

**UNIVERSIDAD
DON BOSCO**



**AUTOMATIZACIÓN DE LOS PROCESOS CONTABLES Y DE CONTROL
PRESUPUESTAL DE LA CIUDADELA DON BOSCO**



**TRABAJO DE GRADUACIÓN
PREPARADO PARA LA
FACULTAD DE INGENIERIA
PARA OPTAR AL GRADO DE**

**INGENIERO EN CIENCIAS
DE LA COMPUTACIÓN**

POR

ANGEL ALFONSO GUERRERO FLORES

EDWIN ALBERTO PEREZ LOPEZ

ROBERTO CARLOS NAVAS PORTILLO

ASESOR

LIC. JOSÉ RHIN SALAZAR

SEPTIEMBRE DE 1997

SOYAPANGO

-

EL SALVADOR

-

CENTROAMERICA

UNIVERSIDAD DON BOSCO

RECTOR

ING. FEDERICO MIGUEL HUGUET RIVERA

SECRETARIO GENERAL

PBRO. PEDRO GARCÍA S.D.B.

DECANO DE LA FACULTAD DE INGENIERIA

ING. ORLANDO CABRALES

ASESOR DEL TRABAJO DE GRADUACION

LIC. JOSÉ RHIN SALAZAR

JURADO CALIFICADOR

SR. JAVIER SERRANO

LIC. GUILLERMO VÁSQUEZ

UNIVERSIDAD DON BOSCO

FACULTAD DE INGENIERIA

DEPARTAMENTO DE INGENIERIA EN

CIENCIAS DE LA COMPUTACION

JURADO EVALUADOR DEL TRABAJO DE GRADUACION

**“AUTOMATIZACIÓN DE LOS PROCESOS CONTABLES Y DE
CONTROL PRESUPUESTAL DE LA CIUDADELA DON BOSCO”**

LIC. GUILLERMO VASQUEZ
JURADO

SR. JAVIER SERRANO
JURADO

LIC. JOSÉ RHIN SALAZAR
ASESOR

Agradecimientos

A Dios Todopoderoso y a la Virgen María, por saber guiarnos a través de la luz del amor infinito.

A Mi Padre, Noel Enrique Navas, y a mi Madre, María Elba Portillo por apoyarme sin condición y con un gran amor.

A mi novia, Sandra Elizabeth Martínez, por haberme soportado durante tanto tiempo, por su gran amor y paciencia.

A mis Compañeros de Tesis, por haber compartido tanto sufrimiento.

A mis Compañeros de Trabajo, porque su ejemplo me a servido de inspiración para emprender con ahínco este proyecto.

A nuestro excelente Asesor, Lic. Rhin Salazar, por haber dedicado tanto tiempo a nuestro proyecto y sin ningún interés.

Al Lic. Guillermo Vásquez, iniciador de un proyecto tan emprendedor y por su apoyo en la realización de nuestra tesis.

A todos ellos y muchos mas, les doy mis mas sinceros agradecimientos.

Roberto Carlos Navas

Agradecimientos

A Dios Todopoderoso y a la Santísima Virgen María por iluminarnos y darnos la fortaleza necesaria para alcanzar nuestra meta.

A mis padres, Pedro Fausto Pérez y Vilma Gloria de Pérez por todo su cariño y comprensión, así como por sus oraciones y todo el apoyo y confianza que me han brindado siempre. Gracias por ser tan especiales.

A mis familiares y amigos, quienes de una forma muy especial me brindaron siempre su apoyo constante e incondicional a fin de lograr mis objetivos y metas.

A mis compañeros de tesis por todo su apoyo, sentido del humor y constancia durante todo el desarrollo de nuestro proyecto.

A nuestro excelente Asesor, Lic. Rhin Salazar, por haber dedicado tanto tiempo a nuestro proyecto y sin ningún interés.

Al Lic. Guillermo Vásquez, iniciador de un proyecto tan emprendedor y por su apoyo y consejos durante la realización de nuestra tesis, y por haber tenido siempre confianza en nosotros.

A todas las personas que de una u otra forma me han brindado su apoyo durante el desarrollo del proyecto.

Edwin Alberto Pérez López

Agradecimientos

A Dios Todopoderoso y a la Santísima Virgen María, por habernos dado la capacidad para culminar nuestro proyecto.

A mi padre Luis Isidoro Guerrero, por todo sus esfuerzos y ejemplos que siempre me motivaron a alcanzar mis objetivos

A mi madre Edmee Miguelina Flores, por todos sus consejos y motivarme a cumplir mis metas.

A mi tío Victor Manuel Guerrero, por toda su comprensión y paciencia durante toda mi carrera universitaria

A mi familia, por haberme orientado siempre por el buen camino.

A mis compañeros, por todo el esfuerzo y dedicación que hicieron posible la conclusión del proyecto

A nuestro excelente Asesor, Lic. Rhin Salazar, por haber dedicado tanto tiempo a nuestro proyecto y sin ningún interés.

Al Lic. Guillermo Vásquez, iniciador de un proyecto tan emprendedor y por su apoyo en la realización de nuestra tesis.

A todas aquellas personas que siempre me motivaron sinceramente muchas gracias.

Angel Alfonso Guerrero.

“Trate de encontrarlo en la Cruz Cristiana, pero Él no estaba ahí; fui al Templo de los Hindúes y a las viejas Pagodas, pero no pude encontrar ni un rastro de Él en ninguna parte.

He buscado en las montañas y en los valles, pero ni en las alturas ni en las profundidades he sido capaz de encontrarlo. Luego fui a Cabaa en la Mecca, pero tampoco estaba ahí.

Le pregunte a los Intelectuales y Filósofos, pero Él estaba mas allá de su entendimiento.

Fue entonces cuando busque en mi corazón y Él estaba ahí, donde me concedió verlo.

Él estaba en ningún otro lugar donde encontrarlo.”

Jelaluddin Rumi

INDICE

INDICE	6
INTRODUCCION	10
DESCRIPCIÓN DEL TEMA	12
DEFINICIÓN DEL PROBLEMA	14
JUSTIFICACIONES	16
OBJETIVOS DEL PROYECTO	17
OBJETIVO GENERAL	17
OBJETIVOS ESPECÍFICOS	17
ALCANCES Y LIMITACIONES	19
ALCANCE DEL SISTEMA CONTABLE.	19
ALCANCE DEL MÓDULO DE CONTROL PRESUPUESTAL.	19
LIMITACIONES DEL SISTEMA CONTABLE.	19
LIMITACIONES DEL MÓDULO DE CONTROL PRESUPUESTAL.	20
ANTECEDENTES INFORMÁTICOS	21
MARCO TEÓRICO	22
METODOLOGÍA DE INVESTIGACIÓN Y DESARROLLO	24
ETAPAS DEL CICLO DE VIDA PARA EL DESARROLLO DE SISTEMAS.	24
CONCEPCIÓN DE LA IDEA O PROYECTO.	24

INVESTIGACIÓN PRELIMINAR	24
DETERMINACIÓN DE LOS REQUERIMIENTOS DEL SISTEMA.	27
DISEÑO DEL SISTEMA.	28
DESARROLLO DEL SOFTWARE O PROGRAMA.	29
PRUEBA DEL SISTEMA.	29
SITUACIÓN ACTUAL	30
PROCESO ADMINISTRATIVO CONTABLE	31
DESCRIPCIÓN GENERAL.	31
DELIMITACIÓN EN EL ESPACIO.	31
FUNCIONES Y OBJETIVOS.	31
ENTIDADES INVOLUCRADAS.	31
PROCESOS.	32
LIMITACIONES DEL PROCESO ADMINISTRATIVO CONTABLE.	44
PROCESO DE ASIGNACIÓN Y CONTROL PRESUPUESTAL	46
DESCRIPCIÓN DEL PROCESO DEL CONTROL PRESUPUESTAL.	46
DELIMITACIÓN EN EL ESPACIO.	46
FUNCIONES Y OBJETIVOS.	46
ENTIDADES INVOLUCRADAS.	46
PROCESOS.	47
LIMITACIONES DEL PROCESO DE ASIGNACIÓN Y CONTROL PRESUPUESTAL.	52
DESCRIPCIÓN DEL SISTEMA CONTABLE INFORMÁTICO ACTUAL.	53
CARACTERÍSTICAS TÉCNICAS DEL SISTEMA CONTABLE ACTUAL.	53
PROPUESTA Y SOLUCIÓN	63
INTRODUCCIÓN	63
DISEÑO DEL SISTEMA PROPUESTO	64
DESCRIPCIÓN GENERAL DEL SISTEMA PROPUESTO.	64
CONSIDERACIONES GENERALES PARA EL MODULO DE CONTABILIDAD GENERAL	66
CATÁLOGOS DE CUENTAS	66
CUENTAS CONTABLES	66
INGRESO DE PARTIDAS	66
REPORTES CONTABLES	67
VENTAJAS Y LIMITACIONES DEL SISTEMA	68
VENTAJAS DEL SISTEMA	68
LIMITACIONES DEL SISTEMA	68
REQUERIMIENTOS TÉCNICOS PARA EL DESARROLLO Y FUNCIONAMIENTO DE LA APLICACIÓN	70
LA BASE DE DATOS.	70
LA APLICACIÓN	71
LA RED	72
EL HARDWARE	73
DISEÑO DE LA BASE DE DATOS PARA EL SISTEMA CONTABLE.	74
DISEÑO DE LA BASE DE DATOS PARA EL SISTEMA DE CONTROL PRESUPUESTAL.	76
DISEÑO DEL REPOSITORIO DE APLICACIONES Y ENTIDADES INSTITUCIONALES.	77
DEFINICIÓN DE LA BASE DE DATOS.	78
DEFINICIÓN DE TABLAS	80
DEFINICIÓN DE CAMPOS O COLUMNAS	83

TABLA: COP_CONTABILIDAD	84
TABLA: COP_CUENTA_CONTAB	85
TABLA: COP_CUENTA_CONTAB_H	86
TABLA: COP_PARTENC_1	87
TABLA: COP_PARTDET_1	88
TABLA: COP_PARTENC_2	89
TABLA: COP_PARTDET_2	90
TABLA: COP_PARTENC_H	91
TABLA: COP_PARTDET_H	92
TABLA: COP_PERÍODO	93
TABLA: COP_SALDO	94
TABLA: COP_SALDO_H	95
TABLA: COP_T_CUENTA	96
TABLA: COP_T_PARTIDA	97
TABLA: PRE_RUBRO	98
TABLA: PRE_RUBRO_CUENTA	99
TABLA: PRE_RUBRO_CUENTA_H	100
TABLA: PRE_RUBRO_H	101
TABLA: PRE_SALDO	102
TABLA: PRE_SALDO_H	103
PROCESOS DEL SISTEMA	104
DISEÑO DE LOS PRINCIPALES TRIGGERS DEL SISTEMA	105
DISEÑO DE LOS PROCEDIMIENTOS ALMACENADOS	111
DISEÑO DE LA APLICACIÓN.	123
PROGRAMACIÓN DE LA APLICACIÓN GRÁFICA	123
FORMAS GENERALES	130
ESTRUCTURA DE LA APLICACIÓN COP:	145
EL SISTEMA CENTRAL DE REPOSITORIO DE APLICACIONES Y ENTIDADES INSTITUCIONALES CDBREP.	147
PROCESO DE INICIALIZACIÓN DEL SISTEMA COP	149
MENÚ SISTEMA	150
MENÚ: CONTABILIDAD	171
MENÚ: PARTIDAS	187
MENÚ: CONSULTAS	195
MENÚ: REPORTES	198
MENÚ: REPORTES CONSOLIDADOS	207
MENÚ: REPORTES DE PRESUPUESTO	209
MENÚ: PRESUPUESTO	212
MENÚ: AYUDA	223
CONCLUSIONES	224
FUNCIONES DEL CLIENTE, DE LA ADMINISTRACIÓN Y DEL DESARROLLO	227
ASOCIACIÓN CON EL CLIENTE	227
FUNCIÓN DE LA ADMINISTRACIÓN	228
ANÁLISIS DE COSTES/BENEFICIOS Y ADMINISTRACIÓN DE PROYECTOS	228
UTILIZACIÓN DE LOS ESPECIALISTAS DE LA PLANTILLA	230
RESPONSABILIDAD DEL DESARROLLADOR	231
DÓNDE ENCAJAN LOS CONSULTORES?	232
PLANIFICACIÓN Y DESARROLLO CONCURRENTES	233

FORMACIÓN	234
POR QUÉ FALLAN LOS MÉTODOS TRADICIONALES DE DISEÑO?	235
FILOSOFÍA DE DISEÑO E IMPLEMENTACIÓN RECOMENDABLE	236
DISEÑO ITERATIVO	237
DISEÑO EVOLUTIVO	238
DISEÑO AJUSTADO AL PROBLEMA DE NEGOCIOS ORIGINAL	238

GLOSARIO	239
-----------------	------------

TÉRMINOS CONTABLES	239
TÉRMINOS INFORMÁTICOS	240

BIBLIOGRAFÍA	245
---------------------	------------

ANEXOS	246
---------------	------------

ANEXO A: AREAS ADMINISTRATIVAS CONSIDERADAS EN EL PLAN DE DESARROLLO INFORMÁTICO DE LA CIUDADELA DON BOSCO.	247
ANEXO B: APLICACIONES A DESARROLLAR PARA EL ÁREA DE ADMINISTRACIÓN DE LOS RECURSOS ACADÉMICOS	248
ANEXO C: APLICACIONES A DESARROLLAR PARA EL ÁREA DE ADMINISTRACIÓN DE LOS RECURSOS FINANCIEROS.	249
ANEXO D: APLICACIONES A DESARROLLAR PARA EL ÁREA DE ADMINISTRACIÓN DE LOS RECURSOS HUMANOS.	250
ANEXO E: APLICACIONES A DESARROLLAR PARA EL ÁREA DE ADMINISTRACIÓN DE LOS RECURSOS MATERIALES.	251
ANEXO F: REPORTES GENERADOS POR EL ACTUAL SISTEMA DE CONTABILIDAD DE LA CIUDADELA DON BOSCO.	252
ANEXO G: REPORTES DEL SISTEMA COP.	253
ANEXO H: TRIGGERS Y PROCEDIMIENTOS ALMACENADOS DEL SISTEMA COP.	254

INTRODUCCION

La Ciudadela Don Bosco, un proyecto realizado por la obra Salesiana en El Salvador, nació con la idea de brindar educación en todos los niveles en un solo lugar, a través de la integración de las diferentes instituciones educativas que lo componen: Centro de Formación Profesional, Colegio Don Bosco, Oratorio Juvenil, Parroquia “San Juan Bosco”, Universidad Don Bosco y el Centro de Investigaciones y Transferencia de Tecnología. Este propósito se está logrando desde 1990, y las metas alcanzadas han sido bastantes, sin embargo, la evolución de este gran proyecto ha ido a pasos agigantados lo cual ha planteado dificultades y metas más grandes que lograr.

Debido a ello, y al crecimiento de la población a la cual se le presta servicio, las necesidades a nivel administrativo han ido en evolución y crecimiento, y es ahora que se ha llegado al punto de utilizar toda la tecnología y recursos que la Ciudadela ha producido y creado para solventar sus propios problemas y retos futuros. Fue así como se proyectó el Plan Informático de la Ciudadela Don Bosco¹, el cual, entre otros aspectos, plantea la meta de crear aplicaciones informáticas para ayudar a solventar las necesidades administrativas de la Institución, utilizando los recursos humanos, tecnológicos y administrativos de la manera más óptima y eficiente para minimizar la inversión económica y de tiempo que el proyecto requiera.

Esta Tesis es parte de ese gran proyecto y plantea la automatización de los Procesos de Contabilidad General y de Control Presupuestal de la Ciudadela Don Bosco, como parte del área de administración de los recursos financieros. Para lograr este propósito, la Ciudadela optó por capacitar a los alumnos de la Universidad Don Bosco en proceso de graduación para que diseñaran e implementarán los sistemas informáticos que se plantearon en el “Plan de Desarrollo Informático de la Ciudadela Don Bosco”.

Nosotros como parte de ese grupo, presentamos a continuación el avance que ha tenido nuestro proyecto, plasmando en este documento los resultados de la Investigación Preliminar y los Rasgos de la Propuesta para el control de los Procesos Contables y de Control Presupuestal. Consideramos que ésta es la culminación de la primera etapa de nuestro

¹ Ver Anexos

proyecto, y a la vez, la más importante, ya que nos dará las bases para el desarrollo del software necesario y su implementación tomando como fundamento los procesos administrativos, problemas y requerimientos que deben ser considerados en la aplicación a realizar.

Después de esta etapa, continuaremos con el desarrollo de la aplicación, el período de pruebas y su implementación, lo cual se ha venido realizando en trabajo conjunto con los otros grupos de Tesis que tienen a su cargo desarrollar las demás aplicaciones administrativas.

Es importante recalcar, que el éxito o fracaso de cada proyecto, dependerá en gran medida del éxito de los demás proyectos, ya que no podemos juzgar a cada proyecto individualmente sin haber analizado los demás. Esto se debe a la naturaleza integrada de las aplicaciones a desarrollar y del Proyecto Global en sí. Puede pensarse en nuestro proyecto como una pieza del rompecabezas, el cual no estará completo y en armonía, si alguna pieza no encaja o falta.

DESCRIPCIÓN DEL TEMA

El proyecto consta básicamente del análisis e implementación del Sistema Contable actual de la Universidad Don Bosco e incorpora el Módulo de Control Presupuestal, verificando procesos, documentos, datos, etc., a fin de elaborar luego los diseños lógicos y físicos del nuevo Sistema Informático que sustituirá al existente.

Concretamente el Sistema Informático Contable dará apoyo al manejo de las operaciones contables necesarias para la adecuada administración de los recursos financieros que posee la Universidad Don Bosco. Para lograr esto se cuenta con las siguientes fuentes de datos:

- Definición de Cuentas en el Catálogo de Cuentas de la Institución.
- Movimientos contables realizados durante el ejercicio. Estos son proveídos por las partidas realizadas, las cuales al ser mayorizadas actualizan los saldos de las respectivas cuentas.

A partir de esta información serán generados todos los informes, consultas, libros contables y estados financieros necesarios, como son:

- Libro Mayor General
- Libro Diario (Registro de Partidas Contables).
- Saldos de Cuentas a la Fecha
- Histórico de Partidas y Resumen de los ejercicios efectuados en años anteriores.
- Generación de Informes Financieros:
 - Balance General
 - Estado de Resultados
 - Balanza de Comprobación
 - etcétera

Por otro lado el Sistema Contable dará soporte a los siguientes procesos:

- Procesamiento de partidas, aunque no sean del mes en proceso (en el caso de atrasos en la actualización de partidas).
- Cierres contables por períodos.

- Balances y Cierres anuales.
- Reportes Provisionales antes del Cierre Mensual.

El Módulo de Control Presupuestal será desarrollado de forma íntimamente relacionada con la Contabilidad, obteniendo sus datos de las siguientes fuentes:

- Definición de Rubros Presupuestales (relacionados con cuentas contables).
- Definición o Asignación de Saldos Presupuestados anualmente para cada rubro.
- Saldos de las Cuentas Contables asociadas a cada rubro para su respectivo control.

El Sistema contendrá el registro del Presupuesto para cada Unidad, el cual debe haber sido previamente definido por los contadores y administradores para ser ingresado al programa. La información provista por éste y por los elementos mencionados anteriormente permitirá elaborar informes y consultas en línea que facilitarán el control de los gastos e ingresos de la Institución, así como la toma de decisiones con bases reales y al día, ya que los saldos serán continuamente alimentados con datos provenientes de la Contabilidad.

DEFINICIÓN DEL PROBLEMA

En una empresa la administración de recursos es fundamental para la consecución de sus metas. Es función de la Contabilidad el llevar a cabo un control riguroso tanto sobre estos recursos como sobre las obligaciones de la empresa. El realizar dicha función requiere gran cantidad de procesos y usualmente del manejo de grandes volúmenes de información, lo que hace tedioso su control con métodos manuales.

Uno de los métodos más eficientes para realizar dicha tarea es la mecanización de procesos y el procesamiento electrónico de la información utilizando como herramienta la computadora. Sin embargo el mal uso y diseño de sistemas Informáticos puede llegar a ocasionar más problemas de los que resuelve por diversas causas.

La Universidad Don Bosco no ha sido la excepción en implementar dicha solución. Actualmente el Departamento Contable de esta institución cuenta con un sistema computarizado, el cual es utilizado para apoyar las distintas actividades y tareas realizadas. Sin embargo existen ciertos problemas o deficiencias tanto en el programa en sí, como en el uso que se le da, lo cual limita considerablemente la utilidad de Éste para la institución. Entre las principales deficiencias y problemas existentes pueden citarse:

- Las políticas y el diseño de la Contabilidad a nivel institucional (Ciudadela Don Bosco), no se adecua muchas veces a la diversidad de necesidades en cada unidad, lo que aumenta su grado de complejidad.
- El Sistema no es capaz de satisfacer de manera global y eficiente los requerimientos actuales.
- Debido a la tecnología utilizada para su elaboración y funcionamiento, el programa no es capaz de expandirse en la medida en que los requerimientos y volúmenes de información crecen.
- El sistema posee un diseño poco flexible que hace complicado y tedioso los cambios en los

procesos necesarios.

- Dependencia Técnica: Se necesita continuamente la asistencia de quien desarrolló el programa para su frecuente mantenimiento.
- Incapacidad del programa para interactuar en línea con otras aplicaciones utilizadas en la Universidad. No es posible con las herramientas actuales lograr la integración de los sistemas desarrollados.
- El Sistema actual no posee suficientes procesos que puedan regular adecuadamente el flujo de efectivo en base a un presupuesto previamente determinado. Por otro lado aún los pocos procesos soportados no son utilizados como debieran por la administración de la Institución.
- Por otro lado, la Universidad Don Bosco no cuenta con herramientas eficientes para el Control Presupuestal de la misma, lo que le ha dificultado administrar de manera efectiva los recursos financieros con los que cuenta. Por ello, se hace necesario incluir dentro de nuestro proyecto los procesos y datos necesarios para que el Sistema sea capaz de llevar registro y control sobre lo presupuestado, de manera que la Institución pueda obtener la información necesaria para poder ver el flujo de efectivo con respecto a lo que se ha presupuestado en un año determinado.
- Debido a que no se cuenta con datos actualizados sobre los movimientos que van teniendo los saldos contables el Presupuesto sólo puede ser comparado con datos hasta cierto punto demasiado lejanos y retrasados, por lo que no puede ser utilizado adecuadamente como una herramienta de control efectivo de los gastos e ingresos, ya que no permite proyectarse y corregir las fallas que se van dando en el transcurso del Ejercicio en el momento justo.

JUSTIFICACIONES

Consideramos justificable el desarrollo del proyecto por las siguientes razones:

- El cúmulo de nuevos retos y necesidades futuras, inclusive algunas presentes, sobrepasa la capacidad que tiene el actual sistema Informático de solventarlas eficientemente. El rediseño del Sistema Informático de Contabilidad y Control Presupuestal se hace necesario, dado que ni la plataforma en que se basa, ni la estructura lógica del programa permiten realizar los cambios necesarios en el momento justo.
- El sistema debe ser implementado en un lenguaje de 4ª. Generación, utilizando por lo mismo nueva tecnología, plataforma y arquitectura del programa, ya que son éstos requisitos indispensables para asegurar tanto un alto rendimiento y eficiencia del sistema, así como la posibilidad de una mayor vida útil de este.
- La necesidad del cambio de plataforma y estructura, no sólo de Contabilidad General, sino de todas las aplicaciones utilizadas en la UDB, es inminente. Por razones económicas y de tiempo es conveniente para la institución utilizar a los profesionales que ha formado para colaborar en la solución de esta fuerte necesidad.

OBJETIVOS DEL PROYECTO

Objetivo General

Rediseñar e implementar el Sistema de Contabilidad de la Universidad Don Bosco adicionando a éste el Control Presupuestal, dentro del marco y objetivos del Plan de Desarrollo Informático de la misma Institución.

Objetivos Específicos

- Mejorar la eficiencia y corregir las fallas que posee actualmente el Sistema Mecanizado de Contabilidad², realizando el análisis y diseño del Sistema de tal forma que Éste satisfaga adecuada y eficientemente las necesidades y requerimientos del Departamento de Contabilidad de la Institución, tanto internos como legales.
- Cumplir con los objetivos del Plan de Desarrollo Informático de la Universidad, brindando al Programa la capacidad de integrarse con otras Aplicaciones (todas aquellas que generen partidas contables) para obtener información consistente, integrada y oportuna para aquellas Instancias encargadas de la administración de los recursos financieros de la UDB.
- Crear el Sistema con una estructura flexible, la cual permita posteriormente realizar cambios o actualizaciones necesarios de una manera sencilla, afectando al mínimo a los demás procesos existentes.
- Utilizar para su Desarrollo y Funcionamiento herramientas de Software y Hardware que permitan implementar Sistemas seguros tanto en la integridad de la información como en las restricciones de acceso (Seguridad de Usuarios).
- Producir un Sistema amigable con el Usuario, incluyéndole las características necesarias

para hacer fácil su uso, como son interfaces gráficas estándar, ayuda en línea, mensajes adecuados y oportunos, manuales, etc.

- Optimizar el uso de Documentos, Reportes y Formularios, a través de una mejor y mayor utilización del Computador personal para la obtención y manipulación de la información. Esto ayudará a disminuir y simplificar el flujo de documentos en las diferentes unidades.
- Proporcionar un sistema bien documentado, que incluya manuales para el usuario y para el sistema, de manera que proporcionen una guía fácil y eficiente a los usuarios del mismo, y que además documenten su diseño y programación para facilitar su mantenimiento futuro, incluso por personas que no estuviesen involucradas en su desarrollo.

ALCANCES Y LIMITACIONES

Alcance del Sistema Contable.

El Sistema Contable inicia desde el ingreso y registro de las Partidas Contables, ya sea que hayan sido generadas por sistemas Auxiliares a la Contabilidad (ej.: Cuentas por Pagar, Colecturía, Activo Fijo, etc.) o introducidas manualmente por un operador del sistema. Contemplará desde la Mayorización de las partidas, actualización de saldos, generación de reportes financieros, consolidación de contabilidades, cierres de ejercicio, hasta el traslado de toda esta información a históricos.

Alcance del Módulo de Control Presupuestal.

Este módulo trabajará conjuntamente con la Contabilidad General, permitiendo la definición de un control Presupuestal por cada Contabilidad definida dentro del sistema.

Su campo de acción abarca el registro de los rubros presupuestados para un período específico y la relación de estos rubros de presupuesto con las cuentas contables de las cuales se quiera llevar control. En ningún momento incluye el conjunto de políticas y tomas de decisiones que implican la asignación y autorización de estos rubros a los Centros de Costos.

Su objetivo principal será facilitar a los usuarios la información necesaria para llevar un adecuado control de sus Ingresos y Gastos, permitiendo en su momento la reasignación de fondos a los Centros de Costos involucrados.

Limitaciones del Sistema Contable.

No se incluye dentro del diseño del Sistema Contable los módulos correspondientes a sistemas considerados como Auxiliares de la Contabilidad, como son Cuentas por Pagar, Cuentas por Cobrar, Planillas, Activo Fijo, etc., sin embargo tendrá una estructura abierta que permita comunicarse con estos módulos cuando sean implementados, para poder así aceptar de manera automática las partidas contables que sean generadas por Éstos.

El Sistema servirá a todas las Instituciones o Unidades dentro de la Ciudadela, por lo que se diseñará de manera que cumpla con los requerimientos generales y primordiales para todas ellas. Debido a esto y a las limitantes de tiempo y recursos humanos y técnicos disponibles se dejarán fuera del actual diseño algunos informes y consultas especiales que son muy específicas para ciertas Unidades. Sin embargo la estructura flexible del programa permitirá en un futuro apoyar dichas necesidades y solicitudes.

Limitaciones del Módulo de Control Presupuestal.

Este módulo permitirá efectuar un control adecuado de los rubros definidos relacionándolos con la Contabilidad de la Unidad respectiva. Sin embargo, dicha relación limita o restringe hasta cierto punto la efectividad del Sistema, ya que existe una dependencia directa entre los saldos utilizados en el programa y los datos que sean ingresados en Contabilidad para dichas cuentas. En caso que alguna Contabilidad esté retrasada por algún motivo, el módulo de presupuestos hará uso tan sólo de los datos existentes, mostrando una situación financiera no actualizada. En vista de esto el programa permitirá hacer uso de las partidas aún no mayorizadas a fin de hacer ciertas estimaciones o aproximaciones a la realidad.

ANTECEDENTES INFORMÁTICOS

Con el propósito de lograr los objetivos antes mencionados, se ha considerado el desarrollo de la aplicación de acuerdo a los criterios y diseños planteados en el Plan de Desarrollo Informático de la CDB³. Este plan ha dispuesto el diseño y desarrollo de las aplicaciones clasificándolas según el tipo de recursos a los que están orientados a administrar, esto es:

- **Recursos Financieros:** Incluye los Módulos de Contabilidad, Cuentas por Pagar, Cuentas por Cobrar, Registro del IVA, Control Presupuestario, Caja y Bancos (Colecturía), y Proveedores/Pagaduría.
- **Recursos académicos:** Para su administración, se consideran las aplicaciones de Registro académico, Administración académica (Pénsums, Horarios, Contenido, etc.), Control de Trabajos de Graduación, Control de Cursos Libres/Capacitaciones, Administración de Biblioteca, y el Registro del Servicio Social.
- **Recursos Materiales:** Está integrado por las siguientes aplicaciones de Inventarios: de Recursos Consumibles, Inventarios de Activo Fijo, Inventario de Recursos Técnicos, Inventario de Libros (Biblioteca), e Inventario de Materiales en Venta.
- **Recursos Humanos:** Incluye los Sistemas de Registro de Empleados, Expediente de Personal, Selección y Reclutamiento de Personal y Planillas.

El Plan contempla la integración de estas aplicaciones, compartiendo aquella información común entre ellas, evitando así la redundancia de Datos.

Por esta razón el enfoque de nuestro Proyecto está regulado por las normas de diseño que dicho Plan impone en cuanto a Estructuración de Bases de Datos, Programas, y Herramientas de Desarrollo a utilizar.

³ Ver Anexos.

MARCO TEÓRICO

Las Herramientas a utilizar en el desarrollo e implementación de los Sistemas serán un Manejador de Bases de Datos Relacionales como gestor de la información y Lenguajes de Programación de 4a. Generación (4GL) para la construcción de las aplicaciones. La causa principal de ello se debe a las múltiples ventajas que poseen respecto a los Lenguajes y Manejadores de Bases de Datos utilizados comúnmente. Los tres modelos populares que han sido utilizados para organizar una Base de Datos son: *Jerárquico*, de *Redes* y *Relacional*. Estos tres modelos han evolucionado gradualmente a medida que los Usuarios y los Profesionales en Computación han ganado experiencia en el uso de Sistemas Manejadores de Bases de Datos (DBMS). Ellos difieren en términos de costos de implementación, velocidad, grado de redundancia de datos, facilidad con la que se puede satisfacer los requerimientos de información, y la facilidad con la que las Aplicaciones pueden ser actualizadas.

Usados desde finales de la década de los 60's, los modelos para Bases de Datos Jerárquico y de Redes, fueron primeramente desarrollados y utilizados en computadores de tipo Mainframe. Los conceptos tras el Modelo de Bases de Datos Relacional fueron introducidos a principios de los 70's por E. F. Codd y desarrollados a través de esa década. Este Modelo de Bases de Datos ha sido usado ampliamente en Microcomputadoras (PC: Computadoras Personales) y está también siendo usado más y más en computadores más poderosos para aplicaciones a mayor escala.

Por otro lado, los Lenguajes 4GL permiten acelerar la creación de Sistemas, trabajando en estrecha relación con los usuarios y logrando que el Programador se concentre más en el análisis que en los tecnicismos de la programación, de manera que logre satisfacer adecuadamente las necesidades y requerimientos. Además permite lograr fácilmente la integración de distintas aplicaciones.

Se trabajará bajo una topología o estructuración de Cliente-Servidor, esto es, centralizando los Datos en un Computador (Servidor) que manipulará dicha información para poderla compartir con otras terminales (aplicaciones clientes). Esto trae diversas ventajas entre las que podemos mencionar:

- Se reduce el tráfico en las redes, ya que el Cliente (PC que ejecuta la aplicación) solicita la información o la ejecución de un proceso al servidor, y este a su vez lo hace y envía una respuesta (la información solicitada, estado de ejecución de un proceso, o un error), de tal forma que solo la información (respuesta del servidor) y las solicitudes del cliente viajan a través de la red.
- Permite compartir recursos, debido a que ciertos procesos son ejecutados por el Computador cliente (PC) y otros por el Servidor, por tanto se tiene menores tiempos de respuesta y más seguridad.
- Hay mayor seguridad manteniendo la integridad de los datos en caso de una falla, ya que el servidor es el único encargado de manipular en forma directa la información. Aún en caso de fallas en la red o en el PC Cliente, el servidor continuará procesando hasta la última petición hecha por el cliente antes que ocurriera la falla.
- Reduce los Costos, ya que al ser el Servidor el único Computador (puede existir más de un servidor) que contiene los datos y los manipula directamente, solo hay que proteger una máquina. En el caso de varias PC's con diferentes Sistemas, y con los datos distribuidos en estas, se hace necesario proteger cada uno de estos computadores para no perder la información en caso de fallas, lo cual incrementa los costos.
- Facilita la Integración de varios Sistemas. Al estar centralizada la información, se tiene la ventaja de poder tener información de diversos sistemas a la vez, permitiendo el flujo de datos entre ellos.

METODOLOGÍA DE INVESTIGACIÓN Y DESARROLLO

La Metodología utilizada tanto para la investigación como para el desarrollo del proyecto es el llamado método del Ciclo de Vida para el Desarrollo de Sistemas o Método de la Cascada, el cual incluye todas las actividades que los analistas, diseñadores y usuarios realizan para desarrollar e implantar un sistema. Las diversas partes del proyecto pueden encontrarse al mismo tiempo en distintas fases del desarrollo.

Se muestran en esta sección las etapas que componen dicha metodología, incluyendo tanto su concepción teórica como el avance realizado por el grupo en su desarrollo.

Etapas del ciclo de Vida para el Desarrollo de Sistemas.

Concepción de la idea o proyecto.

La idea del desarrollo de la aplicación de Contabilidad General surgió a partir de la necesidad de la Administración de la UDB de tener aplicaciones integradas y de fácil actualización.

Investigación Preliminar

La investigación preliminar incluye un planteamiento o una definición más precisa de la idea global, así como un pequeño estudio de factibilidad, basado principalmente en el juicio y experiencia de los realizadores del proyecto, así como en consultas con personal de la Universidad. El estudio de factibilidad básicamente se concentra en los siguientes puntos:

- **Factibilidad Técnica:** aquí se dará respuesta a ciertas interrogantes, como si existe equipo disponible, tecnología existente de Software y el personal disponible.
- **Factibilidad Económica:** en este caso se establece principalmente si al crear el sistema los beneficios que se obtendrán serán suficientes para aceptar los costos.

- **Factibilidad Operacional:** se debe determinar si en caso de que se desarrolle e implante el sistema en realidad se hará uso de éste, y si no hay factores que pudiesen incidir negativamente en el uso y operación normal del mismo.

Durante esta etapa se realizaron varias entrevistas y reuniones con las personas responsables del proyecto por parte de la Institución, como son los Administradores y Contadores de cada una de las Unidades de la Ciudadela, así como con el Administrador General de ésta. A partir de tales entrevistas pudo obtenerse suficiente información para lograr una perspectiva general de la factibilidad del proyecto en sus distintas áreas:

Factibilidad Técnica:

Para lograr el desarrollo e implementación del Sistema son necesarios diversos elementos, los cuales son:

Hardware

- Computadores con suficiente capacidad para el desarrollo de las aplicaciones.
- Computadores que ejecutarán la aplicación una vez este en producción.
- El Computador principal que manejará las Bases de Datos
- El Medio de comunicación entre el Computador principal y los computadores clientes.

Software

- El Software de Administración de Bases de Datos (RDBMS)
- Herramientas de Desarrollo de Aplicaciones
- Módulos y Librerías necesarias para la comunicación con la Base de Datos.
- Un sistema operativo con Interfaz Gráfico para el desarrollo y ejecución de las aplicaciones.

De los elementos anteriormente mencionados, la Ciudadela Don Bosco posee con todos los recursos excepto el medio físico para interconectar las PC's clientes con el Computador Central, sin embargo la implementación de este proyecto está contemplado para

inicios del año 1997.

Por otro lado, es obvio que será necesario los conocimientos y el soporte tanto en el área de Desarrollo de aplicaciones como en la administración de las Bases de Datos. Actualmente el desarrollo se hará en una red local (LAN), utilizando Visual BASIC como herramienta de desarrollo y Sybase SQL Server X como Servidor de Bases de Datos, instalado en un Computador IBM RS/6000 con Sistema Operativo AIX (Unix).

Factibilidad Económica:

Normalmente los costos de realizar un proyecto de tal magnitud como es el Proyecto Informático de la Ciudadela (incluyendo dentro de éste al Sistema de Contabilidad y Control Presupuestal) son bastante altos. Sin embargo la institución no tendrá que realizar todo el monto de gastos normal, sino sólo una parte de ellos, ya que tanto el Manejador de la Base de Datos como ciertas herramientas de desarrollo han sido donadas a la Ciudadela, lo que disminuye enormemente los costos del Proyecto.

Por otro lado los gastos respecto al personal de desarrollo, incluyendo tanto capacitación como remuneración económica son nulos, ya que la capacitación también fue donada a la Ciudadela y el personal que desarrollará el Sistema seremos estudiantes egresados de la Universidad realizando un trabajo de tesis.

El único costo obligatorio por el momento es la inversión necesaria para la implementación de la Red que interconectará las diferentes dependencias en el Campus de la Ciudadela, de lo cual ya existe un proyecto en proceso de implementación.

En cuanto a las ventajas que esto traerá, la eficiencia y ahorro de tiempo en la ejecución de procesos administrativos, permitirá a la institución agilizar e incrementar los servicios prestados a los alumnos, así como a otros clientes, de manera que pueda ir incrementando la población y por tanto los ingresos.

Factibilidad Operacional:

Los principales elementos a considerar una vez inmersos en la implementación y puesta en producción de los sistemas, serán los recursos humanos que administrarán tanto la Base de Datos como las aplicaciones desarrolladas; y la capacitación oportuna del personal que hará uso de las aplicaciones para asegurar el buen uso de estas. La Ciudadela Don Bosco

ya cuenta con el personal que estará a cargo de la operación de las aplicaciones, de quienes podríamos decir, su nivel de conocimientos en cuanto al uso de PC's y aplicaciones administrativas es aceptable. Por otro lado, el personal para la administración de aplicaciones y Bases de Datos aún no ha sido asignada, y tampoco existe actualmente la cantidad necesaria de recursos humanos que puedan suplir esta necesidad, sin embargo esto se ha planeado para un futuro en la etapa de implementación de las aplicaciones.

Como se ha determinado brevemente la institución cuenta con todos los recursos necesarios para el desarrollo e implementación de los sistemas, tanto en las áreas técnicas, económicas y operacionales, lo cual nos permite continuar en el proceso del desarrollo de éstos, pasando luego a la siguiente etapa del Ciclo de Vida.

Determinación de los requerimientos del sistema.

Se estudian los procesos necesarios para dar respuesta a las siguientes interrogantes:

- Qué es lo que se hará?
- Cómo se hará?
- Quien lo hará?

Básicamente se utilizaron tanto entrevistas personales como reuniones grupales con las personas que actualmente realizan las tareas concernientes a la Contabilidad y el Control Presupuestal, y que posteriormente estarían encargadas de la operación del sistema.

Se estudiaron los datos sobre requerimientos con la finalidad de identificar las características que debe tener el sistema, incluyendo datos de entrada, procesos a realizar y salidas de información que deberá producir el nuevo sistema.

Debido al tipo de técnicas de desarrollo y programación ésta es la parte fundamental del proyecto, ya que es aquí donde se concibe el entorno y el contenido del Sistema a realizar, dedicándole por lo mismo el mayor tiempo disponible, buscando siempre la claridad y exactitud en el estudio del sistema actual. A partir de este estudio se ha elaborado una serie de diagramas y flujos de los procesos administrativos encontrados, los cuales son presentados en la siguiente sección del documento. Se realizó también un análisis sobre el Sistema Contable en uso actualmente, ya que es en base a éste que se realizan la gran mayoría de procesos

contables en todas las Unidades, por lo que el nuevo sistema debe tomarlo muy en cuenta a la hora del diseño, ya que debe contener al menos las opciones y capacidades de éste, adicionándole nuevas potencialidades y características que mejoren el desempeño de los departamentos contables de la Ciudadela.

Toda la información aquí recopilada fue debidamente organizada y presentada en su momento a las personas responsables de la Administración y Contabilidad de las distintas Unidades de la Ciudadela, así como a un grupo de asesores estadounidenses que realizan un estudio en todas las áreas de la Institución, constituyéndose así en el núcleo del presente documento.

Diseño del Sistema.

Se detallará en esta etapa la forma en que el sistema cumplirá con los requerimientos identificados durante la fase de análisis. Se identifican todos los tipos de salidas que producirá el sistema y los datos específicos para cada reporte y cada salida, se harán los bosquejos de pantallas que aparecerán cuando el sistema está terminado, se especificarán los datos de entrada, los datos a ser calculados y la codificación de los datos que serán almacenados. Las especificaciones se representarán por medio de diagramas, tablas y símbolos especiales.

Actualmente se ha iniciado esta etapa del desarrollo del proyecto, comenzando por definir y estandarizar adecuadamente los datos de los que hará uso el sistema, así como su estructura y relaciones. Se han considerado asimismo las variantes y mejoras respecto al sistema actual, tanto a nivel técnico como operacional de manera que se facilite lo más posible su uso y brinde un mejor control de los datos y procesos contables y presupuestales. Se presenta al final del documento una serie de dichas consideraciones generales sobre el diseño propuesto, de tal manera que sirva como una introducción a la siguiente etapa evaluativa del desarrollo del sistema.

Desarrollo del Software o programa.

Una vez diseñado el sistema se procederá a su desarrollo utilizando un lenguaje de programación de cuarta generación, haciendo uso además de las herramientas disponibles a fin de acelerar lo más posible el proceso. Se elaborará su respectiva documentación, ya que ésta será indispensable para facilitar el mantenimiento que sea necesario una vez la aplicación esté instalada.

Prueba del sistema.

Se experimentará para asegurarnos que el Software no tenga problemas en su ejecución, es decir que funcione de acuerdo a las especificaciones y la forma en que los usuarios esperan que lo haga.

Se procesarán algunos datos y después se verificarán los resultados.

Se verificará asimismo que los usuarios utilicen el sistema de la forma debida.

