARES

**OBSERVACIONES:** Productos alternativos para uso con equipos existentes.-

08 a	1	Unid	SOFTWARE DE APLICACION PARA DESARROLLO OPEN VISUAL STUDIO NET CD S ORIGINALES - DISPER	650-01567 Visual Studio.NET Pro 2003 Win32 English CD Spcl EdIt. (Presentation Caja)	\$345,00
09 a	1	Unid	SOFTWARE Y LICENCIA MICROSOFT OFFICE PROFESIONAL CD S ORIGINAL	Office Pro 2003 Win32 Spanish OLP NL (Licencia OPEN)	\$440,00
10 a	6	Unid	SOFTWARE Y LICENCIA MICROSOFT OFFICE STANDARD EDITION CD S ORIGINAL	Office 2003 Win32 Spanish OLP NL (Licencia OPEN)	\$360,00



CIB-ESPOL

• Vélez #1002 y Pedro Moncayo • Fax: 04-2516989 • magnani@interactive.net.ec  
 • Edif. Centenario, Sts. Pisco y 502 • Tel: 04-2516997 • Guayaquil - Ecuador

*[Handwritten signature]*

# ANEXO K

Cotización Empresa TECNOPLUS

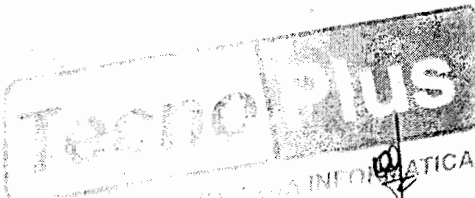


ANEXO "D"  
TABLA DE CANTIDADES Y PRECIOS

ITEM	CANT.	UND	DETALLE	P/UNITARIO	P/TOTAL
1	7	UND	ESTACION DE TRABAJO DIGITALIZACION Clones Pentium IV	899,00	6.293,00
2	2	UND	ESTACION DE TRABAJO REEMPLAZO Clones Pentium IV	899,00	1.798,00
3	5	UND	IMPRESORAS MATRICIALES 80 COLUMNAS Impresoras Marca Epson FX-880	365,00	1.825,00
4	1	UND	SCANNER DE ALTA VELOCIDAD DIGITALIZACION Escaner Marca Epson GT30000	4.300,00	4.300,00
5	1	UND	IMPRESORA INYECCION Impresora HP Deskjet 3745	70,00	70,00
6	2	UND	IMPRESORA LASER Impresora HP Laser Jet 1022	240,00	480,00
7	7	UND	UPS UPS marca Tripp Lite SmartPro USB	135,00	945,00
8	1	UND	Licencia OLP de Microsoft Visual Studio .NET Professional Edition 2003. Incluye medio de instalación original	580,00	560,00
9	1	UND	Licencia OLP de Microsoft Office Professional Edition 2003. Incluye medio de instalación original	480,00	480,00
10	6	UND	Licencia OLP de Microsoft Office Estándar Edition 2003. Incluye medio de instalación original	390,00	2.340,00
13	3	UND	Licencia OLP de Microsoft Visual Studio .NET Professional Edition 2003. Incluye medio de instalación original	560,00	1.680,00

Subtotal	20.771,00
IVA 12%	2.492,52
Total	23.263,52

Son: Veinte y tres mil doscientos sesenta y tres; 52/100 Dólares Americanos



Teodoro Wolf N16-07 y Padre Sojano (esq.) • Telf.: 2548 654 • Telefax: 2900 598  
e-mail: [tecnoplus@interactive.net.ec](mailto:tecnoplus@interactive.net.ec) • Quito - Ecuador  
[www.tecnoplus-ec.com](http://www.tecnoplus-ec.com)

# ANEXO L

## Cotización Empresa MC System

**MC**  
S T E M  
V I V S . A .

CLIENTE : DIRECCION GENERAL DEL PERSONAL DE LA ARMADA  
ATENCION: Calm. Valdemar Sánchez V  
FECHA : 30-Dic-04  
PROFORMA # : 2034

CANTIDAD	DESCRIPCION	P.UNITARIO	TOTAL
3	Licencias de SQL Server 2000	\$ 660.00	\$ 1.980,00
3	Licencias de Windows 2003 Server	\$ 712.45	\$ 2.137,35
1	Medio de instalación de SQL Server 2000 (CD)	\$ 28.00	\$ 28,00
1	Medio de instalación de Windows Server 2003 (CD)	\$ 28.00	\$ 28,00

Validez de la oferta : 15 días

Forma de pago: Contraentrega


Tiempo de entrega: Licencias Microsoft 5 días

Licencias y medio de instalación de Symantec 15 días

Medio de instalación de SQL Y Windows 15 días

Subtotal	\$ 4.173,35
12% IVA	\$ 500,80
<b>TOTAL</b>	<b>\$ 4.674,15</b>

Atentamente,

  
Srta Carmen Vaca Villón  
Cel 094640251

# ANEXO M

Cotización Empresa ORCOIN S.A.

**ORCOIN S.A.**  
Comercializadora de Consultoría Informática S.M.L

**Principal Guayaquil:** Ciudadela Kennedy Norte  
Calle Principal del Salado Solar 5 Manzana 907  
Telefax: 2680610 - 2680584

**Sucursal Bucay:** García Moreno s/n Intersección  
Sargento Eduardo Seis Teléfono: 2727474  
Bucay - Ecuador

**OrcoIn-Italia:** Via Monte S. Gabriele 30-33054  
Lignano Sabbiadoro (UD) Italy  
Phone: +39 (0)431-70523 Telefax: +39(0)431-70523

## COTIZACION 2004-12-28B

Atencion: DIRECCION GENERAL DEL PERSONAL  
Fecha: 28 DE DICIEMBRE DEL 2004

EN ATENCION A VUESTRO GENTIL REQUERIMIENTO, NOS PERMITIMOS COTIZAR LO SIGUIENTE

CANT.	DESCRIPCIÓN	P.UNITARIO	P.TOTAL
5	LICENCIA SOL SERVER 2000 ESTÁNDAR ESPAÑOL OL'PNC No. PARTE 228008	728,00	3.640,00
3	LICENCIA DE WINDOWS 2003 SERVER	786,00	2.358,00
86	NORTON CORPORATIVO NOVENA EDICION PARA EXTENSIONES QUE INCLUYE EL MAINTENANCE GOLD POR DOCE MESES ENTRA EN EL RANGO DE 25 / 99 LICENCIA QUE INCLUYE ESTACIONES Y SERVIDOR	36,00	3.096,00
1	MEDIO DE INSTALACION	30,00	30,00
1	CD ORIGINAL DE SOL SERVER 2000	28,00	28,00
1	CD ORIGINAL WINDOWS 2003 SERVER	28,00	28,00
	<b>SUBTOTAL</b>		<b>9.180,00</b>
	<b>IVA 12%</b>		<b>1.101,60</b>
	<b>TOTAL</b>		<b>10.281,60</b>

### CONDICIONES DE VENTA

VALIDEZ DE LA OFERTA : 25 DIAS  
FORMA DE PAGO : AL CONTADO  
TIEMPO DE ENTREGA : INMEDIATA

Atentamente

JOHNNY PASTORIZA  
GERENTE NACIONAL DE VENTAS

**ORCOIN S. A.**  
RUC. 0991511028001

E-mail: orcoincv@iud.it

# ANEXO N

## Cotización Empresa TECMOCOMP

TECMOCOMP

Dirección: El Morlán # 1270 y César Frank.

Tele-Fax: 233774 - 2409345

### PROFORMA

**Cilante:** DIGPER  
**Atención:** Calm. Valdemar SANCHEZ Vera.  
**Cargo:** Director del Personal de la Armada.  
**Ciudad:** Guayaquil

**Fecha:** 30 de Diciembre del 2004  
**Moneda:** Dólares  
**No.:** gr-721-140

ITEM	DESCRIPCION	CANT.	PRUNIT	P/TOTAL
1	Licencias de SQL Server 2K	5	\$690.00	\$3,450.00
2	Licencias de Win 2k3 Server	3	\$750.00	\$2,250.00
3	Licencias Symantec Antivirus V 9.0 La Gold	86	\$49.00	\$4,214.00
4	Medios Magnéticos de Instalación	3	\$33.00	\$99.00
	1 medio magnético para SQL Server 2000, 1 para Windows 2003 server y uno para Antivirus Symantec.			
SUBTOTAL				\$10,013.00
12% IVA				\$1,201.56
TOTAL				\$11,214.56

\* Validez de la oferta: 5 días

\* Forma de pago: 70% a la orden de compra, 30% contra entrega.

\* Tiempo de entrega: 15 días a partir de la orden de compra.

\* Garantía: 12 meses

Ing. Gustavo Ron Toledo  
Gerente General  
TECMOCOMP



CIB-ESPOL



# ANEXO O

## FACTURA MC SYSTEM

Francisco de Paula Icaza #407 y General Córdova,  
Edificio El Alfil, Piso 4, Oficina 407  
Teléfonos: 2304195 - 2304829  
2300022 - 099742672  
email: mcsystem@easynet.net.ec  
Guayaquil - Ecuador

**MC SYSTEM**  
VIV S.A.

R.U.C. 0992273321001

<b>FACTURA</b>
001-001- N° 0002389
Autorización SRI 1095971603

FECHA:	30-Dic-04		
CLIENTE:	DIRECCION GENERAL DEL PERSONAL DE LA ARMADA		
	DIRECCION AV-25 DE JULIO VIA PUERTO MARITIMO		
RUC/ICI:	0968562100001	TELEFONO:	GUIA REMISION:

CANTIDAD	DESCRIPCION	PUNITARIO	TOTAL US\$
3	Licencias de SQL Server 2000	660.00	1,980.00
3	Licencias de windows 2003 Server	712.45	2,137.35
1	Medio de instalación de SQL Server 2000 (CD)	28.00	28.00
1	Medio de instalación de windows Server 2003 (CI)	28.00	28.00

Favor emitir cheque a nombre de MC SYSTEM

MC SYSTEM  
CANCELADO  
Humberto Salazar  
26/1/05

che  
907

Son: Cuatro mil seiscientos setenta y cuatro con 15/100	<b>SUB - TOTAL US\$</b>	4,173.35
	Dólares USA	I.V.A. 0%
		I.V.A. 12%
	<b>TOTAL US\$</b>	500.80
		4,674.15

FORMA DE PAGO	RECIBI CONFORME
---------------	-----------------

Bertha Changkuón González, Imprenta Colonial RUC 0968282922001 Aut.1085  
Fecha de impresión: 08/2004, Fecha Caducidad: Agosto del 2005, #1801-2000

Original: Adquiriente / Copia colorada: Emisor / Copia amarilla: SRI, sin derecho a crédito tributario

# ANEXO P



## TRIGGER DE LA TABLA PERSONA

**CIB-ESPOL**

Para actualización (escritura) .....	1
Para un reingreso (sobre-escritura).....	48
Para un campo en especial de consulta (lectura).....	53

Estos trigger son de autoría de Escuela Politécnica Nacional, al igual que el Sistema integrado de Personal de la Armada

### Para actualización (escritura)

/\* Trigger para registrar los cambios en los campos realizados en sobre la tabla PERSONA

-----  
Para cada actualización realizada sobre algún campo de la tabla persona se guarda un registro en la tabla TRANSACCION con la información del campo actualizado y el valor anterior y el actual para este campo. Además se guarda el usuario responsable de la actualización y la fecha de la misma. Fecha de modificación : 01/08/2000