El método del ciclo de vida puede ser combinado con otra técnica o metodología del desarrollo de sistemas, llamada "Construcción de Prototipos", la cual consiste en ir construyendo distintos diseños o prototipos sencillos basados en la información que se tenga en ese momento, para presentarlo al usuario y que sea éste quien vaya evaluando si vamos en la dirección correcta, o en su caso que haga observaciones a este diseño. Con estas observaciones se puede ir puliendo el sistema, a fin de adecuarlo en lo posible con lo que el usuario necesita.

Esta es la metodología en la que nos basamos para el desarrollo del presente proyecto, construyéndolo a través de pasos concretos y sistemáticos, por medio de los cuales el Software a crear será hecho de forma ordenada y precisa, siguiendo una estructura iterativa o cíclica debido a la frecuente retroalimentación por parte del usuario, lo cual permite que el Sistema avance en la dirección correcta y mejore su calidad, satisfaciendo los requerimientos de forma óptima.

SITUACIÓN ACTUAL

El núcleo del presente Capítulo consiste en la presentación de los resultados de la Investigación Preliminar realizada en las Unidades que conforman la Ciudadela Don Bosco para el Proyecto de Automatización de los Procesos Contables y de Control Presupuestal de la misma Institución.

Su objetivo principal es sentar las bases para las siguientes etapas que forman dicho proyecto, las cuales son las de formular y aprobar una alternativa de solución y proceder a su desarrollo. Al final del mismo, se presentan los rasgos de la Propuesta de Solución que se ha definido y que aún están sujetas a cambio debido a la constante retroalimentación de los usuarios.

Por otro lado, el contenido plasmado en las siguientes páginas presenta la situación actual de cómo son realizados los procesos administrativos principales involucrados en el Proyecto de Automatización, sin centralizarse inicialmente en los procesos operativos o mecanizados, sino más bien en el Proceso en General, en donde cualquiera de sus procedimientos puede estar mecanizado o no. La información presentada en el documento es basada en gran parte en la información proporcionada por los mismas Personas involucradas en los procesos analizados, tales como Administradores de Unidad, Jefes de Área, Contadores y otros.

Nuestra visión ha sido siempre la de hacer un buen diseño basado enteramente en la realidad de cómo son llevados los procesos actualmente y no en suposiciones o información errónea, es por ello que consideramos importante someter los resultados de nuestra investigación a la constante revisión de las mismas personas que serán beneficiadas con el proyecto, no sólo con el fin de evaluarnos, sino también para analizar las acciones a tomar con todos aquellos aspectos que no son claros o no han sido debidamente definidos a nivel operativo y administrativo.

Se detallarán a continuación los procedimientos u operaciones que componen los Procesos Administrativos de Contabilidad y Control Presupuestal, especificándose además para cada uno su lugar en la secuencia de procesos, sus pre-requisitos, funciones, objetivos y flujo de actividades.

Proceso Administrativo Contable

Descripción General.

Conjunto de Normas y Procedimientos para ordenar, analizar y registrar las operaciones financieras practicadas por las unidades que forman la Ciudadela Don Bosco.

Delimitación en el Espacio.

Se da desde el Nivel de Administración General de la CDB hasta todas las unidades que la forman.

Funciones y objetivos.

- Establecer un control riguroso sobre cada uno de los recursos y las obligaciones de la Institución.
- Registrar, en forma clara y precisa, todas las operaciones efectuadas por la Institución durante el período fiscal o Ejercicio Contable.
- Proporcionar, en cualquier momento, una imagen clara y verídica de la situación financiera de las distintas Unidades de la Ciudadela.
- Prever con bastante anticipación el futuro de la Institución.
- Servir como comprobante y fuente de información, ante terceras personas, de todos aquellos actos de carácter jurídico en que la contabilidad puede tener fuerza probatoria conforme a lo establecido por la ley.

Entidades Involucradas.

Involucra las siguientes Unidades:

- Centro de Formación Profesional.
- Centro de Investigación y Transferencia de Tecnología.
- Colegio Don Bosco, Oratorio y Parroquia.
- Universidad Don Bosco.

- Comunidad Salesiana.

Procesos.

Las operaciones o actividades que conforman el Proceso administrativo contable son:

- Mantenimiento del Catálogo de Cuentas
- Procesamiento de Partidas
- Mayorización
- Generación de Informes
- Cierre Contable
- Consolidación

Estos se tratan a continuación basándonos en la investigación realizada.

Mantenimiento del Catálogo.

Definición:

Consiste en la Creación, Modificación o Anulación de Cuentas Contables a causa de las necesidades imperantes para llevar un mejor control financiero.

Ubicación Temporal:

Este proceso puede realizarse en cualquier momento del ejercicio contable.

Objetivos:

Mantener una estructura de cuentas acorde a las necesidades administrativas de la Institución.

Pre-requisitos:

Análisis de las necesidades administrativas y financieras de la empresa.

Flujo del Proceso:

Adición de Cuentas Contables:

1. Análisis de la situación administrativa para determinar que cuentas se van a crear.

2. Avisar al Economato sobre la creación de la cuenta (cuando sea necesario).
3. Consenso entre las unidades para la codificación y utilización de dichas cuentas. (Cuando sea necesario).
4. Verificaciones para la creación de la Cuenta:
5. Que exista la cuenta padre (Si es necesaria).
6. La cuenta padre debe tener saldo igual a cero.
7. No debe existir previamente otra cuenta con el mismo código.
8. El código de la cuenta debe apegarse a las reglas establecidas por el Economato Salesiano.
9. Si cumple con las condiciones anteriores se crea la cuenta.

Modificación de Cuentas contables:

1. Verificar que el saldo de la cuenta sea cero y que no pertenezca a las cuentas definidas por el Economato.
2. Modificación del concepto de la cuenta.

Anulación de cuentas contables:

1. Verificar que el saldo de la cuenta sea cero.
2. Verificar que esa cuenta sea de detalle y que no pertenezca a las cuentas definidas por el Economato.
3. Poner fuera de uso la cuenta.

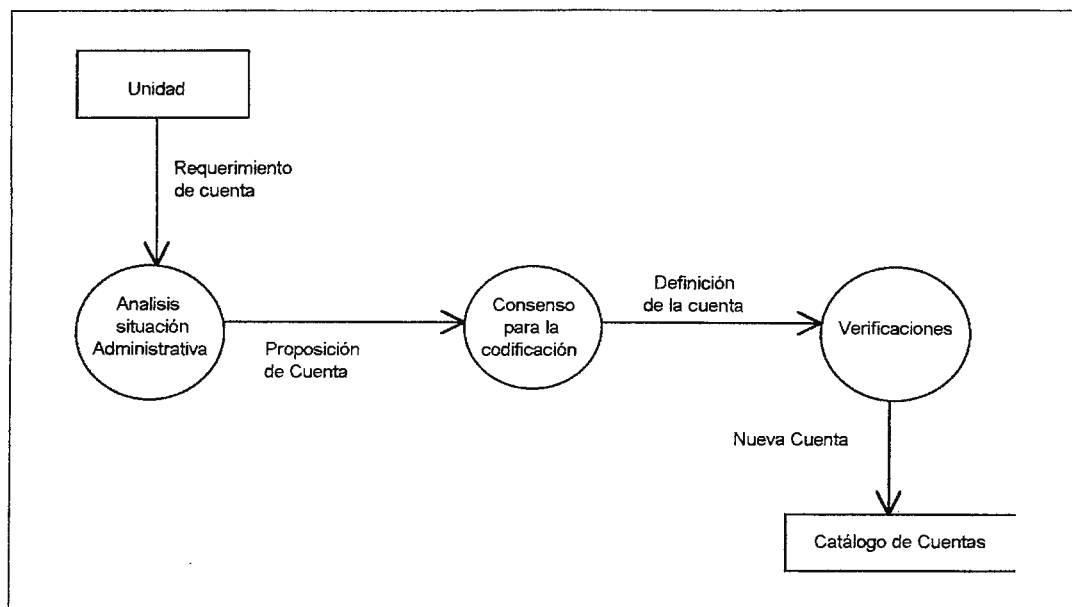


Ilustración 1: Diagrama de Flujo de Datos para el proceso de adición de cuentas contables.

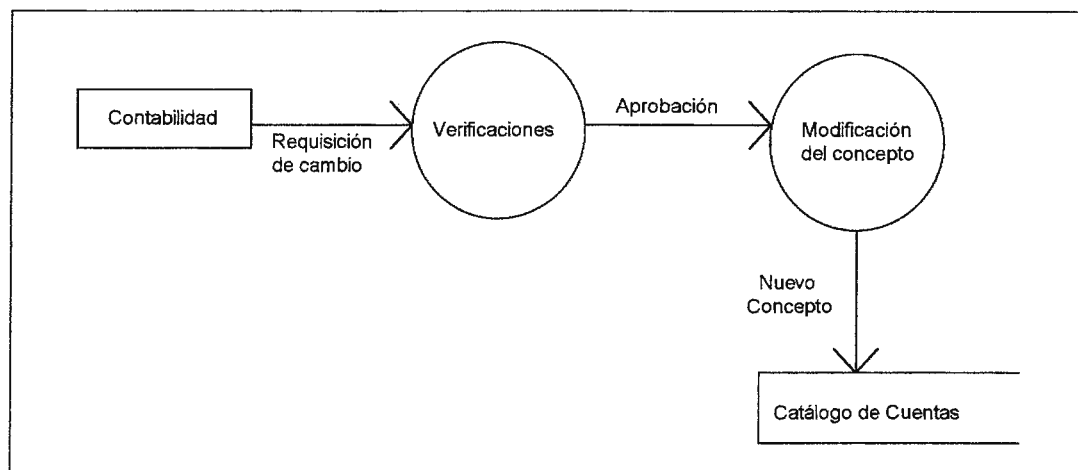


Ilustración 2: Diagrama de Flujo de Datos de la modificación del concepto de una cuenta contable.

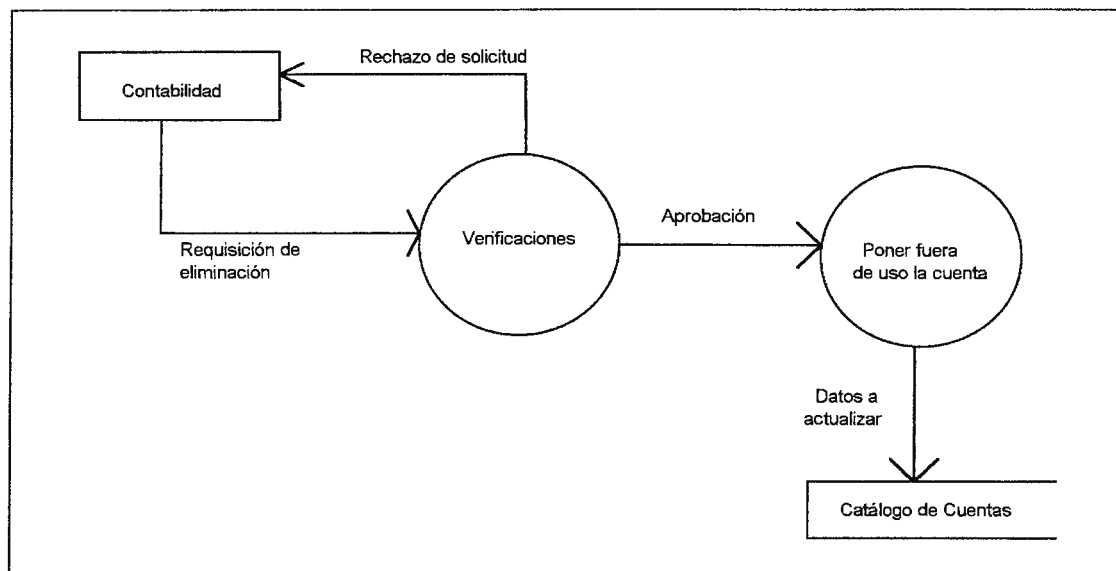


Ilustración 3: Diagrama de Flujo de Datos de la anulación de una cuenta contable.

Procesamiento de Partidas.

Definición:

Consiste en la creación de documentos de registro para las operaciones financieras realizadas en la Institución.

Ubicación temporal:

Este proceso se realiza a lo largo de todo el ejercicio contable.

Objetivos:

Registrar las transacciones comerciales y todas las operaciones que afecten de alguna manera los Activos, Pasivos o el Capital de la Institución.

Pre-requisitos:

- Ocurrencia de una transacción comercial o una operación financiera.
- Análisis de la transacción.

- Existencia de documentos de respaldo de la transacción.
- Existencia de las cuentas contables dentro del Catálogo referidas a los conceptos involucrados en la transacción.

Flujo del Proceso:

1. Se crea la partida como efecto de una operación realizada, cuyos movimientos son resultado del análisis del contador.
2. Para poder ingresar una partida se verifica:
3. Que la fecha de la partida no pertenezca a un período cerrado.
4. Que sus movimientos hagan referencia a cuentas de detalle existentes y en uso, dentro del catálogo de la unidad.
5. Que cuadre el Debe con el Haber.
6. Mientras no haya sido mayorizada, la partida puede ser modificada o eliminada.

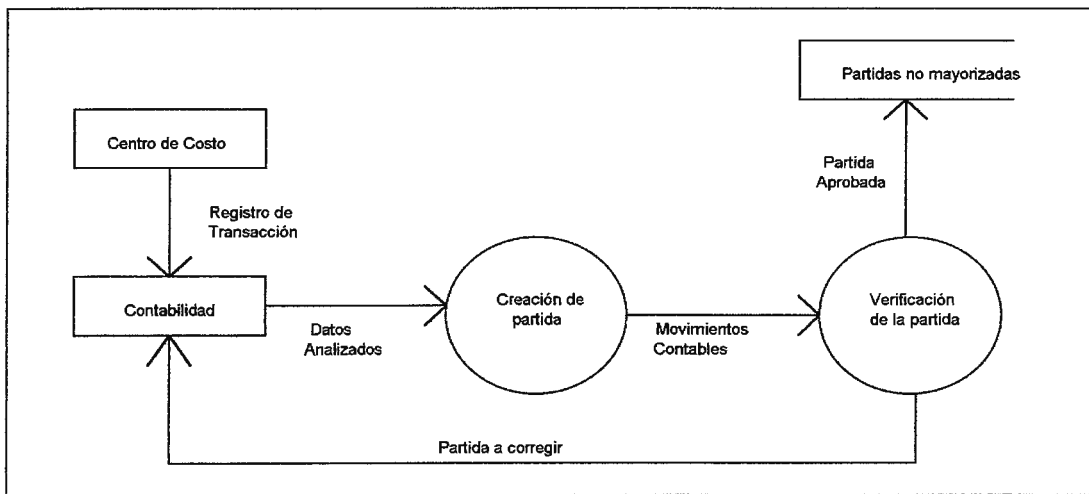


Ilustración 4: Diagrama de Flujo de Datos del Procesamiento de Partidas contables.

Mayorización.

Definición:

Consiste en la actualización de los saldos de las cuentas en el Libro Mayor a partir de los valores registrados en las partidas.

Ubicación temporal:

El proceso puede realizarse durante todo el ejercicio contable, y es repetido cada vez que se autoriza una partida.

Objetivos:

Mantener actualizados los saldos de las cuentas contables, de manera que reflejen los valores de las transacciones financieras y operaciones registradas en las partidas.

Pre-requisitos:

- Procesamiento de la partida.
- Revisión y Autorización de la partida.

Flujo del Proceso:

1. Para poder mayorizar una partida se verifica:
2. que la fecha de la partida no pertenezca a un periodo cerrado.
3. Actualizar los saldos de las cuentas involucradas en los movimientos de la partida en el Libro Mayor.

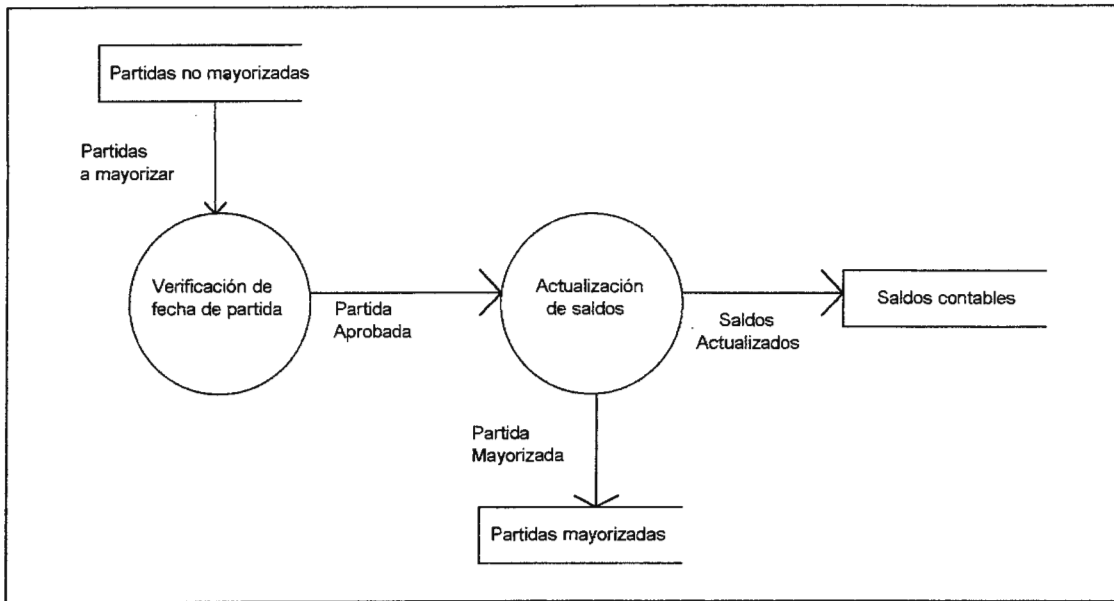


Ilustración 5: Diagrama de Flujo de Datos del Proceso de mayorización de Saldos Contables.

Generación de Informes.

Definición:

Consiste en la generación de los distintos informes que reflejan la situación contable de la institución.

Ubicación temporal:

Puede realizarse en cualquier momento del ejercicio contable, siempre que se desee conocer la situación contable de la institución.

Objetivos:

Informar a Instancias tanto internas como externas de la situación financiera de la Institución.

Pre-requisitos:

Registro de las operaciones a través de partidas, hayan sido mayorizadas o no.

Flujo del Proceso:

1. Determinación de requerimientos de información contable.
2. Recopilación de la información almacenada necesaria.
3. Estructuración del informe (cálculos, distribución, agrupamientos, etc.).

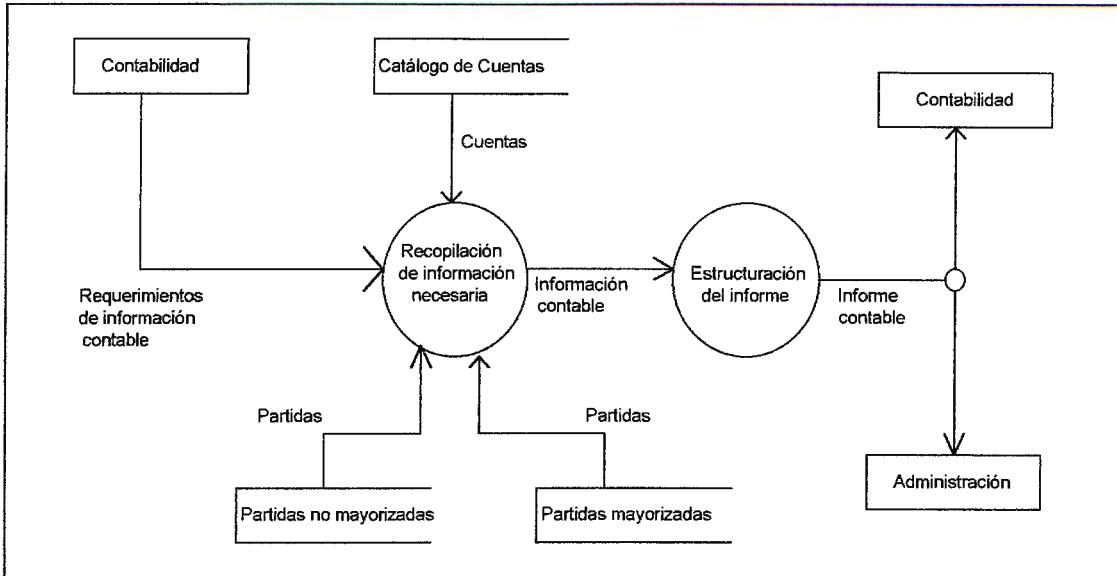


Ilustración 6: Diagrama de Flujo de Datos de la generación de reportes contables.

Impresión del Informe.

Los Reportes más importantes dentro de la Contabilidad se detallan a continuación.

Balance General o Estado de Situación Financiera.

Definición:

Es el documento contable que representa la situación financiera de la Institución a una fecha determinada.

Ubicación Temporal:

Puede ocurrir en cualquier período del Ejercicio Contable.

Objetivos:

Dar a conocer la situación financiera a la administración de la Institución, contribuyentes o

donantes (Empresa Privada, Gobiernos Extranjeros, etc.) y principalmente al estado.

Pre-requisitos:

Haber Actualizado las partidas pertenecientes al período del Informe.

Flujo del Proceso:

1. Elaborar el Encabezado del Balance
2. Clasificación de las cuentas de Activo.
3. Clasificación de las cuentas de Pasivo.
4. Determinación del Capital mediante la formula Activo - Pasivo.
5. Autorización de parte del Contador y/o del Gerente.

Estado de Pérdidas y Ganancias o Estado de Resultados

Definición:

Es el documento contable que muestra detallada y ordenadamente la utilidad o pérdida del ejercicio.

Ubicación Temporal:

Puede ocurrir en cualquier periodo del Ejercicio Contable.

Objetivos:

Dar a conocer la forma como se ha obtenido la utilidad o pérdida del ejercicio detalladamente.

Pre-requisitos:

Haber Actualizado las partidas pertenecientes al período del Informe.

Flujo del Proceso:

1. Elaborar el encabezado del Estado de Pérdidas y Ganancias.
2. Analizar todos los elementos que entran en la compraventa de bienes y/o servicios.
3. Determinar la utilidad o pérdida de ventas mediante la diferencia entre el precio de costo y de venta de los bienes y/o servicios prestados.

4. Analizar detalladamente los gastos de operación, así como los gastos e ingresos que no corresponden a la actividad principal de cada una de las Instituciones dentro de la Ciudadela, y determinar el valor neto que debe restarse de la utilidad bruta, para obtener la utilidad o la pérdida líquida del Ejercicio.
5. Autorización de parte del Contador y/o del Gerente.

Balance de Comprobación

Definición:

Es el documento contable que muestra detalladamente los movimientos tanto deudores como acreedores de todas las cuentas.

Ubicación Temporal:

Se puede hacer en cualquier momento del Ejercicio, realizándolo cuando sea necesario, aunque se acostumbra formularlo a fin de mes.

Objetivos:

Comprobar si todos los cargos y abonos de los asientos del Diario han sido registrados en las cuentas del Mayor respetando la partida doble, así como facilitar la preparación de la Balanza que se presenta a fin de año.

Pre-requisitos:

Haber Actualizado ya sea algunas o todas las partidas pertenecientes al período del Informe.

Flujo del Proceso:

1. Elaborar el encabezado de la Balanza de Comprobación.
2. Anotar folio, nombre y movimientos tanto deudor como acreedor de cada una de las cuentas.
3. Determinar el tipo de saldo de cada una de las cuentas.
4. Sumar los movimientos para determinar el equilibrio entre los movimientos acreedores y deudores.

Cierre Contable.**Definición:**

Consiste en el cúmulo de actividades y operaciones necesarias para dar por finalizados los procesos contables de un período determinado.

Ubicación temporal:

Al final de cada período contable definido.

Objetivos:

Consolidar la información financiera de la Institución para un período dado.

Pre-requisitos:

Registro y Mayorización de todas las operaciones pertinentes al período a cerrar.

Revisión y ajuste de la información contable del período.

Flujo del Proceso:

1. Verificar que todas las partidas del periodo a cerrar hayan sido mayorizadas.
2. Se elaboran estados financieros y reportes preliminares para efectos de consulta y revisión.
3. Se realizan todas las partidas de ajuste necesarias para que los saldos de las cuentas contables correspondan con la realidad.
4. Se mayorizan las partidas de ajuste.
5. Se elaboran los estados financieros definitivos
6. En el caso del cierre anual:
7. Se saldan las cuentas del ejercicio.
8. Se establecen los saldos para el nuevo ejercicio.

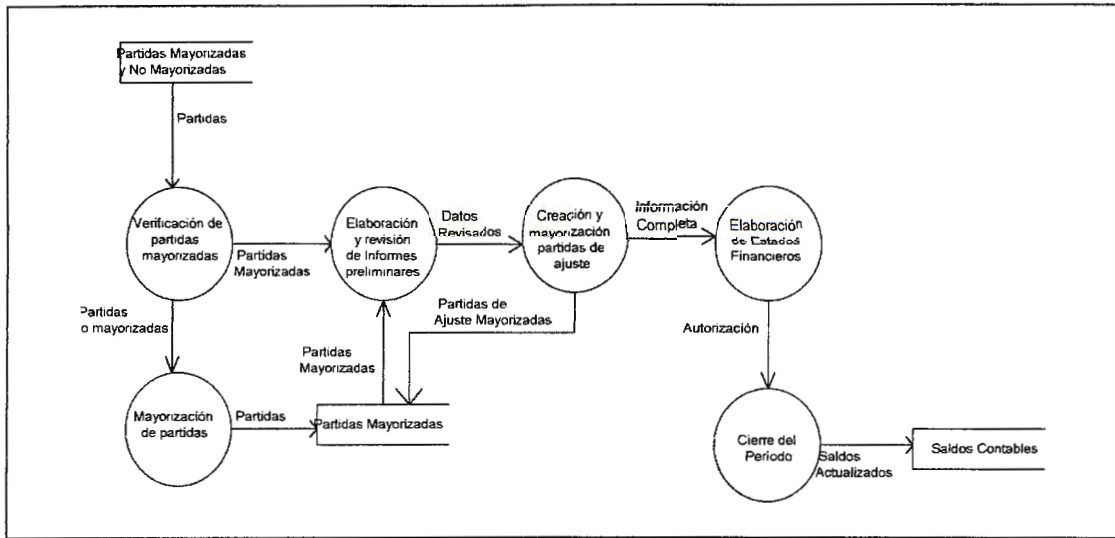


Ilustración 7: Diagrama de Flujo de Datos del Proceso de Cierre Contable de un periodo.

Consolidación.

Definición:

Integrar la información contable de varias unidades, para efectos de determinar la situación financiera de todo el conjunto para un periodo determinado.

Ubicación temporal:

Al finalizar un periodo contable.

Objetivos:

Informar y controlar la situación financiera del conjunto de unidades.

Pre-requisitos:

Que las contabilidades de cada unidad hayan cerrado hasta el periodo de consolidación.

Flujo del Proceso:

1. Obtención de los informes contables de cada unidad a consolidar a la fecha.
2. Suma de los saldos de las cuentas en los niveles en común.
3. Elaboración de los informes.

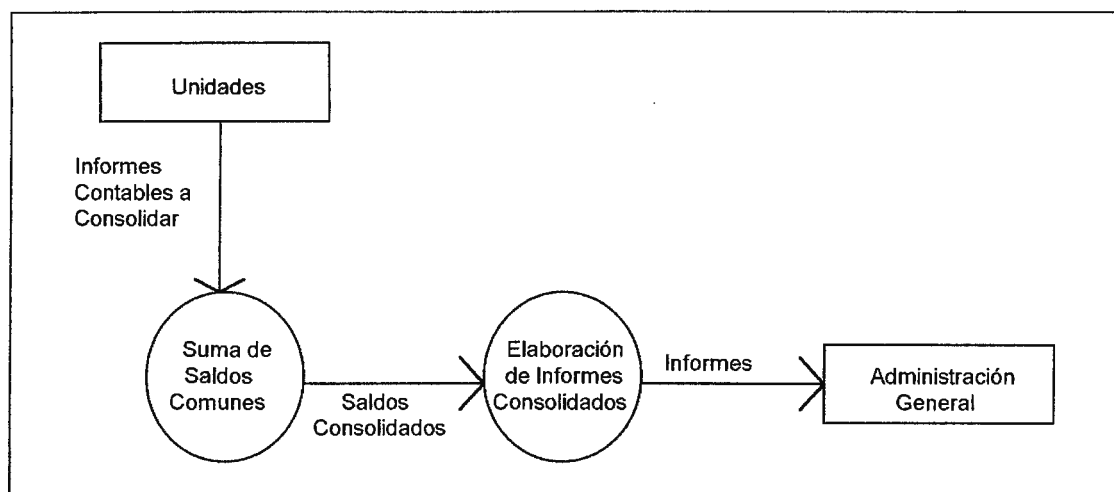


Ilustración 8: Diagrama de Flujo de Datos del Proceso de Consolidación de Contabilidades.

Limitaciones del Proceso Administrativo Contable.

Se han encontrado las siguientes limitantes del Sistema durante las entrevistas realizadas:

- Insuficiente cantidad de niveles para la codificación de las Cuentas de detalle necesarias en cada Unidad.
- El sistema no permite realizar Estados Financieros previos al Cierre, o generar ciertos reportes y consultas necesarias utilizando las partidas que aún NO han sido mayorizadas.
- Hay confusión con la asignación del número correlativo TEMPORAL que se asigna a las partidas antes de ser mayorizadas, ya que se asigna según vayan siendo introducidas, sin tener en cuenta el orden de las partidas por FECHA, lo que dificulta ciertas consultas previas a los Cierres.
- Dependencia técnica para ciertas operaciones que deberían ser exclusivas del

departamento contable y de la Administración de la Unidad.

- El Catálogo que rige las Instituciones Salesianas no se adecua a las actividades y necesidades concretas de cada Unidad dentro de la Ciudadela.
- Existe falta de coordinación y eficiente comunicación entre las Unidades para efectos del control contable.
- Se desconoce la existencia de algún manual de operaciones para las actividades contables (a excepción del Manual de Aplicación de Cuentas).
- La carencia de una actualización en línea de los controles auxiliares a la Contabilidad provoca muchas veces atrasos al Proceso, lo cual dificulta el reflejar la situación financiera real.
- La falta de una adecuada planeación provoca un continuo reajuste de los controles contables, debido a que los problemas surgen con poco margen de tiempo para darles la debida solución.

Proceso de Asignación y Control Presupuestal

Descripción del proceso del control Presupuestal.

Este proceso tiene un papel preponderante dentro de las estrategias de planeación realizadas por la Administración de las distintas unidades de la Ciudadela. Implica la asignación de recursos financieros para las distintas unidades, a fin de cubrir las necesidades proyectadas por éstas para el nuevo ejercicio contable, así como la proyección de ingresos a percibir en el transcurso de éste. A través del ejercicio se realizan los controles necesarios para verificar el cumplimiento de tales proyecciones.

Delimitación en el espacio.

Parte desde los Centros de Costo en cada unidad hasta la Administración General de la Ciudadela.

Funciones y objetivos.

- Establecer metas concretas a las Unidades a través de la proyección de ingresos.
- Regular los gastos de los Centros de Costo en cada Unidad.
- Controlar el déficit o el superávit de la Institución.
- Proporcionar una visión global de la situación financiera actual de la Institución, comparando ésta con las proyecciones realizadas, a fin de poder tomar decisiones oportunas.

Entidades Involucradas.

Involucra las siguientes Unidades:

- Centro de Formación Profesional.
- Centro de Investigación y Transferencia de Tecnología
- Colegio Don Bosco, Oratorio y Parroquia

- Universidad Don Bosco
- Comunidad Salesiana.
- Economato Salesiano

Y sus respectivos Centros de Costo.

Procesos.

Las operaciones o actividades que conforman el Proceso administrativo de Asignación y Control Presupuestal son:

- Asignación Presupuestal.
- Control Presupuestal.
- Reasignación de Presupuestos.

Asignación Presupuestal.

Definición:

Consiste en asignación de los montos estimados que deberán ingresar y egresar dentro del ejercicio contable y su aprobación.

Ubicación Temporal:

Se da al inicio del ejercicio.

Objetivos:

Crear el marco de referencia al cual deberá ceñirse la unidad para controlar sus gastos e ingresos en el ejercicio contable.

Pre-requisitos:

Estar finalizado el Ejercicio contable inmediatamente previo al ejercicio a presupuestar de manera que se puedan tener informes de como estuvo la situación contable en dicho ejercicio en cada unidad.

Flujo del Proceso:

1. Análisis de saldos de ingresos y gastos del ejercicio previo al de presupuesto si están disponibles.
2. Proyección de gastos e ingresos para cada centro de costos de la unidad.
3. Consolidación de estas proyecciones para elaborar el presupuesto a asignar en cada unidad.
4. Revisión, discusión y corrección de los presupuestos.
5. Autorización de los Presupuestos.
6. Registro de los montos presupuestados para cada actividad si están definidas en el catálogo contable.

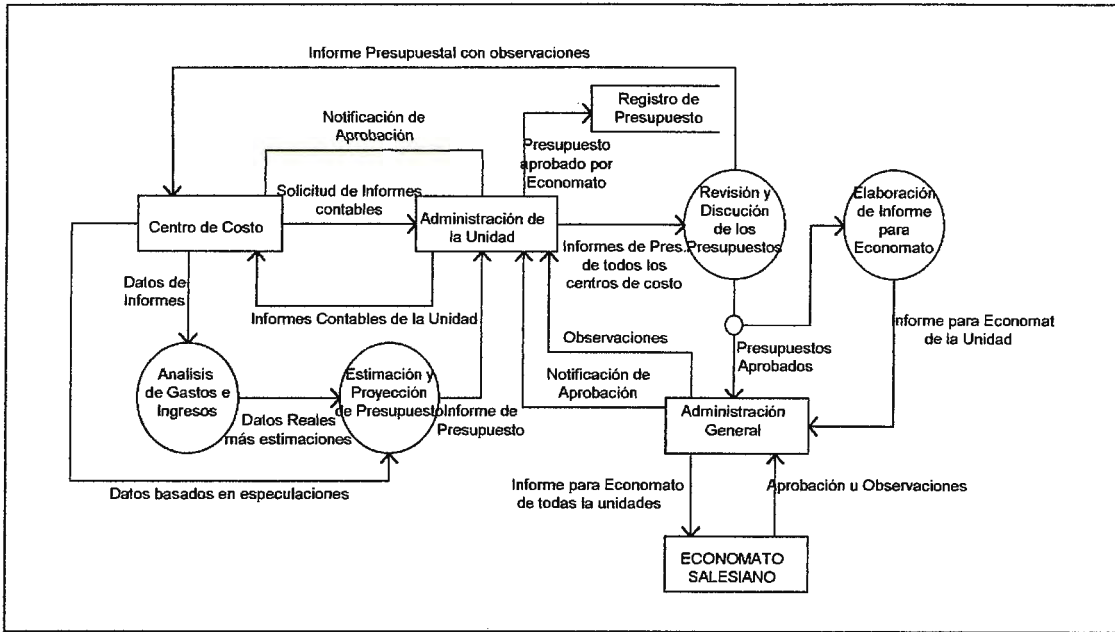


Ilustración 9: Diagrama de Flujo de Datos del Proceso de Asignación de Presupuesto.

Control Presupuestal.

Definición:

Consiste en la comparación de los montos presupuestados contra los saldos reales a una fecha determinada.

Ubicación Temporal:

Durante todo el ejercicio.

Objetivos:

Controlar el comportamiento de la situación financiera de la unidad a medida transcurre el tiempo, para poder tomar acciones correctivas y preventivas.

Pre-requisitos:

Haber asignado y aprobado los montos de presupuesto.

Contar con el registro actualizado de las transacciones financieras hasta la fecha a controlar.

Flujo del Proceso:

1. Generación de informes del estado financiero actual.
2. Comparación entre los montos asignados y los actuales.
3. Crear Informe que refleje el déficit o superávit y su proporción.

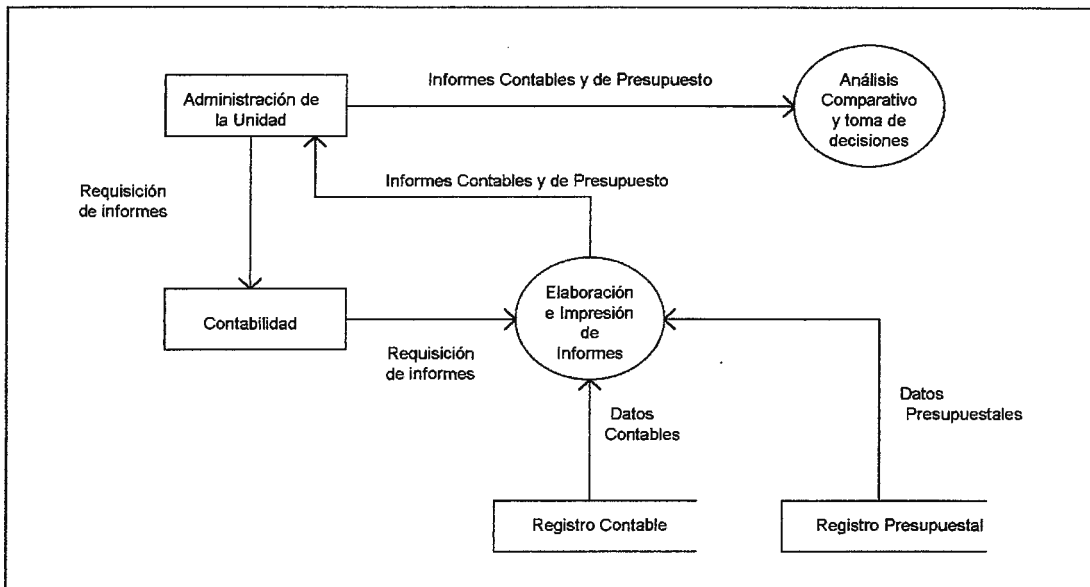


Ilustración 10: Diagrama de Flujo de Datos del Proceso de Control Presupuestal.

Reasignación Presupuestal.

Definición:

Es el proceso de redistribuir los montos asignados de presupuesto a cada unidad debido al cambio constante de las necesidades financieras.

Ubicación Temporal:

Se da arbitrariamente durante el ejercicio dependiendo de las necesidades que surgen.

Objetivos:

Redistribuir el Capital presupuestado entre los centros de costo para solventar una necesidad financiera determinada y/o balancear de manera más equitativa el presupuesto.

Pre-requisitos:

- Haber realizado la asignación Presupuestal.
- Análisis de la situación financiera actual comparado con el presupuesto.

Flujo del Proceso:

1. Análisis de alternativas para solventar necesidad.
2. Distribución del capital presupuestado y su aprobación.

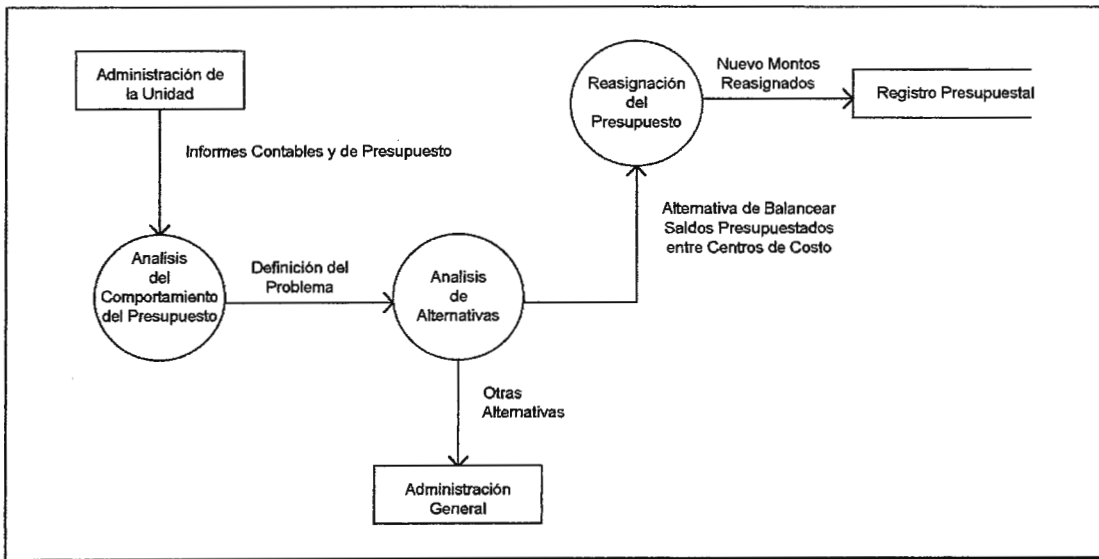


Ilustración 11: Diagrama de Flujo de Datos del Proceso de reasignación Presupuestal.

Limitaciones del Proceso de Asignación y Control Presupuestal.

Se han encontrado las siguientes limitantes del Sistema durante las entrevistas realizadas:

1. No existe suficiente información actualizada acerca de los ingresos y gastos de cada centro de costo, que brinde una base para que éstos puedan realizar proyecciones adecuadas.
2. No existen políticas ni una metodología estándar definidas para la asignación y control del presupuesto.
3. No todas las Unidades llevan un control Presupuestal periódico, ya que existen retrasos en la Contabilidad que no permiten realizarlo.

Descripción del Sistema Contable Informático Actual.

Actualmente la Ciudadela Don Bosco cuenta con un Sistema Contable Informatizado, el cual es utilizado en todas sus Unidades. Cada Unidad posee una copia particular del Sistema en una computadora personal en el cual ingresa y controla su Contabilidad.

El sistema a realizar dará soporte a todas las operaciones, consultas y reportes que el sistema actual realiza, además de otras opciones que serán especificadas por los administradores y contadores de las distintas Unidades. Debido a esto es necesario estudiar detenidamente el programa existente para lograr realizar un programa más completo y eficiente.

Características Técnicas del Sistema Contable Actual.

1. El programa fue realizado utilizando el lenguaje de programación FoxPlus.
2. Funciona en una computadora personal.
3. Sólo puede llevarse una sola Contabilidad por copia del programa.
4. No se trabaja en línea con los sistemas auxiliares. La comunicación entre sistemas se da por medio de discos flexibles y la intervención del técnico.
5. El programa no es ejecutable directamente desde el sistema operativo, sino que se necesita tener en la máquina el núcleo de FoxPlus para correrlo.

Opciones del Sistema.

El Sistema actual cuenta con distintas opciones para realizar los procesos, consultas y generación de reportes necesarios para el control de la Contabilidad, las cuales han sido utilizadas por las distintas Unidades de la Ciudadela hasta ahora.

Las principales opciones con que cuenta el Sistema son:

1. Libro Mayor.

2. Menú de Partidas.
3. Salir del Sistema.
4. Actualización.
5. Reportes Financieros.
6. Utilitarios.

Cada una de estas opciones se detallan en una serie de procesos representados por submenús del programa. A continuación se dará una descripción general del contenido de cada una de ellos:

1. Libro Mayor.

1.1 Mantenimiento del Libro Mayor

Este contiene ciertas opciones para la creación y mantenimiento de la Cuentas pertenecientes al Catálogo de Cuentas de la Institución, controlando datos como los códigos, descripciones de cuentas, tipo de cuenta, saldos actuales y anteriores, y el saldo presupuestado anual.