\*/

```
CREATE trigger upd_persona_ins
on PERSONA
for update
as
declare @rows int, @ANTERIOR_VALOR varchar(50), @ACTUAL_VALOR
varchar(50),
        @TRA_ID int, @PER_ID int, @USR_MOD varchar(10), @TRA_TAB_ID int,
        @LoginConexion varchar(128),
        @Usuario varchar(128),
        @Maquina varchar(128)

select @rows = @@rowcount
if @rows = 0
return

select @TRA_ID = isnull((select MAX(TRA_ID) from TRANSACCION),0)

-- Información del usuario y máquina de conexión
SELECT @LoginConexion = SYSTEM_USER
SELECT @Usuario = USER
```

```
SELECT @Maquina = HOST_NAME()
```

```
-- Código de Diplomado
```

```
if update (DIP_COD)
```

```
begin
```

```
    select @ANTERIOR_VALOR = isnull(d.DIP_COD, "")
```

```
    from deleted d
```

```
    select @ACTUAL_VALOR = isnull(i.DIP_COD, ""), @PER_ID = i.PER_ID,
```

```
@USR_MOD = i.PER_USR_MOD
```

```
    from inserted i
```

```
    if @ANTERIOR_VALOR <> @ACTUAL_VALOR
```

```
    begin
```

```
        select @TRA_ID = @TRA_ID + 1
```

```
        insert into TRANSACCION
```

```
        (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,  
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,  
USUARIO, TRA_OPERACION)
```

```
        values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "DIP_COD",
```

```
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
```

```
ltrim(@Maquina), ltrim(@Usuario), "M")
```

```
        -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
```

```
        update DES1.PERSONA set PER_WRST_MOD = @Maquina
```

```
        where PER_ID = @TRA_TAB_ID
```

```
    end
```

```
end
```

```
-- Estado Civil
```

```
if update (EST_CIV_COD)
```

```
begin
```

```
    select @ANTERIOR_VALOR = isnull(d.EST_CIV_COD, "")
```

```
    from deleted d
```

```
    select @ACTUAL_VALOR = isnull(i.EST_CIV_COD, ""), @PER_ID = i.PER_ID,
```

```
@USR_MOD = i.PER_USR_MOD
```

```
    from inserted i
```

```
    if @ANTERIOR_VALOR <> @ACTUAL_VALOR
```

```
    begin
```

```
        select @TRA_ID = @TRA_ID + 1
```

```
        insert into TRANSACCION
```

```
        (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,  
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,  
USUARIO, TRA_OPERACION)
```

```
        values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "EST_CIV_COD",
```

```
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
```

```
ltrim(@Maquina), ltrim(@Usuario), "M")
```

```

-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

-- Tipo de Sangre
if update (TIP_TIP_SAN_COD)
begin
select @ANTERIOR_VALOR = isnull(d.TIP_TIP_SAN_COD,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.TIP_TIP_SAN_COD,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"TIP_TIP_SAN_COD", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

```

```

-- Código de Brevet
if update (TIP2_TIP_BRE_COD)
begin
select @ANTERIOR_VALOR = isnull(d.TIP2_TIP_BRE_COD,"") CIB-ESPOL
from deleted d
select @ACTUAL_VALOR = isnull(i.TIP2_TIP_BRE_COD,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION

```



```

(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"TIP2_TIP_BRE_COD", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

-- Código de Tipo de Empleado

if update (TIP3\_TIP\_EMP\_COD)

begin

```

select @ANTERIOR_VALOR = isnull(d.TIP3_TIP_EMP_COD, "")
from deleted d
select @ACTUAL_VALOR = isnull(i.TIP3_TIP_EMP_COD, ""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin

```

```

select @TRA_ID = @TRA_ID + 1

```

```

insert into TRANSACCION

```

```

(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)

```

```

values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"TIP3_TIP_EMP_COD", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")

```

-- SE AFECTA CAMPO: ESTACION QUE MODIFICA

```

update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID

```

end

end

-- Código de Perfeccionamiento

if update (PER\_PERF\_COD)

begin

```

select @ANTERIOR_VALOR = isnull(d.PER_PERF_COD, "")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_PERF_COD, ""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i

```

```

if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
    select @TRA_ID = @TRA_ID + 1
    insert into TRANSACCION
    (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
    TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
    USUARIO, TRA_OPERACION)
    values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_PERF_COD",
    @ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
    ltrim(@Maquina), ltrim(@Usuario), "M")
    -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
    update DES1.PERSONA set PER_WRST_MOD = @Maquina
    where PER_ID = @TRA_TAB_ID
end
end

```



**CIB-ESPOL**

```

-- Código de Tipo de Sexo
if update (SEX_SEX_COD)
begin
    select @ANTERIOR_VALOR = isnull(d.SEX_SEX_COD, "")
    from deleted d
    select @ACTUAL_VALOR = isnull(i.SEX_SEX_COD, ""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
    from inserted i
    if @ANTERIOR_VALOR <> @ACTUAL_VALOR
    begin
        select @TRA_ID = @TRA_ID + 1
        insert into TRANSACCION
        (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
        TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
        USUARIO, TRA_OPERACION)
        values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "SEX_SEX_COD",
        @ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
        ltrim(@Maquina), ltrim(@Usuario), "M")
        -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
        update DES1.PERSONA set PER_WRST_MOD = @Maquina
        where PER_ID = @TRA_TAB_ID
    end
end

```

```

-- Código de País
if update (PAI_PAI_COD)
begin
    select @ANTERIOR_VALOR = isnull(d.PAI_PAI_COD, "")

```

```

from deleted d
select @ACTUAL_VALOR = isnull(i.PAI_PAI_COD,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PAI_PAI_COD",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Código de Condición de Trabajo
if update (CON_CON_TRA_COD)
begin
select @ANTERIOR_VALOR = isnull(d.CON_CON_TRA_COD,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.CON_CON_TRA_GOD,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"CON_CON_TRA_COD", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Código de División Política Ecuador
if update (DIV_DIV_COD)
begin
  select @ANTERIOR_VALOR = isnull(d.DIV_DIV_COD,"")
  from deleted d
  select @ACTUAL_VALOR = isnull(i.DIV_DIV_COD,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
  from inserted i
  if @ANTERIOR_VALOR <> @ACTUAL_VALOR
  begin
    select @TRA_ID = @TRA_ID + 1
    insert into TRANSACCION
      (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
      TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
      USUARIO, TRA_OPERACION)
      values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "DIV_DIV_COD",
      @ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
      ltrim(@Maquina), ltrim(@Usuario), "M")
    -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
    update DES1.PERSONA set PER_WRST_MOD = @Maquina
    where PER_ID = @TRA_TAB_ID
  end
end

```



**CIB-ESPOL**

```

-- Código de Calificación de Seguridad
if update (CAL_CAL_SEG_COD)
begin
  select @ANTERIOR_VALOR = isnull(d.CAL_CAL_SEG_COD,"")
  from deleted d
  select @ACTUAL_VALOR = isnull(i.CAL_CAL_SEG_COD,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
  from inserted i
  if @ANTERIOR_VALOR <> @ACTUAL_VALOR
  begin
    select @TRA_ID = @TRA_ID + 1
    insert into TRANSACCION
      (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
      TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
      USUARIO, TRA_OPERACION)
      values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
      "CAL_CAL_SEG_COD", @ANTERIOR_VALOR, @ACTUAL_VALOR,
      ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
    -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
    update DES1.PERSONA set PER_WRST_MOD = @Maquina
  end
end

```



```

    where PER_ID = @TRA_TAB_ID
end
end

-- Identificador de Gra ...
if update (PRO_PRO_GRA_IDEN)
begin
    select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PRO_PRO_GRA_IDEN),"")
    from deleted d
    select @ACTUAL_VALOR =
isnull(convert(varchar(128),i.PRO_PRO_GRA_IDEN),""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
    from inserted i
    if @ANTERIOR_VALOR <> @ACTUAL_VALOR
    begin
        select @TRA_ID = @TRA_ID + 1
        insert into TRANSACCION
        (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
        values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PRO_PRO_GRA_IDEN", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
        -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
        update DES1.PERSONA set PER_WRST_MOD = @Maquina
        where PER_ID = @TRA_TAB_ID
    end
end

-- Código de clasificación especialdad
if update (CLA_CLA_ESP_COD)
begin
    select @ANTERIOR_VALOR = isnull(d.CLA_CLA_ESP_COD,"")
    from deleted d
    select @ACTUAL_VALOR = isnull(i.CLA_CLA_ESP_COD,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
    from inserted i
    if @ANTERIOR_VALOR <> @ACTUAL_VALOR
    begin
        select @TRA_ID = @TRA_ID + 1
        insert into TRANSACCION

```

```

(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"CLA_CLA_ESP_COD", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

```

```

-- Código de grupo de persona
if update (GRU_GRU_PER_COD)
begin
select @ANTERIOR_VALOR = isnull(d.GRU_GRU_PER_COD,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.GRU_GRU_PER_COD,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"GRU_GRU_PER_COD", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

```

```

-- Identificador de Unidad Administrativa
if update (UNI_UNI_ADM_IDEN)
begin
print 'unidad_administrativa'
print @ANTERIOR_VALOR
select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.UNI_UNI_ADM_IDEN),"")
from deleted d

```



**CIB-ESPOL**

```

select @ACTUAL_VALOR =
isnull(convert(varchar(128),i.UNI_UNI_ADM_IDEN),""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"UNI_UNI_ADM_IDEN", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

-- Código de Tipo de Instrucción

```

if update (TIP4_TIP_INS_COD)
begin
select @ANTERIOR_VALOR = isnull(d.TIP4_TIP_INS_COD,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.TIP4_TIP_INS_COD,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"TIP4_TIP_INS_COD", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Código de Grupo Especialidad
if update (GRU2_GRU_ESP_COD)
begin
select @ANTERIOR_VALOR = isnull(d.GRU2_GRU_ESP_COD,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.GRU2_GRU_ESP_COD,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"GRU2_GRU_ESP_COD", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```



**CIB-ESPOL**

```

-- Código de Nivel
if update (NIV_NIV_COD)
begin
select @ANTERIOR_VALOR = isnull(d.NIV_NIV_COD,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.NIV_NIV_COD,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "NIV_NIV_COD",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina

```

```
where PER_ID = @TRA_TAB_ID
end
end
```

-- Código de Categoría

```
if update (CAT_CAT_COD)
```

```
begin
```

```
select @ANTERIOR_VALOR = isnull(d.CAT_CAT_COD, "")
```

```
from deleted d
```

```
select @ACTUAL_VALOR = isnull(i.CAT_CAT_COD, ""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
```

```
from inserted i
```

```
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
```

```
begin
```

```
select @TRA_ID = @TRA_ID + 1
```

```
insert into TRANSACCION
```

```
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
```

```
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "CAT_CAT_COD",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
```

```
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
```

```
update DES1.PERSONA set PER_WRST_MOD = @Maquina
```

```
where PER_ID = @TRA_TAB_ID
```

```
end
```

```
end
```

-- Cédula de Identidad

```
if update (PER_CED_IDE)
```

```
begin
```

```
select @ANTERIOR_VALOR = isnull(d.PER_CED_IDE, "")
```

```
from deleted d
```

```
select @ACTUAL_VALOR = isnull(i.PER_CED_IDE, ""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
```

```
from inserted i
```

```
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
```

```
begin
```

```
select @TRA_ID = @TRA_ID + 1
```

```
insert into TRANSACCION
```

```
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
```

```
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_CED_IDE",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
```

```
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end
```

```
-- Fecha del Ascenso Actual
if update (PER_FEC_ASC_ACT)
begin
select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_FEC_ASC_ACT),"")
from deleted d
select @ACTUAL_VALOR =
isnull(convert(varchar(128),i.PER_FEC_ASC_ACT),""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_FEC_ASC_ACT", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end
```



**CIB-ESPOL**

```
-- Número de Cuenta Bancaria
if update (PER_NO_CUE_BCO)
begin
select @ANTERIOR_VALOR = isnull(d.PER_NO_CUE_BCO,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_NO_CUE_BCO,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
```

```

begin
  select @TRA_ID = @TRA_ID + 1
  insert into TRANSACCION
    (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
    TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
    USUARIO, TRA_OPERACION)
    values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
    "PER_NO_CUE_BCO", @ANTERIOR_VALOR, @ACTUAL_VALOR,
    ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
  -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
  update DES1.PERSONA set PER_WRST_MOD = @Maquina
  where PER_ID = @TRA_TAB_ID
end
end

```

-- Grado para Sueldos

```

if update (PER_GRA_SUE)
  begin
    select @ANTERIOR_VALOR =
    isnull(convert(varchar(128),d.PER_GRA_SUE), "")
    from deleted d
    select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_GRA_SUE), ""),
    @PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
    from inserted i
    if @ANTERIOR_VALOR <> @ACTUAL_VALOR
    begin
      select @TRA_ID = @TRA_ID + 1
      insert into TRANSACCION
        (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
        TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
        USUARIO, TRA_OPERACION)
        values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_GRA_SUE",
        @ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
        ltrim(@Maquina), ltrim(@Usuario), "M")
      -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
      update DES1.PERSONA set PER_WRST_MOD = @Maquina
      where PER_ID = @TRA_TAB_ID
    end
  end
end

```

-- Grado Actual

```

if update (PER_GRA_ACT)
  begin
    select @ANTERIOR_VALOR = isnull(d.PER_GRA_ACT, "")

```

```

from deleted d
select @ACTUAL_VALOR = isnull(i.PER_GRA_ACT,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_GRA_ACT",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```



**CIB-ESPOL**

```

-- Nombres
if update (PER_APE)
begin
select @ANTERIOR_VALOR = isnull(d.PER_APE,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_APE,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_APE",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```



```

-- Fecha de Nacimiento
if update (PER_FEC_NAC)
begin
select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_FEC_NAC),'')
from deleted d
select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_FEC_NAC),''),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_FEC_NAC",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Fecha de Ingreso
if update (PER_FEC_ING)
begin
select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_FEC_ING),'')
from deleted d
select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_FEC_ING),''),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_FEC_ING",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")

```

```

-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

-- Número de Pasaporte
if update (PER_NRO_PAS)
begin
select @ANTERIOR_VALOR = isnull(d.PER_NRO_PAS,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_NRO_PAS,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_NRO_PAS",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

-- Ubicación en el Grado
if update (PER_UBI_GRA)
begin
select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_UBI_GRA),"")
from deleted d
select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_UBI_GRA,""),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION

```



**CIB-ESPOL**

```

(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_UBI_GRA",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Indicador de A Disposición
if update (PER_A_DIS)
begin
select @ANTERIOR_VALOR = isnull(d.PER_A_DIS, "")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_A_DIS, ""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_A_DIS",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Indicador de Baja
if update (PER_IND_BAJ)
begin
select @ANTERIOR_VALOR = isnull(d.PER_IND_BAJ, "")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_IND_BAJ, ""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i

```

```

if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
    select @TRA_ID = @TRA_ID + 1
    insert into TRANSACCION
        (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
        TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
        USUARIO, TRA_OPERACION)
        values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_IND_BAJ",
        @ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
        ltrim(@Maquina), ltrim(@Usuario), "M")
    -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
    update DES1.PERSONA set PER_WRST_MOD = @Maquina
    where PER_ID = @TRA_TAB_ID
end
end

```

-- Fecha de Ultima Baja

```

if update (PER_FEC_ULT_BAJ)
begin
    select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_FEC_ULT_BAJ), "")
    from deleted d
    select @ACTUAL_VALOR =
isnull(convert(varchar(128),i.PER_FEC_ULT_BAJ), ""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
    from inserted i
    if @ANTERIOR_VALOR <> @ACTUAL_VALOR
    begin
        select @TRA_ID = @TRA_ID + 1
        insert into TRANSACCION
            (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
            TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
            USUARIO, TRA_OPERACION)
            values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
            "PER_FEC_ULT_BAJ", @ANTERIOR_VALOR, @ACTUAL_VALOR,
            ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
        -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
        update DES1.PERSONA set PER_WRST_MOD = @Maquina
        where PER_ID = @TRA_TAB_ID
    end
end

```



**CIB-ESPOL**

-- Fecha de Ultimo Ingreso

```

if update (PER_FEC_ULT_ING)

```

```

begin
select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_FEC_ULT_ING),"")
from deleted d
select @ACTUAL_VALOR =
isnull(convert(varchar(128),i.PER_FEC_ULT_ING),""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_FEC_ULT_ING", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

-- Indicador de ha sido Comandante
if update (PER_IND_COM)
begin
select @ANTERIOR_VALOR = isnull(d.PER_IND_COM,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_IND_COM,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_IND_COM",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina

```

```

where PER_ID = @TRA_TAB_ID
end
end

```

-- Indicador de Autorización de Matrimonio

```

if update (PER_IND_MAT)
begin
select @ANTERIOR_VALOR = isnull(d.PER_IND_MAT, "")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_IND_MAT, ""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_IND_MAT",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```



**CIB-ESPOL**

-- Indicador de Número de Ascenso

```

if update (PER_IND_NO_ASC)
begin
select @ANTERIOR_VALOR = isnull(d.PER_IND_NO_ASC, "")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_IND_NO_ASC, ""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)

```

```

        values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_IND_NO_ASC", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
        update DES1.PERSONA set PER_WRST_MOD = @Maquina
        where PER_ID =@TRA_TAB_ID
        end
    end

-- Indicador de Pro ...
if update (PER_IND_PRO)
    begin
        select @ANTERIOR_VALOR = isnull(d.PER_IND_PRO, "")
        from deleted d
        select @ACTUAL_VALOR = isnull(i.PER_IND_PRO, ""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
        from inserted i
        if @ANTERIOR_VALOR <> @ACTUAL_VALOR
            begin
                select @TRA_ID = @TRA_ID + 1
                insert into TRANSACCION
                (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
                values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_IND_PRO",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
                -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
                update DES1.PERSONA set PER_WRST_MOD = @Maquina
                where PER_ID =@TRA_TAB_ID
            end
        end

-- Contador de Suspensiones
if update (PER_IND_SUS)
    begin
        select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_IND_SUS), "")
        from deleted d
        select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_IND_SUS), ""),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
        from inserted i
        if @ANTERIOR_VALOR <> @ACTUAL_VALOR
            begin

```

```

select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_IND_SUS",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Indicador de Fallecido
if update (PER_IND_FAL)
begin
select @ANTERIOR_VALOR = isnull(d.PER_IND_FAL, "")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_IND_FAL, ""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_IND_FAL",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Fecha de Fallecimiento
if update (PER_FEC_FAL)
begin
select @ANTERIOR_VALOR =
isnull(convert(varchar(128), d.PER_FEC_FAL), "")
from deleted d

```



```

select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_FEC_FAL),""),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_FEC_FAL",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID=@TRA_TAB_ID
end
end

```

-- Calificación Anual

```

if update (PER_CAL_ANU)
begin
select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_CAL_ANU), "")
from deleted d
select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_CAL_ANU),""),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_CAL_ANU",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID=@TRA_TAB_ID
end
end

```

```

-- Número de Partida Actual
if update (PER_NO_PAR_ACT)
begin
select @ANTERIOR_VALOR = isnull(d.PER_NO_PAR_ACT,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_NO_PAR_ACT,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_NO_PAR_ACT", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```



**CIB-ESPOL**

```

-- Número de Afiliación
if update (PER_NO_AFIL)
begin
select @ANTERIOR_VALOR = isnull(d.PER_NO_AFIL,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_NO_AFIL,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_NO_AFIL",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina

```

```

    where PER_ID =@TRA_TAB_ID
end
end

-- Reconocimiento Anterior
if update (PER_REC_ANT)
begin
select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_REC_ANT),"")
from deleted d
select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_REC_ANT),""),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_REC_ANT",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

-- Tiempo de servicio Personal en Baja
if update (PER_TIE_SER_BAJ)
begin
select @ANTERIOR_VALOR = isnull(d.PER_TIE_SER_BAJ,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_TIE_SER_BAJ,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)

```

```

values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_TIE_SER_BAJ", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

-- Tiempo de Servicio Reconocido Evaluación
if update (PER_TIE_SER_REC)
begin
select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_TIE_SER_REC),"")
from deleted d
select @ACTUAL_VALOR =
isnull(convert(varchar(128),i.PER_TIE_SER_REC,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_TIE_SER_REC", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

-- Esp ces ...
if update (PER_ESP_CES)
begin
select @ANTERIOR_VALOR = isnull(d.PER_ESP_CES,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_ESP_CES,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR

```



**CIB-ESPOL**

```

begin
  select @TRA_ID = @TRA_ID + 1
  insert into TRANSACCION
    (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
    TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
    USUARIO, TRA_OPERACION)
    values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_ESP_CES",
    @ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
    ltrim(@Maquina), ltrim(@Usuario), "M")
  -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
  update DES1.PERSONA set PER_WRST_MOD = @Maquina
  where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Fecha de Graduación
if update (PER_FEC_GRA)
  begin
    select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_FEC_GRA),"")
    from deleted d
    select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_FEC_GRA),""),
    @PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
    from inserted i
    if @ANTERIOR_VALOR <> @ACTUAL_VALOR
    begin
      select @TRA_ID = @TRA_ID + 1
      insert into TRANSACCION
        (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
        TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
        USUARIO, TRA_OPERACION)
        values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_FEC_GRA",
        @ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
        ltrim(@Maquina), ltrim(@Usuario), "M")
      -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
      update DES1.PERSONA set PER_WRST_MOD = @Maquina
      where PER_ID = @TRA_TAB_ID
    end
  end
end

```

```

-- Gra gra ...
if update (PER_GRA_GRA)
  begin
    select @ANTERIOR_VALOR = isnull(d.PER_GRA_GRA,"")

```

```

from deleted d
select @ACTUAL_VALOR = isnull(i.PER_GRA_GRA,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_GRA_GRA",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DESI.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID=@TRA_TAB_ID
end
end

```

```

-- Fecha de pro asc ...
if update (PER_FEC_PRO_ASC)
begin
select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_FEC_PRO_ASC),"")
from deleted d
select @ACTUAL_VALOR =
isnull(convert(varchar(128),i.PER_FEC_PRO_ASC),""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_FEC_PRO_ASC", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DESI.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID=@TRA_TAB_ID
end
end

```



**CIB-ESPOL**

```

end

-- Fecha ...
if update (PER_FEC_FIN)
begin
select @ANTERIOR_VALOR = isnull(convert(varchar(128),d.PER_FEC_FIN),"")
from deleted d
select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_FEC_FIN),""),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_FEC_FIN",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Indicador de Subjude
if update (PER_IND_SUBJ)
begin
select @ANTERIOR_VALOR = isnull(d.PER_IND_SUBJ,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_IND_SUBJ,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)

```

```
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_IND_SUBJ",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
```

```
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID=@TRA_TAB_ID
end
end
```

-- Promoción

```
if update (PER_PROM)
```

```
begin
```

```
select @ANTERIOR_VALOR = isnull(d.PER_PROM,"")
```

```
from deleted d
```

```
select @ACTUAL_VALOR = isnull(i.PER_PROM,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
```

```
from inserted i
```

```
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
```

```
begin
```

```
select @TRA_ID = @TRA_ID + 1
```

```
insert into TRANSACCION
```

```
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
```

```
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_PROM",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
```

```
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
```

```
update DES1.PERSONA set PER_WRST_MOD = @Maquina
```

```
where PER_ID=@TRA_TAB_ID
```

```
end
```

```
end
```

-- TM Total ...

```
if update (PER_TM_TOT)
```

```
begin
```

```
select @ANTERIOR_VALOR = isnull(d.PER_NO_PAR_ACT,"")
```

```
from deleted d
```

```
select @ACTUAL_VALOR = isnull(i.PER_NO_PAR_ACT,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
```

```
from inserted i
```

```
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
```

```
begin
```

```
select @TRA_ID = @TRA_ID + 1
```



**CIB-ESPOL**



```

insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_NO_PAR_ACT", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

```

```

-- Cuota de Eliminación
if update (PER_CUO_ELI)
begin
select @ANTERIOR_VALOR = isnull(d.PER_CUO_ELI,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_CUO_ELI,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i

```

```

if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_CUO_ELI",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

```

```

-- Estado
if update (PER_EST)
begin
select @ANTERIOR_VALOR = isnull(d.PER_EST,"")
from deleted d

```



**CIB-ESPOL**

```

select @ACTUAL_VALOR = isnull(i.PER_EST,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
if ltrim(@ACTUAL_VALOR) = "E"
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_EST",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "E")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
--update DES1.PERSONA set PER_WRST_MOD = @Maquina
--where PER_ID =@TRA_TAB_ID
else
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_EST",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

```

-- Es Docente

```

if update (PER_DOCENTE)
begin
select @ANTERIOR_VALOR = isnull(d.PER_DOCENTE,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_DOCENTE,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION

```



**CIB-ESPOL**

```
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,  
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,  
USUARIO, TRA_OPERACION)
```

```
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_DOCENTE",  
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),  
ltrim(@Maquina), ltrim(@Usuario), "M")
```

```
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
```

```
update DES1.PERSONA set PER_WRST_MOD = @Maquina
```

```
where PER_ID = @TRA_TAB_ID
```

```
end
```

```
end
```

```
-- Tiempo ...
```

```
if update (PER_TCON_SUE)
```

```
begin
```

```
select @ANTERIOR_VALOR =
```

```
isnull(convert(varchar(128),d.PER_TCON_SUE),"")
```

```
from deleted d
```

```
select @ACTUAL_VALOR =
```

```
isnull(convert(varchar(128),i.PER_TCON_SUE),""), @PER_ID = i.PER_ID,
```

```
@USR_MOD = i.PER_USR_MOD
```

```
from inserted i
```

```
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
```

```
begin
```

```
select @TRA_ID = @TRA_ID + 1
```

```
insert into TRANSACCION
```

```
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,  
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,  
USUARIO, TRA_OPERACION)
```

```
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_TCON_SUE",  
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),  
ltrim(@Maquina), ltrim(@Usuario), "M")
```

```
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
```

```
update DES1.PERSONA set PER_WRST_MOD = @Maquina
```

```
where PER_ID = @TRA_TAB_ID
```

```
end
```

```
end
```

```
-- gru_gru_esp_cod
```

```
if update (gru_gru_esp_cod)
```

```
begin
```

```
select @ANTERIOR_VALOR = isnull(d.gru_gru_esp_cod,"")
```

```
from deleted d
```

```

select @ACTUAL_VALOR = isnull(i.gru_gru_esp_cod,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "gru_gru_esp_cod",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```



**CIB-ESPOL**

```

-- Cuota de eliminación para evaluación
if update (per_cuo_eli_eva)
begin
select @ANTERIOR_VALOR = isnull(d.per_cuo_eli_eva,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.per_cuo_eli_eva,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "per_cuo_eli_eva",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

-- Número llamada con ...