1.2 Impresión de Saldos del Mayor y Auxiliar

Se genera un informe que contiene el Catálogo de Cuentas completo, y presenta el saldo del mes especificado para cada una de las cuentas.

1.3 Regresar al Menú Anterior

1.4 Impresión Catálogo de Cuentas

Permite tanto consultar el Catálogo de Cuentas en pantalla como imprimirlo completo.

1.5 Inicializar Mes

Este procedimiento consiste básicamente en cerrar el mes en proceso, con lo que los datos del archivo de trabajo pasan a otro archivo definitivo. Una vez hecho esto ninguna otra partida de ese mes puede ser introducida, convirtiéndose el siguiente mes en el nuevo mes en proceso.

1.6 Inicializar Año

Este procedimiento consiste en una preparación de los archivos de trabajo al realizarse el

Cierre del Ejercicio, dejándolos listos para iniciar el nuevo Año.

1.7 Consulta Estados de Cuentas

Muestra los movimientos en que ha intervenido la cuenta especificada de las partidas. Las partidas a tomar en cuenta serán las que se encuentren en un rango de fechas digitado.

1.8 Acumulaciones Mensuales por Cuenta

Esta opción presenta en pantalla los saldos mensuales para una cuenta especificada.

1.9 Consulta de Saldos del Mayor

Muestra los saldos del mes especificado para las cuentas de mayor.

2. Menú de Partidas.

2.1 Consulta de Partidas

Especificando el número de una partida puede visualizarse su contenido. Sólo se incluyen en esta opción las partidas no actualizadas.

2.2 Ingreso de Partidas

Se digitan aquí las partidas tanto del mes en proceso como de los siguientes meses (en caso de atrasos). No pueden digitarse partidas pertenecientes a los meses ya cerrados.

Los datos incluidos en la partida son: Número de Partida, Fecha de la Partida y los Movimientos de ésta. Para los movimientos se digitan el Código de Cuenta, Concepto, Código de Contracuenta, y el Monto, especificando si es un cargo o un abono.

2.3 Regresar al Menú Anterior

2.4 Modificación de Partidas

Permite modificar los datos de las partidas que no hayan sido actualizadas, pertenezcan al mes en proceso o a posteriores.

2.5 Eliminación de Partidas

Permite eliminar las partidas digitadas mientras aún no hayan sido actualizadas.

2.6 Impresión Partidas no Actualizadas

Esta opción envía a impresor todas las partidas no actualizadas para el mes en proceso.

2.7 Consulta de partidas ya Actualizadas

Permite visualizar las partidas que ya hayan sido actualizadas, especificando el número de la partida.

2.8 Menú de Cheques

Este submenú contiene las opciones necesarias para realizar la emisión de cheques, imprimiendo al mismo tiempo el correspondiente voucher, así como también crea automáticamente las partidas necesarias.

Las opciones son las siguientes:

2.8.1 Emisión de Cheques

Utiliza formas continuas para imprimir tanto el cheque como el correspondiente voucher. Crea al mismo tiempo la partida para el cheque.

2.8.2 Emisión de Cheques con formato

Sirve para imprimir los vouchers (y crear la partida) cuando no se utiliza la forma continua para los cheques, sino cheques comunes y corrientes.

2.8.3 Regresar al Menú Anterior

2.8.4 Hacer copias de Partidas

Permite hacer una copia de la partida especificada. Prácticamente no es utilizada actualmente.

2.8.5 Estados de Cuentas por actualizar

Funciona de igual forma que la opción 1.7 del Sistema, mostrando los movimientos de las partidas (no actualizadas) especificadas en que se utilice una cuenta determinada.

2.8.6 Actualizar cuentas en partidas

Permite modificar automáticamente en todas las partidas no actualizadas un código de cuenta, especificándosele el código de la cuenta actual y el código de la cuenta con la cual quiere

reemplazar al primero en todos los movimientos que la utilicen.

2.9 Menú de Conciliaciones

Contiene varias de las opciones anteriormente mencionadas, pero trabajan sólo con las partidas no actualizadas que no pertenecen al mes en proceso.

Contiene las siguientes opciones:

2.9.1 Consulta de Partidas

2.9.2 Impresión de Partidas

2.9.3 Regresar al Menú Anterior

2.9.4 Modificación de Partidas

2.9.5 Consulta Estados de Cuentas

2.9.6 Impresión Estados de Cuentas

2.9.7 Transferir partidas del mes

Sirve para actualizar el correlativo de todas aquellas partidas que pertenecen al mes en proceso, pero que fueron digitadas o introducidas cuando la contabilidad aún no se llevaba actualizada hasta este mes. Este proceso no actualiza dichas partidas.

2.9.8 Cargar Cheques de Planilla

Archiva en el Sistema Contable las partidas generadas automáticamente por el Sistema de Planillas. La comunicación entre los sistemas se realiza por medio de disquetes.

2.9.9 Cargar Partidas de Colecturía

Archiva en el Sistema Contable las partidas generadas automáticamente por el Sistema de Colecturía. La comunicación entre los sistemas se realiza por medio de disquetes.

3 Salir del Sistema.

4 Actualización.

Al seleccionar esta opción son mayorizadas o actualizadas todas las partidas (no actualizadas) que hayan sido introducidas para el mes en proceso. Se afectan tanto los saldos de las cuentas de detalle como las de sus correspondientes cuentas padre. Incluye partidas digitadas por el operador o generadas por otros sistemas.

5 Reportes Financieros.

El Sistema Contable genera una serie de Reportes y Estados Financieros para su impresión. Para algunos de éstos puede especificarse el nivel de detalle con el que se desean. No es posible visualizar estos reportes previamente en la pantalla.

5.1 Balance de Comprobación

- 5.1.1 Con todas las cuentas
- 5.1.2 Sólo cuentas con saldos

5.2 Balance General

5.3 Regresar al Menú Anterior

5.4 Estados de Cuentas

- 5.4.1 Con detalle
- 5.4.2 Sin detalle

5.5 Cuadro Comparativo de Gastos

5.6 Estado de Pérdidas y Ganancias

- 5.6.1 Con todas las cuentas
- 5.6.2 Sólo cuentas con saldos

5.7 Impresión Partidas Actualizadas

5.8 Presupuesto de cuentas

5.9 Saldos del Presupuesto

6 Utilitario

Consiste en una serie de opciones para el mantenimiento general del Sistema contable.

6.1 Mantenimiento Fechas

Define el rango de fechas para los períodos contables. Para el caso de la Ciudadela los períodos equivalen a los meses del Año.

6.2 Mantenimiento Nombres Casas Salesianas

Contiene ciertos datos para la identificación de la Unidad, como son los nombres de la Unidad, del Administrador de la Unidad, del Contador, del Auditor, y otros. Estos datos son utilizados más que todo en la emisión de reportes.

6.3 Regresar al Menú Anterior

6.4 Fecha inicio período contable

Se utiliza para indicar el día o fecha en que da inicio el Ejercicio Contable. Para el caso de la Ciudadela es el 1 de Enero.

6.5 Mantenimiento clave de Ingreso al Sistema

Mantenimiento para las claves de ingresos de los usuarios del Sistema Contable.

6.6 Backup de Archivos de Datos

Realiza una copia de seguridad de los datos del Sistema a la fecha.

6.7 Recuperar Archivos de Datos

Permite recuperar los datos del Sistema de alguna copia previamente realizada.

6.8 Reindexar las bases de datos

Actualiza los archivos de índice de los datos del sistema, si por alguna razón no se hubiesen actualizado durante la ejecución normal del programa.

6.9 Actualizar correlativo de partida

Sirve para definir un número inicial para el correlativo de las partidas a introducir, si por alguna circunstancia se haya perdido el correlativo en el programa.

Limitantes Técnicas del Sistema Actual.

Se han determinado las siguientes limitaciones del Sistema actual, debidas a factores técnicos:

- El sistema sólo puede trabajar con una Contabilidad por copia del programa.
- Debido a que el sistema se está utilizando en Computadoras Personales, la comunicación entre la contabilidad y sus Sistemas Auxiliares (Planillas, Colecturía, etc.) debe realizarse por medio de disquetes, lo que limita en gran medida la funcionalidad del sistema.
- La plataforma en que está desarrollado el Sistema actual no permite establecer un diseño de datos complejo, como para controlar múltiples contabilidades relacionadas, que responda a las necesidades de la Institución y sea eficiente.
- La plataforma de desarrollo no dispone de la capacidad de brindar protección a los datos. Los archivos de datos de FoxPlus pueden ser accedidos y manipulados fácilmente fuera del Sistema de Contabilidad.
- Existe demasiada dependencia técnica de la persona que desarrolló el programa para realizar tareas que debieran haber sido automatizadas.

Limitantes Funcionales del Sistema Actual.

Se han determinado las siguientes limitantes del Sistema durante las entrevistas realizadas:

- Insuficiente cantidad de niveles para la codificación de las Cuentas de detalle necesarias en cada Unidad.
- El sistema no permite realizar Estados Financieros previos al Cierre, o generar ciertos reportes y consultas necesarias utilizando las partidas que aún NO han sido mayorizadas.
- Hay confusión con la asignación del número correlativo TEMPORAL que se asigna a las partidas antes de ser mayorizadas, ya que se asigna según vayan siendo introducidas, sin tener en cuenta el orden de las partidas por FECHA, lo que dificulta ciertas consultas previas a los Cierres.
- El Catálogo de Cuentas (de la UDB) en el Sistema no corresponde al Catálogo original de la Institución Salesiana, ya que se cambió la forma de codificación de las Cuentas al INSERTARSE el Código del Centro de Costo en el segundo nivel del Código de la Cuenta.
- Esto hace que ciertos requerimientos de información basados en el CONCEPTO de ciertos grupos de cuentas sean más difíciles de obtener, ya que el Catálogo está codificado respecto a Centros de Costo y no respecto al CONCEPTO CONTABLE en ciertas cuentas.
- La Generación de algunos reportes en el Sistema toma en cuenta la estructuración actual de las cuentas, sin embargo, al agregar o quitar cuentas que hoy figuran en el Catálogo, estas no se verán reflejadas en los reportes sin previa actualización de los programas.
- No ha habido una capacitación formal en el uso del sistema ni en la aplicación práctica de sus capacidades, incluyendo también ciertos procesos de Cierre del Ejercicio, a pesar que están contemplados dentro del Manual de Aplicación de Cuentas del Economato

Salesiano.

- No se cuenta con manuales actualizados con todos los cambios y adiciones realizados al Sistema, tanto para su uso como para referencia técnica.
- Debido a que el Sistema está diseñado para llevar control de las operaciones contables durante un solo ejercicio, es necesaria la intervención técnica cuando se inicia un nuevo ejercicio.

Propuesta y Solución

Introducción

El contenido del presente capítulo consiste en la presentación del Diseño de Datos y de la Aplicación del Sistema de Contabilidad y Control Presupuestal para la Ciudadela Don Bosco (COP), la cual constituye nuestra propuesta de solución a la situación planteada en el primer capítulo de este documento.

Se incluyen dentro del mismo una descripción general del Sistema propuesto, mostrando la estructura completa de opciones que seguirá el Programa, y especificando tanto las Ventajas del Sistema propuesto sobre la Aplicación actual, como las Limitaciones que éste posee.

Posteriormente se explican asimismo los Diseños y Diagramas de los distintos esquemas de Datos, procesos y verificaciones de integridad de la información implementados por el Servidor de la Base de Datos, así como los procesos, formatos estándar de Pantallas (Ventanas) de Entrada y Salida de datos, y los reportes principales utilizados en la Aplicación Cliente.

Se explicará detalladamente la forma general de uso u operación de los distintos tipos de ventanas utilizadas en el Sistema, para así tener una idea global de como funcionarán los diferentes módulos de entradas de datos, consultas y emisión de reportes que el Programa contendrá.

DISEÑO DEL SISTEMA PROPUESTO

Descripción General del Sistema propuesto.

- Será un sistema que permita llevar la Contabilidad General para “ n ” empresas con una sola copia del programa.
- Cada Empresa o Contabilidad definida en el sistema tendrá la opción de llevar paralelamente un control presupuestal, a través de la definición de un Catálogo de Rubros Presupuestales.
- El Control Presupuestal de una Empresa se realizará a través de la relación entre cuentas contables y rubros presupuestales, tomando en cuenta de que cada rubro puede controlar una o más cuentas, pero no viceversa.
- Permitirá establecer n niveles de relaciones o jerarquías entre distintas Contabilidades, permitiendo a varias contabilidades 'hijas' actualizar o consolidarse en una contabilidad “padre”.
- El Sistema dará la libertad a cada empresa (Unidad en el caso de la Ciudadela) de llevar un Catálogo de Cuentas propio, con la estructura y codificación de cuentas que se desee, pero con la capacidad de consolidarse en un Catálogo "Base" común.
- Para el caso de la Ciudadela, esto quiere decir que cada Unidad (CITT, UDB, Colegio Don Bosco, CFP, etc.) pudiera llevar su propio y único Catálogo con las cuentas, codificación y estructura que cada una necesite, y consolidándose todas ellas en una sola “Contabilidad Padre”, para el caso representada por el Catálogo de Cuentas de la Institución Salesiana.
- Cada Empresa podrá definir un catálogo de rubros presupuestales independiente de las otras Empresas.
- El Sistema permitirá la generación de reportes y consultas previos a los cierres contables, para poder tener en cualquier momento una visión preliminar de la situación financiera de la Unidad, sin necesidad de haber mayorizado todas las partidas.
- El Sistema dará soporte a todas las operaciones, consultas y reportes que el sistema actual realiza, además de otras opciones que fueron solicitadas y definidas oportunamente por los futuros usuarios del sistema.

- El Sistema tendrá la capacidad de trabajar en línea con otros sistemas a desarrollarse, alimentándose automáticamente de las partidas generadas por las operaciones realizadas por éstos.
- El Sistema posee dos niveles de seguridad, el primero proporcionado por el Manejador de la Base de Datos, y el segundo por la integración a un Sistema central, llamado “Repositorio de Aplicaciones y Entidades Institucionales”.
- El Sistema será una aplicación cliente - servidor, las bases de datos estarán localizadas en un servidor central manejadas por un Software administrador de Bases de Datos Relacionales (RDBMS), y la aplicación estará localizada en un Computador PC cliente, la cual funcionará como una interfaz entre el usuario y la Base de Datos. La comunicación entre el Servidor de Bases de Datos y la Aplicación Cliente se hará a través de una red de área local o “LAN”, utilizando TCP/IP como protocolo de comunicación.

Consideraciones Generales para el Modulo de Contabilidad General

Catálogos de Cuentas

- Se establecerá un Catálogo de Cuentas por cada Empresa (Unidad).
- Un catálogo puede tener hasta “n” catálogos “hijos”, pero sólo puede tener un único catálogo padre (estructura de árbol).
- Una contabilidad “padre” no puede realizar su cierre contable hasta que hayan cerrado todos sus “hijos”.
- No puede eliminarse un Catálogo si éste aún tiene catálogos hijos.

Cuentas Contables

- Actualmente el Código de Cuenta de Ingresos utiliza cinco niveles, los cuales en su orden indican: la Cuenta de Mayor, el Centro de Costo, la Actividad, y los dos niveles de Detalle.
- El saldo de una cuenta de detalle (sin disgregaciones) será acumulado automáticamente en su cuenta padre (y superiores) dentro del Catálogo al mayorizarse las partidas que la utilicen.
- En el caso de Catálogos hijos, una cuenta de detalle podrá también actualizar otra cuenta de detalle en el Catálogo padre.
- Sólo se podrán modificar, agregar o subdividir cuentas cuando su saldo sea CERO.
- No podrán borrarse cuentas contables de ningún catálogo, si éstas tienen movimientos, sus saldos no son distintos a cero, o si existen otras cuentas que se acumulen en ellas, ya sean del mismo o de otro catálogo.

Ingreso de Partidas

- Las partidas serán ingresadas al sistema tanto manualmente, como por su generación en

otros sistemas relacionados.

- Se podrá modificar o eliminar una partida mientras no haya sido mayorizada, o sea, mientras no haya actualizado saldos.
- No podrán introducirse partidas contables en que no cuadre el debe con el haber.
- Sólo se podrán efectuar movimientos con cuentas de último nivel.
- Sólo podrá "anularse" una partida mayorizada mediante una contrapartida.

Reportes Contables

- Todos los reportes podrán ser impresos en papel, o únicamente visualizados en pantalla.
- El Sistema permite generar los siguientes reportes o Estados Financieros principales utilizando ya sea las Partidas Mayorizadas y/o las No Mayorizadas:
 - Libro Diario
 - Libro Mayor
 - Balance de Comprobación
 - Balance General
 - Estado de Resultados
- Sólo podrán ejecutarse los Reportes Comparativos para los Períodos Contables que ya hayan sido Cerrados. Aunque un Período no cerrado tenga Partidas Mayorizadas, el Sistema no permitirá utilizarlo para dichos reportes.

Ventajas y Limitaciones del Sistema

Muchas de las Ventajas y Limitaciones del Sistema desarrollado fueron planteadas desde el Proyecto original de la Tesis, sin embargo, debido a los constantes cambios en los procesos administrativos de la Ciudadela durante el desarrollo de la aplicación, surgieron nuevas limitaciones que estaban fuera del contenido de la Propuesta planteada originalmente.

Ventajas del Sistema

- Cuenta con una interfaz gráfica amigable al usuario. Puede ser ejecutado en los ambientes Windows 3.xx y Windows 95.
- Es una aplicación diseñada para ser utilizada simultáneamente por múltiples usuarios en una red.
- Posee dos esquemas de seguridad, el primero en el ámbito de la Base de Datos y el segundo en el ámbito de Aplicación Cliente.
- Está integrado con un Sistema central de Repositorio, con la posibilidad de integrarse con otras Aplicaciones Auxiliares de la Contabilidad.
- Posee un estándar en el diseño de los distintos módulos de manipulación de datos, lo que facilita su aprendizaje y uso.
- Posee verificaciones para mantener la integridad de los datos tanto a nivel del Manejador de la Base de Datos como de la Aplicación Cliente.
- Genera poco tráfico de red, debido a que es una Aplicación Cliente-Servidor.

Limitaciones del Sistema

- Sólo es posible ejecutarlo utilizando un ambiente de red, es decir, que al menos el Servidor de Base de Datos y el Cliente se encuentren interconectados por algún medio de comunicación, y utilizando el mismo protocolo de red.

- Debido a la Herramienta empleada en el Desarrollo del Sistema, existen limitantes en la cantidad de datos que pueden ser manipulados en la aplicación cliente, ya que la mayoría de controles VBX pueden utilizar hasta un límite de 64 Kilobytes de memoria.
- Es necesario tener instaladas en el Cliente soporte para dos tipos de comunicación con la base de datos, la primera a través de las “DB-LIBS” de Sybase, y la segunda a través del estándar ODBC de 16 bits. Esto se debe a que los reportes están diseñados en la herramienta “Crystal Reports”.
- Debido a que el proceso de mayorización se realiza en la base de datos haciendo uso de un Trigger de Actualización, el cual se ejecuta recursivamente, no es posible utilizar más de 16 niveles en la codificación de las Cuentas Contables.
- A diferencia de los Catálogos Contables, no es posible consolidar Presupuestos entre varias Empresas.
- A diferencia de los Informes Financieros Contables no es posible generar reportes comparativos de Presupuesto utilizando datos preliminares (de partidas no mayorizadas).

Requerimientos Técnicos para el Desarrollo y Funcionamiento de la Aplicación

La Base de Datos

El Software Manejador de la Base de Datos Relacional (RDBMS) a utilizar será Sybase SQL Server System X, corriendo en un Computador IBM RISC/6000 con sistema Operativo AIX versión 3.2.5 (UNIX de IBM). El Manejador de Base de Datos correrá como un proceso del Sistema Operativo, y en modalidad Cliente - Servidor, es decir, esperando requisiciones de clientes a través de la red y enviándoles respuestas. En específico, el Proceso del Manejador de la Base de Datos, estará “escuchando” peticiones a través del puerto lógico 2025 de los servicios del Protocolo TCP/IP.

La Base de Datos es capaz de soportar múltiples usuarios simultáneamente, ya sea realizando múltiples procesos y/o accedendo la misma información, de tal manera que el acceso a la información en forma concurrente será tarea del RDBMS. Por otro lado, el RDBMS, posee mecanismos de seguridad implícitos en las Bases de Datos, de tal manera que es posible definir usuarios y poder definir derechos de acceso a cada uno de ellos.

Los objetos que se podrán definir dentro de las Bases de Datos serán:

- Tablas
- Vistas
- Reglas
- Triggers
- Defaults
- Procedimientos Almacenados
- Usuarios

La Aplicación

La Aplicación será desarrollada en Microsoft Visual Basic 3.0 para Windows, la cual, correrá en computadoras PC compatibles con Sistemas Operativos DOS - Windows 3.1 ó Windows 95. Será desarrollada y operará en un ambiente Gráfico basado en Ventanas, específicamente Windows, lo cual permitirá que su operación sea más fácil y a la vez más intuitiva. Sin embargo, basándonos en observaciones hechas durante el período de Investigación, esto también constituirá un reto, ya que se necesitará cambiar la forma de trabajo de los usuarios, empezando por pasar de ambientes y aplicaciones en modo carácter a modos gráficos. Por otro lado, los elementos que constituirán la aplicación a desarrollar serán:

- Menús Desplegables
- Menús de Aparición Súbita (Pop-Up Menús)
- Ventanas de Diálogo
- Cajas de Texto
- Botones de Comando
- Cajas de Selección
- Botones de Selección exclusiva (radio buttons)
- Listas
- Cajas Combo (Combo Box)

Todos estos elementos de Interfaz gráfica con el usuario son un estándar en los sistemas como Windows, los cuales hacen posibles la interacción entre el Usuario y la Aplicación. Su operación puede hacerse a través de dispositivos de entrada convencionales, como el teclado y el Mouse, sin embargo, debido a su naturaleza gráfica, es más fácil y útil usar este último.

La Red

Con el objeto de transmitir la información entre la Base de Datos y la Aplicación, será necesario la utilización de medios de comunicación entre computadoras, en este caso en específico, hablaremos de una red de área local, o LAN. Existen dos aspectos a considerar en la Red que se necesita implementar: Los Protocolos de Comunicación y los medios físicos de transmisión de datos.

La Ciudadela Don Bosco posee un Campus bastante amplio, y las distancias entre los diferentes puntos que necesitarán usar aplicaciones de este tipo es considerablemente grande, de tal manera que se hace necesario pensar en una red de comunicaciones confiable, rápida en cuanto a velocidades de transmisión y que permita extenderse sobre áreas grandes. Por ello, ya existe un proyecto para la Interconexión del Campus de la Ciudadela Don Bosco, que tentativamente iniciará a finales de 1997, y consistirá en la conexión de las diferentes Redes de área local existentes en cada edificio a través de Fibra Optica, la cual cumple y excede los requerimientos de la red.

En cuanto al aspecto de Software para la red, se utilizará el protocolo TCP/IP, el cual es un estándar ampliamente aceptado en el mundo de la transmisión de datos, actualmente es el protocolo utilizado por Internet, la red mundial más grande de Computadoras. Las razones que conducen a seleccionar dicho protocolo son las siguientes:

- Dado que la Ciudadela Don Bosco ya está conectada a Internet, éste es el protocolo de elección.
- TCP/IP es un protocolo adoptado por los sistemas operativos UNIX como su propio estándar, y dado que el servidor de base de datos trabajará en ese ambiente operativo, se hace obvia su elección.
- TCP/IP es un protocolo abierto, es decir permite la comunicación entre diferentes tipos y marcas de computadoras, así como de Sistemas Operativos.

Esto nos lleva a la necesidad de instalar en cada PC Cliente el software necesario para comunicarse en una Red TCP/IP, por lo cual recomendamos la utilización de Windows95

La Aplicación

La Aplicación será desarrollada en Microsoft Visual Basic 3.0 para Windows, la cual, correrá en computadoras PC compatibles con Sistemas Operativos DOS - Windows 3.1 ó Windows 95. Será desarrollada y operará en un ambiente Gráfico basado en Ventanas, específicamente Windows, lo cual permitirá que su operación sea más fácil y a la vez más intuitiva. Sin embargo, basándonos en observaciones hechas durante el período de Investigación, esto también constituirá un reto, ya que se necesitará cambiar la forma de trabajo de los usuarios, empezando por pasar de ambientes y aplicaciones en modo carácter a modos gráficos. Por otro lado, los elementos que constituirán la aplicación a desarrollar serán:

- Menús Desplegables
- Menús de Aparición Súbita (Pop-Up Menús)
- Ventanas de Diálogo
- Cajas de Texto
- Botones de Comando
- Cajas de Selección
- Botones de Selección exclusiva (radio buttons)
- Listas
- Cajas Combo (Combo Box)

Todos estos elementos de Interfaz gráfica con el usuario son un estándar en los sistemas como Windows, los cuales hacen posibles la interacción entre el Usuario y la Aplicación. Su operación puede hacerse a través de dispositivos de entrada convencionales, como el teclado y el Mouse, sin embargo, debido a su naturaleza gráfica, es más fácil y útil usar este último.

como sistema operativo para los PC's que tendrán las aplicaciones, ya que este posee soporte completo y transparente para redes de este tipo.

El Hardware

Servidor de Base de Datos: Actualmente ya se cuenta con el Computador que realizará dicha función, cuyas especificaciones se detallan a continuación:

- Marca: IBM
- Modelo: RISC System 6000 250
- Procesador: PowerPC
- Memoria Física: 32 MB RAM
- Capacidad de Almacenamiento: 2 GB (2048 MB)
- Sistema Operativo: AIX Versión 3.2.5

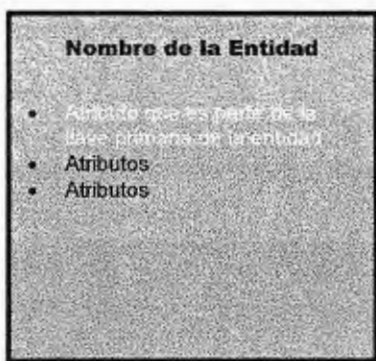
PC's Clientes: La Ciudadela Don Bosco ya posee infraestructura instalada en cuanto a PC's se refiere, sin embargo será necesario expandir las capacidades de sus Computadoras Personales en ciertas áreas. Las especificaciones recomendadas para la óptima operación de las aplicaciones son:

- Computadoras IBM PC Compatibles
- Procesadores: 486 DX2 o mayor
- Memoria RAM: 8 Mb o más
- Monitores y Vídeo: VGA o mayor
- Disco Duro: 120 Mb o más
- Tarjeta de Red.
- Mouse
- Sistema Operativo: DOS con Windows 3.x ó Windows95 (recomendado)

Diseño de la Base de Datos para el Sistema Contable.

En la siguiente página se presenta el Diagrama Entidad - Relación de la Base de Datos del Sistema Contable Propuesto. Nótese que se ha utilizado una simbología y nomenclatura que sean fáciles de entender y no extremadamente técnica, por lo cual no se han seguido ciertos estándares para la diagramación de Entidad - Relación.

Descripción de la Simbología utilizada:



Entidad:

Representación Lógica de una tabla en la Base de Datos, en donde los atributos pueden ser interpretados como campos.



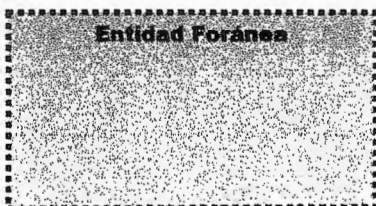
Relación:

Indicador que describe una relación entre 2 entidades. Esta relación servirá para garantizar la integridad referencial entre tablas.



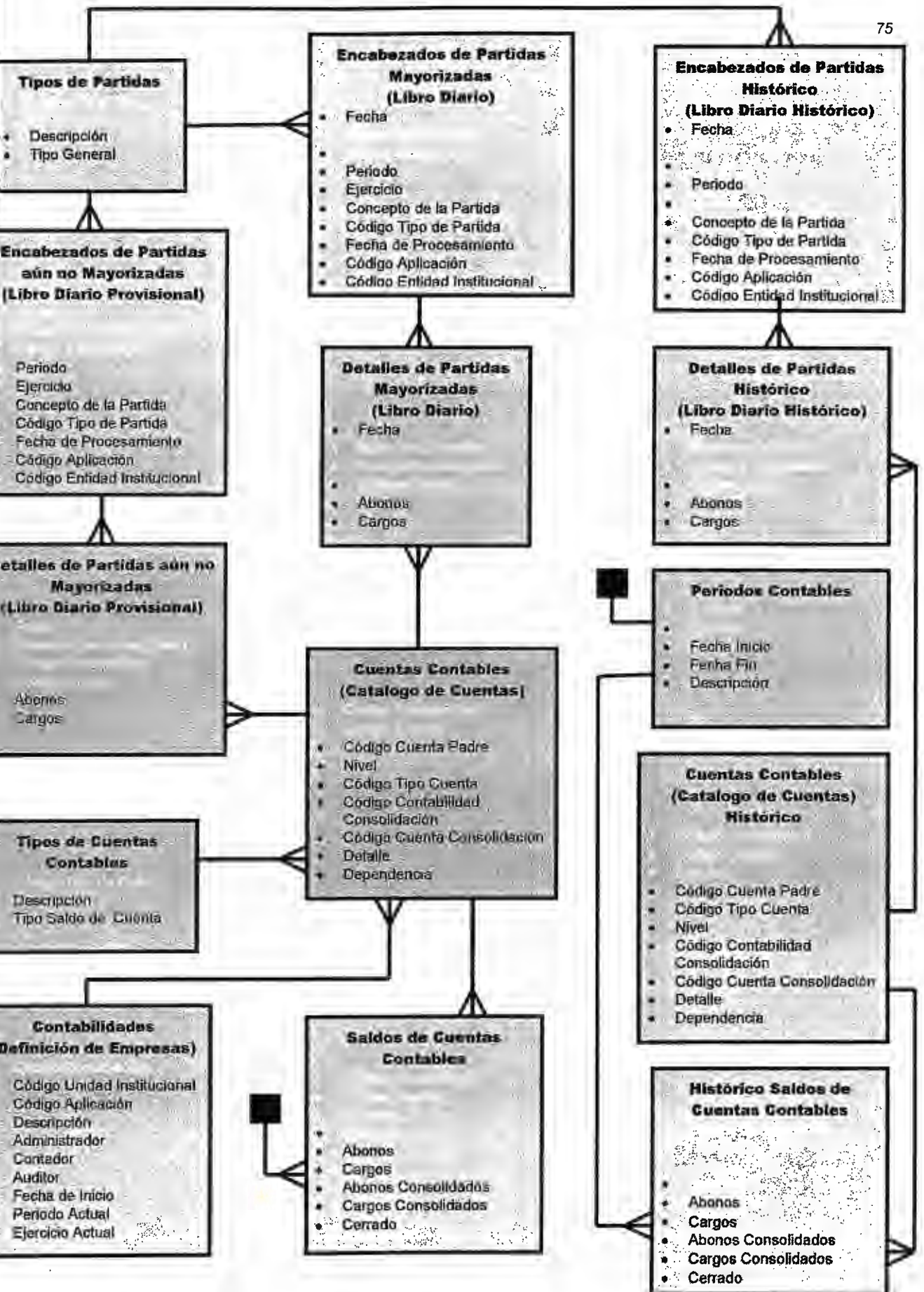
Conector:

Símbolo que se utilizará como ayuda a la distribución del diagrama, e indicará que hay una relación entre ambos conectores



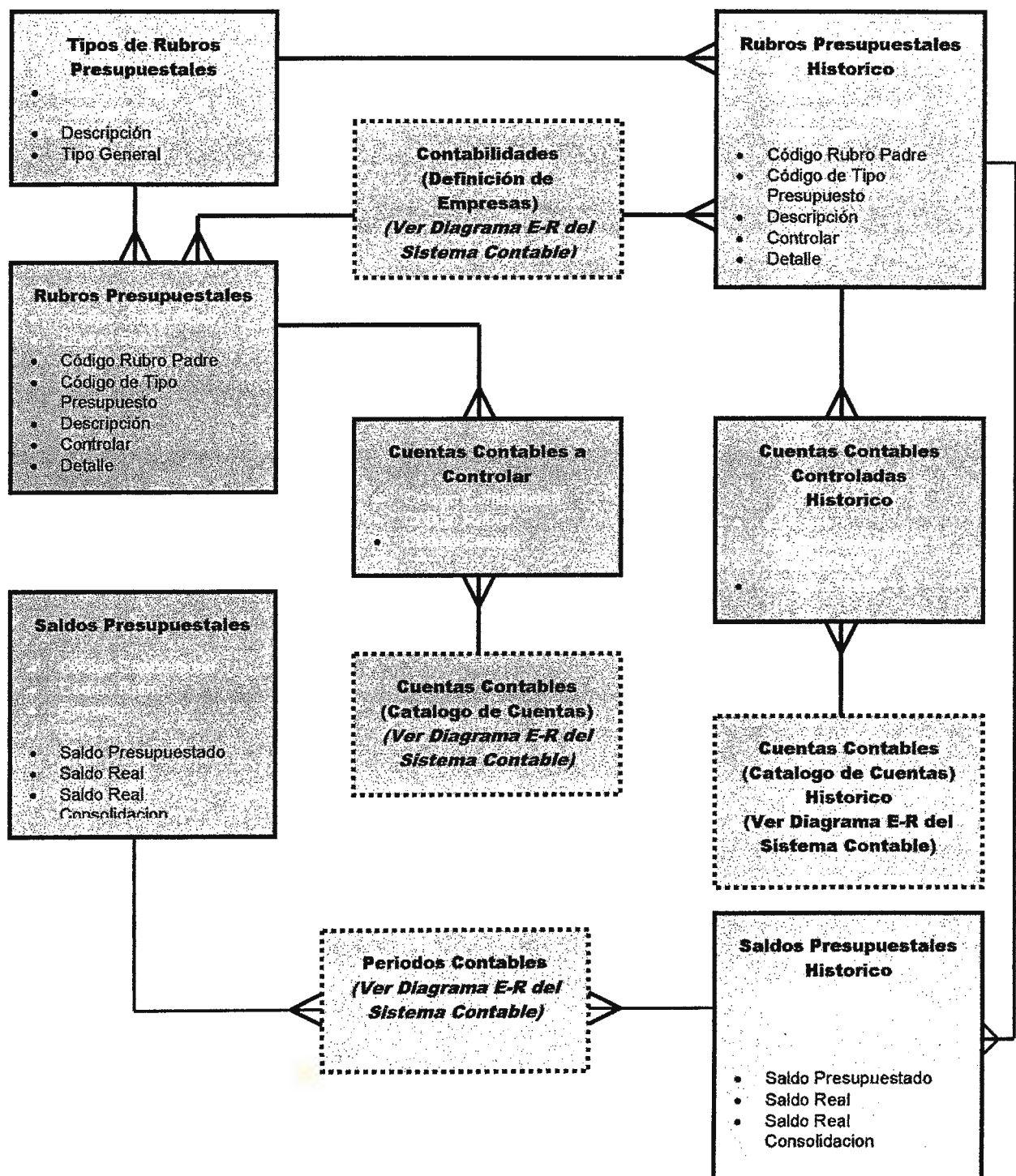
Entidad Foránea:

Se utilizará este símbolo cuando se haga referencia a una entidad que pertenece a otro sistema.



Diseño de la Base de Datos para el Sistema de Control Presupuestal.

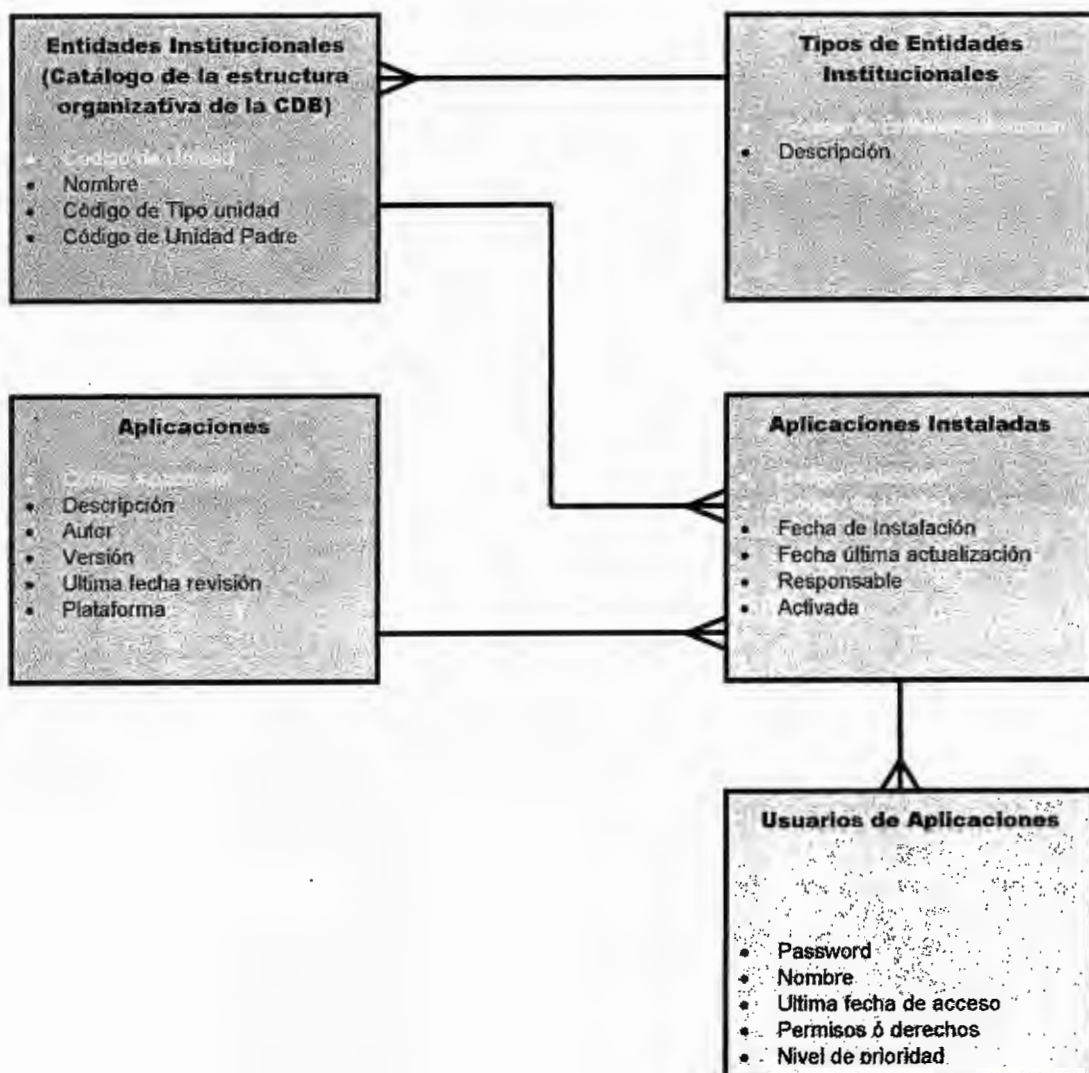
A continuación se presenta el Diagrama Entidad - Relación de la Base de Datos para el Sistema de Control Presupuestal propuesto, siguiendo la misma simbología antes presentada.



Diseño del Repositorio de Aplicaciones y Entidades Institucionales.

A continuación se presenta el Diagrama Entidad - Relación del repositorio de aplicaciones y catálogo de entidades institucionales propuesto, el cual será compartido por todas las demás bases de datos y sistemas administrativos desarrollados bajo los lineamientos del Plan de Desarrollo Informático de la Ciudadela Don Bosco.

El Propósito principal de está base de datos es llevar un registro y control de todas las aplicaciones que sean desarrolladas, así como, un catálogo que permita llevar registro de todas las instituciones, áreas, unidades, departamentos, sub-departamentos, etc. que representan la estructura organizacional de la Ciudadela Don Bosco, de manera que este repositorio, proporcione información que es utilizada por la mayoría de sistemas a desarrollar, incluyendo el nuestro.



Definición de la base de datos.

El diseño de la estructura de datos del Sistema de Contabilidad y Control Presupuestal “COP”, con sus respectivas tablas, Triggers, reglas de validación, procesos almacenados y todos los objetos que interactúan con dicha estructura, constituyen la base de datos del sistema. A continuación se presenta la descripción de la Base de datos utilizada en el sistema para el manejo de la información, indicando todos sus componentes así como también la forma en que se relacionan y el modo de funcionamiento para los procesos y Triggers utilizados.

El Sybase SQL Server al igual que cualquier Manejador de bases de datos relacional moderno, maneja la información en estructuras de datos llamadas tablas, las cuales pueden ser concebidas como cuadros divididos en filas y columnas, donde el programador o diseñador de la base de datos, define que columnas contendrá el cuadro o tabla. Luego, esta tabla crece, aumentando filas al “cuadro”, estas filas son llamadas “registros” o “tuplas”. Cada columna o campo de la tabla es de un “tipo de dato” predeterminado, es decir, sólo puede contener un tipo de información, ya sea, sólo números, datos alfanuméricos, cantidades monetarias, fechas, etc. Cada Manejador de Base de Datos incluye un número finito de tipos de datos que puede administrar. Adicionalmente, cada columna o campo que componen una tabla, puede ser opcionalmente restringida aun más, a través de mecanismos de validación de datos llamados “Reglas”, las cuales son utilizadas para restringir el valor que contiene una columna en un registro específico a una lista o rango de valores predeterminados. Cada tabla puede contener una cantidad indefinida de registros, que sólo es limitada por las capacidades del software y los recursos informáticos disponibles, por esta razón, debe existir (según el modelo relacional) una manera de identificar cada registro en una tabla. Para ello, Sybase implementa lo que se le denomina “Llave Primaria”, definida como el conjunto de campos (uno o más) cuyos valores en cualquier registro, y de manera combinada no se repetirán nunca en la tabla, y por tanto podrán ser tomados como “Llaves” para identificar un registro.

Una base de datos, por lo general contiene más de una tabla, y es la interacción entre ellas la que compone un sistema de información. Para lograr dicha “interacción”, el Manejador de Base de Datos cuenta con un mecanismo de relacionar dos tablas. A este mecanismo se le conoce como relaciones, las cuales sirven para mantener la integridad referencial.

Específicamente, una relación se define a través del campo o conjunto de campos que intervienen entre las dos tablas. A estos campos se les conoce como “Llave Foránea” que no es mas que el conjunto de campos en una tabla dada, que hacen referencia a la llave primaria de otra tabla.

Sybase SQL Server, implementa muchas características que van mas allá del modelo relacional, las cuales, son de gran ventaja para la implementación de aplicaciones complejas. Estas características son los Procedimientos Almacenados y los Triggers, que no son mas que “programas” que realizan una tarea específica, y que afectan de alguna manera la información dentro de la base de datos. Los procedimientos almacenados, son procesos que son ejecutados explícitamente, es decir, tras invocación del usuario, además de que pueden recibir parámetros, que pueden variar el comportamiento del proceso. Por otro lado, Un Trigger es un proceso que no se ejecuta al ser invocado, sino que automáticamente se ejecuta cuando se inserta, borra o edita la tabla a la cual esta relacionado.

Dichos estos conceptos, a continuación se presentan una descripción de cada componente de la base de datos que conforman el sistema COP.

Definición De Tablas

Las tablas que componen la Base de Datos del Sistema COP, constituyen la estructura de datos principal del sistema de información. Cada una de ellas esta identificada por un conjunto de campos que forman su “Llave Primaria”, es decir lo campo que en conjunto identificaran de manera única cualquier registro de datos que contenga. A continuación se define el catálogo de tablas del sistema, donde se presentan el nombre “real” de la tabla en la base de datos, una breve descripción de su propósito, y la lista de campos que forman su llave primaria.

Nombre Tabla	Descripción	Llave Primaria
cop_contabilidad	Archivo de datos generales de las empresas o unidades de las cuales se procesará la contabilidad. Contiene la definición de todas las Contabilidades del sistema COP, así como sus parámetros generales.	cod_contab
cop_cuenta_contab	Archivo del catálogo de cuentas contables para cada unidad o empresa registrada en el sistema. Esta tabla contiene los catálogos de cuentas de todas las contabilidades del sistema para el ejercicio en proceso.	Cod_contab Cod_cuenta
cop_cuenta_contab_h	Archivo de históricos del catálogo de cuentas contables para los ejercicios que ya fueron cerrados. De esta manera se va guardando el catálogo de cada contabilidad a través de los años, ya que los catálogos de cuentas tienden a cambiar a través del tiempo.	Ejercicio cod_contab cod_cuenta
Cop_partenc_1	Archivo maestro de partidas que no han actualizado saldos de cuentas contables, es decir, aquellas que ya fueron introducidas al sistema y que están en espera de ser mayorizadas. Esta tabla sólo contiene los encabezados de esas partidas.	Cod_contab Fecha_partida num_corr_d
cop_partdet_1	Archivo de transacciones contables que no han	cod_contab

	sido mayorizadas y que forman las partidas en la tabla anterior. Por cada registro de la tabla cop_partenc_1, existen al menos 2 registros en esta tabla.	fecha_partida num_corr_d num_corr_movim
cop_partenc_2	Archivo maestro de partidas que actualizaron saldos de cuentas, es decir que ya fueron mayorizadas y por tanto ya no pueden ser modificadas o borradas. Esta tabla sólo contiene los encabezados de esas partidas.	cod_contab num_corr_ejerc
cop_partdet_2	Archivo de transacciones que han sido mayorizadas. Contiene los movimientos que forman las partidas almacenadas en la tabla anterior.	cod_contab num_corr_ejerc num_corr_movim
cop_partenc_h	Histórico del maestro de partidas para los ejercicios terminados. Contiene los encabezados de las partidas que fueron procesadas en ejercicios anteriores, para poder generar informes históricos.	cod_contab num_corr_ejerc ejercicio
cop_partdet_h	Histórico de transacciones contables para los ejercicios terminados. Contiene cada uno de los movimientos que forman las partidas históricas.	cod_contab ejercicio num_corr_ejerc num_corr_movim
cop_período	Catálogo de Períodos contables por cada ejercicio contable o año desde que se inicio a utilizar el sistema COP hasta la fecha.	Ejercicio Período
cop_saldo	Saldos de cuentas contables por periodos en el ejercicio. Contiene el total de cargos y abonos acumulados por cada período contable del ejercicio en procesos por cada cuenta del catálogo.	Cod_contab Cod_cuenta Ejercicio Período
cop_saldo_h	Histórico de saldos para los ejercicios contables terminados. Contiene la misma información de la tabla anterior pero para ejercicios que ya fueron procesados.	cod_contab cod_cuenta ejercicio periodo
cop_t_cuenta	Catálogo de Tipos de cuentas contables para la clasificación de las cuentas en el catálogo.	cod_t_cuenta

cop_t_partidas	Catálogo de Tipos de partidas contables, para la clasificación de partidas.	cod_t_partida
pre_rubro	Catálogo de rubros de presupuesto. Contiene los rubros presupuestales para cada contabilidad definida en el sistema.	cod_contab cod_rubro
pre_rubro_cuenta	Rubros de presupuesto asociados a las cuentas contables que se desean controlar. Contiene las relaciones entre cada cuenta contable y rubro presupuestal para realizar comparaciones.	cod_contab cod_rubro cod_cuenta
pre_rubro_cuenta_h	Histórico de rubros de presupuesto que estaban asociados a cuentas contables	Ejercicio cod_contab cod_rubro cod_cuenta
pre_rubro_h	Histórico del catálogo de rubros de presupuesto. Contiene los rubros presupuestales para cada contabilidad tal y como fueron definidos en ejercicios anteriores.	Ejercicio cod_contab cod_rubro
pre_saldo	Saldos de los rubros en los diferentes periodos. Contiene el saldo real acumulado y el saldo presupuestado para cada rubro de presupuesto en cada período contable del ejercicio en proceso.	cod_contab cod_rubro ejercicio período
pre_saldo_h	Histórico de saldos de presupuesto para los ejercicios terminados. Contiene la misma información de la tabla anterior, pero para los ejercicios que ya fueron procesados.	cod_contab cod_rubro ejercicio período

Definición de Campos o Columnas

Cada tabla del sistema esta compuesta por una o más columnas, cada una de ellas con sus propias características.

Nombre del campo: Indica el nombre por el cual es referenciado el registro dentro de la tabla o entidad.

Tipo de dato: Indica cuál es el tipo de información que almacena el campo dentro de la tabla. El Manejador de base de datos Sybase consta de los siguientes tipos de datos:

CHAR: Sólo puede contener datos alfanuméricos, los cuales por si solos no pueden estar involucrados en cálculos matemáticos.

VARCHAR: Similar al anterior, con la diferencia de que provee un manejo del espacio físico en memoria y disco mejorado.

INT: Contiene números enteros hasta el rango de billones.

TYNYINT: Contiene números enteros hasta el rango de cientos y por tanto ocupa menos recursos.

DATETIME: Contiene valores que representan fechas y horas.

SMALLINT: Contiene números enteros en el rango de miles.

BIT: Contiene 0 o 1, es útil para representar valores de falso/verdadero o sí/no.

MONEY: Contiene valores reales redondeados a 2 decimales.

Longitud: Indica el número de posiciones que contiene cada campo dentro de la tabla, en aquellos casos en los cuales es necesario especificar la cantidad de posiciones decimales a la cual se redondeará el dato, la longitud es especificada en el formato *enteros.decimales*.

Nulo: Indica si el valor puede ser nulo, es decir que no requiere que se especifique un dato en específico.

Llave: Indica el identificador único de la tupla dentro de la tabla asegurando que el campo no contendrá información repetida o redundante.

A continuación se describen los campos que conforman cada tabla del sistema.

Tabla: cop_contabilidad

Nombre del Campo	Tipo de Dato	Longitud	Núm	Ulave	Primaria	Ulave	Foránea	Descripción
cod_contab	Char	5			✓			Indica el código de la contabilidad de la empresa o unidad a controlar
cod_unidad	Char	5				✓		Código de la unidad o empresa para la cual se registra su contabilidad
cod_app	Char	3				✓		Código de la aplicación que se relaciona con el sistema de contabilidad.
descrip_contab	Varchar	60						Descripción de la contabilidad para la unidad o empresa especificada
Administrador	Varchar	50		✓				Nombre del administrador de la unidad
Director	Varchar	50		✓				Nombre del director de la unidad
Contador	Varchar	50		✓				Nombre del contador de la unidad
Auditor	Varchar	50		✓				Nombre del auditor de la unidad
Ejercicio	Int	11				✓		Indica el ejercicio o año fiscal en el cual se trabaja la contabilidad
período	Int	11				✓		Indica el período dentro del ejercicio contable actual.
fecha_ini	Datetime	20						Fecha de inicio del ejercicio contable
cuenta_superavit	Varchar	30				✓		Cuenta de pérdidas y ganancias del ejercicio
nivel_mayor	Tinyint	3						Indica el nivel de mayor en el catálogo contable

Tabla: cop_cuenta_contab

Nombre del Campo	Tipo de Dato	Longitud	Nulo	Descripción		
				Llave Primaria	Llave Foránea	
cod_contab	Char	5		✓	✓	Indica el código de la contabilidad de la empresa o unidad a controlar
cod_cuenta	Varchar	30		✓		Código de la cuenta contable utilizada en el catálogo
cod_cuenta_p	Varchar	30	✓		✓	Indica el código de la cuenta padre o la cuenta de mayor de la cual depende una subcuenta determinada
Nivel	Smallint	10				Representa el nivel jerárquico en la estructura del catálogo para una cuenta determinada
cod_t_cuenta	Char	3			✓	código del tipo de rubro al cual pertenece la cuenta dependiendo de la naturaleza de está
cod_contab_c	Char	5	✓		✓	Indica la contabilidad en la cual puede consolidar una contabilidad determinada
cod_cuenta_c	Varchar	30	✓		✓	Indica la cuenta específica en la cual se consolidará.
descrip_cta	Varchar	60				Representa la descripción de la cuenta en el catálogo
Dependencia	Bit	1				Indica la cuenta de consolidación en otra contabilidad para una cuenta determinada
Detalle	Bit	1				Indica si una cuenta no posee más dependientes

Tabla: cop_cuenta_contab_h

Nombre de Campo	Tipo de Dato	Longitud	Nulo	Llave Primaria	Llave Externa	Descripción
Ejercicio	Int	11		✓		Indica el ejercicio o año fiscal histórico en el cual se trabajo la contabilidad
cod_contab	Char	5		✓	✓	Indica el código de la contabilidad de la empresa o unidad a controlar.
Cod_cuenta	Varchar	30		✓		Código de la cuenta contable Utilizada en el catálogo
Cod_cuenta_p	Varchar	30			✓	Indica el código de la cuenta padre o la cuenta de mayor de la cual depende una subcuenta determinada
Nivel	Smallint	10				Representa el nivel jerárquico en la estructura del catálogo para una cuenta determinada
Cod_t_cuenta	Char	3			✓	Indica el tipo de rubro al cual pertenece la cuenta dependiendo de la naturaleza de esta
Cod_contab_c	Char	5	✓		✓	Indica la contabilidad en la cual puede consolidar una contabilidad determinada
Descrip_cta	Varchar	60				Representa la descripción de la cuenta en el catálogo
Cod_cuenta_c	Varchar	30			✓	Indica la cuenta de consolidación en otra contabilidad para una cuenta determinada
Dependencia	Bit	1				Indica si la cuenta consolida en otra contabilidad
Detalle	Bit	1				Indica si una cuenta no posee más dependientes

Tabla: cop_partenc_1

Nombre de Campo	Tipo de Dato	Longitud	Nulló	Llave	Llave Foránea	Descripción
cod_app	Char	3			✓	Indica el código de la aplicación que origino la partida contable
cod_unidad	Char	5			✓	Código de la unidad o empresa para la cual se registra su contabilidad
cod_contab	Char	5		✓	✓	Indica el código de la contabilidad de la empresa o unidad a controlar
fecha_partida	Datetime	8		✓		Indica la fecha de creación de la partida contable
num_corr_d	Smallint	10		✓		Representa el número correlativo de diario de la partida contable
Ejercicio	int	11			✓	Indica el ejercicio o año fiscal en el cual se trabaja la contabilidad
Periodo	int	11			✓	Indica el período contable en el cual se trabaja la contabilidad
concepto_part	Varchar	120				Concepto o descripción general de la partida contable
cod_t_partida	Char	3			✓	Indica la clasificación de la partida contable dependiendo del concepto o transacción que la genero
fecha_proc	Datetime	20				Representa la fecha de ingreso de la partida al sistema es decir la fecha de procesamiento

Tabla: cop_partdet_1

Nombre del Campo	Tipo de Dato	Longitud	Nulo	Clave	Primaria	Ulave	Foranea	Descripción
cod_contab	Char	5			✓	✓		Indica el código de la contabilidad de la empresa o unidad a controlar
fecha_partida	Datetime	8			✓	✓		Indica la fecha de creación de la partida contable
num_corr_d	Smallint	10			✓	✓		Representa el número correlativo de diario de la partida contable
num_corr_movim	Tinyint	3			✓			Número correlativo de los movimientos o transacciones contables que contiene la partida
cod_cuenta	Varchar	30				✓		Código de la cuenta contable a la cual se aplica el movimiento de la partida
Cargos	Money	14.2						Representa el monto de cargo aplicado a una cuenta
Abonos	Money	14.2						Representa el monto de abono aplicado a la cuenta
Descrip_det	Varchar	100						Concepto o descripción de las transacciones contables

Tabla: cop_partenc_2

Nombre del Campo	Tipo de Dato	Longitud	Nulo	Llave	Primaria	Llave Foranea	Descripción
cod_contab	char	5		✓	✓		Indica el código de la contabilidad de la empresa o unidad a controlar
fecha_partida	datetime	8		✓			Indica la fecha de creación de la partida contable
num_corr_ejerc	smallint	10		✓			Indica el número correlativo de la partida en el ejercicio
Ejercicio	int	11				✓	Indica el ejercicio o año fiscal en el cual se trabaja la contabilidad
Período	int	11				✓	Indica el período contable en el cual se trabaja la contabilidad
Concepto_part	varchar	120					Concepto o descripción general de la partida contable
cod_t_partida	char	3				✓	Indica la clasificación de la partida contable dependiendo del concepto o transacción que la genere
fecha_proc	datetime	20					Representa la fecha de ingreso de la partida al sistema es decir la fecha de procesamiento
cod_app	char	3				✓	Indica el código de la aplicación que origino la partida contable
cod_unidad	char	5				✓	Código de la unidad o empresa para la cual se registra su contabilidad

Tabla: cop_partdet_2

Nombre de Campo	Tipo de Datos	Longitud	Nullable	Primary Key	Foreign Key	Descripción
cod_contab	Char	5		✓	✓	Indica el código de la contabilidad de la empresa o unidad a controlar
num_corr_ejerc	Smallint	10		✓	✓	Indica el número correlativo de la partida en el ejercicio
cod_cuenta	Varchar	30				Código de la cuenta contable a la cual se aplica el movimiento de la partida
num_corr_movim	Tinyint	3		✓		Número correlativo de los movimientos o transacciones contables que contiene la partida
Cargos	Money	14.2				Representa el monto de cargo aplicado a una cuenta
Abonos	Money	14.2				Representa el monto de abono aplicado a la cuenta
descrip_det	Varchar	100				Concepto o descripción de las transacciones contables

Tabla: cop_partenc_h

Nombre de Campo	Tipo de Dato	Longitud	Descripción			
			Null	Primary	Foreign	
cod_contab	Char	5		✓	✓	Indica el código de la contabilidad de la empresa o unidad a controlar
Ejercicio	Int	11		✓	✓	Indica el ejercicio o año fiscal histórico en el cual se trabajo la contabilidad
num_corr_ejerc	Smallint	10		✓		Indica el número correlativo de la partida en el ejercicio histórico
fecha_partida	Datetime	8				Indica la fecha de creación de la partida contable
Periodo	Int	11			✓	Indica el período contable histórico en el cual se trabajo la contabilidad
concepto_part	Varchar	120				Concepto o descripción general de la partida contable histórica
cod_t_partida	Char	3			✓	código del tipo de partida contable dependiendo del concepto o transacción que la genero
fecha_proc	Datetime	20				Representa la fecha de ingreso de la partida al sistema es decir la fecha de procesamiento
cod_app	Char	3			✓	Indica el código de la aplicación que origino la partida contable
cod_unidad	Char	5			✓	Código de la unidad o empresa para la cual se registra su contabilidad

Tabla: cop_partdet_h

Nombre de Campo	Tipo de Dato	Longitud	Nulo	Primaria	Foránea	Descripción
			Leave	Leave		
cod_contab	Char	5		✓	✓	Indica el código de la contabilidad de la empresa o unidad a controlar
Ejercicio	Int	11		✓	✓	Indica el ejercicio o año fiscal histórico en el cual se trabajo la contabilidad
num_corr_ejerc	Smallint	10		✓	✓	Indica el número correlativo de la partida en el ejercicio histórico
num_corr_movim	Tinyint	3		✓		Número correlativo de los movimientos o transacciones contables que contiene la partida
cod_cuenta	Varchar	30			✓	Código de la cuenta contable a la cual se aplico el movimiento de la partida
cargos	Money	14.2				Representa el monto de cargo que se aplico a una cuenta
abonos	Money	14.2				Representa el monto de abono que se aplicado a una cuenta
descrip_det	Varchar	100				Concepto o descripción de las transacciones contables que se efectuaron

Tabla: cop_período

Nombre del Campo	Tipo de Dato	Longitud	Null	Leave	Primaria	Leave	Foránea	Descripción
Ejercicio	Int	11			✓			Indica el ejercicio o año fiscal en el cual se trabaja la contabilidad
Periodo	int	11			✓			Indica el período contable en el cual se trabaja la contabilidad
fecha_ini	Datetime	20						Indica la fecha de inicio del ejercicio contable
fecha_fin	Datetime	20						Indica la fecha de finalización del ejercicio contable o cierre del ejercicio contable
descrip_período	Varchar	50						Indica la descripción del período contable
cerrado	Bit	1						Indica si se realizó el cierre del ejercicio contable

Tabla: cop_saldo_h

Nombre del Campo	Tipo de Dato	Longitud	Nulo Llave	Primaria Llave	Foránea	Descripción
cod_contab	char	5		✓	✓	Indica el código de la contabilidad de la empresa o unidad que se controla en el ejercicio
cod_cuenta	varchar	30		✓	✓	Código de la cuenta contable a la cual se aplica el movimiento de la partida
Ejercicio	int	11		✓	✓	Indica el ejercicio o año fiscal histórico en el cual se trabajó la contabilidad
Periodo	int	11		✓	✓	Indica el período contable histórico en el cual se trabajó la contabilidad
Abonos	money	14.2				Representa el monto de cargo que se aplicó a una cuenta
Cargos	money	14.2				Representa el monto de cargo que se aplicó a una cuenta
Cargos_cons	money	14.2				Indica el total de cargos consolidados en el ejercicio histórico de la cuenta
Abonos_cons	money	14.2				Indica el total de abonos consolidados de la cuenta
Cerrado	bit	1				Indica si el período ya ha sido cerrado.

Tabla: cop_t_cuenta

Nombre del Campo	Tipo de Dato	Longitud	Nulo	Primaria	Foránea	Descripción
			Clave	Clave		
Cod_t_cuenta	Char	3		✓		código del tipo de rubro al cual pertenece la cuenta dependiendo de la naturaleza de esta
Descrip_t_cta	Varchar	40				Indica la descripción del tipo de cuenta
Tipoc	Char	1				Tipo de saldo al que pertenece la cuenta
t_cuenta_gral	Char	1				Tipo genérico: activo, pasivo, capital, egresos o ingresos

Tabla: cop_t_partida

Nombre de Campo	Tipo de Dato	Longitud	Nulo	Clave	Primaria	Externa	Descripción
cod_t_partida	Char	3			✓		código del tipo de partida contable dependiendo del concepto o transacción que la genero
descrip_t_part	Varchar	40					descripción del tipo de partidas contables
partida_cierre	Bit	1					Indica si tipo es partida de cierre

Tabla: pre_rubro

Nombre del Campo	Tipo de Dato	Longitud	Nullable	Primary Key	Foreign Key	Descripción
cod_contab	char	5		✓	✓	Indica el código de la contabilidad de la empresa o unidad de la cual se controla el presupuesto
cod_rubro	varchar	30		✓		código de rubro de presupuesto
cod_rubro_p	varchar	30	✓		✓	código de rubro padre o rubro del cual depende un rubro determinado
cod_t_rubro	char	3			✓	Indica el código del tipo de rubro al cual pertenece un rubro determinado
descrip_rubro	varchar	60				Indica la descripción del rubro
Controlar	bit	1				Indicador de control de rubro
Nivel	smallint	10				Indica el nivel dentro de la jerarquía del catálogo de rubros presupuestales
Detalle	bit	1				Indica si el rubro es de detalle

Tabla: pre_rubro_cuenta

Nombre del Campo	Tipo de Dato	Longitud	Descripción			
			Nulló	Llave	Primaria Llave	Foránea
cod_contab	char	5		✓	✓	Indica el código de la contabilidad de la empresa o unidad de la cual se controla el presupuesto
cod_rubro	varchar	30		✓	✓	código de rubro de presupuesto
cod_cuenta	varchar	30		✓	✓	cuenta contable para la cual se controla su presupuesto

Tabla: pre_rubro_cuenta_h

Nombre del Campo	Tipo de Dato	Longitud	Nullable	Primary Key	Foreign Key	Descripción
cod_contab	char	5		✓	✓	Indica el código de la contabilidad de la empresa o unidad de la cual se controlaba el presupuesto
Ejercicio	int	11		✓		ejercicio contable en el cual se controlaron los rubros de presupuesto
cod_rubro	varchar	30		✓	✓	código de rubro de presupuesto histórico
cod_cuenta	varchar	30		✓	✓	cuenta contable para la cual se controló su presupuesto

Tabla: pre_rubro_h

Nombre de Campo	Tipo de Datos	Longitud	Null	Unique	Primaria	Foreign	Descripción
Ejercicio	Int	11			✓		ejercicio contable en el cual se controló el presupuesto
Periodo	Int	11			✓		período contable en el cual se controló el presupuesto
cod_contab	Char	5			✓		Indica el código de la contabilidad de la empresa o unidad de la cual se controló el presupuesto
cod_rubro	Varchar	30			✓		código de rubro de presupuesto en histórico
cod_rubro_p	Varchar	30	✓			✓	código de rubro padre o rubro del cual depende un rubro determinado
cod_t_rubro	Char	3				✓	Indica el código del tipo de rubro al cual pertenece un rubro determinado
descrip_rubro	Varchar	60					Indica la descripción del rubro
controlar	Bit	1					Indicador de control de rubro
nivel	Smallint	10					Indica el nivel dentro de la jerarquía del catálogo de rubros presupuestales
detalle	Bit	1					Indica si el rubro es de detalle

Tabla: pre_saldo

Nombre de Campo	Tipo de Dato	Longitud	Nulo	Clave	Primaria	Externa	Descripción
cod_contab	char	5		✓	✓		Indica el código de la contabilidad de la empresa o unidad de la cual se controla el presupuesto
cod_rubro	varchar	30		✓	✓		código de rubro de presupuesto
Ejercicio	int	11		✓	✓		ejercicio contable en el cual se controla el presupuesto
Periodo	int	11		✓	✓		período contable en el cual se controla el presupuesto
saldo_presup	money	14.2					Indica el valor del saldo presupuestado para un rubro Determinado
saldo_real	money	14.2					Indica el saldo real del rubro en un período contable
saldo_real_cons	Money						Contiene el saldo acumulado entre todas las contabilidades consolidadas.

Tabla: pre_saldo_h

Nombre de Campo	Tipo de Dato	Longitud	Nulo	Primaria	Foránea	Descripción
cod_contab	Char	5		✓	✓	Indica el código de la contabilidad de la empresa o unidad de la cual se controla el presupuesto
cod_rubro	Varchar	30		✓	✓	código de rubro de presupuesto
Ejercicio	Int	11		✓	✓	ejercicio contable en el cual se controla el presupuesto
Período	Int	11		✓	✓	período contable en el cual se controla el presupuesto
saldo_presup	Money	14.2				Indica el valor del saldo que se presupone para un rubro determinado
saldo_real	Money	14.2				Indica el saldo real del rubro en un período contable histórico
saldo_real_cons	Money					Contiene el saldo acumulado entre todas las contabilidades consolidadas.

Procesos del Sistema

Dentro del Sistema se manejan tres tipos distintos de procesos:

- Procedimientos Almacenados en la Base de Datos
- Triggers de Inserción, Modificación y Eliminación
- Procesos en la Aplicación Cliente

Los Procedimientos Almacenados son conjuntos de instrucciones en lenguaje SQL, los cuales residen en el Servidor y son ejecutados por el Manejador de la Base de Datos (en este caso Sybase SQL Server), y que pueden ser invocados ya sea desde la Aplicación residente en el Cliente o desde otros procesos o Triggers en la Base.

Los Triggers son asimismo procesos (en SQL) residentes en la Base de Datos, con la diferencia que están relacionados directamente a una tabla específica, y son ejecutados automáticamente por el SQL Server cuando se inserta, modifica o eliminan registros de la tabla a la cual se relacionan.

El tercer tipo de procesos son los distintos procedimientos y funciones de los cuales se compone la Aplicación Cliente, los cuales han sido realizados utilizando Visual Basic. Estos procesos consisten principalmente en validaciones de los datos de entrada, búsqueda y presentación de información requerida (tanto a nivel de consultas y reportes), y todo aquello necesario para implementar una robusta pero muy comprensible interfaz para el usuario.

A continuación serán presentados los principales Triggers y Procedimientos Almacenados utilizados en el Sistema. Se dará asimismo una breve descripción de las acciones que éstos ejecutan, a fin de tener una idea general de la implementación de cada uno de ellos, así como su representación utilizando flujogramas.

Diseño de los principales Triggers del Sistema

La principal función de todos los Triggers utilizados en el Sistema es establecer las distintas relaciones entre tablas, a fin de mantener la integridad de la información en las distintas tablas de la Base de Datos. Se verifica en ellos la existencia de los datos pertinentes en las tablas que están relacionadas, a fin de determinar si es posible insertar o eliminar filas de una tabla según el caso.

Además de esta función, hay ciertos Triggers en el sistema que realizan otras operaciones más que únicamente verificar la integridad de los datos, ejecutando ya sea actualizaciones en saldos, creación de registros de saldos en caso que no existan, cambios de estado de Cuentas, consolidación de contabilidades, etc.

Las operaciones y funciones principales de estos Triggers serán explicados a continuación :

Trigger de Inserción de Períodos Contables

1. Valida que el rango de fechas de inicio y final de Ejercicio no esté invertido, es decir, la fecha posterior como fecha de inicio.
2. Verifica que el rango de fechas no se traslape con otro ya definido.

Trigger de Inserción de Cuentas Contables.

1. Verifica que exista la Cuenta Padre de la Cuenta a insertar.
2. Valida también que la Cuenta Padre NO sea la Cuenta de Superávit/Déficit del Ejercicio, ya que esta no puede descomponerse en subcuentas.
3. Verifica que la Cuenta Padre posea un Saldo igual a Cero, o sea, que tenga un mismo valor de Cargos y Abonos realizados.
4. Verifica también que la Cuenta Padre realmente posea un nivel igual al nivel de la nueva Cuenta menos 1 (-1), y que sea del mismo tipo que el de la nueva cuenta.

- Si puede insertar la nueva cuenta, entonces quita la marca de "Cuenta de Detalle" de la Cuenta Padre.

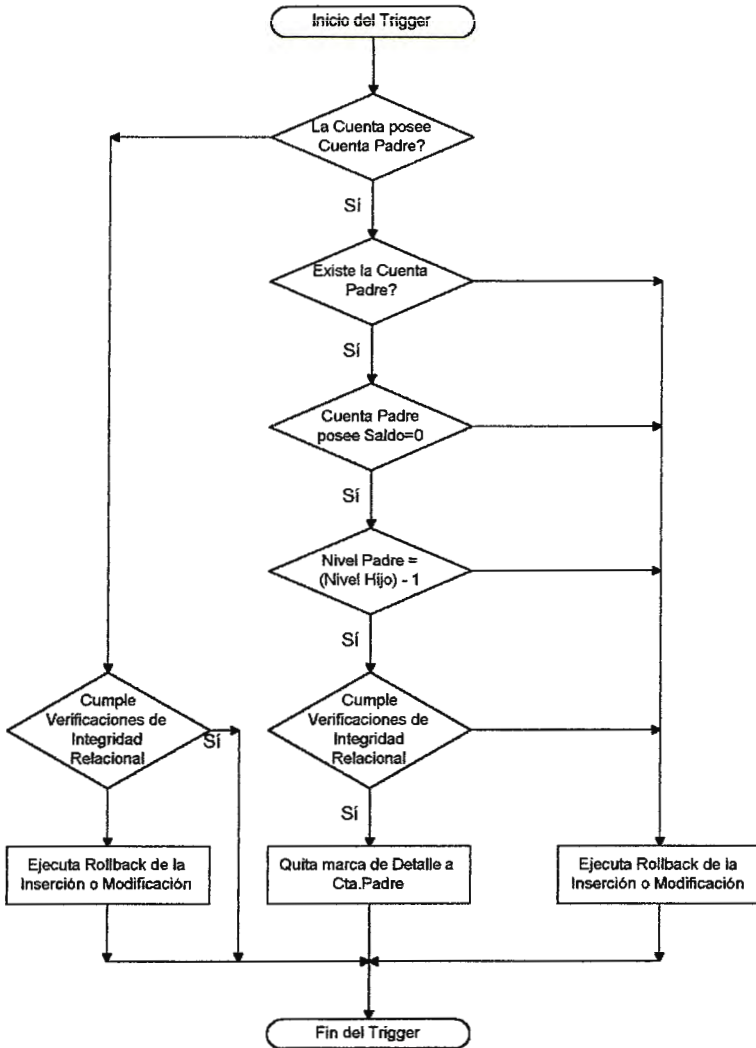


Ilustración 12: Diagrama de Flujo del Trigger de Inserción de Cuentas Contables.

Trigger de Borrado de Cuentas Contables

- Verifica que la Cuenta a Borrar no se esté utilizando en las Partidas. Si lo está, entonces no permite borrarla.
- Verifica si la Cuenta Padre de la cuenta recién borrada tiene más cuentas hijas. Si no tiene más, entonces vuelve a marcar la Cuenta Padre como "de Detalle".

Trigger de Inserción de Detalle de Partidas Mayorizadas

1. Se verifica que la partida corresponda al Período Actual para poder ser mayorizada.
2. Verifica que en la tabla de Saldos existan registros para las Cuentas de Detalle utilizadas para el Período Actual. Si no existen, entonces se crean, con lo cual se activa su Trigger de inserción.
3. Actualiza los saldos de las Cuentas de Detalle correspondientes al detalle de la partida que se está mayorizando.

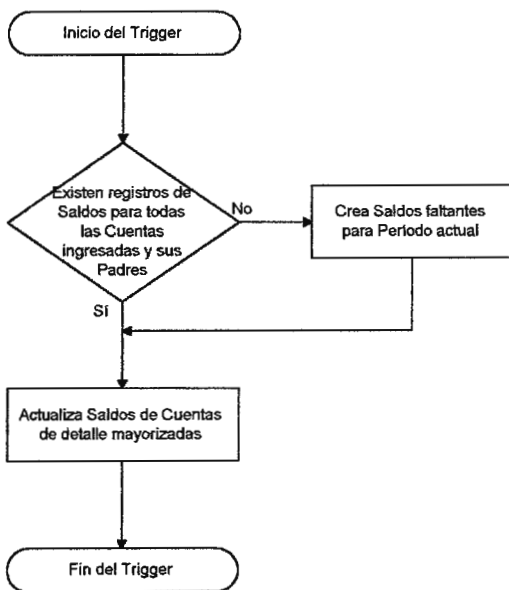


Ilustración 13: Diagrama de Flujo del Trigger de Inserción de Detalles de Partidas Mayorizadas.

Trigger de Inserción de Saldos Contables

1. Se crea un Cursor con la estructura de los Saldos Contables.
2. Se toman los montos de los Saldos de Detalle insertados y se almacenan en el cursor
3. Se inicia un lazo que recorrerá una a una todas los Saldos en el Cursor
4. Se toma el valor del Saldo de una Cuenta, y se determina cual es su Cuenta Padre inmediata.

5. Se actualiza la Cuenta Padre con el nuevo valor. Para cada Cuenta Padre actualizada se determinará también cual es su Cuenta Padre inmediata, para así ir actualizando el Catálogo hasta llegar al primer Nivel.
6. Si existe una Cuenta relacionada a la Cuenta en el Cursor, ésta se actualiza también, así como sus correspondientes Cuentas Padre. Con esto se logra la Consolidación de Contabilidades.
7. Se actualiza asimismo los Saldos Reales de los Rubros Presupuestales que tengan asociada la Cuenta utilizada. Los correspondientes Rubros Padres también son actualizados.
8. Para cada una de los Saldos insertados se seguirá el mismo proceso de Actualización.

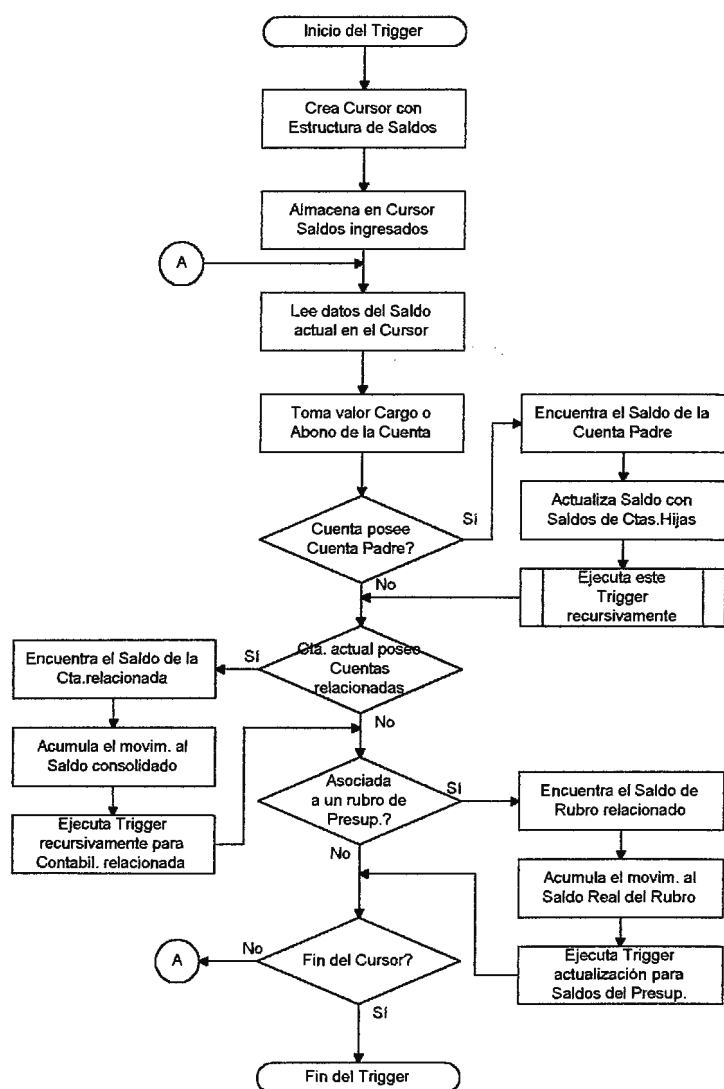


Ilustración 14: Diagrama de Flujo del Trigger de inserción y actualización de saldos contables.

Trigger de Inserción de Rubros de Presupuesto.

1. Verifica que exista el Rubro Padre del Rubro a insertar.
2. Verifica que el Rubro Padre posea un monto presupuestado igual a 0 o que no este aun asignado.
3. Verifica también que el Rubro Padre realmente posea un nivel igual al nivel del nuevo Rubro menos 1 (-1), y que sea del mismo tipo que el del nuevo rubro.
4. Si puede insertar el nuevo rubro, entonces quita el atributo de “Rubro de Detalle” del Rubro Padre.

Trigger de Inserción de Saldos de Presupuesto

1. Se crea un Cursor con la estructura de la tabla de Saldos de Presupuesto.
2. Se empiezan a actualizar recursivamente los saldos de todos los rubros Padre, tanto los saldos reales (locales y consolidados) como los presupuestados.

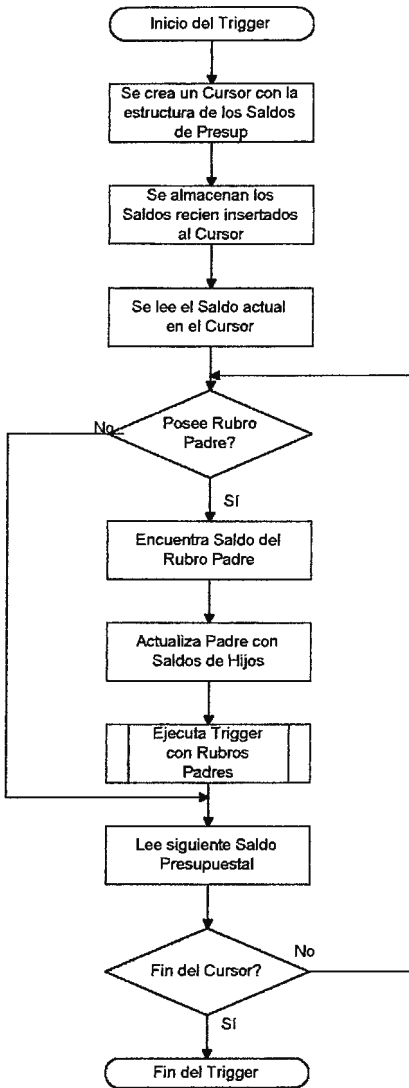


Ilustración 15: Diagrama de Flujo del Trigger de Inserción y actualización de los saldos de presupuesto.

Diseño de los Procedimientos Almacenados

El Sistema cuenta con varios Procesos Almacenados, con los cuales se realizan varias de las operaciones más complejas del Programa.

Los más importantes de éstos son:

- Proceso de Mayorización de Partidas Contables
- Proceso de Cierre Mensual
- Proceso de Cierre Anual
- Procesos de Creación de Reportes (Estados Financieros, Libro Mayor, etc.)

A continuación se explicará el esquema de operaciones básica que tales procesos realizan:

Proceso de Mayorización de Partidas Contables :

Este proceso trabaja en conjunto con el Trigger de Inserción de la tabla de Detalle de Partidas Mayorizadas y el de Saldo Contables, y es el encargado de realizar el traslado de las Partidas No Mayorizadas a las tablas de Partidas Mayorizadas. Al mayorizar se actualizan las correspondientes Cuentas Padres de las Cuentas utilizadas en el detalle de una Partida (la actualización es realizada por el Trigger mencionado). Se actualiza tanto las Cuentas Padres en la Contabilidad a la cual pertenece la partida, así como aquellas cuentas externas (de otras Contabilidades) que estén relacionadas con las cuentas utilizadas.

El algoritmo básico del Proceso es el siguiente:

1. El Sistema toma las Partidas a mayorizar y crea una copia de sus Encabezados en la tabla de Encabezados de Partidas Mayorizadas.
2. Luego crea una copia de sus Detalles, con lo cual se activa el Trigger de inserción y se actualizan los saldos de las respectivas cuentas de detalle, sus Cuentas Padres y las Cuentas externas relacionadas a ellas.

3. Si la inserción se realizó con éxito, se eliminan de las tablas de Partidas No Mayorizadas.

Dentro del Trigger de Inserción de la tabla de Detalles de Partidas Mayorizadas se realizan las siguientes operaciones:

1. Se verifica que la partida corresponda al Período Actual para poder ser mayorizada.
2. Verifica que en la tabla de SalDOS existan registros para las Cuentas de Detalle utilizadas para el Período Actual. Si no existen, entonces se crean, con lo cual se activa su Trigger de inserción.
3. Actualiza los saldos de las Cuentas de Detalle correspondientes al detalle de la partida que se está mayorizando.

Dentro del Trigger de Inserción de la tabla de SalDOS Contables se realizan las siguientes operaciones:

1. Se crea un Cursor con la estructura de los SalDOS Contables.
2. Se toman los montos de los SalDOS de Detalle insertados y se almacenan en el cursor
3. Se inicia un lazo que recorrerá una a una todas los SalDOS en el Cursor
4. Se toma el valor del SalDO de una Cuenta, y se determina cual es su Cuenta Padre inmediata.
5. Se actualiza la Cuenta Padre con el nuevo valor. Para cada Cuenta Padre actualizada se determinará también cual es su Cuenta Padre inmediata, para así ir actualizando el Catálogo hasta llegar al primer Nivel.
6. Si existe una Cuenta relacionada a la Cuenta en el Cursor, ésta se actualiza también, así como sus correspondientes Cuentas Padre. Con esto se logra la Consolidación de Contabilidades.
7. Se actualiza asimismo los SalDOS Reales de los Rubros Presupuestales que tengan asociada la Cuenta utilizada. Los correspondientes Rubros Padres también son actualizados.
8. Para cada una de los SalDOS insertados se seguirá el mismo proceso de Actualización.

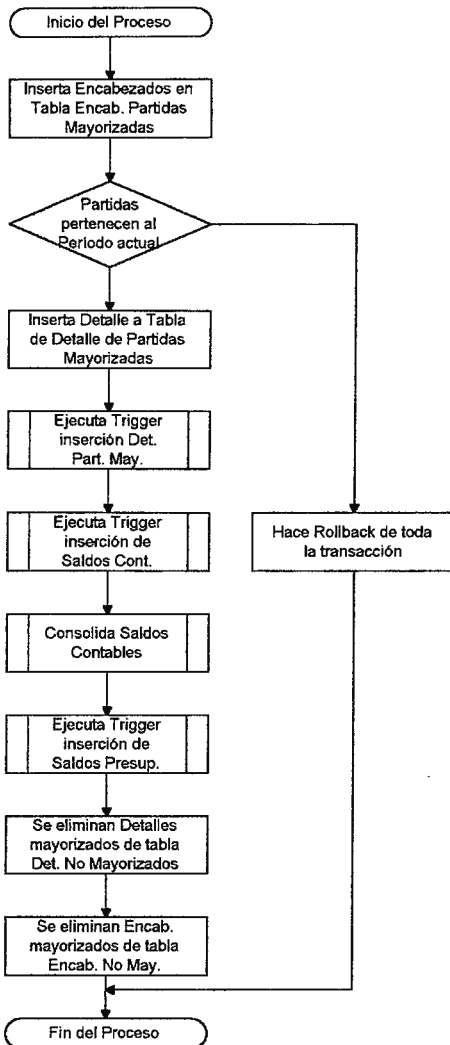


Ilustración 16: Diagrama de Flujo del Proceso de Mayorización de Partidas.

Proceso de Cierre Contable Mensual:

Este proceso se encarga de “marcar” un Período Contable como “Cerrado”, con lo cual se indicará que sus Saldos ya no sufrirán ningún cambio de ahí en adelante, o lo que es lo mismo, ya no se mayorizarán más partidas para ese Período.

Las operaciones que realiza el proceso son las siguientes:

1. Verifica si hay Partidas Mayorizadas para el Período a cerrar.
2. Si no hay Partidas Mayorizadas cancela el proceso, de lo contrario continúa.
3. Verifica si hay Partidas No Mayorizadas para el Período a cerrar.