```

if update (per_num_lla_con)
begin
select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.per_num_lla_con),"")
from deleted d
select @ACTUAL_VALOR = isnull(convert(varchar(128),i.per_num_lla_con),""),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "per_num_lla_con",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

-- Indicador Tec ...

```

if update (per_ind_tec)
begin
select @ANTERIOR_VALOR = isnull(convert(varchar(128),d.per_ind_tec),"")
from deleted d
select @ACTUAL_VALOR = isnull(convert(varchar(128),i.per_ind_tec),""),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "per_ind_tec",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina

```

```

    where PER_ID = @TRA_TAB_ID
  end
end

-- Promoción de ascenso
if update (per_prom_asc)
  begin
    select @ANTERIOR_VALOR = isnull(convert(varchar(128),d.per_prom_asc),"")
    from deleted d
    select @ACTUAL_VALOR = isnull(convert(varchar(128),i.per_prom_asc),""),
    @PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
    from inserted i
    if @ANTERIOR_VALOR <> @ACTUAL_VALOR
      begin
        select @TRA_ID = @TRA_ID + 1
        insert into TRANSACCION
        (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
        TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
        USUARIO, TRA_OPERACION)
        values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "per_prom_asc",
        @ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
        ltrim(@Maquina), ltrim(@Usuario), "M")
        -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
        update DES1.PERSONA set PER_WRST_MOD = @Maquina
        where PER_ID = @TRA_TAB_ID
      end
    end

-- per_apt_asc ...
if update (per_apt_asc)
  begin
    select @ANTERIOR_VALOR = isnull(d.per_apt_asc,"")
    from deleted d
    select @ACTUAL_VALOR = isnull(i.per_apt_asc,""), @PER_ID = i.PER_ID,
    @USR_MOD = i.PER_USR_MOD
    from inserted i
    if @ANTERIOR_VALOR <> @ACTUAL_VALOR
      begin
        select @TRA_ID = @TRA_ID + 1
        insert into TRANSACCION
        (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
        TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
        USUARIO, TRA_OPERACION)

```



**CIB-ESPOL**

```

    values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "per_apt_asc",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
    update DES1.PERSONA set PER_WRST_MOD = @Maquina
    where PER_ID=@TRA_TAB_ID
    end
end

-- Fecha de Ingreso para Sueldos
if update (PER_FEC_ING_SUE)
    print 'error'
    begin
        select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_FEC_ING_SUE), "")
        from deleted d
        select @ACTUAL_VALOR =
isnull(convert(varchar(128),i.PER_FEC_ING_SUE), ""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
        from inserted i
        if @ANTERIOR_VALOR <> @ACTUAL_VALOR
        begin
            select @TRA_ID = @TRA_ID + 1
            insert into TRANSACCION
            (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
            values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_FEC_ING_SUE", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
            update DES1.PERSONA set PER_WRST_MOD = @Maquina
            where PER_ID=@TRA_TAB_ID
            end
        end

-- Reparto de Cobro
if update (PER_REP_COB)
    begin
        select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_REP_COB), "")
        from deleted d
        select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_REP_COB), ""),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD

```

```

from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_REP_COB",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```



**CIB-ESPOL**

```

-- Indicador de Cobro
if update (PER_IND_COB)
begin
select @ANTERIOR_VALOR = isnull(d.PER_IND_COB, "")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_IND_COB, ""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR .
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_IND_COB",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Indicador de Subsidio Matrimonial
if update (PER_SUB_MAT)
begin

```



```

select @ANTERIOR_VALOR = isnull(d.PER_SUB_MAT,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_SUB_MAT,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION

(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_SUB_MAT",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

-- Indicador de Subsidio Matrimonial
if update (PER_SUB_FAM)
begin
select @ANTERIOR_VALOR = isnull(d.PER_SUB_FAM,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_SUB_FAM,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION

(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_SUB_FAM",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

```

```

end

-- Tipo de Rancho
if update (PER_TIP_RAN)
begin
select @ANTERIOR_VALOR = isnull(d.PER_TIP_RAN,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_TIP_RAN,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_TIP_RAN",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DESI.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

```



**CIB-ESPOL**

```

-- Forma de Pago
if update (PER_FOR_PAG)
begin
select @ANTERIOR_VALOR = isnull(d.PER_FOR_PAG,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_FOR_PAG,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_FOR_PAG",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")

```

```

-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
  update DES1.PERSONA set PER_WRST_MOD = @Maquina
  where PER_ID =@TRA_TAB_ID
  end
end

-- Tiempo de Servicio para sueldos (imposiciones )
if update (PER_TREC_SUE)
  begin
  select @ANTERIOR_VALOR = isnull(d.PER_TREC_SUE,"")
  from deleted d
  select @ACTUAL_VALOR = isnull(i.PER_TREC_SUE,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
  from inserted i
  if @ANTERIOR_VALOR <> @ACTUAL_VALOR
  begin
  select @TRA_ID = @TRA_ID + 1
  insert into TRANSACCION
  (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
  values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_TREC_SUE",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
  -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
  update DES1.PERSONA set PER_WRST_MOD= @Maquina
  where PER_ID =@TRA_TAB_ID
  end
end

-- PER_UBI_SIN_VAC
if update (PER_UBI_SIN_VAC)
  begin
  select @ANTERIOR_VALOR = isnull(d.PER_UBI_SIN_VAC,"")
  from deleted d
  select @ACTUAL_VALOR = isnull(i.PER_UBI_SIN_VAC,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
  from inserted i
  if @ANTERIOR_VALOR <> @ACTUAL_VALOR
  begin
  select @TRA_ID = @TRA_ID + 1
  insert into TRANSACCION

```

```
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
```

```
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_UBI_SIN_VAC", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
```

```
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
```

```
update DES1.PERSONA set PER_WRST_MOD = @Maquina
```

```
where PER_ID =@TRA_TAB_ID
```

```
end
```

```
end
```

```
-- Calificación de Tripulantes
```

```
if update (CAL_TRI_COD)
```

```
begin
```

```
select @ANTERIOR_VALOR = isnull(d.CAL_TRI_COD,"")
```

```
from deleted d
```

```
select @ACTUAL_VALOR = isnull(i.CAL_TRI_COD,""), @PER_ID =
```

```
i.PER_ID, @USR_MOD = i.PER_USR_MOD
```

```
from inserted i
```

```
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
```

```
begin
```

```
select @TRA_ID = @TRA_ID + 1
```

```
insert into TRANSACCION
```

```
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
```

```
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "CAL_TRI_COD",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
```

```
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
```

```
update DES1.PERSONA set PER_WRST_MOD = @Maquina
```

```
where PER_ID =@TRA_TAB_ID
```

```
end
```

```
end
```

```
-- Leva de Conscripción
```

```
if update (PER_LEVA)
```

```
begin
```

```
select @ANTERIOR_VALOR = isnull(d.PER_LEVA,"")
```

```
from deleted d
```

```
select @ACTUAL_VALOR = isnull(i.PER_LEVA,""), @PER_ID = i.PER_ID,
```

```
@USR_MOD = i.PER_USR_MOD
```

```
from inserted i
```



**CIB-ESPOL**

```

if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
    select @TRA_ID = @TRA_ID + 1
    insert into TRANSACCION
        (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
        TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
        USUARIO, TRA_OPERACION)
        values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_LEVA",
        @ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
        ltrim(@Maquina), ltrim(@Usuario), "M")
    -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
    update DES1.PERSONA set PER_WRST_MOD = @Maquina
    where PER_ID = @TRA_TAB_ID
end
end

```

```

-- Pensión de Retiro
if update (PER_PEN_RET)
begin
    select @ANTERIOR_VALOR = isnull(d.PER_PEN_RET, "")
    from deleted d
    select @ACTUAL_VALOR = isnull(i.PER_PEN_RET, ""), @PER_ID = i.PER_ID,
    @USR_MOD = i.PER_USR_MOD
    from inserted i
    if @ANTERIOR_VALOR <> @ACTUAL_VALOR
    begin
        select @TRA_ID = @TRA_ID + 1
        insert into TRANSACCION
            (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
            TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
            USUARIO, TRA_OPERACION)
            values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_PEN_RET",
            @ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
            ltrim(@Maquina), ltrim(@Usuario), "M")
        -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
        update DES1.PERSONA set PER_WRST_MOD = @Maquina
        where PER_ID = @TRA_TAB_ID
    end
end

```

```

-- Fecha de Posesión
if update (PER_FEC_POS)
begin

```

```

select @ANTERIOR_VALOR =
isnull(convert(varchar(128),d.PER_FEC_POS),"")
from deleted d
select @ACTUAL_VALOR = isnull(convert(varchar(128),i.PER_FEC_POS),""),
@PER_ID = i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_FEC_POS",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

```



**CIB-ESPOL**

```

-- Tiempo muerto entre contratos y licencias sin sueldo
if update (PER_TIE_MUE_LIC)
begin
select @ANTERIOR_VALOR = isnull(d.PER_TIE_MUE_LIC,"")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_TIE_MUE_LIC,""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_TIE_MUE_LIC", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID =@TRA_TAB_ID
end
end

```

```

end

-- Tiempo reconocido en servicio anterior
if update (PER_AUX)
begin
select @ANTERIOR_VALOR = isnull(d.PER_AUX, "")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_AUX, ""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "PER_AUX",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
update DES1.PERSONA set PER_WRST_MOD = @Maquina
where PER_ID = @TRA_TAB_ID
end
end

-- Promoción de Ascenso Empleados Civiles
if update (PER_PRO_ASC_CIV)
begin
select @ANTERIOR_VALOR = isnull(d.PER_PRO_ASC_CIV, "")
from deleted d
select @ACTUAL_VALOR = isnull(i.PER_PRO_ASC_CIV, ""), @PER_ID =
i.PER_ID, @USR_MOD = i.PER_USR_MOD
from inserted i
if @ANTERIOR_VALOR <> @ACTUAL_VALOR
begin
select @TRA_ID = @TRA_ID + 1
insert into TRANSACCION
(TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID,
"PER_PRO_ASC_CIV", @ANTERIOR_VALOR, @ACTUAL_VALOR,
ltrim(@LoginConexion), getdate(), ltrim(@Maquina), ltrim(@Usuario), "M")

```

```

-- SE AFECTA CAMPO: ESTACION QUE MODIFICA
  update DESI.PERSONA set PER_WRST_MOD = @Maquina
  where PER_ID =@TRA_TAB_ID
  end
end