4. Si hay Partidas No Mayorizadas que pertenezcan al período actual, muestra un mensaje indicando que éstas deben ser eliminadas o mayorizadas antes de proseguir el Cierre, y cancela el proceso. Si no encuentra Partidas No Mayorizadas para el Período a cerrar, entonces continúa.
5. Determina el Código del siguiente Período Contable.
6. Calcula la Pérdida o Ganancia correspondiente al Período a Cerrar.
7. Actualiza la Cuenta de Pérdidas/Ganancias a fin de almacenar su valor para tenerlo disponible al realizar ciertos reportes. Esto no requiere saldar las cuentas de Ingresos y Gastos, ya que sólo se ha realizado un cálculo. Dichas cuentas se saldarán únicamente en el Cierre del Ejercicio.
8. Marca el Período Contable actual como “Cerrado”.
9. Actualiza el siguiente Período contable como el Período Actual de la Contabilidad activa.

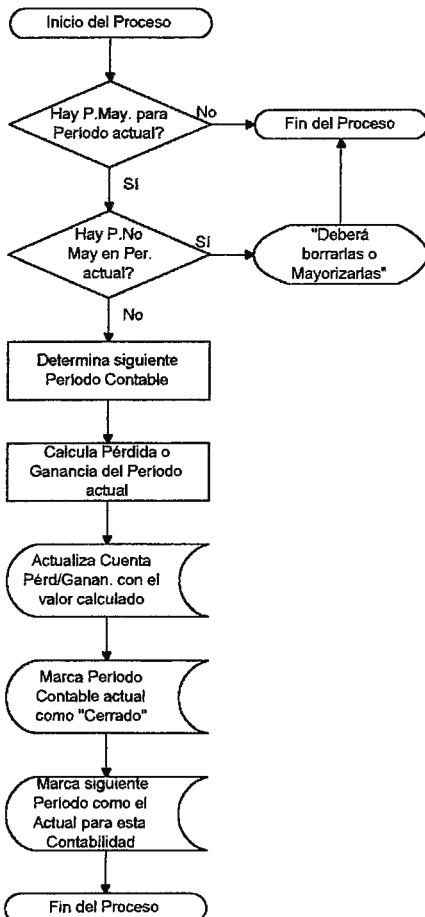


Ilustración 17: Diagrama de Flujo del Proceso de Cierre mensual o de Periodo contable.

Proceso de Cierre del Ejercicio.

Este proceso se realizará al final del Ejercicio Contable, a fin de completar el proceso contable, almacenando la información referente al Catálogo, los Saldos y las Partidas utilizadas en el año en las Tablas Históricas, y preparando los archivos (tablas) de trabajo para la información del siguiente Ejercicio.

Los pasos que sigue el proceso son los siguientes:

1. Realiza varias verificaciones, entre ellas si ya se cerró el último Período del Ejercicio.
2. Calcula el Debe y el Haber de la Cuenta de Pérdidas o Ganancias del Ejercicio.
3. Crea la Partida de Cierre de Cuentas de Resultados, con la cual saldará las cuentas de ingresos y gastos.
4. Crea la Partida de Cierre de Cuentas del Balance, con lo cual saldará las cuentas restantes.
5. Actualiza las tablas históricas con la información del Ejercicio a cerrar.
6. Prepara las tablas de trabajo para empezar el siguiente Ejercicio Contable.

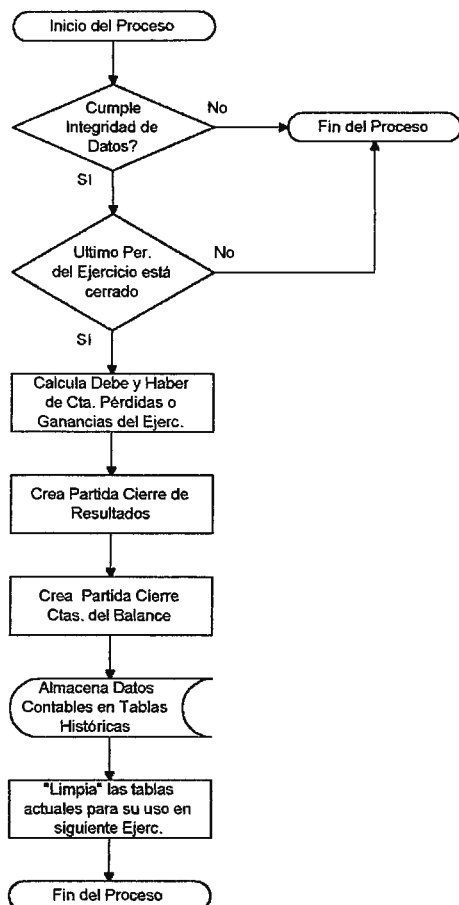


Ilustración 18: Diagrama de Flujo del Proceso de Cierre del Ejercicio.

Proceso de Copia de Catálogo de Cuentas base.

Este proceso permite crear las cuentas para un nuevo Catálogo utilizando las cuentas de otro Catálogo ya existente en el Sistema. Se especifica únicamente el código de las Contabilidades origen y destino, así como un rango de cuentas de la contabilidad origen y el nivel hasta el cual se desea crear las cuentas. Esto provee una manera rápida de crear Catálogos a partir de ciertas cuentas base, ya que no es necesario volver a digitar cientos de cuentas para una nueva contabilidad si existe otra con la misma estructura de cuentas, o que por lo menos sean iguales hasta cierto nivel.

La secuencia de las operaciones que realiza es la siguiente:

1. Verifica que existan ya los registros de ambas contabilidades, tanto la de origen como la de destino.

2. Verifica también que la contabilidad origen posea el rango de cuentas que se especificó.
3. El proceso empieza a insertar las cuentas hacia el Catálogo destino comenzando por las de nivel 1 hasta las del nivel especificado, es decir, las cuentas padre antes que las hijas. Esto es necesario por cuestiones de integridad de los datos, ya que el Trigger de inserción de la tabla de Cuentas Contables no permite agregar cuentas cuya cuenta padre no exista previamente.
4. En caso que ya existan algunas de las cuentas especificadas en la Contabilidad destino, el sistema enviará un mensaje de error y abortará el proceso.

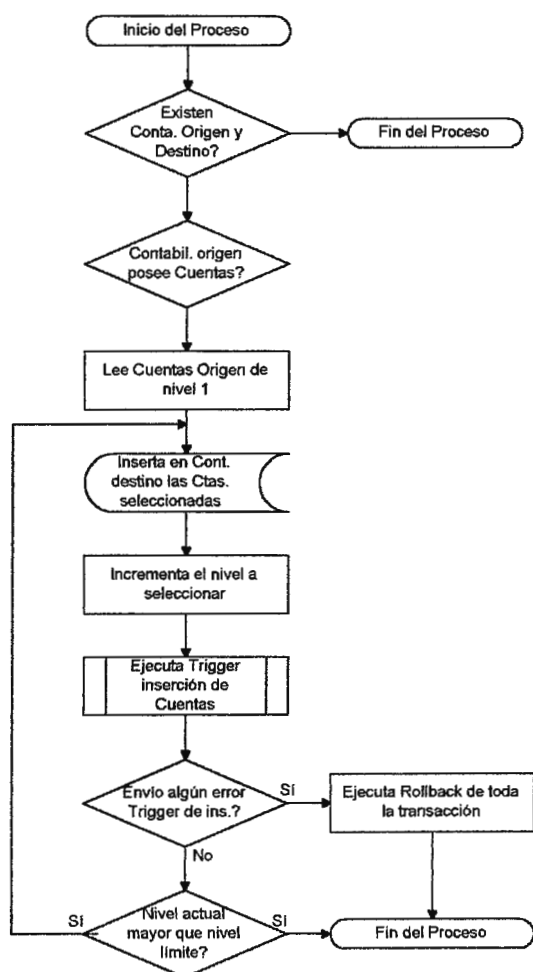


Ilustración 19: Diagrama de Flujo del Proceso de Copiado de Cuentas Contables.

Procesos para la Creación de Reportes

Los reportes del Sistema son creados utilizando una de dos fuentes de datos:

Consultas o “Queries” utilizando instrucciones SELECT para obtener los datos de distintas tablas de la base de datos; ó

Alimentando una tabla especial, la cual tiene una estructura sumamente general y a la cual son agregados ciertos registros seleccionados, valores calculados y/o agrupados, los cuales son obtenidos a través de un proceso complejo que no puede realizarse a través de una simple instrucción SELECT. El sistema lee esta tabla (y otras relacionadas a ella) y crea el reporte.

Se explicará a continuación cada uno de estos procesos utilizados principalmente para la creación de los Reportes Contables, y cuyos resultados son almacenados en la tabla mencionada anteriormente. Esta tabla es llamada “cop_saldo_pnom”.

Proceso de Creación de Reporte de Saldos Comparativo

El proceso prepara los datos (cuentas y saldos) para crear el reporte de saldos comparativos de dos períodos contables ya cerrados:

1. Verifica inicialmente si los dos períodos especificados realmente existen, y si ya han sido cerrados. El sistema busca los períodos tanto en las tablas de trabajo como en las históricas.
2. Toma todas las cuentas utilizadas en el período 1, y los agrega a una tabla temporal con sus correspondientes saldos.
3. Luego tomará todas las cuentas que hayan sido utilizadas en el período 2, y por cada una verificará si existe en el período 1. Si una cuenta existe, se añadirá únicamente su segundo saldo, y si no existe, entonces se creará un nuevo registro para ella en la tabla temporal, asignando CERO como su saldo del período 1.

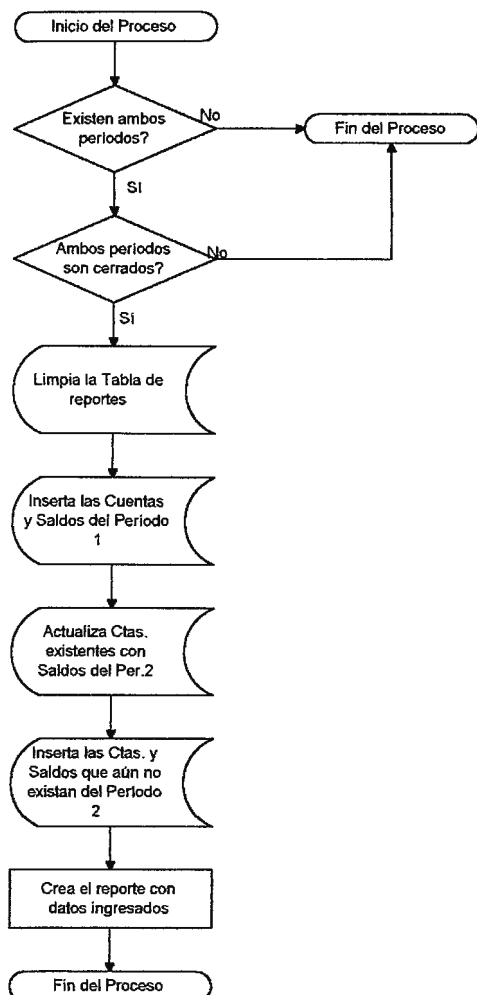


Ilustración 20: Diagrama de Flujo del Proceso de Generación del Reporte de SalDOS Comparativos.

Creación del Libro Mayor

Este procedimiento recibe seis parámetros:

- El código de contabilidad
- La fecha mínima y la máxima (rango de fechas)
- La cuenta mínima y la máxima (rango de cuentas)
- Si se va a utilizar también las Partidas No Mayorizadas

Con ellos realiza las siguientes operaciones:

1. Limpia la tabla cop_saldo_pnom a fin de utilizarla para el reporte solicitado
2. Inserta en la tabla los Movimientos de las Partidas agrupados por Cuenta y Fecha que pertenezcan al Catálogo especificado y que se encuentren dentro del rango de cuentas y fechas pasados como parámetros.
3. Para cada Cuenta por cada fecha, se empieza a buscar e ingresar las respectivas Cuentas Padres, utilizando un lazo que va desde la fecha mínima hasta la fecha máxima.
4. Se procede a actualizar los saldos de las Cuentas para el período especificado.
5. Se procede luego a calcular los cargos y abonos para los períodos anteriores al especificado, o sea, desde el 1 de Enero del Ejercicio actual hasta la fecha mínima especificada menos 1 día.
6. A los datos en la tabla únicamente se aplica el filtro del nivel de cuentas máximo a utilizar (ya en la aplicación cliente) para determinar qué filas serán utilizadas en el reporte.

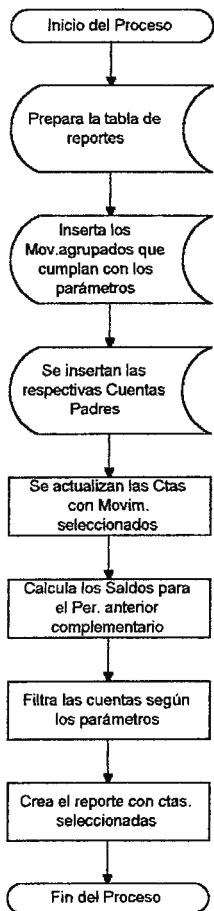


Ilustración 21: Diagrama de Flujo de la Creación del Reporte de Libro Mayor.

Creación de los Estados Financieros para el Ejercicio actual

Se utiliza el mismo proceso para la creación tanto del Balance General como para el Estado de Resultados. Se distingue entre ambos a través de los parámetros que son pasados al procedimiento:

- Código de la Contabilidad
- Fecha mínima y máxima del período a utilizar (rango de fechas)
- El tipo de reporte, es decir, si es un Balance General o un Estado de Resultados
- Si se desea utilizar también los saldos de las Partidas No Mayorizadas entre ambas fechas

A partir de estos datos el proceso realizará las siguientes operaciones:

1. Borra de la tabla `cop_saldo_pnom` los registros (filas) que existan.
2. Inserta en la tabla las cuentas correspondientes para cada reporte: las de Activo, Pasivo y Capital para el Balance General, y las de Ingresos y Gastos para el Estado de Resultados.
3. Con las partidas mayorizadas que pertenezcan al rango de fechas se calcula los saldos para las Cuentas que fueron insertadas a la tabla.
4. En caso de haberse especificado, el proceso también utiliza las partidas no mayorizadas que pertenezcan al rango de fechas y calcula sus correspondientes saldos, para así sumárselos a los valores anteriormente calculados.
5. Si el reporte es un Balance, entonces se calcula el total de Gastos e Ingresos para el rango de fechas especificado, para así saber cual fue la Ganancia o Pérdida correspondiente a ese período. También se utilizan las partidas no mayorizadas, si se hubiera especificado.
6. Se “asigna” el valor calculado a la Cuenta de Ganancias o Pérdidas del Ejercicio, a fin de cuadrar el Balance.
7. En la aplicación cliente se hará el respectivo reporte con los datos obtenidos.

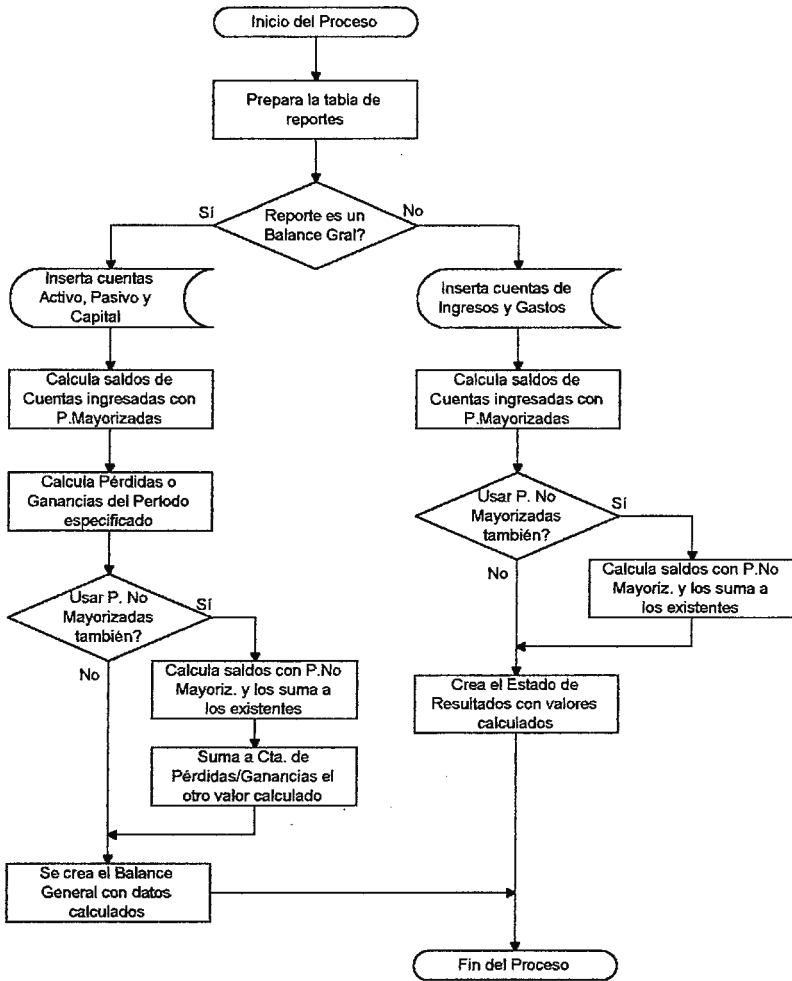


Ilustración 22: Diagrama de Flujo del Proceso de Generación de Estados Financieros.

Diseño de la Aplicación.

Para lograr nuestro objetivo de diseñar una aplicación Cliente/Servidor, es necesario tomar en consideración dos grandes áreas en la programación: El programa que servirá de interfaz entre el usuario y la base de datos, y la programación del Administrador de Bases de Datos para mantener la integridad de la información, así como también para realizar los procesos necesarios para satisfacer las necesidades de Control.

En consecuencia, para el análisis del desarrollo de la aplicación, el presente capítulo se divide en dos áreas: Programación de la Aplicación Gráfica y Programación de la Base de Datos.

Programación de la Aplicación Gráfica

La aplicación que hará de interfaz con el usuario final del Sistema, proporcionará todas las facilidades y restricciones para manipular los datos que se encuentran en el Servidor, asegurando que la integridad de la Base de Datos sea mantenida y a la vez, permitiendo que el usuario pueda manipular la información de manera eficiente, dando la posibilidad de ajustar el comportamiento del programa a las necesidades específicas de sus Sistema Contable-Presupuestal. Habiendo expuesto el anterior objetivo de la aplicación, a continuación encontrará una guía de como ha sido desarrollada, por lo que la información que encontrará a continuación fue escrita pensando en el área técnica que en un futuro dará mantenimiento o expansión a la aplicación, no obstante, ciertos puntos podrán ser de utilidad para el usuario final, en el camino de una mejor comprensión del funcionamiento general de la aplicación.

Plataforma de Desarrollo:

Durante la etapa de desarrollo del sistema, muchas herramientas han sido utilizadas para producir el sistema completo, hay que mantener en mente que muchas de ellas poseen funciones específicas de la base de datos y otras de la herramienta de programación utilizada. Para efectos de presentar una visión técnica del sistema gráfico, nos enfocaremos en las

funciones específicas de Visual BASIC 3.0 para Windows y OpenSnap 1.2.17. Todas estas herramientas han sido utilizadas en el Sistema Operativo Windows 95 de Microsoft a pesar de que las aplicaciones pueden correr en Windows 3.x. Debido a que las herramientas de desarrollo necesitan comunicarse con la base de datos utilizando la red y el protocolo TCP/IP, Windows 95 fue la elección óptima por su soporte integrado para redes TCP/IP.

Herramienta de Programación:

Visual BASIC 3.0 para Windows ha sido nuestra herramienta principal de desarrollo, ya que permite la creación de aplicaciones Windows basados en Ventanas, Eventos, Controles y Propiedades, lo cual facilita al programador la producción de interfaces gráficas para el usuario a la vez que su integración con bases de datos locales y remotas. El lenguaje debajo de esta herramienta es un estilo de BASIC Estructurado y ampliado, que permite la manipulación de objetos, propiedades, y eventos propios del ambiente Windows.

La versión 3.0 de Visual BASIC puede ser ejecutada en ambientes Windows 3.1, Windows 3.11 y Windows 95, produce programas ejecutables (.EXE) que pueden ser ejecutados en cualquiera de los ambientes antes mencionados, con la ayuda de uno o varias Librerías Dinámicas o DLL (Dynamic Link Libraries) dependiendo de que controles y librerías de funciones compongan el programa.

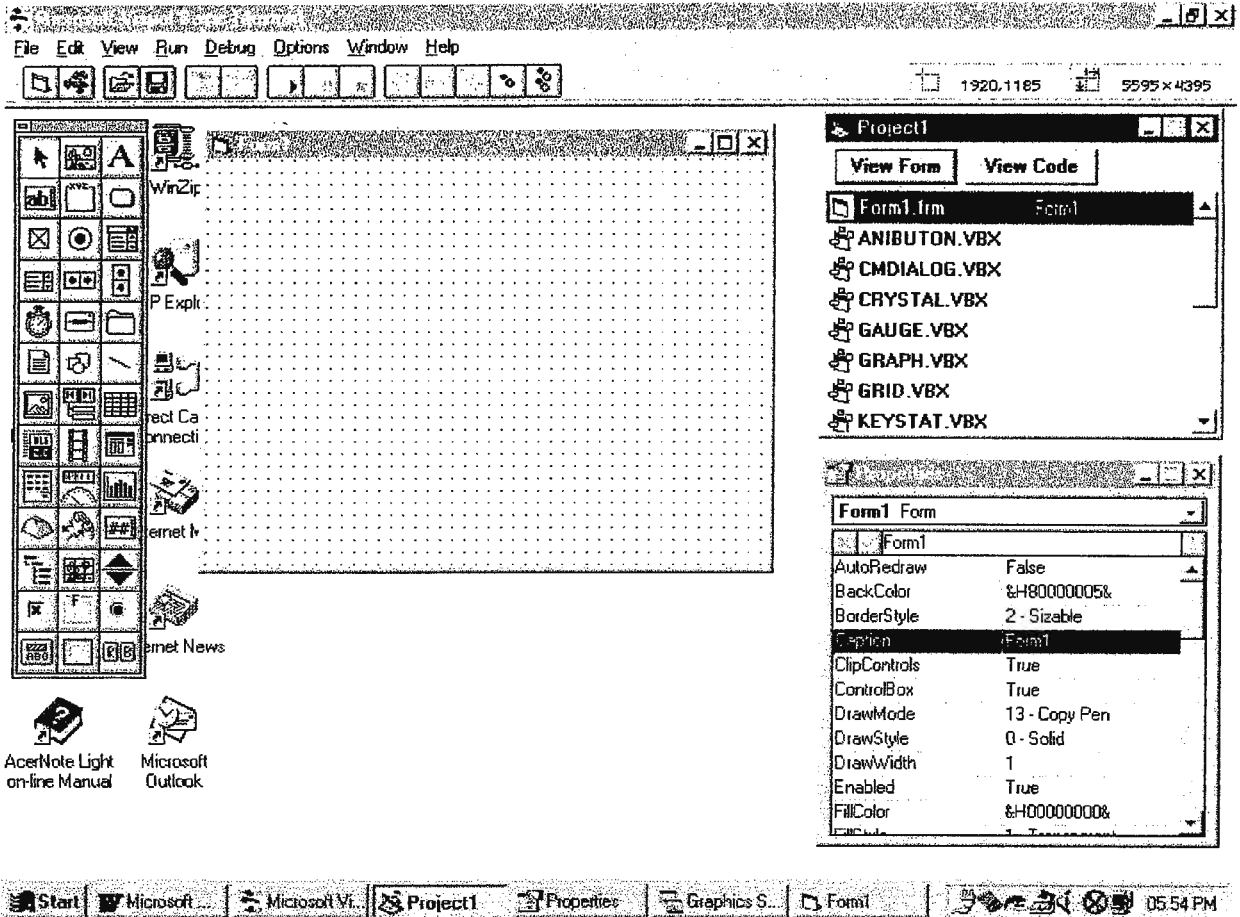


Ilustración 23: Ambiente de Desarrollo de Visual BASIC 3.0 Professional.

Un programa típico en Visual BASIC esta compuesto por los siguientes componentes:

- Formas:** En la aplicación final, cada forma representa una ventana, la cual sirve como interfaz entre el usuario y los datos y/o procesos del programa.
- Controles:** Un control es un componente que puede ser añadido a una aplicación y que posee un conjunto de características programáticas específicas, estos son añadidos a las formas y proveen de capacidades para presentar/manipular las entradas del usuario o las salidas del sistema. Un control puede o no tener una representación gráfica en la aplicación, ya que algunos sólo añaden funciones que extienden las capacidades del Lenguaje de Programación.
- Propiedades:** Una propiedad es una característica de una forma o control, la cual puede tomar diversos valores, dependiendo de su tipo de dato, puede ser considerado como una variable asociada a una forma o control, la cual cambia la apariencia, función o comportamiento de un objeto.

Eventos: Son Procedimientos o Rutinas de Código que son ejecutadas por el sistema y no por una llamada explícita en el programa. Se ejecutan si se da una circunstancia específica en el Sistema. Los eventos están asociados a toda la aplicación, a una forma o aun control.

Lenguaje: Como cualquier lenguaje estructurado, Visual BASIC provee de funciones y procedimientos, es decir, porciones de código que realizan una tarea específica y que pueden ser llamados desde otros programas o procedimientos. Las características de estos componentes son similares a las de cualquier lenguaje estructurado, como Pascal por ejemplo. También provee de una amplia variedad de Tipos de Dato, así como también de las estructuras de control típicas: (if, while, for, case, etc.). También tiene soporte para recursividad, arreglos, funciones matemáticas, manipulación de strings, algebra booleana, etc.

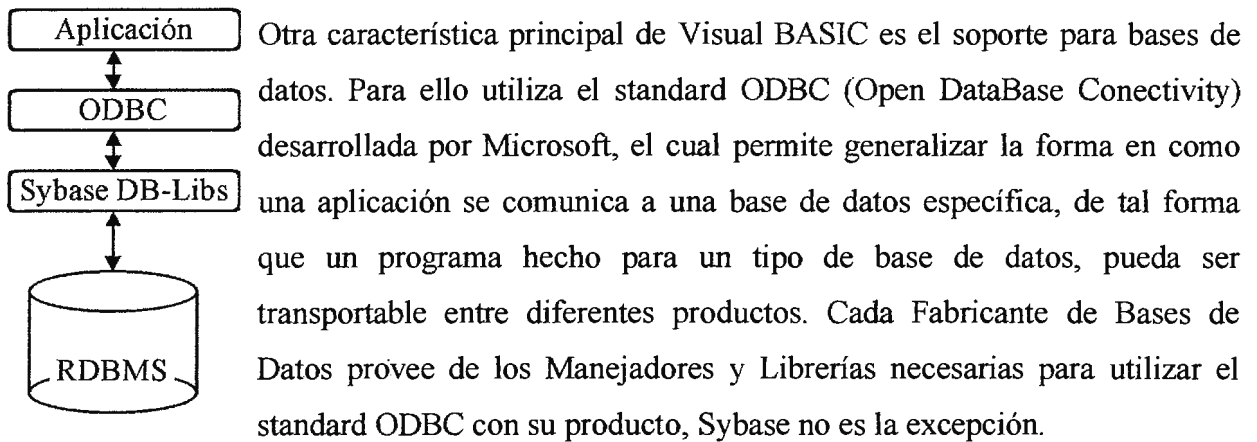


Ilustración 24: Esquema de Conexión al RDBMS utilizando ODBC

Sin embargo, en el caso de Sybase SQL Server System X y similares, ODBC no es la alternativa más eficiente, ya que su arquitectura de 3 capas hace que la comunicación entre la aplicación y la base de datos sea más

lenta, puesto que hay pre-procesamientos a lo largo del camino.

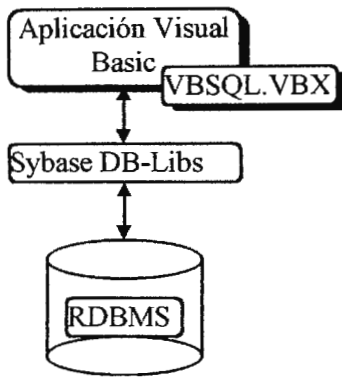


Ilustración 25: Esquema de Conexión al RDBMS utilizando DB-Libs.

Comunicación con el Manejador de Base de Datos (Servidor):

Con el fin de comunicar la aplicación con la base de datos de la manera más eficiente, se ha utilizado un control de Visual BASIC que encapsula todas las funciones que forman las DB-Libraries de Sybase, el cual permite hacer llamadas directas al servidor de bases de datos sin ningún pre-procesamiento intermedio. Debido a ello, también provee mecanismo para el manejo de errores, tanto de la aplicación como de los mensajes que el servidor de bases de datos pudiera enviar.

El Control se llama VBSQL.VBX, desarrollado también por Microsoft para proveer de una manera eficiente de conectividad con el Producto Microsoft SQL Server. Este Manejador de Bases de Datos comparte la misma tecnología con el Sybase SQL Server, sin embargo a medida que ambos productos han ido evolucionando, estas tecnologías se han distanciado. Aun así, desde el punto de vista del desarrollo de aplicaciones Cliente servidor en Visual BASIC, el control es 100% funcional tanto para Sybase SQL Server X como para el Microsoft SQL Server 6.0.

Herramienta CASE.

OpenSnap 1.28.1 ha sido utilizado para acelerar y facilitar el desarrollo tanto de la Base de Datos como de la Aplicación Visual BASIC. Este producto desarrollado por la empresa SOIN de Costa Rica, está diseñado para trabajar exclusivamente con Sybase SQL Server X, y Visual BASIC 3.0 para Windows con VBSQL.VBX. Su filosofía de desarrollo está basada en un repositorio central, en el cual se guardan todas las definiciones de objetos de la Base de Datos al igual que de objetos en el ámbito de la Aplicación Visual BASIC, dicho repositorio no es más que una base de datos en Sybase SQL Server X que contiene la definición y documentación de todos estos objetos para su posterior generación.

OpenSnap provee soporte para la definición de uno o más sistemas, cada uno de ellos consisten de uno o más objetos de la base de datos y de la aplicación Visual BASIC. Cada Sistema puede contener:

Objetos de la Base de Datos:

- Tablas
- Campos
- Reglas
- Relaciones entre Tablas
- Triggers
- Procedimientos Almacenados

Objetos de las Aplicaciones Visual BASIC:

- Formas de Consulta (CONLIS)
- Formas de Consulta Gráfica (CONGRA)
- Formas de Selección por Navegación (SELNAV)
- Formas de Mantenimiento (MANSAB)
- Formas de Mantenimiento Simple (MANMAE)
- Formas de Mantenimiento Maestro-Detalle (MANDET)
- Formas de Menú (MENU)

Generación de Reportes

- Procedimientos Almacenados para Generación de Reportes
- Formato de los Reportes

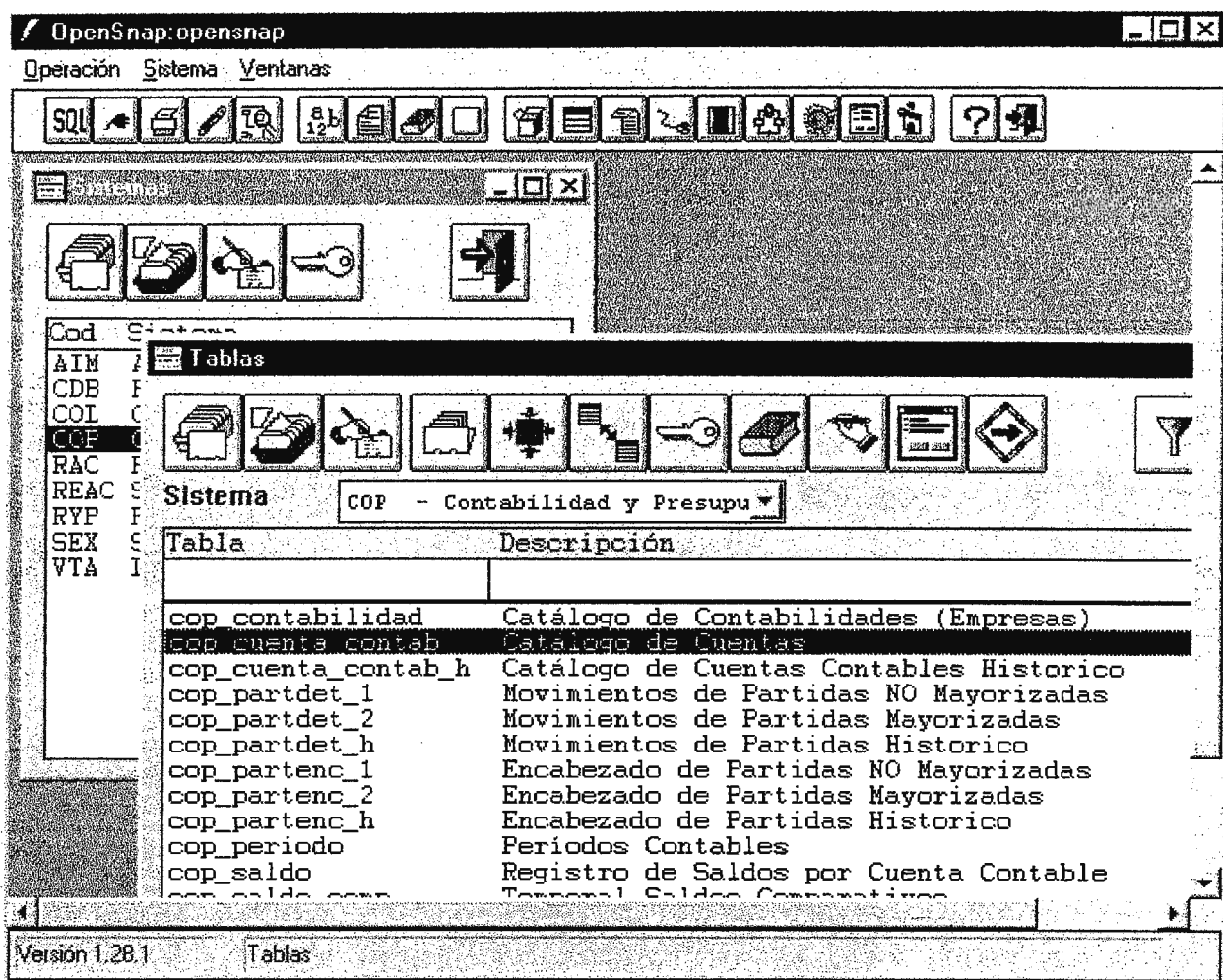


Ilustración 26: Ambiente Gráfico de OpenSnap versión 1.28.1.

Con el fin de desarrollar una aplicación completa, OpenSnap proporciona facilidades para relacionar los tipos de Formas antes mencionadas, de tal manera que el mantenimiento de una Tabla, es la combinación de un SELNAV y un MANSAB, con la posibilidad de que este último puede invocar a una Forma de Consulta o CONLIS, luego de igual manera, una forma de menú invoca a tantas Formas SELNAV como mantenimientos existan.

Por otro lado, OpenSnap provee de un Generador de Reportes, los cuales constan de 2 partes: Un Procedimiento Almacenado y Un Formato de Reporte. El Procedimiento Almacenado contiene el Código SQL que generará los datos necesarios para elaborar el Reporte y el Formato tiene las especificaciones de como estos datos serán impresos. Para el manejo de impresión de reportes, OpenSnap posee un "Spooler" o Cola de Impresión, el cual

administra los reportes que se han impreso en un determinado momento, por lo que este programa debe ser integrado y distribuido junto con la aplicación generada.

Durante el desarrollo encontramos que el soporte para la creación y generación de objetos en la base de datos de OpenSnap era lo suficientemente útil y flexible, sin embargo, el soporte para la creación de objetos de la aplicación Visual BASIC fue considerado poco flexible y en algunos casos, tedioso, ya que en el 90% de los casos había que modificar el código Visual BASIC generado por OpenSnap para adaptarlo a nuestras necesidades, lo cual involucra un conocimiento a fondo de como OpenSnap ha sido diseñado. Para no desperdiciar las capacidades de la Herramienta, se optó por crear Código en Visual BASIC (Formas Generales) que reemplazan algunas de las Formas generadas por OpenSnap y que a su vez tienen la capacidad de integrarse con el código generado por OpenSnap.

Después de haber sido creadas las formas generales que reemplazarían las de OpenSnap, sólo se utilizaron 2 formas generadas por este: Formas de Menú y Formas de Mantenimiento Simple.

De igual manera, encontramos que el Generador de Reportes de OpenSnap no se adecuaba a nuestras necesidades, ya que el programa para crear formatos era poco flexible, por lo que se optó en ese caso, utilizar la Aplicación Crystal Reports, la cual viene incluida con Visual BASIC 3.0 Professional. Este generador de reportes utiliza el standard ODBC para obtener los datos del RDBMS, y provee de un diseñador de formatos para reportes poderoso, que permite incluso la creación de “scripts” o pequeñas rutinas de control en un lenguaje propio de Crystal Reports. Por otra parte, esta herramienta provee de un nivel de integración con Visual BASIC adecuado, utilizando controles VBX.

Debido a las decisiones tomadas respecto a las herramientas de desarrollo, la aplicación emplea los 2 esquemas de comunicación con la base de datos, anteriormente explicadas, es decir: ODBC para la generación de reportes y Sybase DB-Libs para el resto de la aplicación.

Formas Generales

Se cuenta en el Sistema con distinto tipos de Formas Standard para las Ventanas utilizadas en cada uno de las opciones del sistema. Estas formas son utilizadas para el

mantenimiento, consultas y reportes de las distintas tablas del sistema, y serán explicadas detalladamente a continuación.

Formas de Menú

Las Formas de Menú generalmente son utilizadas como la Ventana Principal del Sistema, la cual contiene una barra de menú en la parte superior de la ventana, la cual posee las diferentes opciones principales del sistema, cada opción despliega un menú estilo “Pop-Up” que a su vez contiene otras opciones, estos niveles de sub-menú pueden extenderse hasta 4 niveles de ramificación.

En la parte inferior de la forma de menú se encuentra la Barra de Estado, que contiene información útil para el usuario. Esta se divide en 4 paneles, el primero muestra la Contabilidad o Empresa Activa, ya que el sistema puede trabajar con varias contabilidades, es necesario mostrar esta información. El segundo panel muestra el Usuario del Sistema que esta utilizando la aplicación, cabe mencionar que este no es el mismo usuario de la Base de Datos, recordando que el Sistema tiene un mecanismo de doble seguridad. El tercer y cuarto panel contienen la Fecha y Hora Actual respectivamente.

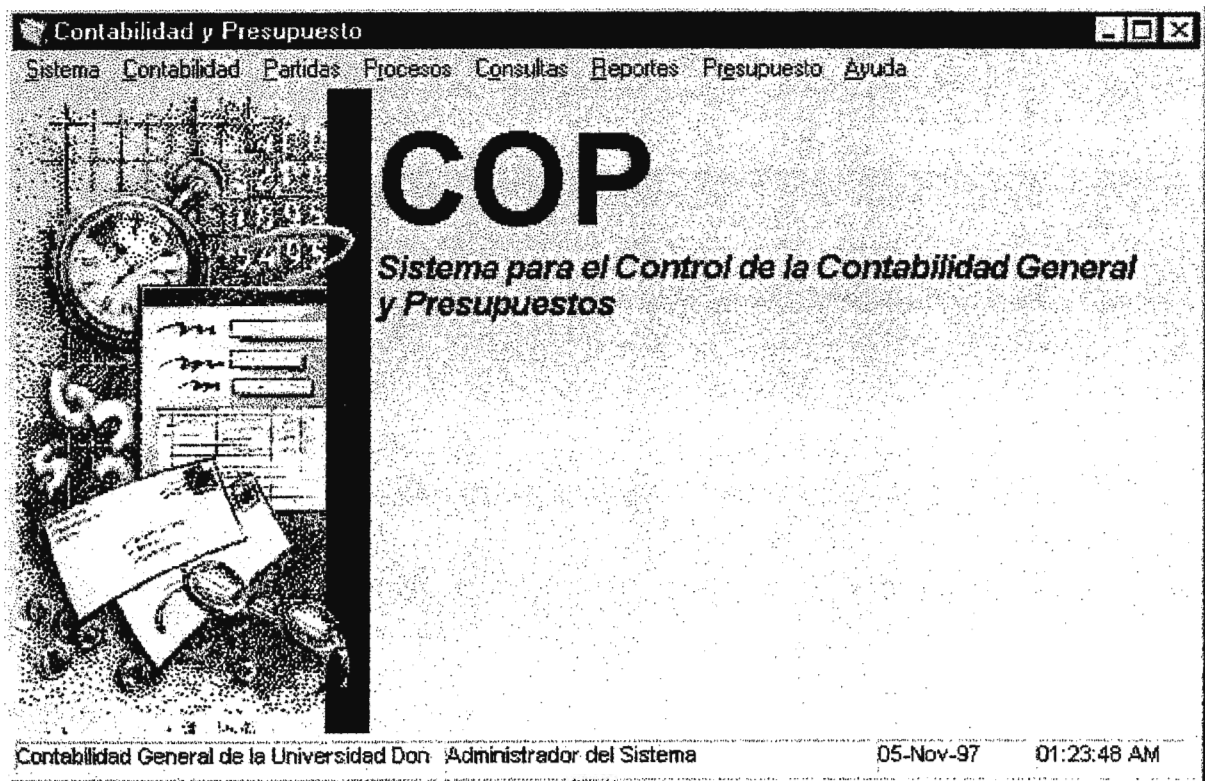


Ilustración 27: Forma de Menú Principal de la Aplicación COP.

Los menús Pop-Up consisten de varias opciones o sub-menús los cuales invocan a otras formas, la invocación de estas formas se hace mediante el siguiente código en el Evento Click de cada opción de menú:

```
Dim F as New <Nombre de la  
Forma>  
F.Show  
Set F=Nothing
```

Formas de Selección por Navegación (SELNAV)

OpenSnap genera estas formas para dar mantenimiento básico a una tabla en el sistema, estas formas permiten visualizar los datos o registros contenidos en una tabla y permite hacer búsquedas y ordenamientos. Por otro lado permite Adicionar, Modificar o Borrar un registro determinado a través de la invocación de otra forma generada por OpenSnap, estas son los MANSAB, de esta manera, el mantenimiento de una tabla del sistema esta compuesto básicamente de un SELNAV y un MANSAB. Sin embargo, la funcionalidad de los SELVNAV generados por OpenSnap está limitada por una deficiencia, la cual no permite visualizar mas campos que los que quepan en el ancho de la ventana, ya que las listas de visualización de datos carecen de una Barra de Desplazamiento Horizontal. En nuestra aplicación en específico, muchas tablas constan de varios campos, y algunos de ellos posee una gran longitud (30 o más caracteres) por lo que muchas veces el ancho de la pantalla no era suficiente para visualizar todos los campos de la tabla. Esto nos impulsó a crear nuestros propios SELNAV, que se integran fácil y eficientemente a los MANSAB generados por OpenSnap.

Estos SELNAV son ocupados extensivamente por toda la aplicación a excepción de los mantenimientos del Catálogo de Cuentas Contables y del Catálogo de Rubros Presupuestales. Debido a que los datos contenidos dentro del Catálogo de Cuentas y el Catálogo de Rubros de Presupuesto se relacionan jerárquicamente, se hicieron formas de mantenimiento que reflejaran de manera gráfica esta estructura. Los MANSAB de estas dos tablas fueron adaptadas para trabajar con estas formas de Navegación.

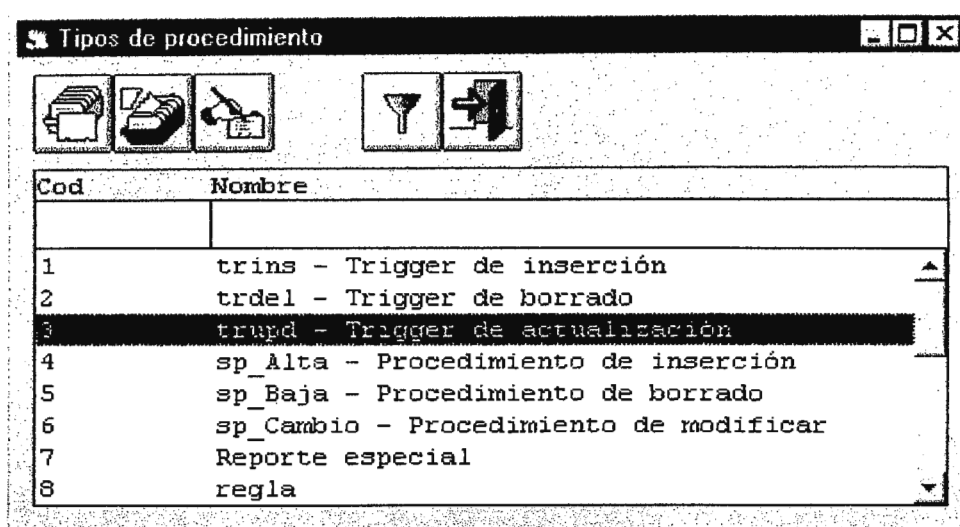


Ilustración 28: Forma SELNAV generada por OpenSnap.

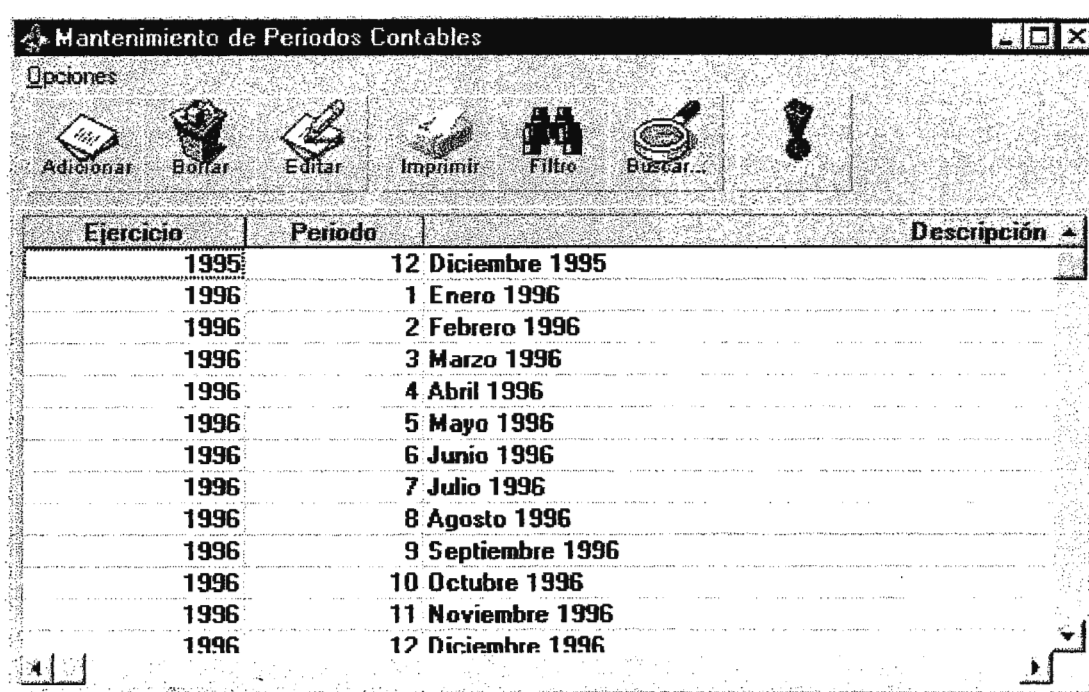


Ilustración 29: Forma SELNAV rediseñada para sustituir a la que genera OpenSnap.

Cada SELNAV posee un menú y una barra de Botones que contienen las siguientes opciones básicas:

Adicionar Permite Adicionar un nuevo registro a la tabla, a través de la invocación de la respectiva forma MANSAB, en el modo de adición (MODO_ALTA).

- Modificar** Permite Editar la información del registro seleccionado en la ventana del SELNAV. También invoca al respectivo MANSAB en modo de edición (MODO_CAMBIO).
- Borrar** Permite Borrar el registro seleccionado en la ventana del SELNAV, para ello invoca al MANSAB en modo de eliminación (MODO_BAJA)
- Filtro** Invoca a una forma general de filtro, la cual permite establecer condiciones de búsqueda y ordenamientos para seleccionar los registros que se quieren ver en la ventana de Navegación.
- Buscar** Permite buscar los registros que contengan un valor determinado para el campo seleccionado por el usuario.
- Salir** Permite Salir del Mantenimiento y regresar a la Ventana que la invocó.

En algunos casos, hay opciones adicionales, dependiendo de la tabla o del propósito del mantenimiento, también es posible invocar otros SELNAV o mantenimientos de tablas desde un botón o opción de menú del SELNAV, estos casos especiales serán descritos en detalle mas adelante. El código necesario para invocar al MANSAB correspondiente desde un Botón de Adición, Modificación o Borrado, es colocado en su respectivo Evento "Click".

```

Dim F as New <Forma de MANSAB>
Modo = [ MODO_ALTA | MODO_BAJA | MODO_CAMBIO]
Push <Campo Llave Primaria 1>
Push <Campo Llave Primaria 2>
:
Push <Campo Llave Primaria n>
F.Show MODAL
If Pop() then
    Leer
EndIf

```

Formas de Mantenimiento Simple de Adición y Borrado (MANSAB)

Las formas de MANSAB son generadas por OpenSnap en conjunto con los SELNAV, sin embargo, debido a que las formas de SELNAV han sido sustituidas por nuestras propias versiones, sólo se necesitan generar los MANSAB para luego modificar las formas de SELNAV e integrarlas con las formas recién generadas.

Ilustración 30: Forma MANSAB mostrando Cajas de Texto y Lista de Selección tipo Combo (Combo Box).

Estas formas por lo general son invocadas desde un SELNAV, y muestran todos los campos que tiene la tabla a la que se le está dando mantenimiento, en forma de TextBox, para los campos que requieren la entrada de un dato carácter, numérico o de fecha; en forma de CheckBox, cuando el campo es de tipo Booleano, es decir sólo puede tener 2 valores: verdadero o falso, en el caso específico de Sybase SQL Server X, este dato puede ser representado por el tipo “bit”, el cual sólo toma dos valores: 0 o 1. En los casos en que uno o más campos son parte de una llave foránea, es decir, que hacen referencia a la llave primaria de otra tabla, OpenSnap ofrece dos maneras de seleccionar los datos posibles. Puede ser utilizando una Lista de Selección estilo Combo o a través de una caja de texto convencional con la diferencia que permite invocar a otra forma en la cual se muestran todas las posibles opciones de las que se puede seleccionar. Estas últimas, son llamadas CONLIS o Formas de Consulta por Lista. Por lo general todos los MANSAB poseen dos botones en la parte inferior derecha de la forma, estos botones se leen: “Aceptar” y “Cancelar”, su función es realizar o no los cambios en la base de datos.

Debido a que el SELNAV invoca al MANSAB ya sea para Adicionar, Modificar o Borrar un registro, es necesario que exista un mecanismo para indicar la “modalidad” en que

ha sido invocado el MANSAB. Este mecanismo esta establecido por OpenSnap, y es realizado utilizando una variable Global llamada “Modo”, la cual puede tomar valores enteros representados por Constantes definidas también por la herramienta CASE, estas constantes son: MODO_ALTA, MODO_BAJA, MODO_CAMBIO. Los valores enteros que representan cada una de estas constantes no importan, pero si hay que tomar en cuenta que antes de invocar un MANSAB hay que establecer en que modalidad se hará, asignada a la variable global “Modo” su valor correspondiente.

En los casos en que un MANSAB ha sido invocado para modificar o borrar un registro específico, existe un mecanismo para indicarle a la forma invocada cual es el registro a modificar o borrar. Para ello el SELNAV pasa las llaves primarias del registro seleccionado como parámetros a la forma Invocada (MANSAB), dado que el MANSAB esta hecho para dar mantenimiento a una tabla específica y tiene implícito que campos son su llave primaria, esta es toda la información que necesita. Sin embargo “pasar parámetros entre formas” no es del todo cierto, pues Visual BASIC no provee de ningún mecanismo interno para hacer esto, por lo que se hace necesario implementar nuestro propio método. Con el objetivo de resolver este problema, OpenSnap implementa una Pila o “Stack” de tipo LIFO (Last IN, First OUT) utilizando arreglos de tipo “Variant”, un tipo de dato único de Visual BASIC que permite asignarle cualquier dato (número, fecha, caracteres, etc.) y cambia o “muta” al tipo de dato correspondiente.

Código Contabilidad	COP02	Nivel Jerárquico	3
Código de Rubro	E19650		
Código de Rubro Padre	6186		
Código de Tipo de Rubro	A01		
Descripción Rubro	INGRESOS VARIOS		
Es Detalle?	1 - Sí	<input checked="" type="checkbox"/>	Controlar este Rubro?
Editando		Aceptar	Cancelar

Ilustración 31: Forma MANSAB modificada después de su generación por OpenSnap, Notar pequeño botón para invocar un CONLIS.

Utilizando este mecanismo, el SELNAV introduce a la pila los valores de los campos que forman la llave primaria del campo seleccionado, luego establece el modo (Borrar o Modificar), después el MANSAB es cargado y este a su vez, extrae de la pila los valores en el orden inverso al que fueron introducidos. Una vez obtenido los valores pasados por el SELNAV, el MANSAB construye una instrucción "SELECT" para leer los datos del registro correspondiente y mostrarlo en la forma. Dependiendo de la modalidad del MANSAB, este Adicionará, Modificará o Borrará el registro en mantenimiento una vez el usuario presione el botón "Aceptar", en el caso contrario, si el usuario presiona el botón "Cancelar", no se hará ningún cambio en la base de datos.

En ambos casos, la forma MANSAB se cerrará y regresará el control a la rutina y forma que lo invocó, pero antes de eso, los MANSAB pasan un parámetro a través de la pila, llamado "Aceptar", el cual contiene True o False (Verdadero o Falso), esto le indicará al SELNAV si se hizo algún cambio al registro y por tanto, si es necesario leer los registros de la tabla y refrescar el SELNAV.

Por lo general los MANSAB no poseen ningún otro botón, aunque es posible agregarlo en Visual BASIC. Este fue nuestro caso, debido a que nos pareció poco funcional la forma en como se invocan los CONLIS para un campo específico en el MANSAB, por lo que decidimos agregar un pequeño botón para indicar que ese campo tiene una lista de posibles valores que el usuario puede consultar. Este botón contiene el siguiente código con el fin de invocar a su respectivo CONLIS.

```
Dim F as New <Forma CONLIS>
F.Show MODAL
If Pop() then
    <Caja de Texto>.Text = Pop()
EndIf
```

Formas de Consulta por Lista (CONLIS)

Los CONLIS son formas invocadas generalmente desde un MANSAB que contiene un campo que es llave foránea de otra tabla, por lo que la función principal del CONLIS es mostrar todos los posibles valores que puede tomar ese campo. Esto lo hace mostrando todos los registros de la tabla a la cual hace referencia el campo. Generalmente el CONLIS muestra como mínimo todos los campos que forman la llave primaria de la tabla referenciada, y un campo descriptivo.

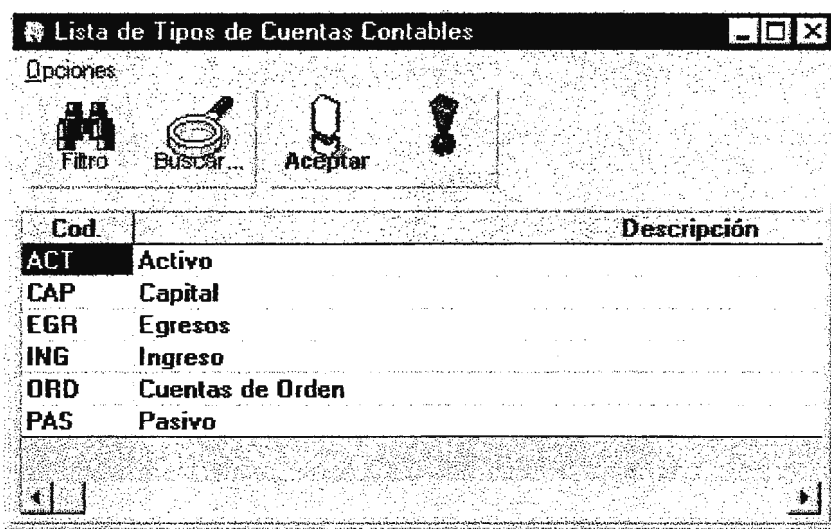


Ilustración 32: Forma CONLIS mostrando los tipos de Cuentas contables a seleccionar.

La apariencia y código fuente del CONLIS son muy similares al de un SELNAV, con la diferencia de que no tiene las opciones y botones para Adicionar, Borrar o Editar, pues su finalidad es sólo mostrar información para seleccionar y devolver el valor del registro seleccionado al procedimiento que lo invocó. Los Botones y opciones que un CONLIS incluye son los siguientes:

- Filtro** Invoca a una forma de Filtro para establecer las condiciones y los ordenamientos de los datos que se quieren mostrar.
- Buscar** Esta opción ha sido adicionada en los CONLIS y no aparece en aquellos directamente generados por OpenSnap. Permite realizar una búsqueda en la tabla utilizando el campo seleccionado como llave de búsqueda.
- Aceptar** Permite devolver el valor seleccionado en el CONLIS al Procedimiento que lo invocó.

Salir Regresa al Procedimiento que invocó el CONLIS sin devolver ningún valor.

El CONLIS no recibe ningún parámetro de la forma que lo invoca, pero si retorna el valor de las llaves primarias del registro seleccionado sólo cuando el usuario presiona el botón “Aceptar” o utiliza otro método para seleccionar el registro deseado. En ambos casos, ya sea

```
Dim F as New <Forma del CONLIS>
F.Show MODAL
If Pop() then
    <Caja de Texto>.Text = Pop()
EndIf
```

que se seleccionó “Aceptar” o “Salir”, esta forma retorna otro valor adicional a través de la Pila, este se llama “Aceptar” el cual contiene true o false (Verdadero o Falso), con el fin de

indicarle a la forma que lo invocó, si se seleccionó o no un dato. El código Visual BASIC necesario para invocar a un CONLIS desde un campo en el MANSAB, se coloca en el evento “Db1Click” (Doble Click) de la etiqueta que acompaña la Caja de Texto. Debido a que se ha diseñado la aplicación para que sea fácil de usar, se han colocado varios mecanismos para realizar la misma acción, en el caso del CONLIS, el usuario puede también presionar la tecla [F4] mientras se esta digitando en la Caja de Texto o puede hacer “Click” en el botón que se encuentra a un lado del campo.

Forma de Filtro

Esta es una forma utilizada por los SELNAV y CONLIS a través de la opción “filtro”, cuyo propósito es proporcionar al usuario de una interfaz con la cual este pueda establecer condiciones de búsqueda y ordenamientos de la tabla que esta siendo mostrada por el SELNAV o CONLIS. Utiliza operadores lógicos para establecer una o más condiciones de selección entre un campo y valor específico, también posee una lista que permite seleccionar por que campos estará ordenado el resultado. Internamente, esta forma sólo requiere de un parámetro, este es el nombre de la Tabla a ser filtrada, dicho parámetro es pasado a través de la pila. Por otra parte, OpenSnap mantiene una bitácora de todas las instrucciones SQL que la aplicación va realizando, con el fin de poder establecer cuales tablas y campos están involucrados en un SELNAV o CONLIS y así poder mostrar la información correcta en la Forma de FILTRO. La forma de filtro crea una tabla temporal que contiene los campos y alias o nombres de tablas involucrados en la última instrucción SELECT que se realizó, luego busca en las tablas especiales ENTIDADES y CAMPOS que han sido generados automáticamente por OpenSnap, los nombres descriptivos de cada campo en el SELECT y así poder mostrarlos en las Listas de Selección de Campos del Filtro.

Criterios de Selección

Campos

- Ejercicio Contable (Año)
- Periodo Contable
- Fecha de Inicio
- Fecha Final
- Descripción del Periodo

Operadores

- =
- <>
- <=
- <
- >=
- >
- En
- Como
- Entre
- No

Expresión

Condición

cop_periodo.fecha_fin <= '19970101'

Ordenar

Ilustración 33: Forma Standard de FILTRO.

Hay que hacer énfasis que las tablas ENTIDADES y CAMPOS deben de estar generadas en la Base de Datos de la Aplicación para poder utilizar los filtros en los SELNAV y CONLIS.

La forma de FILTRO ha sido también integrada dentro de las nuevas formas SELNAV y CONLIS para aprovechar al máximo las herramientas que genera OpenSnap de manera automática, sin embargo para hacer búsquedas simples, se ha agregado la opción de Buscar en la barra de botones de los SELNAV y CONLIS.

Formas de Búsqueda (BUSCAR)

La Forma de Búsqueda es utilizada por las formas SELNAV y CONLIS de la misma manera que lo hacen con la forma de FILTRO, la cual permite hacer una búsqueda utilizando como llave el campo que ha sido seleccionado en el CONLIS o SELNAV.

Para utilizar esta opción, se debe seleccionar un campo o columna del SELNAV o CONLIS y luego se puede presionar el botón “Buscar” o elegir la opción “Buscar...” del menú Pop-Up de la Forma, luego aparecerá la forma de Búsqueda, la cual básicamente esta compuesta por una Caja de Texto, donde se digita la llave de búsqueda, y por un grupo de opciones que permiten seleccionar el tipo de operador lógico a ocupar en la búsqueda (menor que..., Mayor que, igual a, etc.).

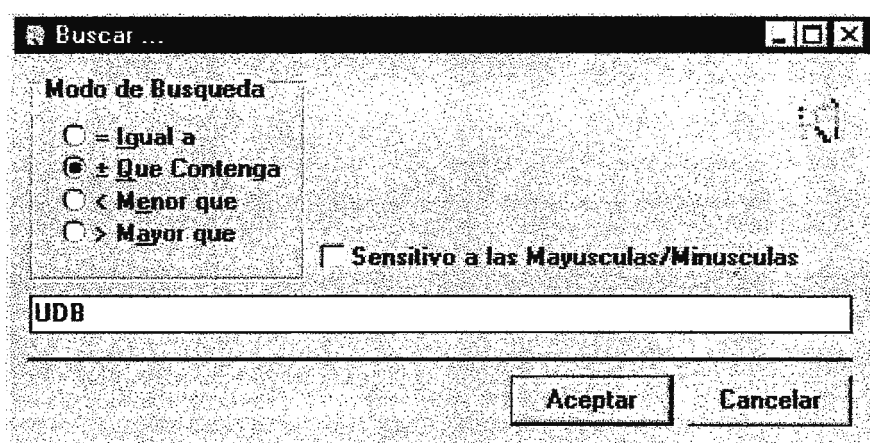


Ilustración 34: Forma de Búsqueda.

Dependiendo del tipo de dato que contiene la columna seleccionada para hacer la búsqueda, así serán las opciones que aparecerán en la forma. En el caso de que el tipo de dato de la columna sea de tipo carácter o similar (VARCHAR, CHAR, TEXT), aparecerán dos opciones adicionales. Estas y otras opciones de la forma de búsqueda se describen a continuación.

- Igual a** Encuentra todos los registros de la tabla, cuyo valor del campo seleccionado es igual al que se ha digitado en la Caja de Texto.
- Que Contenga** Encuentra todos los registros de la tabla, cuyo valor del campo seleccionado contenga el dato digitado en la Caja de Texto. Esta opción sólo aparece cuando el campo seleccionado es de tipo carácter o similar.
- Menor que** Encuentra todos los registros de la tabla cuyo valor del campo seleccionado sea menor que el digitado en la Caja de Texto.

Mayor que Encuentra todos los registros de la tabla cuyo valor del campo seleccionado sea mayor que el digitado en la Caja de Texto.

Sensitivo a las Mayúsculas y Minúsculas

Permite hacer búsquedas tomando en cuenta mayúsculas/minúsculas o insensitivo a este hecho. Sólo es visible cuando el campo seleccionado es de tipo Texto o similar.

Aceptar Este botón hace efectiva la búsqueda.

Cancelar Este botón cancela la búsqueda y regresa a la forma que la invocó sin realizar ningún cambio.

Estructura de la Aplicación COP:

La Aplicación COP, consiste de un Menú principal que contiene todas las posibles opciones del Sistema, su estructura es como sigue:

1. Sistema
 - 1.1. Conectarse
 - 1.2. Verificar Conexión
 - 1.3. Opciones...
 - 1.4. Cambiar Password..
 - 1.4.1. En el Sistema
 - 1.4.2. En la Base de Datos
 - 1.5. Cambiar de Contabilidad
 - 1.6. Programación de Actividades
 - 1.7. Usuarios del Sistema
 - 1.8. Actividades Pendientes
 - 1.9. Salir
2. Contabilidad
 - 2.1. Contabilidades o Empresas
 - 2.2. Catálogo de Cuentas (Jerárquico)
 - 2.3. Catálogo de Cuentas
 - 2.4. Períodos Contables
 - 2.5. Tipos de Cuentas
3. Partidas
 - 3.1. Mtto. de Partidas
 - 3.2. Tipos de Partidas
4. Consultas
 - 4.1. Partidas Mayorizadas
 - 4.2. Partidas Históricas
 - 4.3. Catálogo de Cuentas Históricas
 - 4.4. Saldos de Períodos Históricas
 - 4.5. Catálogo de Presupuesto Histórico

4.6. Saldos Presupuestales Históricos

5. Reportes

5.1. Contabilidad

5.1.1. Catálogo de Cuentas

5.1.2. Libro Auxiliar (Mayorizadas)

5.1.3. Libro Auxiliar (No Mayorizadas)

5.1.4. Libro Mayor

5.1.5. Balance de Comprobación

5.1.6. Estado de Resultados

5.1.7. Balance General

5.1.8. Saldo Consolidado

5.1.9. Comparativo de Saldo de Períodos Cerrados

5.1.10. Reportes Consolidados

5.1.11. Saldo Consolidado

5.1.12. Balance General Consolidado

5.1.13. Estado de Resultados Consolidado

5.1.14. Estados Financieros Históricos

5.1.15. Balance General Histórico

5.1.16. Estado de Resultados Históricos

5.2. Presupuesto

5.2.1. Rubros de Presupuesto

5.2.2. Tipos de Rubro

5.2.3. Montos Presupuestados

5.2.4. Cuentas Asignadas a Rubros

6. Presupuesto

6.1. Rubros Presupuestales

6.2. Tipos de Rubro

7. Ayuda

7.1. Contenido

7.2. Buscar Tópico

7.3. Acerca de...

El Sistema central de Repositorio de Aplicaciones y Entidades Institucionales CDBREP.

Muchas tareas referentes a la seguridad y comunicación con la Base de Datos son realizadas en el proceso de inicialización de la Aplicación COP, así como también algunas tareas que tienen que ver con la integración de las aplicaciones del Proyecto informático de Ciudadela Don Bosco, mas específicamente, con el sistema de Repositorio de Aplicaciones y Entidades Institucionales. Para entender este proceso de inicialización, es necesario primero describir ciertas normas definidas en el Sistema de Repositorio de Aplicaciones y Entidades Institucionales, y que se aplicarán a todos los sistemas en desarrollo. El sistema de Repositorio, cuyo nombre clave es CDBREP, posee control sobre los siguientes aspectos del Proyecto.

Aplicaciones o Sistemas

El sistema CDBREP, posee un registro de todas las aplicaciones que forman parte del proyecto Informático, cada aplicación posee un código identificador de 3 caracteres, una descripción y otros campos de información para llevar registro de las versiones y fechas de revisión de la aplicación. El propósito de este registro, es que exista un sistema central de control de aplicaciones, con el cual, un administrador de sistemas pueda controlar de manera más eficiente las aplicaciones esparcidas por toda la Institución.

Entidades Institucionales

La estructura organizacional de Ciudadela es amplia y compleja, a la vez que dinámica, dicha estructura es (debería ser) universal para toda la institución, es decir, todas las unidades de la institución conocen una estructura organizacional única de Ciudadela. Por ello, esta información que es útil para varios sistemas (Ej. : Recursos Humanos) ha sido centralizada en CDBREP. Este registro es una estructura jerárquica, en donde cada “Entidad” puede poseer

“Sub Entidades”, a la vez que puede ser parte de otra “Super Entidad”. Esta estructura jerárquica puede extenderse “n” niveles, dependiendo de la complejidad organizacional de la institución, además, cada entidad posee un calificativo o tipo de entidad definida por el usuario del sistema en un catálogo de “tipos de entidades institucionales”. Esta clasificación define la naturaleza de la entidad en un nivel jerárquico (Ej. : sección, departamento, sub-sección, unidad, etc.).

Aplicaciones Instaladas

Cada aplicación puede estar instalada en una o más Entidades Institucionales a lo largo de la Institución, por ello se hace necesario llevar un control de que aplicaciones están instaladas en que lugares. Este registro permite llevar este control, por medio de la asignación de una aplicación a una Entidad Institucional previamente registradas en el sistema. Esto además, permite que no se hagan instalaciones “ilegales” o no autorizadas de la aplicación, ya que todas las aplicaciones del proyecto verifican durante la etapa de inicialización, en que lugar están instaladas.

Usuarios y Permisos de las Aplicaciones

A través de este registro, se permite llevar un mecanismo extra de seguridad a las aplicaciones, además del que ya posee el Administrador de Bases de Datos Sybase. Debido a que una misma aplicación puede estar instalada en varias Entidades Institucionales, los usuarios de la aplicación varían también de acuerdo al lugar de instalación. Por ello, el registro de usuarios de los sistemas esta relacionado al catálogo de aplicaciones instaladas y no al catálogo de aplicaciones por sí sólo.

Cada usuario de un sistema instalado posee un identificador único (Login), y una clave secreta (password), además del nombre completo de la persona, última fecha de entrada al sistema, y permisos sobre la aplicación.

Programación de Actividades

Durante la fase de investigación del proyecto, se vió que uno de los graves problemas en la organización de Ciudadela es la falta de eficiencia en la comunicación y asignación de tareas. El sistema CDBREP incluye una herramienta que puede contribuir a disminuir los efectos de este problema, el cual es un mecanismo común para asignar tareas a diversas aplicaciones instaladas desde una aplicación central. Este registro se asemeja a una agenda electrónica compartida entre varias aplicaciones, en donde una entidad central puede asignar tareas específicas que deben ser realizadas por una aplicación en un período de tiempo determinado. Con el fin de poder realizar y coordinar de manera correcta estas actividades, es posible definir cuales son las actividades requisito de la tarea impuesta. Estas actividades requisito, no son mas que actividades previamente asignadas al sistema o a otros sistemas dentro del repositorio. Por otro lado, cada sistema dentro del proyecto, posee una serie de opciones para revisar, y dar mantenimiento a las actividades asignadas por usuarios del sistema o por el Sistema central de repositorio CDBREP.

Proceso de Inicialización del Sistema COP

Cualquier sistema desarrollado en Visual BASIC tiene un procedimiento o forma principal, el cual es ejecutado antes que cualquier otro proceso. En nuestro caso, el proceso de inicialización esta a cargo de la Forma de Menú Principal (MNCOP) la cual se encarga de realizar varias tareas de la etapa de inicialización.

1. Inicializar las librerías de comunicación DB-Libs para la Base de Datos
2. Verificar si se ha ejecutado la aplicación con parámetros de línea, para establecer una conexión con la Base de Datos sin necesidad de pedir un usuario y password.
3. Verificar y leer los parámetros globales de la aplicación, establecidos en la forma de "Opciones"
4. Si no hay datos suficientes para hacer la conexión con la base de datos, mostrar la forma de "Login" para conectarse a la Base de Datos.
5. Conectarse a la Base de Datos y establecer los valores de las variables globales de conexión.

6. Mostrar el diálogo de selección de Contabilidades si es que hay contabilidades definidas en el sistema, sino crear la primera contabilidad del sistema.
7. Luego de haber seleccionado la contabilidad a trabajar, verificar si el sistema ha sido autorizado en el Repositorio de Sistemas, lo cual esta registrado en la computadora local en el archivo COP.INI. Si no se encuentra en éste, la entidad institucional en la cual esta instalado, muestra una lista de posibles selecciones de acuerdo al repositorio de Aplicaciones Instaladas.
8. Después de obtener los datos de cual aplicación instalada es, pedir Login al sistema para establecer los permisos del usuario sobre las opciones del sistema.
9. Mostrar Saludo de entrada al sistema o salir si el login no fue exitoso.
10. Mostrar la forma de menú principal y esperar selección del usuario.

Durante las etapas de verificación, si ocurre algún error, ya sea de la aplicación o de la base de datos, este es capturado y mostrado al usuario en una Ventana de Mensajes. Por lo general si no es posible establecer una conexión con el servidor de Bases de Datos o si el password suministrado es erróneo, el proceso termina y la aplicación no es ejecutada.

Menú Sistema

Este menú posee todas las opciones necesarias para establecer, verificar la conexión con la base de datos, así como también el mantenimiento de parámetros generales de todo el sistema, tales como la Contabilidad de trabajo, cambio de password, actividades asignadas al sistema, usuarios del sistema, etc.

Opción: Conectarse

Esta opción permite conectarse al servidor de Base de Datos ya sea con el fin de conectarse como otro usuario o cuando se perdió la conexión por algún problema de la red o aplicación. Ya sea se este conectado o no con el servidor de bases de datos, esta opción muestra la forma de Login a la Base de Datos que se muestra en el procesos de inicialización del sistema.

The image shows a Windows-style dialog box titled "Inicio de sesión [COP]". It has a standard title bar with minimize, maximize, and close buttons. The dialog contains three input fields: "Usuario" with the text "sa", "Contraseña" with a masked password of asterisks, and "Compañía" with a dropdown menu currently showing "Contabilidad y Presupuesto". At the bottom, there are two buttons: "Aceptar" and "Cancelar". In the top right corner of the dialog area, there is a logo consisting of two overlapping hexagons, one containing "SO" and the other "IN".

Ilustración 35: Forma de Login a la Base de Datos Sybase.

Los componentes que forman esta ventana son los siguientes:

- Usuario** Espacio para digitar el Identificador de Usuario o Login tal y como se ha definido en la configuración del SQL Server.
- Contraseña** Aquí se digita la palabra clave del usuario tal y como se ha establecido en el SQL Server. Los caracteres digitados en esta caja de texto son reemplazados por asteriscos "*" con el fin de mantener la privacidad del password.
- Compañía** Esta caja de selección, permite escoger entre las compañías definidas en "Opciones"->"Compañías" del sistema. El objetivo principal de este mecanismo, es establecer nombres más amigables para el usuario, en lugar del nombre real de la Base de Datos a la cual se esta conectando.
- Aceptar** Descarga la forma de login y procede a la verificación de los datos entrados en el servidor SQL Server, si hay algún dato incorrecto, la forma de login volverá a aparecer para intentarlo de nuevo hasta un máximo de 3 veces, luego de eso, el proceso se aborta.
- Cancelar** Descarga la forma de Login, y no realiza Login al Servidor SQL Server, en el caso que se este en el proceso de inicialización, esto resultará en la cancelación del proceso.

Opción: Verificar Conexión

Esta utilidad del sistema proporciona al usuario de un método para comprobar si la conexión entre la aplicación y el Servidor de Bases de Datos esta funcionando o no. Después

de que el proceso de inicialización de la aplicación COP ha pedido al usuario su identificador de usuario (login) y clave de acceso (password), la aplicación establece un enlace con el Servidor SQL a través de la red. Una vez se ha establecido este enlace el usuario puede ocupar la opción “Verificar conexión” para comprobar si todavía sigue conectado al Servidor SQL. Al seleccionar esta opción del menú, aparecerá una pequeña ventana informando al usuario del estado de la conexión, este puede ser cualquiera de los dos siguientes:

- La Conexión esta Bien o,
- Se Perdió la Conexión.

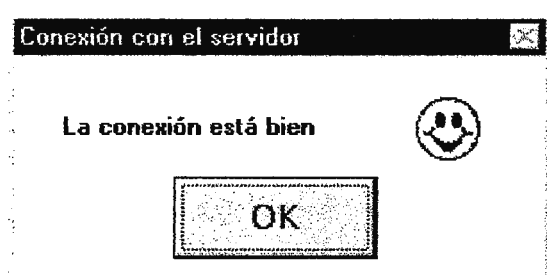


Ilustración 36: Ventana que se muestra al seleccionar “Verificar Conexión”.

Existen diversas causas por las cuales la aplicación puede perder la conexión con el servidor, entre las más frecuentes se enumeran las siguientes:

- A causa de la red física Ej.: cuando se desconecta un cable.
- Se apagó el servidor Sybase SQL Server.
- Se apagó o desconectó de la red el Servidor que contiene el SQL Server.
- Un error fatal en el SQL Server.
- Un error en el Computador que corre la aplicación relacionado al Sistema Operativo
- Se borró accidentalmente un archivo que sirve de soporte o es parte de la aplicación. (Ej.: un archivo DLL o VBX)
- Un error en la aplicación por falta de recursos (memoria, disco, etc.).

Opción: Opciones...

Esta utilidad muestra una ventana de diálogo con diversos parámetros que el usuario puede configurar. Dichos parámetros están relacionados a como la aplicación se enlazará con el Servidor Sybase SQL Server.

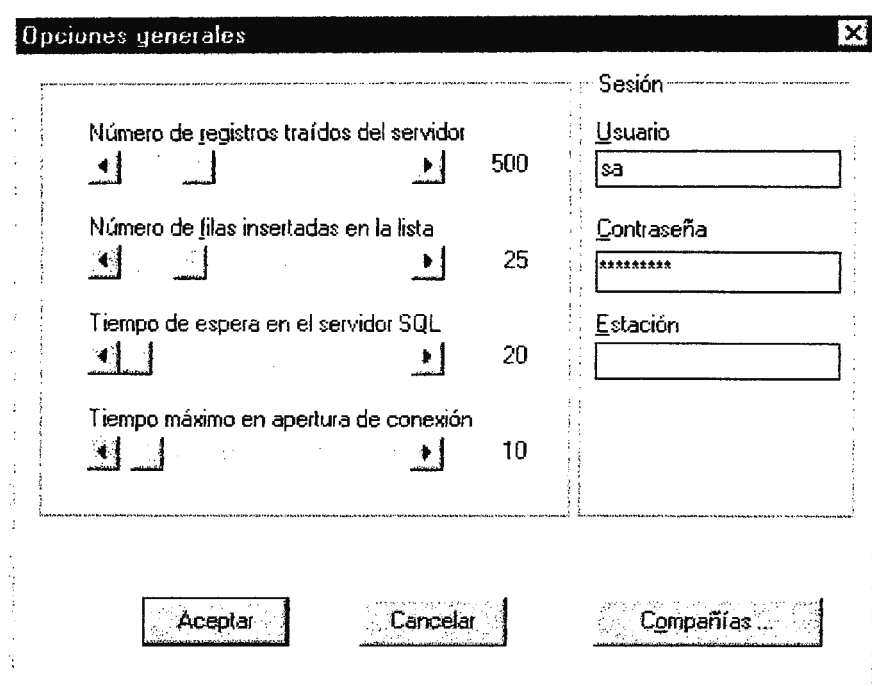


Ilustración 37: Ventana de Diálogo para configuración de Parámetros de comunicación con el Sybase SQL Server.

Las opciones que se muestran en la ventana de diálogo son las siguientes:

Número de registros traídos desde el servidor.

Cada vez que un SELNAV o CONLIS es mostrado, se realiza una consulta a la base de datos, la cual puede retornar un número indefinido de filas o registro, en los casos en que la tabla seleccionada contenga muchas filas, los SELNAV y CONLIS limitarán el número de filas a mostrar al valor seleccionado en este parámetro. Esto se hace debido a que los controles utilizados en Visual BASIC para mostrar los registros tienen un límite de datos que pueden contener, en los casos en que las filas retornadas por el SQL Server sean demasiadas, el control no podrá almacenar toda esa información y generará un error de memoria.

Número de filas insertadas en la Lista.

Similarmenete al caso anterior, algunos MANSAB contienen Listas de Selección que contienen todos los posibles valores que un campo puede tomar, la fuente de estos

datos proviene de una consulta o tabla de la Base de Datos, la cual puede exceder la cantidad de memoria máxima que un control de Lista o Combo puede contener.

Tiempo de espera en el servidor SQL.

Las aplicaciones basadas en VBSQL y DB-Libs de Sybase, manipulan y muestran los datos de la Base de Datos a través de consultas, es decir, que cada vez que la aplicación requiere mostrar información, esta manda un comando SQL de consulta (SELECT) al servidor SQL, el cual procesa y ejecuta el comando, para luego retornar los resultados (datos) a la aplicación. Este parámetro permite establecer el tiempo máximo de espera entre el momento en que se realizó la consulta y el que se empiezan a recibir los resultados; si este tiempo excede al establecido en esta forma la consulta o comando es abortado y un error es mostrado al Usuario.

Tiempo Máximo de apertura de conexión.

Cada vez que se muestra la forma de Login a la Base de datos, la aplicación envía al servidor SQL el nombre del usuario, su password y otro parámetro para que sean autenticados por el Servidor. Una vez enviados estos datos, la aplicación se queda en espera de una respuesta que le indicará si se autoriza o no a conectarse a la base de datos. Este tiempo de espera es limitado por este parámetro, si se excede, se muestra un error, y el proceso de login se repite, es decir, se vuelve a preguntar al usuario por su Login y password, a menos que la aplicación detecte que hubo un error de conexión, en ese caso, el proceso se aborta.

Sesión.

Este recuadro muestra los campos de Login, Password y estación de trabajo, los cuales pueden ser llenados por el usuario, para que el proceso de login a la base de datos este predeterminado. Cuando estos parámetros han sido establecidos, cada vez que se muestre la forma de login a la base de datos, los campos de Usuario y Contraseña aparecerán con los respectivos parámetros, de tal manera que el usuario no tenga que digitar esta información cada vez que se conecta a la Base de Datos.

Aceptar y Cancelar.

Estos botones son útiles para aceptar o rechazar cualquier cambio que se haya hecho a cualquiera de los parámetros mostrados en la forma.

Compañías.

Este botón de comando, muestra una ventana de diálogo adicional, en la cual se puede dar mantenimiento a las asociaciones entre compañía y nombre real de la base de datos. Una compañía no es nada mas que un nombre descriptivo de una base de datos específica y un SQL Server, de tal manera que le sea más fácil al usuario identificar las bases de datos en diferentes Servidores SQL. Las compañías definidas en este diálogo, son las que se mostraran en el campo “compañía” de la forma de login a la base de datos.

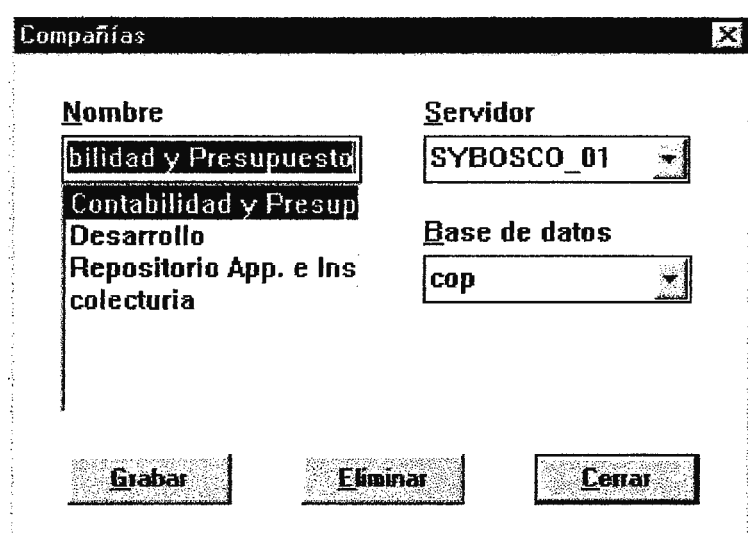


Ilustración 38: Forma de mantenimiento de compañías.

El diseño de este diálogo contiene los siguientes elementos:

Nombre Este elemento esta compuesto por dos controles, una caja de texto y una lista de Selección, esta última, muestra todas las compañías que ya han sido definidas por el usuario. Para modificar una de ellas sólo es necesario dar un Click sobre la compañía deseada. La caja de texto muestra el nombre de la

compañía seleccionada o de la que se está adicionando, aquí el usuario puede digitar cualquier nombre descriptivo.

- Servidor** Esta caja de selección tipo combo, muestra todos los servidores Sybase SQL Server definidos en esa estación, recordando que en un sitio pueden existir varios SQL Servers distribuidos en varios servidores dispersos por la institución, de tal modo que es posible que las aplicaciones (y sus respectivas bases de datos) estén también dispersas a través de diferentes servidores.
- Base de Datos** Muestra el nombre real de la base de datos seleccionada, esta caja de selección, contiene todas las bases de datos que maneja el servidor Sybase SQL Server seleccionado en la opción anterior.
- Grabar** Graba los datos de las compañías definidas en un archivo especial del sistema, para que luego aparezcan disponibles en las Formas de Login a la Base de Datos.
- Eliminar** Permite borrar una compañía definida.
- Cerrar** Cierra el diálogo de mantenimiento de compañías y regresa a la forma que la invocó, en este caso, a la forma de Opciones.

Opción: Cambiar Password

Este sub menú muestra dos opciones principales, las cuales se detallan a continuación:

Opción: En el Sistema

Esta opción muestra un diálogo que le permite al usuario del sistema cambiar su contraseña en el esquema de seguridades propia de la aplicación COP.

The image shows a dialog box titled "Cambio de Password del Sistema". The text inside the dialog box reads: "Cambio de Password del Sistema, para el Usuario: Administrador General de Aplicaciones". Below this text, there are three input fields for text entry. The first field is labeled "Digite su antiguo password", the second is labeled "Digite su Nuevo Password", and the third is labeled "Confirme su nuevo Password". At the bottom of the dialog box, there are two buttons: "Aceptar" and "Cancelar".

Ilustración 39: Cambio de Password del Sistema COP.