-- Per_nom_cor
if update (per_nom_cor)
  begin
  select @ANTERIOR_VALOR = isnull(d.per_nom_cor,"")
  from deleted d
  select @ACTUAL_VALOR = isnull(i.per_nom_cor,""), @PER_ID = i.PER_ID,
@USR_MOD = i.PER_USR_MOD
  from inserted i
  if @ANTERIOR_VALOR <> @ACTUAL_VALOR
  begin
  select @TRA_ID = @TRA_ID + 1
  insert into TRANSACCION
  (TRA_ID, TRA_TABLA, TRA_PER_ID, TRA_TAB_ID, TRA_CAMPO,
TRA_VALOR_ANT, TRA_VALOR_ACT, USR_MOD, FEC_MOD, HOSTNAME,
USUARIO, TRA_OPERACION)
  values (@TRA_ID, "PERSONA", @PER_ID, @PER_ID, "per_nom_cor",
@ANTERIOR_VALOR, @ACTUAL_VALOR, ltrim(@LoginConexion), getdate(),
ltrim(@Maquina), ltrim(@Usuario), "M")
  -- SE AFECTA CAMPO: ESTACION QUE MODIFICA
  update DESI.PERSONA set PER_WRST_MOD = @Maquina
  where PER_ID =@TRA_TAB_ID
  end
end

return

```



**CIB-ESPOL**



## Para un reingreso (sobre-escritura)

```
CREATE trigger trig_ins_persona_upd
/* Trigger de actualización de tablas originado por ingresos y reingresos de Personal
```

-----  
-----  
Al insertar un registro en la tabla persona tenemos un ingreso de personal  
Al modificar la fecha de ultimo ingreso y el indicador de baja de un registro existente tenemos un reingreso de personal.

En estos dos casos se genera un registro en las tablas histórico\_personal, y en la de :

- especialidad y ascensos\_navales para el personal de oficiales y tripulantes
- ascenso\_civil para el personal civil
- ascensos\_navales para el personal de grumetes y guardiamarinas.
- servicio\_anterior al personal civil que fue militar y que reingresa a la institución como civil.

Si se actualiza la partida presupuestaria (campo per\_no\_par\_act) en la tabla persona se actualiza el indicador de cobro(per\_ind\_cob). Si el código de la partida presupuestaria es '01' el indicador de cobro es 'S', caso contrario el indicador de cobro es 'N'. Este dato es importante para el Sistema de Sueldos ya que solo cobra los de partida '01'. Si se actualiza el grupo persona se consulta el tipo de rancho perteneciente y se actualiza el campo de tipo rancho (tip\_ran\_id), dato necesario para sueldos.

-----  
----- \*/

```
--CREATE trigger trig_ins_persona_upd
on PERSONA
for insert, update
as
declare @rows int, @cont int
declare @vper_id integer, @fec_ingreso datetime, @grado varchar(3), @usuario
varchar(15),
        @instruccion varchar(6), @grupoper varchar(2), @partida varchar(2), @nivel
varchar(2),
        @categoria varchar(2), @gruposp varchar(4), @fec_grado datetime,
@ubica_grado int,
        @cedula varchar(10), @ser_ant varchar(10), @est_actual varchar(1),
@fecha_crea datetime,
        @anterior_valor varchar(2), @tip_ran integer, @gruperd varchar(2), @gra_sue
int, @reg_ant int, @tip_con varchar(2)
```

```
select @rows = @@rowcount
if @rows = 0
    return
```

```
if exists(select * from deleted) and @rows > 1
    begin
        -- "No es permitido actualizar varias filas"
        rollback tran
        return
    end
```

-- Leer los datos ingresados

```
select @vper_id = i.per_id,
       @fec_ingreso = i.per_fec_ult_ing,
       @grado = i.per_gra_act,
       @usuario = isnull(i.per_usr_mod,i.per_usr_crea),
       @instruccion = i.tip4_tip_ins_cod,
       @gruposesp = i.gru2_gru_esp_cod,
       @grupoper = i.gru_gru_per_cod,
       @partida = i.per_no_par_act,
       @nivel = i.niv_niv_cod,
       @categoria = i.cat_cat_cod,
       @fec_grado = i.per_fec_gra,
       @ubica_grado = i.per_ubi_gra,
       @cedula = i.per_ced_ide,
       @ser_ant = i.per_tie_ser_baj,
       @est_actual = i.per_est,
       @fecha_crea = i.per_fec_crea,
       @tip_con = i.con_con_tra_cod
```

```
From inserted i, persona p
Where i.per_id = p.per_id
```



--Si se actualiza la partida presupuestaria (per\_no\_par\_act) actualizar indicador de cobro (per\_ind\_cob)

```
if update(per_no_par_act)
    begin
        if @partida = '01'
            update des1.persona set per_ind_cob = 'S' where per_id = @vper_id
        else
            update des1.persona set per_ind_cob = 'N' where per_id = @vper_id
    end
```

-- Si se actualiza el grupo persona

```

if update (GRU_GRU_PER_COD)
begin
select @gruperd = isnull(d.GRU_GRU_PER_COD,"")
from deleted d
if @gruperd <> @grupoper
begin
select @tip_ran = tip_ran_id
from des1.tipo_rancho
where gru_per_cod = @grupoper
print @tip_ran
update des1.persona set tip_ran_id = @tip_ran WHERE PER_ID = @vper_id
end
end

```

```

-- Si se actualiza el grado actual
if update (PER_GRA_ACT)
begin
select @gra_sue = gra_codigo_sueldo from grado where gra_cod = @grado
update persona set per_gra_sue = @gra_sue where per_id = @vper_id
end

```

-- Si se actualiza fecha de último ingreso e indicador de baja se trata de un reingreso

If update (per\_fec\_ult\_ing) and update (per\_ind\_baj)

```

begin
declare @max_id int
-- Insertar registro en historico de ingresos
select @max_id = max(his_per_iden) + 1

from historico_personal
print 'aqui ingreso'
print @fec_ingreso
insert into historico_personal

```



**CIB-ESPOL**

```

(his_per_iden,per_per_id,gru_per_cod,gra_ini,his_per_fec_ing,his_per_est,his_per_u
sr_crea,his_per_fec_crea,his_per_tip_con)

```

values

```

(@max_id,@vper_id,@grupoper,@grado,@fec_ingreso,'A',@usuario,getdate(),@tip
_con)

```

-- Insertar registro en especialidad

```

if @instruccion <> "000000"
begin
select @max_id = max(his_esp_id) + 1
from historico_especialidad

```

```

insert into historico_especialidad

(his_esp_id,per_per_id,tip_ins_cod,tip_doc_cod,his_esp_fec_asi,his_esp_est,his_esp_
_usr_crea,his_esp_usr_fec_crea)
values
(@max_id,@vper_id,@instruccion,'00',@fec_ingreso,'A',@usuario,getdate())
end

-- Insertar registro en ascenso_civil
if @grupoper = '06'
begin
select @max_id = max(asc_civ_iden) + 1
from ascenso_civil

insert into ascenso_civil

(asc_civ_iden,per_per_id,par_pre_cod,niv_niv_cod,cat_cat_cod,asc_civ_fec_niv,asc_
civ_fec_cat,asc_civ_tip_mod,tip_tip_doc_cod,asc_civ_est,asc_civ_usr_crea,asc_civ_
fec_crea)
values
(@max_id,@vper_id,"00",@nivel,@categoria,@fec_ingreso,@fec_ingreso,'5','00','A'
,@usuario,getdate())

--Verificar si ha sido militar, es decir, si tiene registro en la tabla
ascensos_navales
select @cont = count(*) from ascensos_navales where per_per_id = @vper_id

-- Si tiene registros insertar un registro en la tabla servicio_anterior como
tiempo reconocido
if @cont > 0
begin
select @max_id = max(ser_ant_iden) + 1
from servicio_anterior
select @reg_ant = count(*) from desl.servicio_anterior
where per_per_id = @vper_id
and ram_cod = '2'
and cau_cau_ser_cod = '10'
and ser_ant_est = 'A'

if @reg_ant = 0
begin
insert into servicio_anterior

```



**CIB-ESPOL**

```

        (ser_ant_iden,per_per_id,ram_cod,
cau_cau_ser_cod,tip_tip_doc_cod,ser_ant_fec_rec,ser_ant_tie,ser_ant_est,ser_ant_usr
_crea,ser_ant_fec_crea)
        values
(@@max_id,@vper_id,'2','10','00',getdate(),@ser_ant,'A',@usuario,getdate())
    end
    end
    end
else
    if @grupoper <> '03' -- Insertar registro en ascensos_navales
        begin

            select @@max_id = max(asc_nav_iden) + 1
                from ascensos_navales

            insert into ascensos_navales

            (asc_nav_iden,per_per_id,cau_asc_cod,gra_gra_cod,tip_tip_doc_cod,asc_nav_fec_as
c,asc_nav_ubi_gra,asc_nav_est,asc_nav_usr_crea,asc_nav_fec_crea)
                values
(@@max_id,@vper_id,'19',@grado,'00',@fec_ingreso,0,'A',@usuario,getdate())
        end
    end

print 'termino'
return

```

```

    @fec_ult_ing = i.per_fec_ult_ing,
    @gru_per = i.gru_gru_per_cod,
    @fec_ing_sue = i.per_fec_ing_sue,
    @fec_ing = i.per_fec_ing,
    @usuario = i.per_usr_mod
From inserted i, persona p
Where i.per_id = p.per_id
print 'primera fecha'
print @fec_ing_sue
select @fec_eli = d.per_fec_ult_ing,
       @fec_eli_sue = d.per_fec_ing_sue,
       @fec_eli_ing = d.per_fec_ing
from deleted d, persona p
Where d.per_id = p.per_id

```

```

If update (per_fec_ult_ing) and (@fec_eli <> @fec_ult_ing)
begin

```

```

-- busca en histórico personal el registro para actualizar la fecha de último ingreso
-- debe ser igual al registro en el que la fecha de baja y el grado fin es null

```

```

    select @vhis_id = (select his_per_iden
                      from historico_personal
                      where his_per_fec_ing= @fec_eli
                      and gra_fin is null
                      and his_per_est = 'A'
                      and per_per_id = @vper_id)

```

```

-- Una vez encontrado el registro lo actualizamos

```

```

    update historico_personal set
        his_per_fec_ing = @fec_ult_ing,
        his_per_usr_mod = @usuario,
        his_per_fec_mod = getdate()
    where his_per_iden = @vhis_id

```



**CIB-ESPOL**

```

if @gru_per <> '06'
begin

```

```

    select @vasc_nav_id = (select asc_nav_iden
                          from ascensos_navales
                          where asc_nav_fec_asc = @fec_eli
                          and asc_nav_est = 'A'
                          and per_per_id = @vper_id)

```

---

```

if @vasc_nav_id is not null
  begin
    update des1.ascensos_navales
    set asc_nav_fec_asc = @fec_ult_ing,
        asc_nav_usr_mod = @usuario,
        asc_nav_fec_mod = getdate()
    where asc_nav_iden = @vasc_nav_id
  end
end

```

```

--
*****
*****

```

```

-- consulta si la fecha de ingreso y la fecha de ingreso sueldos (aportaciones
-- al seguro es igual a la fecha que se está actualizando, si es así procede
-- a actualizar esos valores también

```

```

if @fec_ing = @fec_eli
  begin
    update persona set
    per_fec_ing = @fec_ult_ing
    where per_id = @vper_id
  end

```

```

if @fec_ing_sue = @fec_eli
  begin
    update persona set
    per_fec_ing_sue = @fec_ult_ing
    where per_id = @vper_id
  end

```

```

EXEC DES1.SP_CALCULA_TIE_SER_TOT @vper_id
end

```

```

-- *****Actualización fecha de aportación (per_fec_ing_sue) *****

```

```

If update (per_fec_ing_sue) and (@fec_ing_sue <> @fec_eli_sue)
begin
  -- busca en histórico personal el registro para actualizar la fecha de primer aporte
  print 'fec_ing_sue'
  print @fec_ing_sue
  print 'fec_eli_sue'
  print @fec_eli_sue

```

```

select @vhis_id = (select his_per_iden
                    from historico_personal
                    where his_per_fec_ing= @fec_eli_sue
                    and per_per_id = @vper_id
                    and his_per_est = 'A')

-- Una vez encontrado el registro lo actualizamos
print 'ahora'
print @fec_ing_sue
update historico_personal set
    his_per_fec_ing = @fec_ing_sue,
    his_per_usr_mod = @usuario,
    his_per_fec_mod = getdate()
where his_per_iden = @vhis_id

--
*****
*****
-- consulta si la fecha de ingreso y la fecha de último ingreso es igual a
-- la fecha que se está actualizando, si es así procede a actualizar esos valores
también

print @fec_ing
print @fec_eli_sue
if @fec_ing = @fec_eli_sue
begin
    update persona set
        per_fec_ing = @fec_ing_sue
    where per_id = @vper_id
end

if @fec_ult_ing = @fec_eli_sue
begin
    update persona set
        per_fec_ult_ing = @fec_ing_sue
    where per_id = @vper_id
end