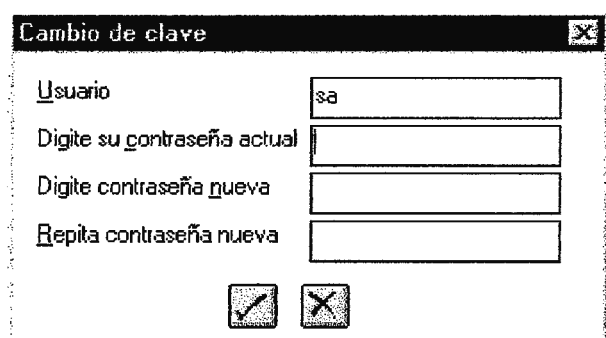
En la parte superior de la ventana se muestra el nombre completo del usuario que esta utilizando el sistema actualmente, y al cual se le cambiará su Password. Para realizar esta operación, el usuario deberá digitar el Password actual en la caja de texto que se lee “Digite su antiguo Password”, luego el sistema verificará si el Password proporcionado coincide con el registrado en la base de datos, y si este es el caso, el programa habilitará las dos cajas de texto que siguen, para dejar que el usuario digite su nuevo Password y lo confirme, de esta manera, el usuario verifica que el Password proporcionado ha sido correctamente digitado. Este último mecanismo de verificación se hace debido a que las tres cajas de texto no muestran el texto real del Password, sino, que sustituye cada carácter digitado por un símbolo “*” (asterisco) para fines de privacidad. Si el nuevo Password digitado fue confirmado correctamente, el Botón Aceptar es activado para permitir que el cambio se haga efectivo, de lo contrario, sólo el botón Cancelar permanece habilitado, el cual puede ser presionado en cualquier momento para abortar la operación.

Cabe resaltar, que los passwords del Sistema COP son guardados en tablas de la base de datos del Repositorio de Aplicaciones y Entidades Institucionales “CDBREP”, por lo que

un usuario experto podría acceder esa información por otros medios. Esta posibilidad indujo a que se diseñará un algoritmo de encriptación para guardar el Password de una manera “no entendible”, es decir, que sólo utilizando el programa adecuado se puede descifrar ese Password y así obtener la cadena de caracteres que lo forman. Este algoritmo de encriptación convierte la cadena de caracteres en otra muy diferente, la cual posee triple longitud que la original y esta compuesta únicamente por caracteres de dígitos (0 – 9). Sin el algoritmo de encriptación es virtualmente imposible descifrar el Password de un usuario, lo que añade un nivel extra de seguridad a la aplicación.

Opción: En la Base de Datos.

Esta sub opción del menú “Cambiar Password...”, permite cambiar la contraseña definida para el usuario en el Sybase SQL Server. Como se mencionó anteriormente, el Manejador de Bases de Datos implementa su propio mecanismo de seguridad, a través de la creación de “Logins”, “Usuarios” y “Passwords”.



The image shows a dialog box titled "Cambio de clave" (Change Password). It contains the following elements:

- A text input field labeled "Usuario" with the value "sa" entered.
- A text input field labeled "Digite su contraseña actual" (Enter your current password).
- A text input field labeled "Digite contraseña nueva" (Enter new password).
- A text input field labeled "Repita contraseña nueva" (Repeat new password).
- At the bottom, there are two buttons: one with a checkmark symbol (representing "Aceptar" or OK) and one with an "X" symbol (representing "Cancelar" or Cancel).

Ilustración 40: Cambio de Clave en el SQL Server.

Este diálogo es automáticamente generado por OpenSnap en todas las aplicaciones creadas con esa herramienta. Su funcionamiento es muy similar a la forma anteriormente descrita, cuya única diferencia estriba en que es posible definir a que usuario se le esta cambiando el Password. Los Botones con los símbolos “✓” y “✕” representan los botones de “Aceptar” y “Cancelar” respectivamente.

Opción: Cambiar de Contabilidad

Permite al usuario cambiar de la contabilidad actual a otra definida en el sistema. Como se explicó en el Proceso de Inicialización de la Aplicación COP, cada vez que se ejecuta el sistema, el usuario debe forzosamente seleccionar en que contabilidad va a trabajar. Sin embargo es posible el caso en el cual el usuario deba cambiar hacia otra contabilidad o “Empresa” por cualquier motivo. En este caso, la opción “Cambiar de Contabilidad” permite hacer esto. Sin embargo, el usuario podría cambiar a una contabilidad para la cual no esté autorizado a utilizar, por ello, cada vez que se selecciona esta opción, el sistema presenta nuevamente la forma de Login al Sistema COP, y realiza la autenticación del usuario. En el caso que ésta falle, el sistema regresa al estado anterior, es decir, no realiza el cambio de contabilidad.

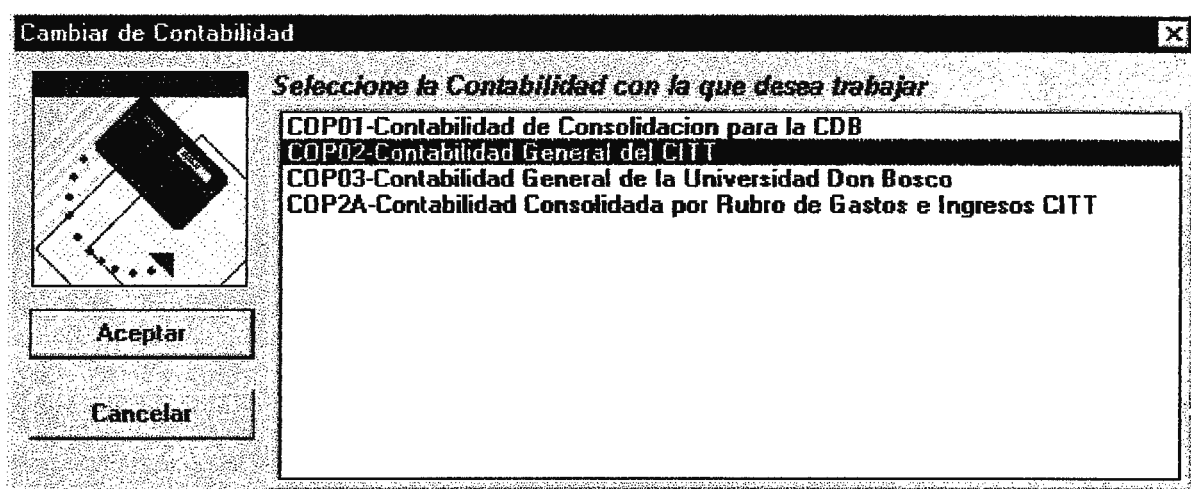


Ilustración 41: Forma que permite cambiar de Contabilidad o “Empresa” en la Aplicación COP.

La estructura de esta forma es bastante intuitiva, y sólo consta de 3 elementos: Una lista de selección, la cual muestra todas las Contabilidades definidas en el sistema, el botón Aceptar y Cancelar. Para realizar el cambio, se puede seleccionar una contabilidad de la lista haciendo un Click y luego presionando el botón aceptar o bien haciendo un Doble-Click directamente sobre la contabilidad deseada. En cualquier circunstancia, luego de seleccionar una contabilidad el sistema realizará el proceso de autenticación (Login) del usuario en el Sistema COP (no confundir con la autenticación en el SQL Server).

Opción: Programación de Actividades

Previamente se ha dado una breve descripción de como el sistema COP se integra con el Sistema Central de Repositorio CDBREP. Esta opción es parte de esa integración y una de las más interesantes. El objetivo principal de este mantenimiento, es adicionar, borrar o modificar actividades para el sistema y Contabilidad en uso. Cada actividad posee una descripción, un rango de fechas a realizarse, y opcionalmente una lista de pre-requisitos que se deben cumplir antes de poder dar como realizada dicha actividad.

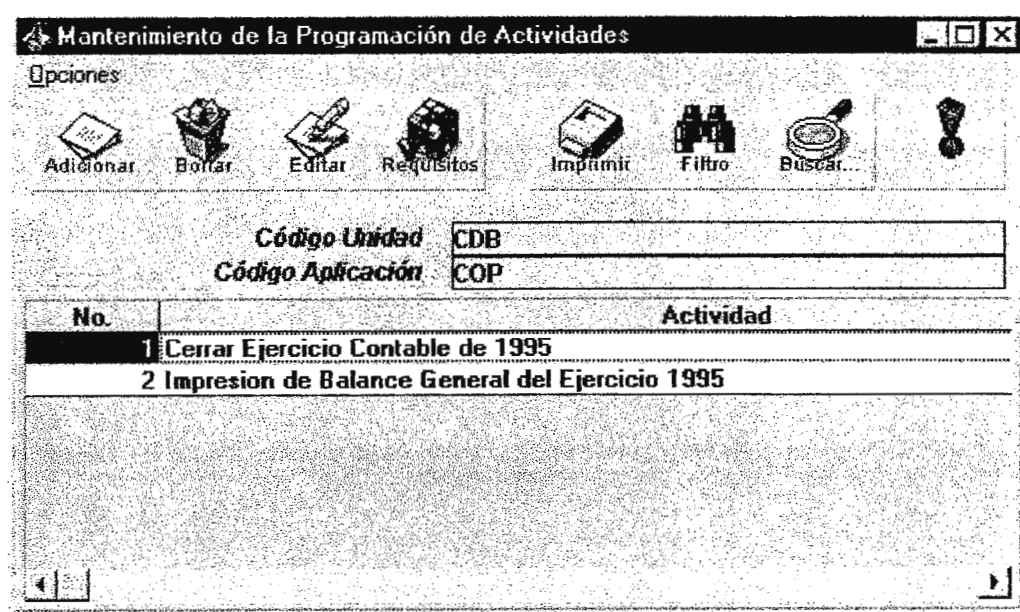


Ilustración 42: Mantenimiento de actividades programadas para la aplicación COP.

Muchas de las opciones que se muestran en los botones, son parte del diseño standard de los mantenimientos basados en formas de SELNAV y MANSAB, a excepción del botón: "Requisitos", el cual invoca a otro mantenimiento de Pre-requisitos para la actividad que ha sido señalada en la lista de actividades. Los elementos principales de este mantenimiento son los siguientes.

Adicionar Permite la asignación de una nueva actividad a realizarse en el futuro, y muestra una forma de MANSAB en la cual se digitan los atributos de la nueva actividad. Nótese que en la parte superior de la pantalla se muestra el código de la Entidad Institucional (Unidad) y de la Aplicación a la cual se le

está asignando la nueva actividad. El código de la Unidad dependerá de la Contabilidad seleccionada.

Borrar Permite borrar la actividad que se ha seleccionado en la ventana de mantenimiento. Esto será permitido por el sistema, si dicha actividad no posee ningún pre-requisito asignado.

Editar Permite Modificar los atributos o propiedades de la actividad seleccionada, para lo cual se invoca a la forma de Mantenimiento de Actividades, para que el usuario cambie dichos atributos.

Código de Unidad	CDB
Código de Aplicación	COP
No. Actividad	1
Fecha Mínima	Jan 1 1995 12:00AM
Fecha Máxima	Jan 31 1995 12:00AM
Descripción	Cerrar Ejercicio Contable de 1995
Opción del Menú	2.2
Macro	Informacion de como cerrar un periodo!
Usuario que realizo	
Fecha de Realización	

Editando

Ilustración 43: Pantalla (MANSAB) de Adición, Borrado o Modificación de una Actividad Programada para la Aplicación COP.

Los atributos de cada actividad programada en un sistema dado son los siguientes:

Código de Unidad Código de la Entidad Institucional a la que pertenece la Aplicación.

Código de Aplicación Código de la Aplicación a la cual se le esta encomendando la tarea.

No. Actividad	Número correlativo de la actividad, el cual, cada vez que se adiciona una nueva aparece de manera automática.
Fecha Mínima	Límite inferior del rango de tiempo destinado para realizar la actividad.
Fecha Máxima	Límite Superior del rango de tiempo destinado para realizar la actividad.
Descripción	Breve descripción de la tarea asignada.
Opción de Menú	Número de la opción del menú en la aplicación que el usuario debería utilizar con el fin de realizar la tarea.
Macro	Descripción detallada de los pasos a seguir para poder ejecutar la actividad exitosamente.
Usuario que la Realizó	Este campo muestra el código del usuario en el sistema que realizó la actividad. Aparece vacío cuando dicha actividad no ha sido aún completada.
Fecha de Realización	Muestra la fecha exacta en la cual se completó la actividad. Estos dos últimos campos no son editables por el usuario, sólo son informativos.

Tanto el Código de Unidad como el Código de Aplicación no pueden ser editados por el usuario, pues estos son determinados a partir de la aplicación que se esta corriendo (COP) y la contabilidad seleccionada (Cod. Unidad).

Cada actividad en el mantenimiento de actividades programadas puede poseer uno o más pre-requisitos. Estos a su vez no son más que otras actividades previamente definidas ya sean en el mismo sistema, o en otros que estén corriendo integradamente utilizando el Repositorio de Aplicaciones y Entidades Institucionales (CDBREP). De esta manera, se puede crear y organizar una red de actividades, las cuales deben ser realizadas en un orden específico por las diferentes Unidades y Aplicaciones de la Institución. Esto colabora a que las tareas se realicen de una manera ordenada y que la información fluya de acuerdo a los procesos administrativos establecidos en Ciudadela Don Bosco.

Requisitos: Esta opción del Mantenimiento de Actividades Programadas, permite al usuario asignar una o más actividades pre-requisito a la actividad que se ha señalado en el SELNAV. Al seleccionar esta opción, el usuario será capaz de adicionar y borrar los pre-requisitos de esa actividad.

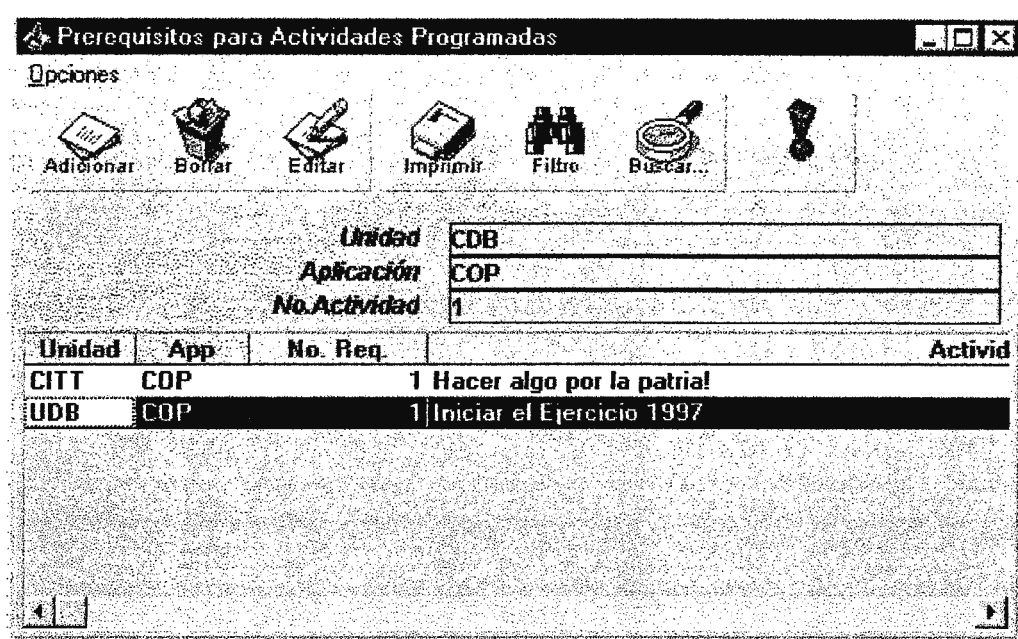


Ilustración 44: Forma de Mantenimiento de pre-requisitos que aparece cuando la opción “requisitos” es seleccionada en el Mantenimiento de Actividades Programadas.

El mantenimiento de pre-requisitos es una forma SELNAV standard, que posee todas las opciones de Adicionar, Borrar e Editar, sin embargo, esta última aparecerá deshabilitada, pues sólo es posible “asignar” o “desasignar” pre-requisitos. En esta forma se muestran todas las actividades que son pre-requisitos para realizar la actividad que fue seleccionada en el Mantenimiento de Actividades Programadas. Cada uno de los registros mostrados, pueden ser borrados a voluntad del usuario, al igual que nuevos pre-requisitos pueden ser asignados.

Mantenimiento de Requisitos para la programación

Código de Unidad: 015

Código de Aplicación: COP

No. Actividad: 1

Actividad Requisito

Codigo de la Unidad: UDB

Codigo de Aplicación: COP

No. de Actividad req: 5

Obligatorio

Cerrar el mes de febrero

Adicionando

Ilustración 45: Asignación de Requisitos.

Al seleccionar Adicionar en las opciones del Mantenimiento de Requisitos, se muestra un MANSAB, el cual permite al usuario definir que otra actividad programada será pre-requisito para la actividad seleccionada. Cada Actividad programada es identificada de manera única por los campos: Código de la Unidad, Código de la Aplicación y No. Correlativo de la Actividad, por lo que para definir la actividad que se convertirá en pre-requisito, es necesario llenar esos tres campos. Los elementos que componen esta forma son los siguientes:

- Código unidad:** Código de Unidad de la Actividad a la que se le esta asignando el pre-requisito.
- Código Aplicación** Código de la Aplicación, a cuya actividad se le esta asignando el pre-requisito
- No. Actividad** No. Correlativo de la actividad a la cual se le esta asignando el pre-requisito.

Actividad Pre-requisito

- Código unidad:** Código de Unidad de la Actividad que será pre-requisito
- Código Aplicación** Código de la Aplicación, cuya actividad será convertida en pre-requisito.

No. Actividad	No. Correlativo de la actividad pre-requisito.
Obligatorio	Indica si se debe realizar primero esta actividad pre-requisito para poder completar la actividad seleccionada.
Aceptar	Asigna el nuevo pre-requisito y se sale de la forma.
Cancelar	Se sale de la forma, sin hacer la asignación.

La actividad Pre-requisito, puede ser digitada en las respectivas cajas de texto de la forma, o puede ser seleccionada de la forma que aparece cuando se presionen los botones que se muestran junto a las cajas de texto. Esta forma es de tipo CONLIS, y muestra una lista de las posibles actividades que se le pueden asignar como pre-requisito.

Opción: Usuarios del Sistema

En un principio se hablo de que la aplicación posee dos niveles de seguridad: el impuesto por el Manejador de bases de datos y el otro implícito en la aplicación COP, a través de su estrecha integración con el Sistema de Repositorio de Aplicaciones y Entidades Institucionales. El segundo mecanismo de seguridad puede ser configurado por el usuario a través de esta opción, en la cual se definen los usuarios que tienen permiso para acceder la aplicación COP y a que opciones de ésta pueden acceder.

Los mecanismos de seguridad definidos por el sistema de repositorio CDBREP son sencillos: cada aplicación instalada posee un número indefinido de usuarios, cada uno con un identificador (Login) y una clave de acceso (password), además de otros datos necesarios para establecer los niveles de seguridad. Entre estos datos, se encuentra un campo que contiene en forma codificada las opciones de menú en la aplicación a las cuales el usuario tiene acceso, y también se define que tipo de acceso posee.

Login	Nombre
DBA	Administrador General de Aplicaciones

Ilustración 46: Forma para mantenimiento de Usuarios en el Sistema COP.

Al seleccionar la opción, aparecerá una forma tipo SELNAV, la cual muestra todos los usuarios definidos en el sistema, específicamente en la contabilidad seleccionada, ya sea en el proceso de inicialización del sistema o después de que el usuario selecciona la opción “Cambiar de Contabilidad” anteriormente definida. En la parte superior de la forma, justo debajo de la barra de botones, aparecen dos códigos, el de la unidad y el de la aplicación, la combinación de estos dos definen la “Aplicación Instalada”. Al igual que las formas standard antes definidas, los botones pueden ser utilizados para Adicionar, Borrar o Editar los datos de los usuarios definidos.

Mantenimiento de Usuarios de Sistemas

Código de Unidad: 008 Ciudadela Don Bosco

Código de Aplicación: 008 Sistema de Contabilidad General y Control Presupuestal

Login: DBA Clave de Acceso: 125704125042123023124

Nivel de Acceso: 1 Derechos de Acceso: [Empty]

Nombre del Usuario: Administrador General de Aplicaciones

Ultima fecha de acceso: May 19 1 307

Buttons: Password, Derechos

Status: Editando | Aceptar | Cancelar

Ilustración 47: Forma para Adicionar, Editar o Borrar Usuarios al Sistema.

Los elementos que componen esta forma son los siguientes:

- Código de Aplicación** Definen la aplicación a la cual se le esta añadiendo usuarios.
- Código Unidad** Definen junto con el código de aplicación, a que aplicación instalada se le están adicionando usuarios.
- Login** Identificador del Usuario, no mayor de 12 caracteres.
- Password** Muestra el password encriptado del usuario, si es que ya posee uno. Para asignar uno nuevo o cambiar el actual, se presiona el botón “Password”, y aparecerá una forma similar a la mostrada en la opción “Cambiar Password ... en el Sistema”.
- Nivel de Acceso** Definen el nivel de acceso que posee el usuario, usualmente entre 1 y 5, donde 1 representa el de más prioridad.
- Derechos de Acceso** Muestra una cadena de caracteres que representa los derechos de acceso del usuario sobre la aplicación. Para darle mantenimiento a estos derechos, es necesario presionar el botón adyacente “Derechos”. Sólo un usuario privilegiado (nivel acceso = 1) o el usuario “DBA” puede acceder a esta opción.
- Nombre del Usuario** Nombre completo del usuario.

Fecha Ultimo Acceso Muestra la última fecha en la cual el usuario entró al sistema. Este campo es mantenido automáticamente por el sistema, por tanto no es editable.

La opción “Derechos” en la forma anterior, al ser seleccionada muestra otra forma que le permite al usuario definir a que opciones de menú de la aplicación tiene derecho a entrar. Debido a que este mantenimiento fue diseñado pensando en que se utilizaría en todas las aplicaciones del “Proyecto Informático de Ciudadela Don Bosco”, cada opción es especificada por una cadena de dígitos separados por punto, y no por el nombre descriptivo de la opción de menú en la aplicación. Esto se debe a que cada aplicación posee su propia estructura de menú y difiere con respecto a las demás, por lo que se buscó una forma más abstracta y generalizada de hacer referencia a una opción de menú específica en una aplicación dada. Esta notación está representada por un número, de tal forma que representa la jerarquía de un menú desplegable: menú No. 1, opción No. 3, sub opción 2 de la aplicación COP, lo cual se representa como: 1.3.2.

Cada opción de menú a la cual se le dará permiso, se le debe de especificar que tipo de permiso. Debido a que la mayoría de opciones de menú de una aplicación llama a una mantenimiento, cuyas opciones por lo general son: Adicionar, Borrar y Consultar, estas opciones han sido incluidas como permisos. Aquellas opciones que impriman algún tipo de información, se les puede restringir su uso, a través del permiso ”Imprimir”. Si una opción de menú no aparece, eso es indicio de que el usuario no posee permiso alguno para ejecutar dicha opción.

Ejemplo: Supongamos que el usuario en cuestión, debe poseer permisos para entrar a la opción “Mantenimiento de Partidas” de la Aplicación COP, pero se requiere que sólo pueda ingresar nuevas partidas y modificar las existentes, mas no borrarlas. Para ello, se verifica que opción es la que queremos restringir; al ver la estructura de menú COP, encontramos que dicha opción es la 3.1, por tanto se escribirá “3.1” en la caja de texto que se lee “Opción del Menú”, luego se marcaran las siguientes cajas de opción: Ejecutar, Adicionar, Editar y todas las demás serán desmarcadas. Luego para hacer efectivo el derecho, se debe presionar el botón “Adicionar” y después el Botón Aceptar, para que la aplicación guarde los cambios en la Base de Datos del Repositorio Central de Aplicaciones y Entidades Institucionales. La

próxima vez que el usuario en cuestión haga “Login” a la aplicación, estos valores (Derechos de Acceso) serán leídos de la Base de Datos y se pondrán en efecto.

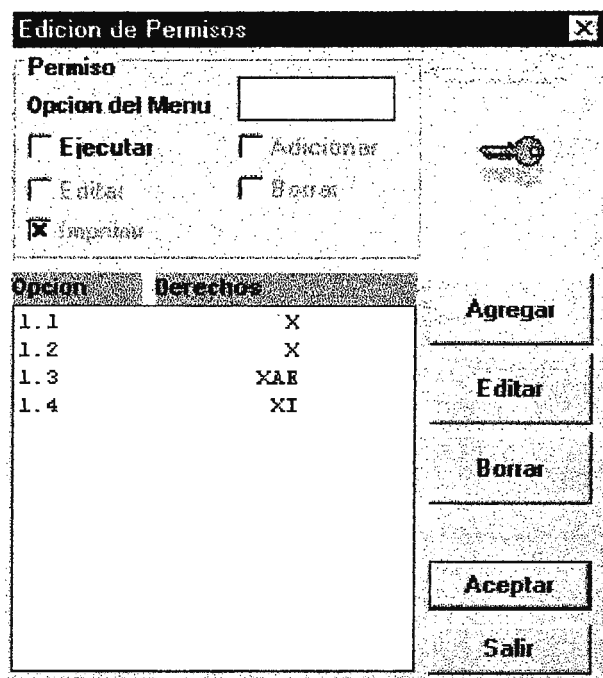


Ilustración 48: Mantenimiento de Derechos de Acceso para los Usuarios del Sistema COP.

Opción: Actividades Pendientes...

A diferencia de la opción “Mantenimiento de Actividades Programadas”, esta opción sólo muestra todas aquellas Actividades programadas que aun no han sido completadas, en el caso que no hubiera alguna, entonces sólo aparecerá un mensaje que indica que no hay actividades por realizar.

Actividades Programadas Pendientes		Pre-Requisitos Información	
Marque cuando haya realizado por completo esta actividad			
<input type="checkbox"/>	1	Cerrar Ejercicio Contable de 1995	
Opción Menú	2.2	Plazo para Realizarse	01-Jan-96 31-Jan-96
<input type="checkbox"/>	2	Impresion de Balance General del Ejercicio 1995	
Opción Menú	4.5	Plazo para Realizarse	01-Jan-96 31-Jan-96

Ilustración 49: Actividades Pendientes en el Sistema COP.

Cuando existen actividades pendientes, estas se muestran en la forma arriba expuesta, cada actividad presenta toda la información pertinente: No. Actividad, Descripción, Opción del Menú, Plazo para realizarse y dos botones: el primero, al ser presionado, muestra en una ventana una descripción breve de como debe realizarse la actividad, esa descripción es la que se especificó en el campo "Macro" en el mantenimiento de actividades programadas. El segundo botón muestra una lista de todas las actividades pre-requisito que han de completarse antes que se pueda cumplir la actividad en cuestión. En el extremo izquierdo de cada actividad se muestra una caja de selección (CheckBox) que permite al usuario definir cuando la actividad ha sido completada. En el caso que el usuario intente marcarla como "Actividad Realizada", el sistema verificará los siguientes requisitos:

- Que esté dentro del plazo de tiempo establecido
- Que todas las actividades pre-requisito hayan sido completadas

Si estos se cumplen, el sistema mostrará una ventana en donde se le pide al usuario que confirme si la actividad ya ha sido completada, pues el sistema guardará información adicional de quien realizó la actividad y en que fecha para efectos de posterior verificación.

Si el usuario confirma, entonces la actividad pasará a ser una "Actividad Realizada" y ya no aparecerá mas en la opción "Actividades Pendientes".

Opción: Salir

Al seleccionar esta opción, el sistema se desconectará de su sesión en la base de datos, borrará todos los archivos temporales que haya creado y cerrará la aplicación. Es altamente recomendado que los usuarios se salgan del sistema a través de esta opción, y no lo hagan a través del sistema operativo (Ctrl – Alt –Del), pues de esa manera se garantiza que no queden procesos “durmiendo” en el Servidor Sybase SQL Server, y que todas las transacciones de datos realizadas mientras la aplicación estuvo funcionando sean realizadas correctamente. Esta opción equivale a presionar la “X” en la esquina superior derecha de la forma principal de Menú de la aplicación COP.

Menú: Contabilidad

Este menú desplegable contiene todas las opciones básicas para el mantenimiento de los parámetros globales de cada contabilidad, desde la definición de contabilidades, pasando por el mantenimiento de catálogos de cuentas, hasta llegar a la definición de periodos contables y tipos de cuentas. Continuación se define cada opción con detalle.

Opción: Contabilidades

Este mantenimiento permite al usuario del sistema definir y dar mantenimiento a las diferentes contabilidades que el sistema maneja. Cada contabilidad posee un conjunto de información que definen sus parámetros globales. Cada contabilidad definida en el sistema COP debe poseer su respectiva “Aplicación Instalada” previamente definida en el sistema CDBREP. Esta opción muestra una forma SELNAV, con la cual se pueden adicionar, editar o borrar contabilidades. Cada contabilidad definida aparece en la lista con sus respectivos datos. Al seleccionar cualquiera de las opciones: Adicionar, Borrar, o Editar, aparecerá una forma de mantenimiento con todos los campos que definen a la contabilidad. Estos campos se definen a continuación:

Código Contabilidad: Cadena de 5 caracteres que identifica de manera única a cada Contabilidad definida en el sistema.

Cod. Unidad y Cod. Aplicación

Estos dos códigos indican a que “Aplicación Instalada” representa dicha contabilidad. Si se presionan los botones adyacentes la aplicación mostrará una forma de Selección tipo CONLIS que contiene todas las “Aplicaciones Instaladas” definidas en el Sistema de Repositorio CDBREP de donde el usuario puede elegir una.

Descripción de Contabilidad

Indica un nombre más descriptivo de la Contabilidad.

Administrador

Nombre que aparecerá en los reportes, como Administrador de la Unidad a la que pertenece esta Contabilidad.

Director

Nombre Completo del Director de la Unidad.

Cod.	Aplic.	Unidad	Descripción de la Contabilidad
COP01	COP	CDB	Contabilidad de Consolidacion para la CDB
COP02	COP	CITT	Contabilidad General del CITT
COP03	COP	UDB	Contabilidad General de la Universidad Don Bosco
COP2A	COP	CITT	Contabilidad Consolidada por Rubro de Gastos e Ingresos CITT

Ilustración 50: Mantenimiento de las Contabilidades Manejadas por el Sistema COP.

Mantenimiento de Catálogo de contabilidades

Código de Contabilidad Código de Aplicación Código de Unidad

Descripción Contabilidad

Administrador

Director

Contador

Auditor

Fecha de Inicio Nivel Cuentas de Mayor

Ejercicio Actual Período Contable Actual

Cuenta Pérdidas/Ganancias del Ejercicio

Desea Borrar este Registro?

Ilustración 51: Forma de Adición, Borrado o Edición de Contabilidades.

- Contador** Nombre completo del Contador que es responsable de esta contabilidad.
- Auditor** Nombre Completo del Auditor.
- Fecha de Inicio** Indica en que fecha fue iniciada o instalada dicha contabilidad.
- Nivel de Cuentas Mayor** Indica hasta que nivel dentro de la jerarquía del catálogo de cuentas es considerado como cuentas de Mayor.
- Ejercicio y Período Actual** Indican el año y período (puede ser mes) en el cual se encuentra la contabilidad. Estos valores van cambiando automáticamente a medida se van cerrando periodos contables, además le indican al sistema de que períodos contables es válido mayorizar partidas, entre otras verificaciones. El período especificado en estos campos tiene que ser previamente definido en el catálogo de períodos contables.

Cuenta de Perdidas y Ganancias del Ejercicio

Este campo indica qué cuenta del catálogo se utilizará para el proceso de cierre del ejercicio. La cuenta que se especifique en este campo deberá ser previamente definida en el catálogo de cuentas contables.

Aceptar

Guarda los cambios.

Cancelar

Cierra la forma sin guardar los cambios.

En el ámbito de seguridades, la aplicación COP sólo permitirá la adición de nuevas contabilidades a los usuarios que posean un nivel de acceso 1 o sea el usuario privilegiado DBA, por otro lado, no se puede borrar una contabilidad definida si ya existen datos relacionados a ésta, tales como: cuentas contables, movimientos, saldos, rubros presupuestales, etc.

Restricciones y Verificaciones

Al Adicionar una nueva Contabilidad.

- Que el código de unidad y el código de aplicación existan en conjunto en la tabla de Aplicaciones Instaladas del Repositorio de Aplicaciones.
- Que el Ejercicio y Período Actual existan en la tabla de periodos contables del sistema COP.
- Que la Cuenta de Pérdidas y Ganancias del Ejercicio especificada por el usuario exista en el Catálogo de cuentas de la contabilidad definida o que sea nula.
- Al Editar una Contabilidad.
- No se podrán editar los siguientes campos: Código de Contabilidad y el Ejercicio y Período Actual, este último podrá ser cambiado si aún no se ha introducido ninguna partida para esa contabilidad.
- Aplican todas las restricciones planteadas en el proceso de adicionar contabilidades arriba expuesto.

Al Borrar una Contabilidad.

- Se podrá borrar una Contabilidad definida, si se cumplen los siguientes requisitos:
- Que no existan datos relacionados a la contabilidad que se quiere borrar en las tablas de Partidas, Cuentas contables y rubros presupuestales.

Opción: Cuentas Contables (Jerárquico)

El sistema COP cuenta con dos opciones para poder dar mantenimiento al catálogo de cuentas contables de una Contabilidad dada, esta opción es la primera. Esta presenta una forma de mantenimiento de cuentas contables, cuya representación gráfica del catálogo de cuentas representa una jerarquía, en donde cada ítem o cuenta se subdivide en un número indefinido de cuentas o ítems, éstos a su vez pueden ser subdivididos, y así sucesivamente.

A diferencia de los demás mantenimientos basados en la forma standard de tipo SELNAV, esta forma presenta algunas variantes debido a la naturaleza de la información a la cual se le quiere dar mantenimiento (de tipo jerárquico). En la parte superior de la ventana se muestra una lista de selección tipo Combo (ComboBox) la cual muestra todas las contabilidades definidas en el sistema, siendo la contabilidad seleccionada por el usuario la que aparecerá seleccionada por omisión. Este elemento permitirá al usuario poder consultar y dar mantenimiento (si los derechos de acceso le permiten) a otras contabilidades diferentes a la actualmente seleccionada. Justo debajo de esta lista de selección, se encuentra la barra de botones tradicionales: Adicionar, Borrar, Editar, Imprimir, etc. Nótese que los botones de filtro y buscar han sido removidos a propósito, debido a que al establecer un patrón de búsqueda en las cuentas contables, podría darse el caso de que las cuentas contables resultantes no incluyan sus respectivas cuentas padre, por lo que sería imposible representar el resultado en una forma jerárquica.

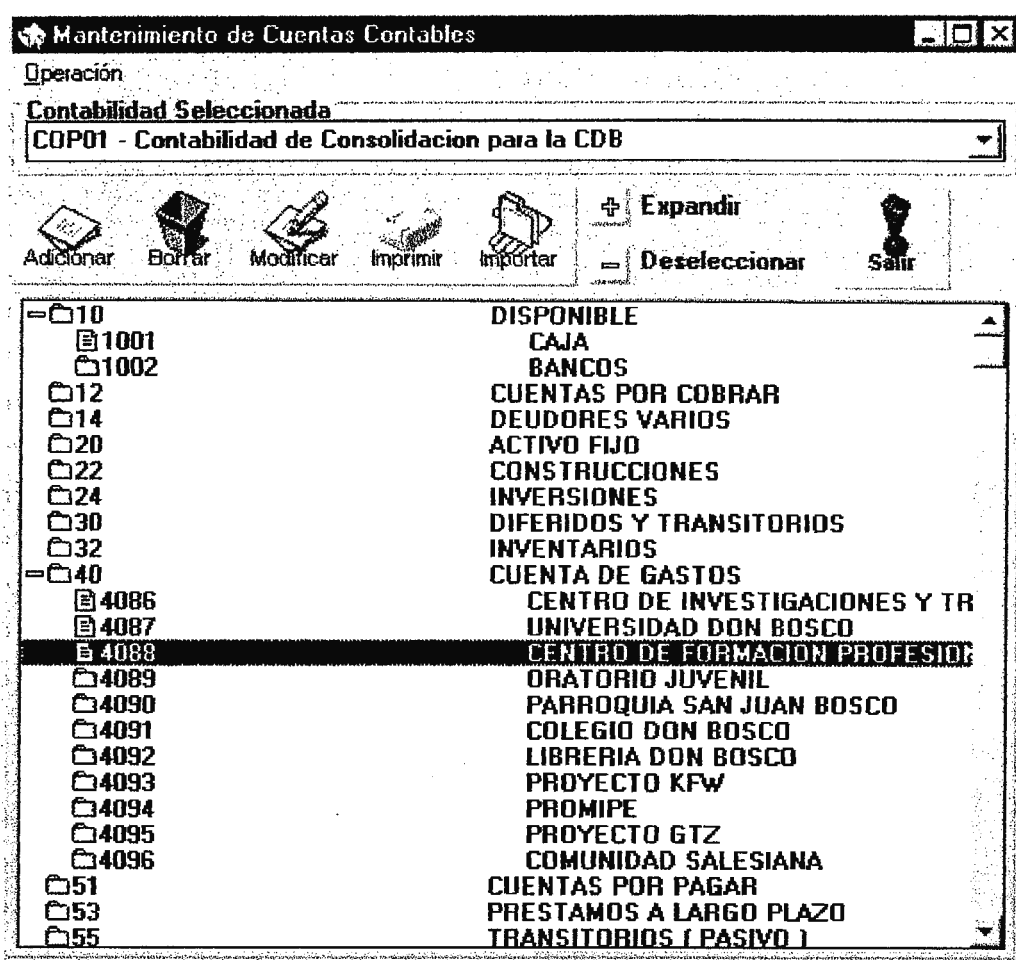


Ilustración 52: Mantenimiento del Catálogo de Cuentas de tipo Jerárquico.

Otra opción importante de este mantenimiento es “Importar”, la cual permite que el usuario copie el catálogo de cuentas de otra contabilidad distinta a la actual. Esto se ha incluido con el fin de facilitar la creación del catálogo de cuentas, pues de esta manera se podrá definir una contabilidad base, y luego sólo copiar sus cuentas contables a las demás contabilidades, para que estas puedan modificar dicho catálogo base a sus necesidades.

Esta forma, al ser cargada, sólo mostrará las cuentas de primer nivel. Cuando el usuario hace Click sobre una de esas cuentas el sistema consulta a la base de datos las cuentas que están directamente debajo de esa jerarquía. Si éstas existen, el símbolo “+” aparecerá a un lado de la cuenta seleccionada, indicando que existen mas cuentas. Luego el usuario podrá hacer Click en ese símbolo para expandir ese nodo y seguir explorando de una forma escalada el catálogo de Cuentas. Esto se hace debido a que Visual Basic impone restricciones de memoria en los controles utilizados, y cargar de una sola vez todo el catálogo de cuentas en el

navegador, podría causar que dichos límites se sobrepasaran. Por tanto la lectura de los datos se va haciendo a medida que el usuario lo necesita. Además, esto proporciona un método de navegación o consulta más rápido y efectivo, tomando en cuenta que los catálogos de cuenta suelen tener miles de registros y que esto demoraría bastante tiempo en ser transmitido a través de la red. El funcionamiento de la barra de botones difiere un poco del que se estipuló en las formas standard, por ello se definen a continuación:

- Adicionar** Permite adicionar una nueva cuenta debajo de la cuenta actualmente seleccionada, es decir una subcuenta de la cuenta señalada. En el caso de que la nueva cuenta que se quiere adicionar es de primer nivel, el usuario debe “deseleccionar” cualquier cuenta de la lista, lo cual es posible utilizando el botón “Deseleccionar” que se encuentra en la barra de botones. Esta opción invoca a una forma MANSAB de mantenimiento.
- Borrar** El botón borrar permite eliminar del catálogo la cuenta seleccionada. Si éste posee subcuentas o su saldo es diferente de 0 (cero) esta acción no será permitida. Esto quiere decir que sólo las cuentas de detalle podrán ser eliminadas, siempre y cuando se cumplan las restricciones de integridad más adelante expuestas.
- Editar** Permite modificar los datos de una cuenta contable, su funcionamiento no posee ninguna variación en especial.
- Imprimir** Permite imprimir el reporte del Catálogo de Cuentas, que también es accesible desde el menú Reportes.
- Expandir/Colapsar** Este botón posee cambia entre dos estados, Colapsar y Expandir, permite al usuario expandir todo el árbol jerárquico de cuentas para que todos los registros en la jerarquía sean visibles, o hacer lo contrario, esconderlas.
- Deseleccionar** Como su nombre lo indica, deselecciona cualquier cuenta que se encuentre señalada, con el fin de poder adicionar una nueva cuenta en el nivel 1.

Salir Permite salir del mantenimiento de cuentas y regresar al menú principal.

Restricciones / Verificaciones

- No se permite que un usuario con nivel mayor a 1 o que no sea el DBA pueda consultar o dar mantenimiento al catálogo de cuentas de otra contabilidad, usando la lista de selección desplegable en la parte superior de la ventana.
- No se puede borrar una cuenta que posea subcuentas.
- Sólo se pueden añadir cuentas al primer nivel de la jerarquía, si se deseleccionan las cuentas primero, de lo contrario el sistema asumirá que se quiere adicionar una subcuenta a la cuenta señalada.

Mantenimiento de Catálogo de cuentas

Cod. Contabilidad: MANSAB Nivel Jerárquico: 2

Código de Cuenta: 4088

Cod. Cuenta Padre: 40

Descripción de Cuenta: CENTRO DE INVESTIGACIONES Y TRANSFERENCIA DE TECNOLOGIA

Cod. Tipo de Cuenta: EGF Egresos

Consolidar esta Cuenta en:

Cod. Contab.	Cod. Cuenta

Es Detalle?: 1 - Si

Posee Dependientes?: 0 - No Posee

Editando

Aceptar Cancelar

Ilustración 53: Forma de Adición, Edición o Borrado de Cuentas Contables.

Después de seleccionar cualquiera de las opciones de Adicionar, Editar o Borrar una cuenta, la forma MANSAB para el mantenimiento de Cuentas Contables aparecerá en pantalla. Esta forma posee diferentes elementos y tiene diferentes comportamientos dependiendo del modo en el que se ha invocado (para adición, edición o borrado). En el caso más simple, cuando es invocada para borrar una cuenta, la forma se muestra, pero ninguno de

los campos es editable, y sólo se puede seleccionar el botón de “Aceptar” o “Cancelar”. En el caso de editar una cuenta, ciertos campos no son editables, tal es el caso del Tipo de Cuentas Contables. A continuación se definen en detalle los elementos de esta forma.