EXEC DES1.SP_CALCULA_TIE_SER_TOT @vper_id
end

-- *****Actualización fecha de ingreso (per_fec_ing) *****

```



**CIB-ESPOL**



```
If update (per_fec_ing) and (@fec_eli_ing <> @fec_ing)
begin
-- busca en histórico personal el registro para actualizar la fecha de primer aporte
select @vhis_id = (select his_per_iden
                    from historico_personal
                    where his_per_fec_ing= @fec_eli_ing
                    and per_per_id = @vper_id
                    and his_per_est = 'A')

-- Una vez encontrado el registro lo actualizamos
update historico_personal set
    his_per_fec_ing = @fec_ing,
    his_per_usr_mod = @usuario,
    his_per_fec_mod = getdate()
where his_per_iden = @vhis_id
end

return
```



**CIB-ESPOL**

## Para un campo en especial de consulta (lectura)

```
CREATE trigger trig_upd_fec_ing_persona
```

```
/* Trigger de actualización de tablas originado por consultas en la tabla Persona
```

```
-----  
Cuando se actualiza uno de los campos  
per_fec_ing,per_fec_ing_sue,per_fec_ult_ing en la tabla persona, se debe registrar el  
cambio en la tabla historico_personal con la fecha correspondiente a la que se está  
modificando y de igual forma si una de las fechas anteriores es igual a la que se está  
modificando.
```

```
Al modificar el campo per_fec_ult_ing o per_fec_ing_sue, fechas de ingresos  
necesarias para cálculos de Antigüedad para el Sistema de Sueldos, se ejecuta el  
procedimiento almacenado DES1.SP_CALCULA_TIE_SER_TOT enviando como  
parámetro el id de la persona
```

```
EJEMPLO: EXEC DES1.SP_CALCULA_TIE_SER_TOT 1  
*/
```

```
on DES1.PERSONA
```

```
for update
```

```
as
```

```
declare @rows int, @fec_eli datetime, @fec_ing datetime, @fec_ing_sue datetime,  
        @fec_eli_sue datetime, @fec_eli_ing datetime,@gru_per  
varchar(2),@vasc_nav_id integer  
declare @vper_id integer, @fec_ult_ing datetime, @vhis_id integer, @usuario  
varchar(15)
```

```
select @rows = @@rowcount
```

```
if @rows = 0
```

```
return
```

```
if exists(select * from deleted) and @rows > 1
```

```
begin
```

```
-- "No es permitido actualizar varias filas"
```

```
-- rollback tran
```

```
return
```

```
end
```

```
-- Leer los datos ingresados
```

```
select @vper_id = i.per_id,
```



**CIB-ESPOL**

---

# ANEXO Z

## MANUAL DE USUARIO DEL SISTEMA OPTIMIZADOR DE BASE DE DATOS

Presentado por: \*

Alex Paúl Tapia Chichande

Luis Miguel Escobar Larenas

Guayaquil – Ecuador

2005

# ÍNDICE GENERAL

INDICE GENERAL .....	I
INDICE DE FIGURAS .....	II
<b>CAPITULO 1 GENERALIDADES .....</b>	<b>1</b>
1.1. A CERCA DE ESTE MANUAL .....	1
1.2. USUARIOS A QUIENES ESTA DIRIGIDO ESTE MANUAL .....	1
1.3. LO QUE EL USUARIO DEBE SABER .....	1
1.4. OBJETIVOS DE LA SOLUCIÓN .....	1
<b>CAPITULO 2 DESCRIPCION DEL CASE.....</b>	<b>3</b>
2.1. OPERACIÓN DEL CASE .....	3
2.2. PARTES DE UNA VENTANA .....	3
<i>Tabla 2.1 Tabla descripción de ventana principal</i> .....	4
2.3. PARTES DEL MENÚ .....	4
2.4. OBJETOS USADOS POR EL CASE .....	5
<i>Tabla 2.2 Tabla descripción de botones</i> .....	6
2.5. REQUERIMIENTOS OPERACIONALES.....	7
<i>Tabla 2.3 Características mínimas del hardware</i> .....	7
<i>Tabla 2.4 Características del software</i> .....	8
<b>CAPITULO 3 INSTALACION Y USO DEL CASE .....</b>	<b>9</b>
3.1. INSTALACIÓN DE PROGRAMAS .....	9
3.2. USO DEL CASE.....	10
3.3. PUNTOS A CONSIDERAR .....	17



**CIB-ESPOL**

# ÍNDICE DE FIGURAS

FIGURA 2.1 PARTES DE LAS VENTANAS DEL CASE.....	3
FIGURA 2.2 PARTES DEL MENÚ.....	5
FIGURA 2.3 CAJAS DE TEXTO PARA INGRESO DE INFORMACIÓN.....	5
FIGURA 2.3 MENSAJE DE BASE ELIMINADA CON ÉXITO.....	6
FIGURA 2.4 MENSAJE DE BASE INGRESADA CON ÉXITO.....	6
FIGURA 2.5 MENSAJE DE RESULTADOS.....	7
FIGURA 2.6 CARACTERÍSTICAS DEL COMPILADOR.....	8
FIGURA 3.1 INICIO DESDE FORTE FOR JAVA 4 CE.....	11
FIGURA 3.2 MENÚ CONSULTA OPCIÓN NUEVA.....	11
FIGURA 3.3 CREACIÓN DE NUEVA BASE.....	12
FIGURA 3.4 INGRESE DE DATOS DE NUEVA BASE.....	12
FIGURA 3.5 MENSAJE BASE INGRESADA EXITOSAMENTE.....	13
FIGURA 3.6 VERIFICACIÓN DE BASE INGRESADA.....	13
FIGURA 3.7 INGRESO DATOS DISCOS.....	14
FIGURA 3.8 ERROR COMETIDO.....	14
FIGURA 3.9 MENSAJE DE DISCO ELIMINADO.....	14
FIGURA 3.10 MENSAJE DE DISCO INGRESADO.....	15
FIGURA 3.11 TABLA INGRESADA.....	15
FIGURA 3.12 GENERACIÓN DE BÚSQUEDA.....	16
FIGURA 3.13 MENSAJE DE RESULTADOS.....	16



**CIB-ESPOL**

# CAPITULO 1 GENERALIDADES

## *1.1.A CERCA DE ESTE MANUAL*

El **Manual de Usuario** del CASE Sistema Optimizador de Base de datos (SYOPBASE), nos brinda una descripción completa para obtener eficacia en la utilización del CASE mencionado.

Su objetivo principal es de capacitar a los usuarios en el manejo del **SYOPBASE**; para de esta manera poder realizar cada uno de los procesos y tareas de manera amigable al usuario.

## *1.2.Usuarios a quienes esta dirigido este manual*

El usuario final de este manual será el DBA encargado de las actividades a realizarse sobre las diferentes bases de datos y es importante mencionar que deberá tener un pleno conocimiento de la misma.

## *1.3.Lo que el usuario debe saber*

Los conocimientos mínimos que deben tener las personas que operarán el **SYOPBASE**, para un buen uso del mismo, son:

- Pleno conocimiento de sus bases de datos, sin llegar a constituirse en un experto.
- Conocimientos sólidos sobre los tipos de transacciones realizadas en la base de datos.
- Conocimientos sólidos sobre los tipos de control de auditorías aplicables a cerca de las bases de datos.
- Conocimientos sólidos sobre el hardware en los cuales se aplicará el mencionado sistema.
- Conocimientos medios sobre históricos de la base de datos y hardware en la cual esta implementada.
- Conocimientos básicos sobre el uso de las computadoras.
- Conocimientos básicos de Informática bajo el ambiente Windows.

## *1.4.Objetivos de la solución*

1. Optimizar la capacidad de almacenamiento de los discos utilizados tanto en sistemas RAID como en los que no los dispongan.

2. Optimizar la velocidad de los procesos realizados en las diferentes transacciones ordinarias de las bases de datos.
3. Disponer de varias alternativas óptimas para elección de almacenamiento.
4. Crear modelos basados en una base de datos.



**CIB-ESPOL**

## CAPITULO 2 DESCRIPCION DEL CASE

### 2.1. Operación del CASE

En este capítulo se describen convenciones, normas y elementos específicos del Sistema, que son empleados en este manual y además algunos elementos adicionales.

### 2.2. Partes de una ventana

Las ventanas, conocidas también como pantallas, representan al área de trabajo limitadas por bordes donde el usuario debe realizar las actividades tales como ingresar datos o procesarlos. Las ventanas del SYOPBASE tienen las siguientes características:

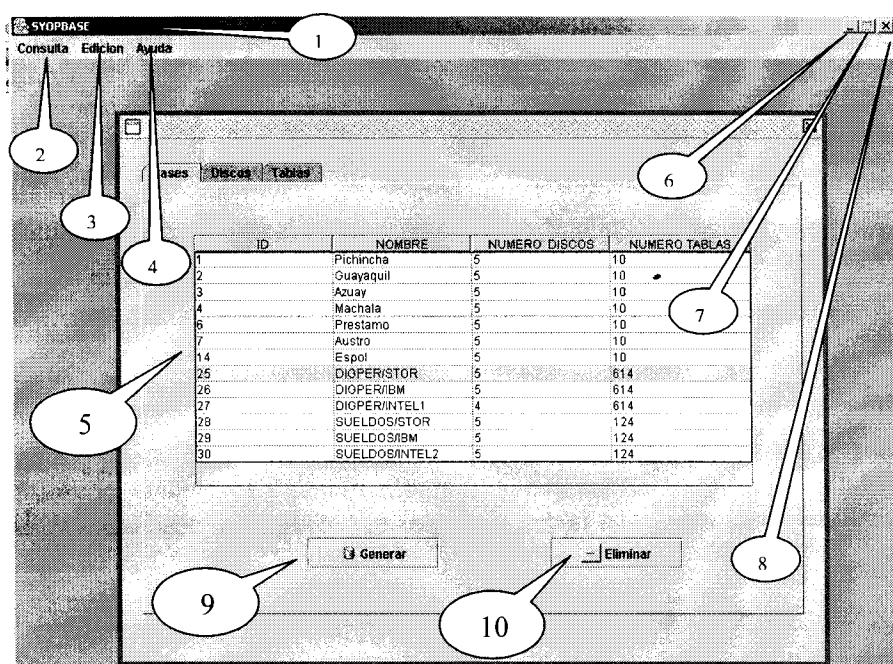


Figura 2.1 Partes de las ventanas del CASE.



**CIB-ESPOL**



Sección	Descripción
1	Título del CASE
2	Menú Consulta
3	Menú Edición
4	Menú Ayuda
5	Área de trabajo
6	Minimizar
7	Restaurar
8	Cerrar
9	Operación de generar
10	Operación de eliminar



**CIB-ESPOL**

Tabla 2.1 Tabla descripción de ventana principal

### 2.3. Partes del menú

Los menús de cada sección que forma el CASE, están diseñados de modo que sean intuitivos, amigables y sencillos para su funcionamiento. Estos han sido divididos de tal manera que el pueda realizar una serie de operaciones de acuerdo a sus necesidades.

Para ingresar a los mismos, **haga clic** sobre el nombre de la opción a la que desea ingresar. Vale la pena mencionar que el nivel máximo de subdivisión de un menú para el CASE es unitario.

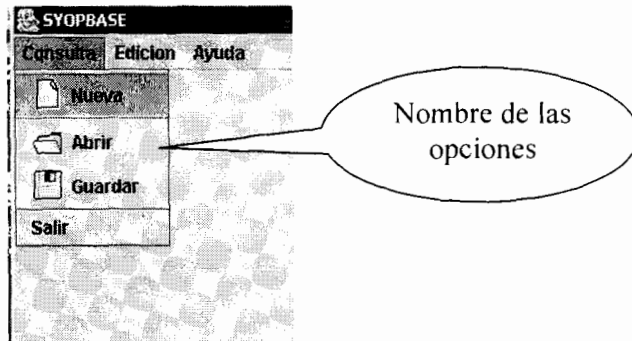


Figura 2.2 Partes del menú

#### 2.4. Objetos usados por el CASE

El CASE posee objetos que son característicos, a los cuales se les ha dado el uso necesario tomando en cuenta las necesidades que se requieren; entre los objetos que hemos empleado se detallan los siguientes:

1. Cajas de texto.- son aquellas que permiten ingresar información al usuario.

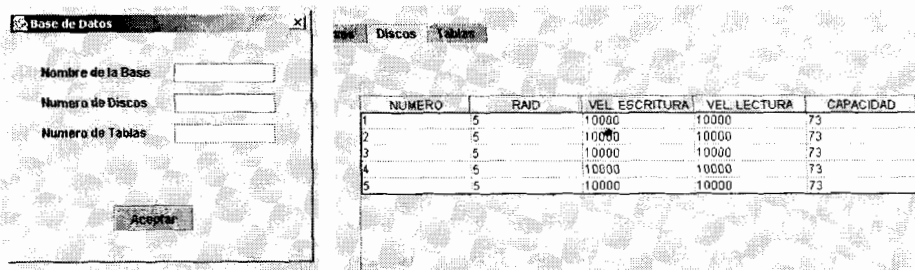


Figura 2.3 Cajas de texto para ingreso de información

2. Botones.- para el diseño del CASE, hemos hecho uso de imágenes y botones los mismos que se describen a continuación:

Botón	Descripción
	Permite el ingreso de nuevas tablas o discos
	Elimina la información ingresada.



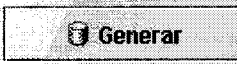
	Permite almacenar todos los cambios hechos en las bases de datos, tablas o discos.
	Permite confirmar la acción realizada y ejecutarla.
	Una vez que se ha ingresado todos los datos, el CASE permite generar la corrida del algoritmo genético.

Tabla 2.2 Tabla descripción de botones

3. Mensajes de información.-estos mensajes son emitidos por el CASE cuando alguna acción ha sido realizada exitosamente. El nombre de la acción realizada se muestra a manera de encabezado.

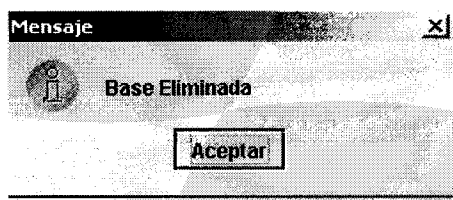


Figura 2.3 Mensaje de base eliminada con éxito

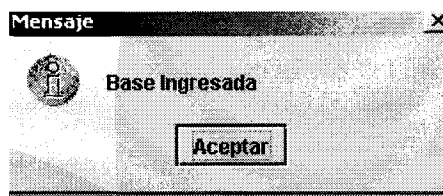


Figura 2.4 Mensaje de base ingresada con éxito



CIB-ESPOL

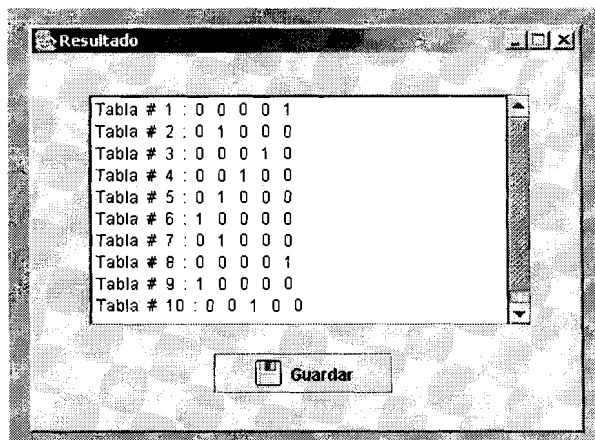


Figura 2.5 Mensaje de resultados

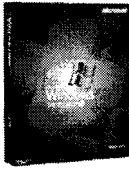

## 2.5. Requerimientos operacionales

### 1. Hardware.-

Características	Descripción
Microprocesador	Pentium IV
Velocidad	2.8 Ghz.
Memoria RAM	512 Mb.
Memoria de Vídeo	8 Mb (de preferencia).
Espacio en el disco duro	100 Mb (j)
Unidad de disquete	1.44 Mb.
Teclado y mouse	Ps2
Tarjeta de video	PCI

Tabla 2.3 Características mínimas del hardware

### 2. Software.-

Sistema Operativo		Windows 2000, ME , XP Professional
Base de Datos		SQL 7.0

Lenguaje de programación		JAVA
--------------------------	---	------

Tabla 2.4 Características del software

3. Compilador.- El sistema ha sido desarrollado en java utilizando como compilador al Forte for Java 4 Community Edition, utilizamos este lenguaje en vista de que es orientado a objetos y adicionalmente Open Source, lo que nos ha permitido un acceso fácil y barato para la implementación del case. En el caso de la base de datos se ha adquirido la licencia de SQL 7.0 por las bondades y facilidades que esta brinda para el desarrollo anteriormente mencionado. Las características principales del compilador se mencionan a continuación:

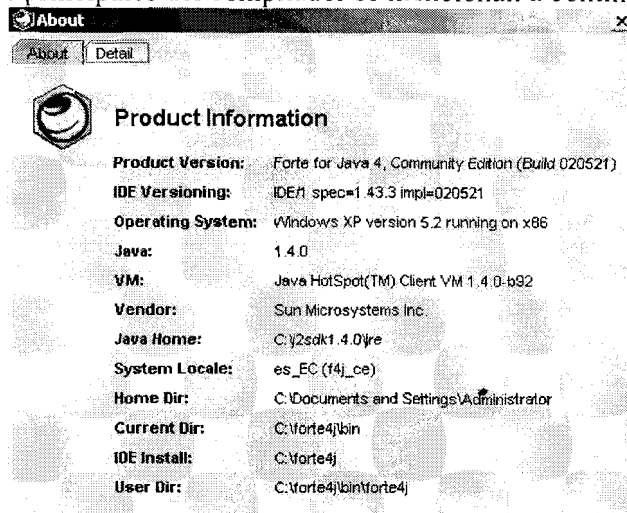


Figura 2.6 Características del compilador

## CAPITULO 3 INSTALACION Y USO DEL CASE

### 3.1. Instalación de programas

Para la instalación del compilador sugerimos seguir los pasos descritos en <http://docs-pdf.sun.com/816-4058/816-4058.pdf>.

Para la instalación de la base de datos SQL 7.0, sugerimos realizar las siguientes acciones (una vez que han sido bajadas las librerías de conectividad y el instalador de la aplicación.):

- 1) Descomprimir el archivo otros.zip en un directorio diferente de donde se encuentra ubicado, generará cuatro archivos exe.
- 2) Descomprimir el archivo InstalaCtriASA703.zip, también en otro directorio, ya que al descomprimirlo creará un subdirectorío llamado Disk1
- 3) Ir al directorio, donde se encuentra los cuatro archivos exe y ejecutar según el orden siguiente:
  - 401COMUPD.EXE(1). Este exe instala la última versión del COMCTL32.DLL; es posible de acuerdo a las versiones anteriores que no sea necesario este paso.
  - DCOM95.EXE(2). Instala el DCOM95, requerido por ODBC 3.5. En algunas configuraciones mandará el mensaje que se cargado una versión más reciente y que por lo tanto, no se puede actualizar las librerías.
  - MDAC\_MIN.EXE(3). Instala el ODBC 3.5.
  - DATAACC.EXE(4). Escoger solamente la opción de Controladores de acceso a datos.
- 4) REINICIAR LA PC.
- 5) En este paso Ud. tendrá que instalar su versión de: Adaptive Server Anywhere 7.0.3, esto se refiere al CD donde se encuentre el software para instalarse, no se refiere si su institución tiene la licencia. En vista de ser un software propietario sugerimos la compra de la licencia del mismo, cuya cotización y valor cancelado de encuentra en el anexo O  
De preferencia instalar en el directorio sugerido, sino Ud. puede hacerlo en otra unidad de disco, pero se sugiere no cambiar la jerarquía de directorios. Sin embargo, si Ud. cambia la unidad de disco, esta ruta deberá ser actualizada en el sistema verificador en el parámetro DBISQL. Solamente debe instalar el producto: SQL Anywhere 7.0, es decir desmarcar todos los productos a instalar, dejando solamente el de SQL Anywhere 7.0. Este paso

debe de realizarse antes de instalar la aplicación del verificador, ya que de lo contrario, no podrá ejecutar el sistema. Verificar que la versión instalada.

6) REINICIAR LA PC.

Esta aplicación se instalará en el siguiente directorio:  
C:\SSF\CTRI\VERIFICADOR\ (Default)

- 7) Ir al directorio, donde Ud. descomprimió el archivo InstalaCtriASA703.zip y al subdirectorio Disk1 y ejecutar setup.exe
- 8) Seguir con las instrucciones de instalación dejando los defaults.
- 9) Ir al menú de Start (Inicio) de windows y dando clic en programs (programas) buscar la opción 'SSF CTRI V.5.00.01 (ASA703)' dentro de esta opción dar clic al shortcut de VERIFICADOR.
- 10) Ingresar el usuario: ssf y password: ssf.
- 11) Ingresar el tipo de verificador.
- 12) Seleccionar el código y el nombre de la institución.
- 13) Ud. puede cambiar el usuario y clave (ssf, ssf), ingresando a la primera opción del toolbar que es de configuración de parámetros y realizar los siguientes pasos:

- En DBUserUID escribir el código de la institución. (Este será el usuario de ingreso)
- En DBUserPWD escribir el código de la institución o la palabra clave que Ud. desee. (esta será la clave de ingreso)
- Si no se instaló en el disco C:\ cambiar todas las opciones que contengan rutas de directorios y sobrescribirlas la unidad en donde se instaló.

A pesar de que este procedimiento nos ha dado buenos resultados es posible que debido a sus configuraciones cause algún tipo de conflicto para lo cual sugerimos ponerse en contacto con su proveedor y de esta manera solventar las dificultades.

### 3.2. Uso del CASE

Una vez que nos encontramos en la pantalla principal del Forte for Java 4 CE, nos colocamos en la viñeta de GUI Editing, activamos la clase Syopbase y ejecutamos:



**CIB-ESPOL**

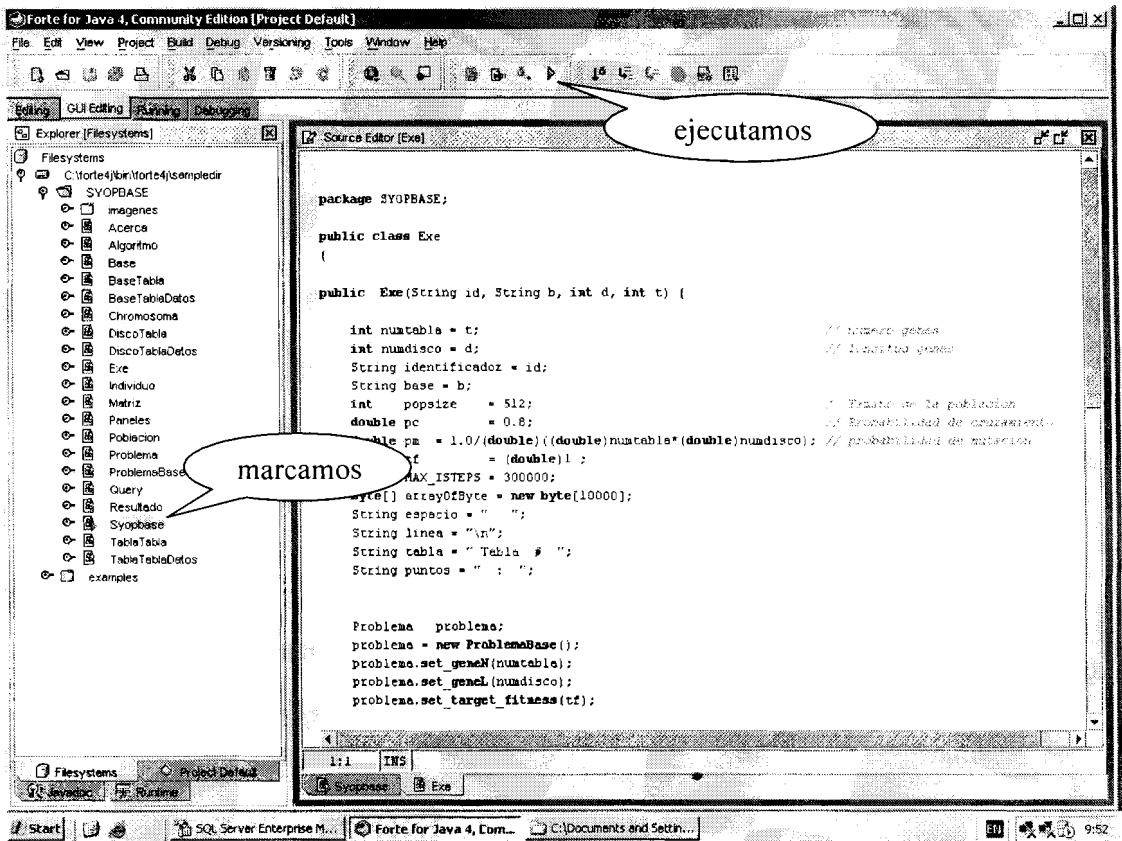


Figura 3.1 Inicio desde Forte for Java 4 CE

Una vez que hemos ingresado al SYOPBASE nos dirigimos al menú Consulta, opción nueva



Figura 3.2 Menú consulta opción nueva



CIB-ESPOL



Base de Datos

Nombre de la Base

Numero de Discos

Numero de Tablas

Aceptar

Figura 3.3 Creación de nueva base

Colocamos los datos requeridos y presionamos el botón aceptar.

Base de Datos

Nombre de la Base

Numero de Discos

Numero de Tablas

Aceptar

Figura 3.4 Ingrese de datos de nueva base

Mensaje de base de datos creada exitosamente

Mensaje

Base Ingresada

Aceptar



CIB-ESPOL

Figura 3.5 Mensaje base ingresada exitosamente

En el menú Consulta opción abrir obtendremos la siguiente pantalla, en donde verificamos la base ingresada:

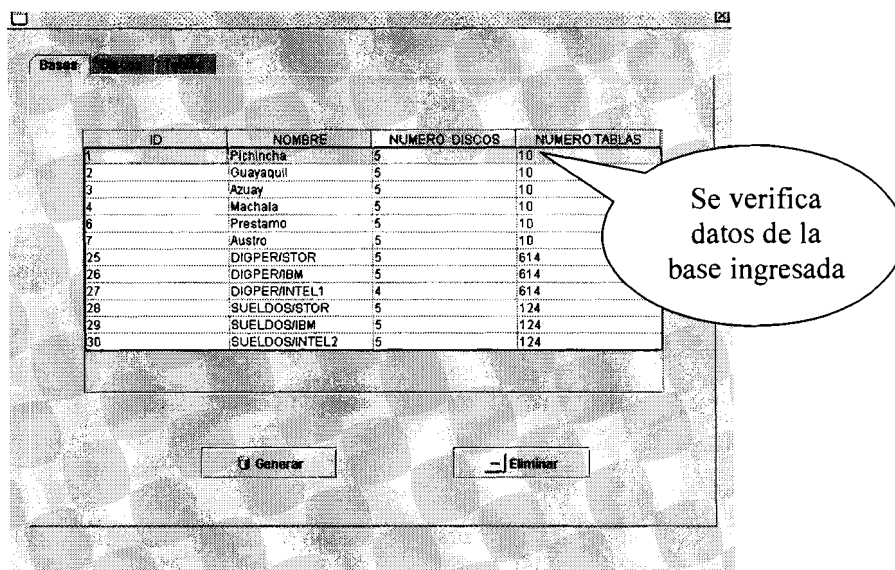


Figura 3.6 Verificación de base ingresada

Escogemos la base ingresada y en la viñeta de discos ingresamos las características de los discos con el botón adicionar:

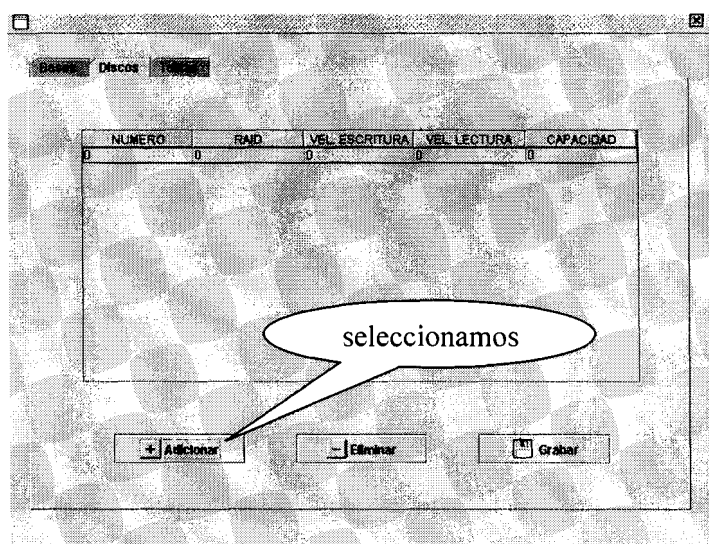


Figura 3.7 Ingreso datos discos

En caso de cometer algún error, escogemos al disco en mención y lo eliminamos:

NUMERO	RAO	VEL. ESCRITURA	VEL. LECTURA	CAPACIDAD
1	0	10000	5000	120
2	0	7200	7200	80
3	0	5200	7200	80
4	0	5200	5200	40
5	0	15000	16000	73
6	0	15000	16000	73

Tabla mal ingresada

Figura 3.8 Error cometido

con el botón eliminar, obteniendo el siguiente mensaje:

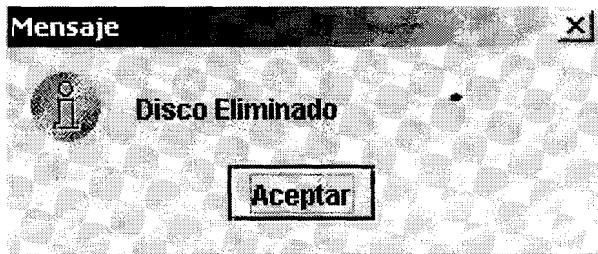
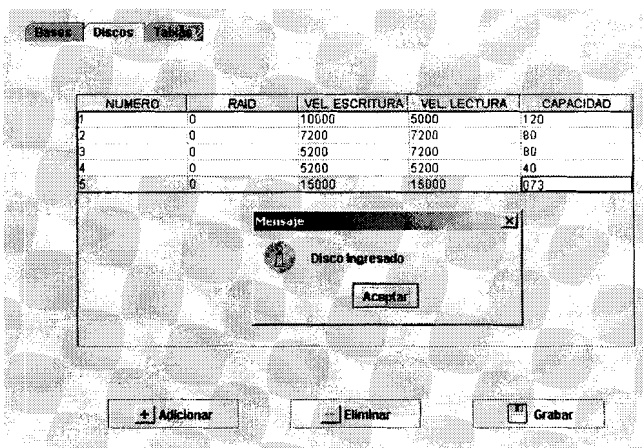


Figura 3.9 Mensaje de disco eliminado

Una vez que presionamos el botón grabar aparece el mensaje de discos ingresado



**CIB-ESPOL**



CIB-ESPOL

Figura 3.10 Mensaje de disco ingresado

Un procedimiento idéntico debemos seguir para el ingreso de datos en las tablas tal como se presenta a continuación:

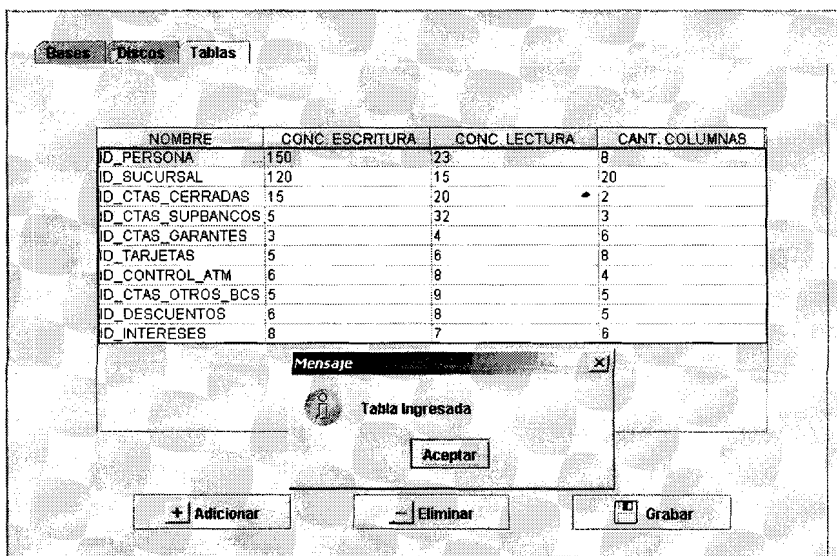


Figura 3.11 Tabla ingresada

Una vez finalizado el proceso de ingreso de discos y tablas nos ubicamos en la pantalla de consulta, abrir y bases donde escogemos a la base ingresada e iniciamos el proceso de generar (el cual iniciara la utilización del algoritmo genético) y este emitirá el repote de resultados, tal como se aprecia aquí:

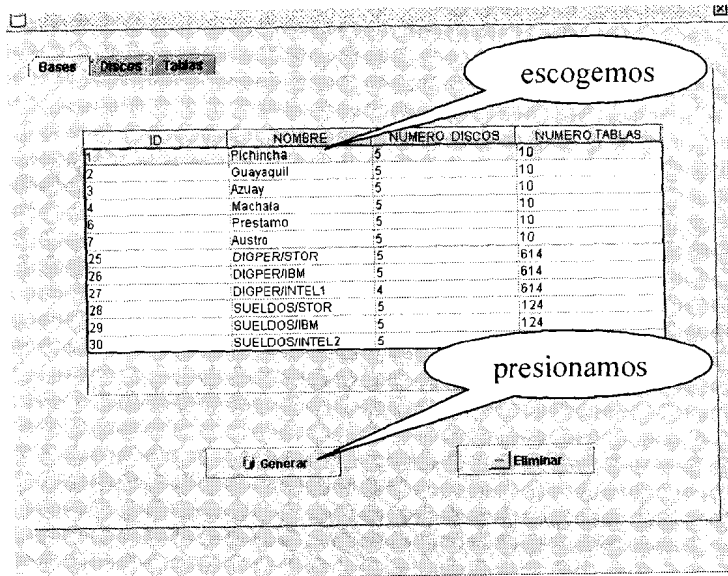


Figura 3.12 Generación de búsqueda

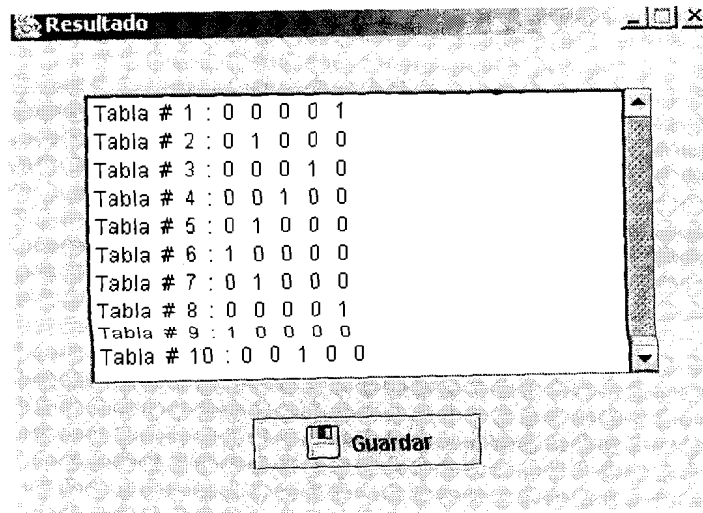


Figura 3.13 Mensaje de resultados

Almacenamos esta disposición, para futuras consultas.

### *3.3.Puntos a considerar*

- 1) Es necesario que la persona que utilice el CASE determine cuales son los rangos de uso de las tablas, para lo cual se ha considerado la implementación de un trigger o disparador en las mismas, los cuales nos indicaran un promedio de utilización, quedando a criterio del usuario la creación de uno o la utilización del propuesto en este manual. Los trigger propuestos se los puede apreciar en el anexo P
- 2) Es necesario que el usuario aplique su criterio para lo cual deberá conocer con bastante certeza su base de datos.
- 3) Como es de esperarse la solución no puede ser única, por lo que las repuestas pueden ser muy variadas, aunque estas sean tendientes a lo mismo.
- 4) La demora de la solución estará dada de acuerdo a la cantidad de tablas y discos que disponga nuestro sistema.
- 5) En el caso de tener una base de datos con gran cantidad de tablas recomendamos introducir las mediante la inserción de archivos planos directamente a las tablas de las bases de datos de esta manera se lograra evitar el ingreso de los mismos uno por uno, mejorando los tiempos de labor.