Cod. Contab.	Este campo define el Código de la Contabilidad en uso, no es editable en ningún momento por el usuario, pues el sistema sabe en cualquier momento, en que contabilidad se está trabajando.
Nivel Jerárquico	Este es un dato de control, que indica en que nivel dentro de la jerarquía del catálogo se encuentra la cuenta. No es editable por el usuario en ningún momento. En el caso de una adición de cuenta, este nivel es calculado automáticamente por la aplicación.
Código de Cuenta	Este es el indicador principal de la cuenta. Sólo se puede editar al adicionar una cuenta. Si ese es el caso, la aplicación automáticamente agregará el código de la cuenta “padre” de la nueva cuenta, al principio de su código, dejando al usuario digitar el resto del código, es decir, los dígitos faltantes.
Código de Cuenta Padre	Contiene el código de la Cuenta “Padre” o de consolidación de la cuenta actual. No es editable en ningún momento, pues sólo sirve para indicar la pertenencia de la cuenta.
Descripción de Cuenta	Contiene el concepto o descripción de la cuenta. Es editable en cualquier momento, excepto en el modo de borrado.
Código Tipo Cuenta	Indica el código que clasifica a dicha cuenta. Estos códigos son definidos en el sistema, a través de la opción: “Tipos de Cuentas”. El usuario puede digitar directamente el código (si lo conoce) o puede elegir uno de la lista que aparecerá si presiona el botón adyacente. Este campo es editable sólo al adicionar y siempre y cuando la cuenta que se esta adicionando sea de nivel 1, es decir, que pertenece al nivel raíz del árbol jerárquico. Esto se hace, debido a que una subcuenta siempre hereda el tipo de cuenta de su padre.

Consolidar Cuenta en...

Cod. Contab y Cod. Cuenta de Consolidación

Estos dos campos en conjunto definen a que cuenta contable (no dentro del mismo catálogo, sino otro) de otra contabilidad se debe de consolidar los saldos de la cuenta. Esto permite que se definan relaciones entre contabilidades distintas. Esta opción dentro del sistema constituye un mecanismo sencillo para lograr la consolidación de varias contabilidades, e incluso crear “redes” jerárquicas entre diferentes Contabilidades o empresas. Al presionar en cualquiera de los dos botones que aparecen junto a estos campos, el usuario tendrá acceso a una lista de todas las cuentas de detalle (o de último nivel en la jerarquía) que pertenecen a otras contabilidades definidas en el sistema. Se limita el contenido de la cuenta de consolidación a cuentas de detalle por el hecho de podrían crearse conflictos al consolidar dos cuentas en el mismo ramal.

Detalle

Esta lista de selección permite indicar al usuario, si la cuenta contable es de detalle (último nivel en la jerarquía) o no. Sin embargo, la base de datos se encargará de actualizar esta propiedad, cada vez que se modifique el catálogo de cuentas.

Dependencias

Esta lista de selección permite definir si la cuenta contable es una consolidadora de otras cuentas en otras contabilidades diferentes de la actual. La Base de datos se encarga automáticamente de mantener actualizada esta variable.

El mantenimiento del Catálogo de cuentas contables es la médula de la contabilidad, pues determina la estructura financiera de la unidad o empresa. Sin embargo, ningún sistema es capaz de ofrecer mecanismos de integridad de datos tales que nunca se cometa un error, ya que es el Contador quien utiliza su experiencia y conocimientos para crear una estructura adecuada a las necesidades de la institución. El mal diseño de un catálogo contable repercutirá

enormemente en el manejo de la contabilidad con o sin sistema. También es importante resaltar el mecanismo único de la aplicación, que permite relacionar una o más contabilidades entre sí, pues de manera bien utilizada, puede ofrecer muchas ventajas. Sin embargo de manera mal utilizada puede crear muchos dolores de cabeza, por lo que se recomienda, que el usuario dedique tiempo a entender a cabalidad los detalles de este mecanismo.

Restricciones y Verificaciones

- Si se adiciona una cuenta, el sistema no permitirá que se modifiquen los campos de: nivel jerárquico, cuenta padre, ni el prefijo del código de cuenta, debido a que son automáticamente llenados.
- No se permitirá que se modifique el código de tipo de cuenta, debido a que cualquier cuenta hereda el tipo o clasificación de su cuenta padre. Esto se aplica en todos los casos excepto cuando se adiciona una cuenta de primer nivel.
- Se verifica que la cuenta especificada en la sección: “cuenta de consolidación” exista en otro catálogo diferente del actual y que sea de detalle.

Opción: Catálogo de Cuentas

Este mantenimiento se ha desarrollado con la misma finalidad que el anterior, con la única diferencia que muestra las cuentas de una forma no jerárquica, sino que en una lista al igual que la mayoría de mantenimientos basados en el SELNAV normal. Debido a que este mantenimiento muestra todas las cuentas de un catálogo sin mostrar ninguna relación entre las cuentas (jerarquía) puede ser útil para consultar todo el catálogo de “un vistazo”. Todas sus funciones son prácticamente las mismas que el mantenimiento jerárquico de cuentas, con la excepción de todas aquellas funciones que son propias del mantenimiento jerárquico, tales como: botón de expandir/colapsar y Deseleccionar.

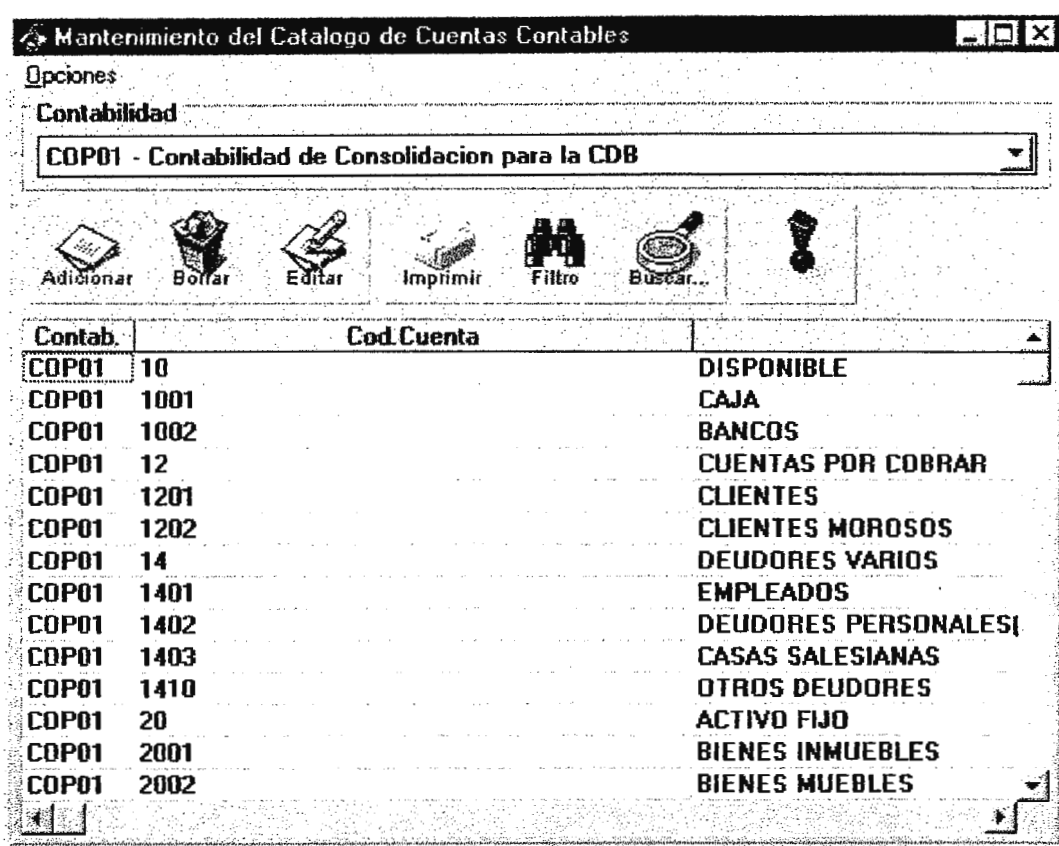


Ilustración 54: Mantenimiento Alternativo del Catálogo de Cuentas.

Opción: Períodos Contables

Esta opción del sistema, permite definir y dar mantenimiento a las tablas de periodos contables. Debido a que se ha desarrollado el sistema pensando en la flexibilidad y parametrización, se ha definido un registro de Períodos Contables, en donde el usuario puede definir cuantos periodos y que duración tiene cada uno de ellos en un ejercicio o año. Esta tabla contendrá la definición de cada período contable, los cuales registrarán todos los procesos de la contabilidad en todas las empresas o contabilidades definidas en el sistema. Esto se debe, a que es imposible poder consolidar varias contabilidades, si están no se rigen por el mismo esquema de tiempo o periodos contables. Por tanto, los cambios hechos a esta tabla afectan a todas las contabilidades, es por ello, que a pesar de ser un mantenimiento aparentemente sencillo y trivial, sus verificaciones son complejas.

Cada periodo contable en un ejercicio dado esta definido por dos números enteros: el año y el número que representa el período dentro de ese año, ya sea este, el mes, el número de la semana, el trimestre, etc. dependiendo de como se definan los periodos. En el caso particular de la Ciudadela Don Bosco, los periodos están definidos por meses, por lo que, dentro de un ejercicio específico la tabla contendrá 12 registros, cada uno representando a un mes del año.

Ejercicio	Periodo	Descripción
1995	12	Diciembre 1995
1996	1	Enero 1996
1996	2	Febrero 1996
1996	3	Marzo 1996
1996	4	Abril 1996
1996	5	Mayo 1996
1996	6	Junio 1996
1996	7	Julio 1996
1996	8	Agosto 1996
1996	9	Septiembre 1996
1996	10	Octubre 1996
1996	11	Noviembre 1996
1996	12	Diciembre 1996

Ilustración 55: Mantenimiento de Periodos Contables.

Al seleccionar el botón de adicionar, borrar o editar, el respectivo MANSAB es invocado para poder dar mantenimiento al catálogo de periodos.

Mantenimiento de Periodos contables

Ejercicio Contable (Año) 1996

Periodo Contable 1

Fecha de Inicio Jan 1 1996 12:00AM

Fecha Final Jan 31 1996 12:00AM

Descripción del Período Enero 1996

Cerrado 1 - Si

Editando Aceptar Cancelar

Ilustración 56: Forma para Adición, Edición o Borrado de Periodos Contables.

La forma MANSAB resultante muestra los siguientes elementos:

Ejercicio Contable	Número entero que identifica el año al que pertenece el período.
Período Contable	Número que identifica el período dentro del año, cuyo valor dependerá de como se subdivida el ejercicio.
Fecha de Inicio	Indica la fecha exacta en la cual comienza el período.
Fecha Final	Indica la fecha exacta en la cual termina el período.
Descripción del Período	Contiene una breve descripción que identifica de una manera más lógica al período.
Cerrado	Indica si el período ya pasó y ha sido cerrado en todas las contabilidades del sistema.

Restricciones y Verificaciones

- Que la fecha inicial y final estén dentro del año indicado en el ejercicio.
- Que el rango de fecha inicial y final no se traslape con otro período previamente definido en la tabla.
- Al intentar borrar un período, se verificará si no existe ningún dato asociado a este, es decir, partidas, saldos, saldo de presupuesto, etc., que hallan sido generados en el período que se quiere borrar.

- Estos periodos no podrán modificarse (su fecha de inicio y fin) si ya están asociados a algún dato en la base de datos, es decir partidas, saldos, etc. debido a que esto podría crear inconsistencias en ciertos procesos, tales como la mayorización, cierre de período o cierre de ejercicio.

Opción: Tipos de Cuentas

Cada cuenta contable dentro de un catálogo dado, está clasificado de cierta manera. Esta clasificación está representada por el “tipo de cuenta” asociado a cada una de ellas. Estos diferentes tipos de cuentas se definen en este mantenimiento, otra vez, a favor de la flexibilidad y parametrización.

Esto permite que el usuario clasifique las cuentas según sus necesidades, y no este restringido a un conjunto de tipos de cuenta predefinido en el sistema.

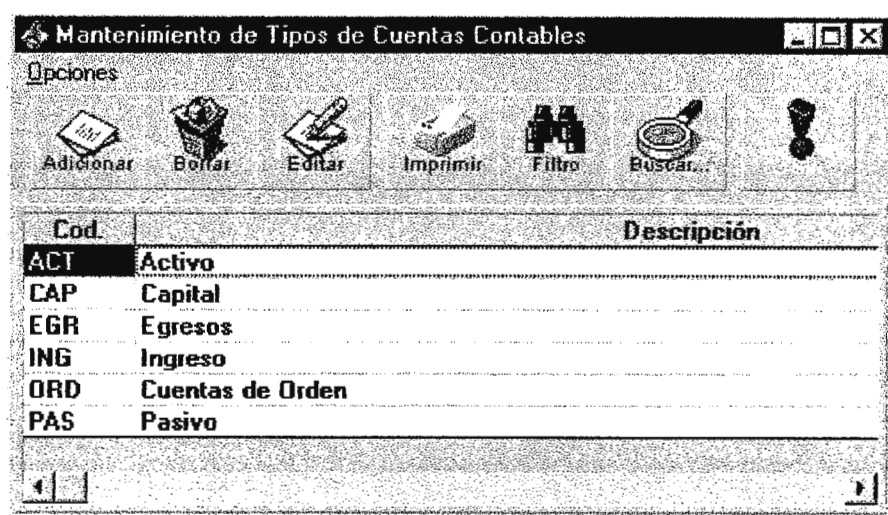


Ilustración 57: Forma para el mantenimiento de tipos de cuentas.

Cada tipo de cuenta es definido por un código alfanumérico y una descripción, además de ciertos parámetros que son editables en la forma MANSAB correspondiente. Al igual que el catálogo de periodos contables, esta es una tabla que contiene parámetros globales para todas las contabilidades del sistema, es decir que afectan a todas las cuentas de cada contabilidad. Al seleccionar las opciones de Adicionar, Editar o Borrar, la respectiva forma de MANSAB aparece en pantalla para permitir realizar la acción seleccionada.

Ilustración 58: Forma para la Adición, Edición o Borrado de Tipos de Cuentas Contables.

La forma MANSAB resultante contiene los siguientes elementos:

Código Tipo de Cuenta	Código alfanumérico (constituido para caracteres) que identifica al tipo de cuenta.
Descripción	Breve descripción para identificar de manera más lógica al tipo de cuenta.
Tipo de Saldo	Indicará como se debe calcular el saldo de una cuenta que este clasificada por este tipo de cuenta. Las opciones son: Crédito o Débito.
Tipo General de Cuenta	Existe una clasificación genérica de las cuentas que no es alterable y que se aplica a la mayoría de contabilidades, por lo que es necesario que el sistema sepa cuales tipos definidos por el usuario representan a los tipos genéricos contables. El tipo genérico al que representa se selecciona en este campo, cuyos posibles valores son: Activo, Pasivo, Capital, Ingresos y Egresos.

Restricciones y Verificaciones

- El código del tipo de cuenta no es modificable.
- No es posible borrar un tipo de cuenta, cuando esta ya ha sido asignada a alguna cuenta contable en cualquier contabilidad.

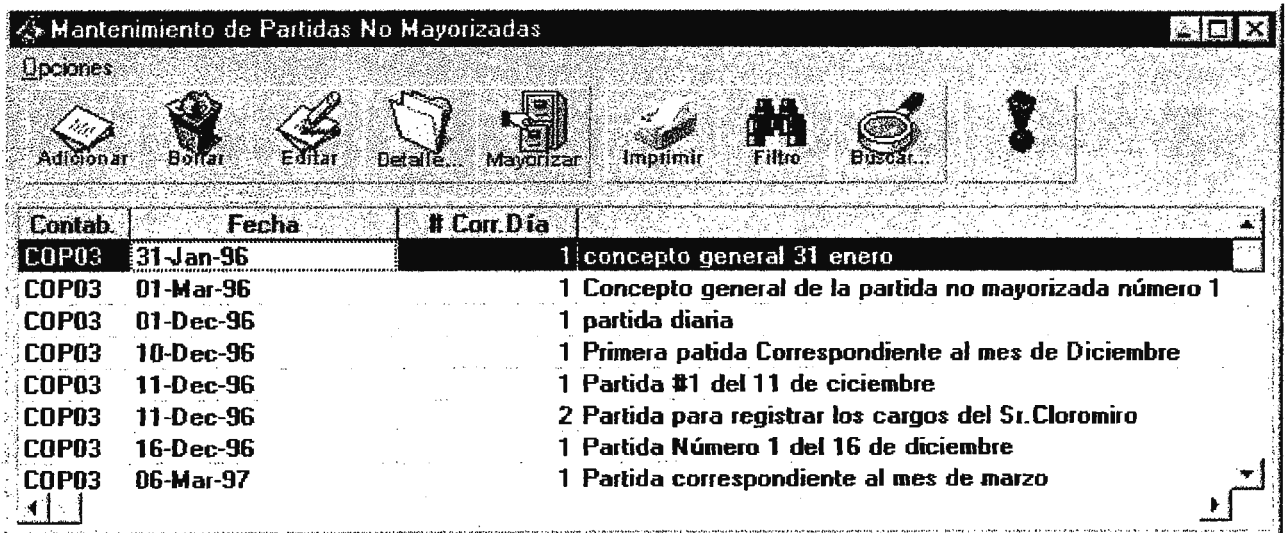
- No es posible editar el tipo de saldo o el tipo genérico de cuenta, cuando este ya fue asociado a una cuenta contable.

Menú: Partidas

El menú de partidas constituye uno de los componentes del sistema al cual se le dará uso mas frecuente, por tanto su estructura es sencilla, para que el usuario encuentre de forma fácil todos los elementos que le son necesarios para la tarea cotidiana de procesar partidas contables y mayorizarlas.

Opción: Mantenimiento de Partidas

El sistema COP cuenta con un mecanismo que permite al usuario dar mantenimiento sin restricciones a las partidas procesadas, con la finalidad de que puedan ser depuradas y verificadas antes de que afecten los saldos de cada cuenta (mayorización). Este mantenimiento representa un registro temporal de partidas que están por mayorizarse, mientras se encuentren en este registro, no se afecta ningún saldo o presupuesto, pero da la ventaja de poder depurar la información antes de que esto ocurra y además de poder generar reportes proforma o estimados de la situación financiera actual.



Contab.	Fecha	# Corr.	Día
COP03	31-Jan-96	1	concepto general 31 enero
COP03	01-Mar-96	1	Concepto general de la partida no mayorizada número 1
COP03	01-Dec-96	1	partida diaria
COP03	10-Dec-96	1	Primera patida Correspondiente al mes de Diciembre
COP03	11-Dec-96	1	Partida #1 del 11 de diciembre
COP03	11-Dec-96	2	Partida para registrar los cargos del Sr.Cloromiro
COP03	16-Dec-96	1	Partida Número 1 del 16 de diciembre
COP03	06-Mar-97	1	Partida correspondiente al mes de marzo

Ilustración 59: Mantenimiento general de partidas.

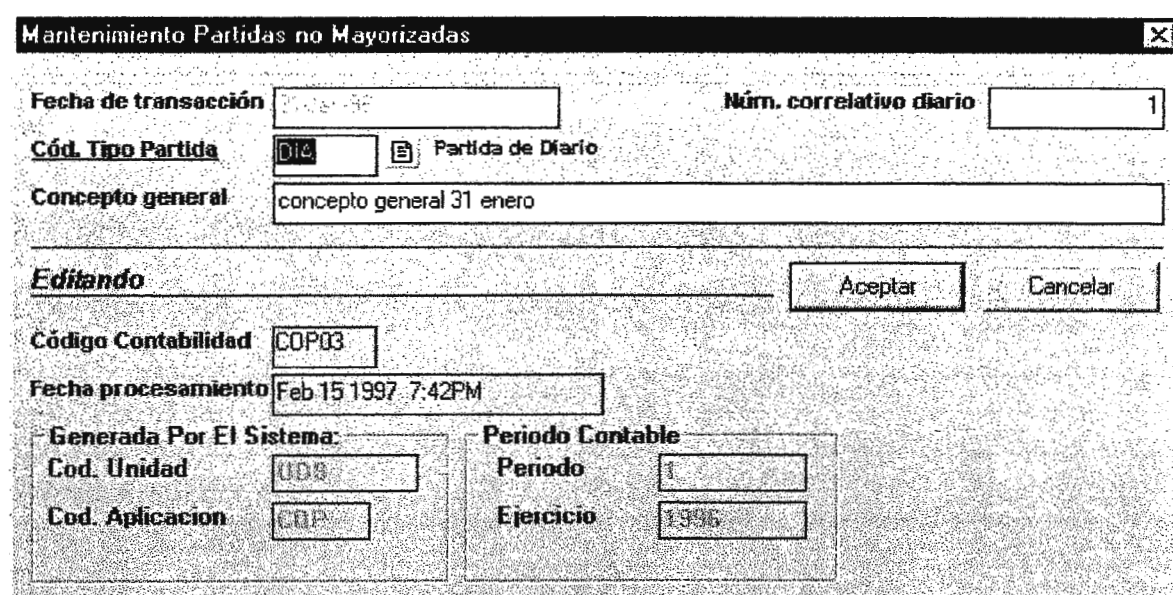
La opción “Mayorizar”, permite actualizar la partida seleccionada en los saldos de las cuentas contables, es decir, afectar los saldos con los movimientos que componen la partida. Esta opción ejecuta un procedimiento almacenado que pasa la partida con sus movimientos de las tablas temporales de partidas (cop_partenc_1 y cop_partdet_1) a las tablas de partidas mayorizadas (cop_partenc_2 y cop_partdet_2), lo cual consiste en borrar de las primeras y luego insertar los datos en las segundas, lo que hará que el Trigger de actualización de saldos se ejecute y actualice los saldos de las cuentas afectadas para el período indicado por la partida contable. Esta opción además permite que el usuario seleccione múltiples partidas para mayorizarlas por lotes, de esta manera se facilita la tarea de actualizar partidas. Antes de mayorizar las partidas seleccionadas, el sistema verificará que estas pertenezcan al período contable en proceso, debido a que es posible introducir partidas que son de periodos adelantados o posteriores al que se esta procesando por la contabilidad seleccionada.

Debido a que las partidas introducidas y aun no mayorizadas no poseen un número correlativo anual que las represente, pues no es posible saber que número se le asignará una vez mayorizada, este mantenimiento tienen su propio método para identificar las partidas procesadas, que es independiente de la numeración de partidas ya actualizadas. Cada una de las partidas introducidas al registro de partidas no mayorizadas, se identifica por la fecha a la que corresponde la transacción y el número correlativo de partida en esa fecha. Es decir, que la numeración se administra por fechas específicas, y se reinicia la cuenta para cada día. Este proceso de numeración es automáticamente generado por el sistema.

La forma SELNAV arriba presentada solamente presenta cada partida como su resumen o encabezado, es decir, no se ven los detalles o movimientos que la componen. Para ello se ha colocado la opción o botón: “Detalle”, el cual presenta otra forma SELNAV (y MANSAB) para poder dar mantenimiento a la partida que fue seleccionada.

Las opciones de Adicionar, Editar y Borrar tienen ciertas variaciones con respecto a las formas estándar de mantenimiento. Los botones de Editar y Borra funcionan tal y como se especifica en las formas estándar del sistema, a excepción, de que al seleccionar Editar, sólo es posible modificar los parámetros del Encabezado de la partida y no sus movimientos. Al seleccionar Borrar, se mostrará el encabezado de la partida para que el usuario confirme si desea borrarla o no, sin embargo, esto causará que se borre toda la partida, es decir su encabezado y todos los movimientos que la componen.

Adicionar es un proceso de 2 pasos, el primero, requiere que el usuario llene los datos en la forma MANSAB standard de l encabezado de la partida, y una vez terminado este paso, el sistema presentará el mantenimiento de movimientos o detalles de partidas, en donde el usuario adicionará, borrará y/o editará los movimientos que constituyen la partida que se esta adicionando.



Mantenimiento Partidas no Mayorizadas

Fecha de transacción: Núm. correlativo diario:

Cód. Tipo Partida:

Concepto general:

Editando

Código Contabilidad:

Fecha procesamiento:

Generada Por El Sistema:

Cod. Unidad: Periodo Contable:

Cod. Aplicacion: Periodo:

Ejercicio:

Ilustración 60: Mantenimiento del Encabezado de Partida.

Los elementos que componen el encabezado o “resumen” de la partida son los siguientes:

- Fecha de Transacción** Representa la fecha en que fue realizada la transacción de la partida. (no la fecha en que fue introducida al sistema)
- No. Correlativo Diario** Este número en conjunto con la fecha de transacción identifica de manera única a la partida, y es calculado automáticamente por el sistema, por tanto no es editable.
- Cod. Tipo de Partida** Código que identifica a que tipo pertenece la partida, esta clasificación por tipos de partida esta definida en un catálogo del sistema, el cual es definido por el usuario. Al presionar en el botón adyacente, un CONLIS aparecerá con todos los tipos de partidas posibles, para que el usuario seleccione una. La

descripción del tipo de partida aparece al lado derecho del campo en color azul.

Concepto General	Contiene una descripción general (resumen) de que transacción(es) representa la partida, es decir un concepto general.
Cod. Contabilidad	Indica a que contabilidad se le asocia esta partida, en todos los casos se selecciona automáticamente la contabilidad actual, seleccionada al iniciar el sistema. No es editable.
Fecha de Procesamiento	Contiene la fecha en que se introdujo al sistema la partida, cuando se esta adicionando, aparecerá la fecha actual, por tanto no es editable por el usuario.

Generada por:

Cod. Unidad	Indica el código de unidad institucional que genero la partida, en el caso de que se hubiera introducido a través del sistema COP, contendrá el código de la unidad a la cual pertenece la contabilidad actual, seleccionada al inicio de la aplicación.
Cod. Aplicación	Contiene el código de la aplicación que fue responsable de generar la partida, si fue introducida por medio del sistema, el código que aparecerá siempre será COP, de lo contrario, si la partida fue generada por otro sistema auxiliar integrado al sistema COP, el código de dicho sistema aparecerá en la casilla. Este campo en conjunto con el código de Unidad, indica que Aplicación instalada en el Repositorio de Aplicaciones fue la que genero la partida, esto se incluye para que el sistema sea abierto a futuros sistemas auxiliares de la contabilidad.

Período Contable:

Ejercicio	Indica a que ejercicio pertenece la transacción de la partida.
Período	Indica a que período pertenece la transacción de la partida, y es calculado en conjunto con el ejercicio de manera automática,

pues el sistema verifica a que período contable pertenece la fecha de transacción. Este dato servirá mas adelante para muchos procesos de generación de consultas y reportes, así como para la mayorización de partidas. De esta manera, el sistema puede verificar si el período al que pertenece la partida es valido y no esta cerrado.

Acceptar

Graba la información del encabezado de la partida que se ha especificado o en el caso de que se este borrando, borra toda la partida, es decir, encabezado y detalles. Cuando se esta adicionando, luego de haber guardado la información del encabezado, automáticamente aparecerá el mantenimiento de detalles de la partida, que se explica mas adelante.

Cancelar

Cancela la operación y regresa a la forma de mantenimiento de partidas.

Después de haber introducido los datos del encabezado de una nueva partida, el sistema mostrará el mantenimiento de detalles o movimientos de la partida, al cual es posible acceder a través de la opción “Detalles” que se muestra como un botón o en el menú de Opciones del mantenimiento de partidas. El Mantenimiento de detalles, muestra una forma standard SELNAV que posee ciertas verificaciones extras y que trabaja de acuerdo al contexto en el cual se le invocó. Este contexto puede ser cualquiera de los siguientes casos: el primero, cuando fue invocada automáticamente en el proceso de adición de partidas, y el segundo, cuando fue invocada explícitamente por el usuario para modificar los movimientos de una partida previamente introducida al sistema. En cualquiera de los dos casos, si el usuario borrar todos los movimientos e intenta salir del mantenimiento de detalles de partidas, el sistema le indicará que confirme esta decisión. En el caso de que la respuesta sea afirmativa, el sistema borrará el encabezado de la partida correspondiente, debido a que no es posible que exista una partida que no tenga al menos 2 movimientos. Por otro lado, si la suma de abonos y la de cargos de todos los movimientos adicionados a la partida, no son iguales, el mantenimiento no

permitirá que el usuario salga de este, hasta que ambas cantidades sean iguales o hasta que borre por completo todos los movimientos, y por ende toda la partida contable.

Mantenimiento de Movimientos en Partidas No Mayorizadas

Opciones

Adicionar, Borrar, Editar, Imprimir, Filtro, Buscar...

Contabilidad: COP03
 Fecha Transacción: 31-Jan-96
 # Corr.Diario: 1

No.	Código de Cuenta	Cargos Cuenta	Abonos Cuenta	
1	10-01-02-001	1000.00	.00	cargo de 1000 a la cuenta 10-01-02-001
2	10-01-02-002	.00	1000.00	Abono de 1000 a la cuenta 10-01-02-002

Cargos: 1,000.00 Abonos: 1,000.00

Ilustración 61: Mantenimiento de los Movimientos de una Partida no Mayorizada aun.

A diferencia de los SELNAV convencionales, este muestra en la parte inferior el total de abonos y cargos, los cuales se actualizan a medida se adición o quitan movimientos. Estos datos le servirán al sistema y al usuario para verificar que la partida contable esta "Cuadrada". Por otro lado, en la parte superior de la ventana, justo debajo de la barra de botones, se puede ver los códigos que identifican la partida a la cual se le esta dando mantenimiento. Al presionar cualquiera de los botones de Adicionar, Borrar o Editar aparecerá una forma MANSAB que le permitirá al usuario dar mantenimiento a un movimiento específico.

Mantenimiento de Movimientos de Partidas No Mayorizadas

No. Correlativo: 1

Cod. de Cuenta: 10-01-02-001

Descripción: Administración UDB

Monto del Movimiento: 1000.00

Descripción del Movimiento: cargo de 1000 a la cuenta 10-01-02-001

Tipo de Movimiento:
 Cargo
 Abono

Editar: Aceptar Cancelar

Ilustración 62: Mantenimiento de un Movimiento Contable. (Detalle de una Partida).

Los elementos que conforman el mantenimiento de movimientos de partidas contables son los siguientes:

No. Correlativo	Indica el número correlativo del movimiento dentro de la partida contable, y es calculado automáticamente por el sistema, aunque es posible modificarlo.
Cod. Cuenta	Código de la cuenta contable a la que afecta el movimiento, esta cuenta contable debe ser de la contabilidad que genero la partida y además debe ser de detalle. Si el usuario no conoce el código de la cuenta, puede seleccionar el botón que aparece justo a la derecha y arriba del campo, el cual invocará una forma CONLIS que contiene todas las cuentas de detalle posibles en el catálogo, para que el usuario seleccione una.
Descripción	Muestra el concepto de la cuenta contable arriba seleccionada, y sólo sirve para informar al usuario, por tanto no es editable.
Monto del Movimiento	Cantidad monetaria que se abonará o cargará a la cuenta especificada en el período definido en la partida.
Tipo de Movimiento	Presenta un “Radio Button”, el cual permite seleccionar cualquiera de las dos opciones: Cargo o Abono. Esto indica si la cantidad especificada en el campo anterior, será un Cargo o un Abono a la cuenta contable.
Descripción del Movimiento	Permite especificar al usuario un concepto del movimiento que le indique a que se debe el cargo o el abono.
Aceptar	Graba el movimiento a la partida y regresa al mantenimiento de movimientos. En el caso de que se este borrando, el movimiento será eliminado de la partida.
Cancelar	Cierra la forma sin realizar ningún cambio.

Restricciones y Verificaciones:

- La cuenta que se afecta con el movimiento debe ser de detalle
- El número correlativo del movimiento, debe ser único dentro de la partida.

- El concepto del movimiento no puede quedar vacío o “nulo”.
- No se podrá salir del mantenimiento de movimientos si la suma de cargo y abonos no coincide.
- Si el usuario intenta salir del mantenimiento cuando no hubiere ningún movimiento registrado, el sistema preguntará si quiere borrar la partida (pues no tiene ningún movimiento). En el caso de responder afirmativamente, el sistema borrará la partida y regresará al mantenimiento de partidas, en caso contrario, el usuario permanecerá en el mantenimiento de movimientos.

Opción: Tipos de Partidas

Al igual que el catálogo de tipos de cuentas, el sistema contempla la clasificación de partidas por tipos, lo que da la flexibilidad de definir tipos de partidas según las necesidades del usuario para luego generar reportes o consultas basándose en esta clasificación.

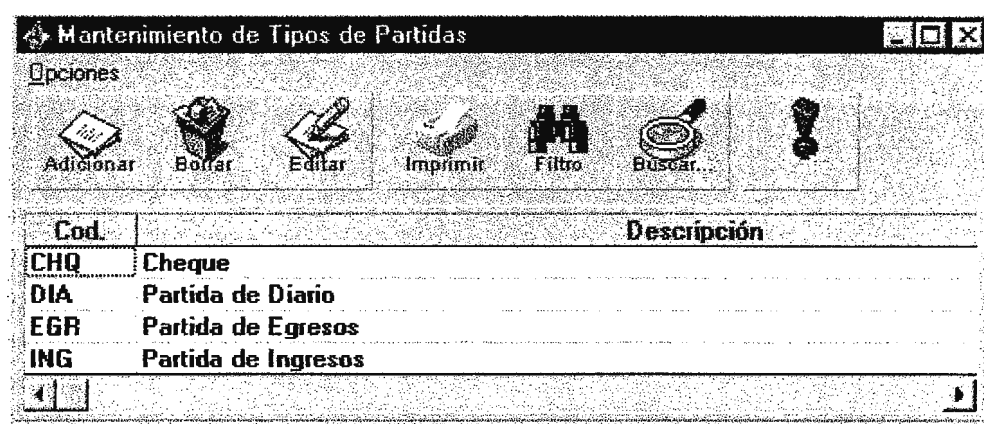


Ilustración 63: Mantenimiento de tipos de partidas.

Los tipos de partidas definidas por el usuario afectaran a todas las contabilidades, de tal manera que todas ellas posean un standard de clasificación de partidas. Las opciones de este mantenimiento se apegan a las formas standard de SELNAV y no poseen ninguna opción fuera de ese esquema. Al seleccionar cualquiera de las opciones de Adicionar, Editar o Borra parecerá la forma MANSAB correspondiente.

The image shows a software dialog box titled "Mantenimiento de Tipos de partidas". It has a standard Windows-style title bar with a close button (X) in the top right corner. Below the title bar, there are two input fields: "Código Tipo de Partida" with an empty text box, and "Descripción del Tipo" with a text box containing the text "Partida de Erent". At the bottom of the dialog, there is a status bar that says "Editando" and two buttons: "Aceptar" and "Cancelar".

Ilustración 64: Forma para la Adición, Edición o Borrado de Tipos de partidas.

Actualmente, el mantenimiento de tipos de partidas cuenta con los siguientes elementos:

Cod. Tipo de Partida	Contiene el código que identifica al tipo de partida, el cual es alfanumérico para facilitar su uso.
Descripción del Tipo	Contiene una breve descripción que define a este tipo de partida.
Aceptar	Guarda los cambios en el modo de adición o edición, y borra el tipo de partida si esta en modo de borrado. Nótese que sólo es posible borrar el tipo de partida si no esta asociado a ninguna partida en cualquier contabilidad.
Cancelar	Cancela la acción y regresa a la forma que lo invocó.

Restricciones y Verificaciones

- Al intentar borrar un tipo de partida, se verificará que no exista ninguna partida contable clasificada por este tipo.
- No se permite modificar el código de tipo de partida.

Menú: Consultas

Este menú posee un conjunto de opciones que invocan a las diferentes formas de consulta del sistema. Muchas de las consultas que dependen de un dato de entrada (ejemplo: Saldos por período para la Cuenta <cualequier cuenta>) están implícitas dentro de los mantenimientos del sistema, y son accesibles seleccionando el registro en una forma de SELNAV y presionando el botón derecho del MOUSE, o seleccionando el menú: "Opciones" que parece en la parte superior de cada forma. El menú resultante desplegará todas las opciones de consultas disponibles para el ítem seleccionado. Para el ejemplo anterior, se

puede seleccionar una cuenta en el mantenimiento del Catálogo de cuentas, y luego al presionar el botón derecho del Mouse aparecerá un menú, el cual a su vez contiene la opción: “Consultas” de donde el usuario puede seleccionar “Saldos por Períodos”. Esta acción provocará que se muestre una forma con los saldos de la cuenta seleccionada para todos los periodos definidos en el sistema.

Sin embargo, como una opción extra, el sistema incluye todas aquellas consultas que son generales o que no dependen de un parámetro de entrada en el menú general: “Consultas”.

Opción: Partidas Mayorizadas

Esta opción presenta una forma de consulta similar al SELNAV, pero sin las opciones de Mantenimiento (Adicionar, Borrar y Editar) para que el usuario pueda ver e imprimir los datos presentados. Esta opción en particular, presentará todas las partidas que ya fueron mayorizadas para el ejercicio en proceso y para la contabilidad seleccionada.

Opción: Partidas Históricas.

Esta opción presenta un SELNAV de Consulta similar al que se detalla en la Opción de Mantenimiento de Partidas, con la única diferencia que no se muestran los botones de Adicionar, Borrar, o Editar.

Esta consulta permite visualizar las partidas que fueron procesadas en algún ejercicio anterior al actual.

Opción: Catálogo de Cuentas Histórico

Esta opción del sistema muestra una forma similar a la del Mantenimiento de Cuentas Contables Jerárquico, con la única diferencia que los botones de adicionar, editar o borrar no son visibles. El objetivo de esta consulta es ver los catálogos de cuentas contables de la contabilidad seleccionada en ejercicios anteriores al actual, de tal manera que se pueda visualizar todas aquellas cuentas que a través del tiempo han sido modificadas o puestas fuera de uso.

Opción: Saldos de Periodos Históricos

A través de esta opción, el usuario podrá consultar los saldos por periodo de cada una de las cuentas contables que estuvieran activas en ejercicios pasados al actual.

Opción: Catálogo de Presupuesto Histórico

Esta opción, invoca a una forma similar a la que se muestra en el Mantenimiento de Rubros Presupuestales, con la única diferencia, que no están visibles los botones de Adicionar, Borrar o Editar, de tal manera que solo sea utilizable como una consulta. El objetivo, es mostrar como estaba estructurado el catalogo de rubros de presupuesto para la contabilidad seleccionada en ejercicios anteriores al que esta en curso.

Opción: Saldos Presupuestales Históricos

A través de esta opción, el usuario puede acceder a una forma de consulta similar a un SELNAV, para poder ver los montos presupuestados y los montos reales de cada rubro de presupuesto en cada periodo de ejercicios pasados.

Menú: Reportes

El Sistema Contable cuenta con varias opciones para elaborar los distintos reportes y Estados Financieros necesarios para llevar un adecuado control de las actividades financieras de la Institución Salesiana.

Todos y cada uno de estos reportes han sido creados utilizando la herramienta Crystal Reports, por lo que su método de Generación e Impresión es bastante parecido en todos los casos. Cuando se solicita uno de estos reportes, primero se crean y se visualizan en una pantalla, a fin de hacer una revisión preliminar de la información completa brindada. Luego puede procederse a imprimir el reporte, en el caso que se requiera de una copia impresa.

Los reportes que el Sistema Contable genera son los siguientes :

- Catálogo de Cuentas Impreso
- Libro Auxiliar (utilizando sólo Partidas Mayorizadas)
- Libro Auxiliar (utilizando sólo Partidas No Mayorizadas)
- Libro Mayor
- Balance de Comprobación
- Estado de Resultados
- Balance General
- Reporte Comparativo de Saldos de Períodos Cerrados
- Reportes Consolidados
- Balance General Consolidado
- Estado de Resultados Consolidado
- Estados Financieros Históricos
- Balance General Histórico
- Estado de Resultados Histórico

Los Reportes del Sistema Presupuestal son :

- Impresión del Catálogo de Rubros de Presupuesto

- Listado de Tipos de Rubro
- Listado de Montos Presupuestados
- Listado de las Cuentas asignadas a los Rubros

En el caso de los Reportes Contables el Sistema permite especificar qué datos específicos se desea imprimir, dando la posibilidad de seleccionar un rango de datos a incluir en el reporte. Para ello se utiliza una ventana especial para cada uno en la cual se especificarán dichos rangos.

Los Reportes Presupuestales no necesitan delimitar el rango de datos a imprimir, por lo que sólo utilizan una ventana donde se indica el destino de la impresión : A pantalla, al impresor, o a un archivo de texto cuyo nombre especificará el usuario.

A continuación serán explicados los dos procesos básicos de generación de reportes :

El proceso de generación para los reportes contables es básicamente el mismo :

1. Se elige el Reporte a emitir en el sub-menú de Reportes Contables.
2. Aparece una ventana en la cual deberá especificar qué datos desea incluir en el reporte. El filtrado de los datos se hará especificando ya sea rangos de fechas, rangos de Cuentas, o los tipos de Partidas Contables que desea incluir en el reporte, por ejemplo.
3. Luego de seleccionar el filtro de los datos a incluir, se presiona el botón “Aceptar” y el Sistema leerá los datos necesarios en la Base a fin de crear el reporte.
4. El reporte aparecerá en pantalla dentro de una ventana especial. Podrá aquí visualizarse toda la información contenida en el reporte, a fin de tener una vista previa de la información a ser impresa. La ventana informa asimismo del número de páginas que componen el informe generado, y el número de registros o filas utilizados.
5. En caso que se desee enviar el informe al impresor, la ventana cuenta con un botón específico para ello (con el icono de un impresor). Al presionarlo aparece un pequeño cuadro de diálogo, en el cual se puede especificar el rango de páginas a imprimir.

Para los reportes presupuestales el proceso es el siguiente:

1. Se elige el Reporte a emitir en el sub-menú de Reportes Presupuestales.

2. Aparece una ventana en la cual únicamente deberá especificar el destino de la impresión: A pantalla, al impresor, o a un archivo de texto cuyo nombre y directorio especificará.
3. Luego de seleccionar el destino, se presiona el botón "Aceptar" y el Sistema leerá los datos necesarios en la Base a fin de crear el reporte. El reporte será enviado ya sea a pantalla, utilizando la misma ventana que se explicó utilizan los reportes contables, o directamente a la impresora predeterminada en Windows, o al archivo de texto especificado.

COP: Sistema para el Control de la Contabilidad y el Presupuesto

Listado de Periodos Contables

ejercicio	periodo	Fecha Inicio	Fecha Fin	Nombre del Periodo
1,995	12	1/12/1995	31/12/1995	Diciembre 1995
1,996	1	1/01/1996	31/01/1996	Enero 1996
1,996	2	1/02/1996	28/02/1996	Febrero 1996
1,996	3	1/03/1996	31/03/1996	Marzo 1996
1,996	4	1/04/1996	30/04/1996	Abril 1996
1,996	5	1/05/1996	31/05/1996	Mayo 1996
1,996	6	1/06/1996	30/06/1996	Junio 1996
1,996	7	1/07/1996	31/07/1996	Julio 1996
1,996	8	1/08/1996	31/08/1996	Agosto 1996
1,996	9	1/09/1996	30/09/1996	Septiembre 1996
1,996	10	1/10/1996	31/10/1996	Octubre 1996

Ilustración 65: Ventana utilizada para mostrar la "Vista Preliminar" de un reporte generado en Crystal Reports.

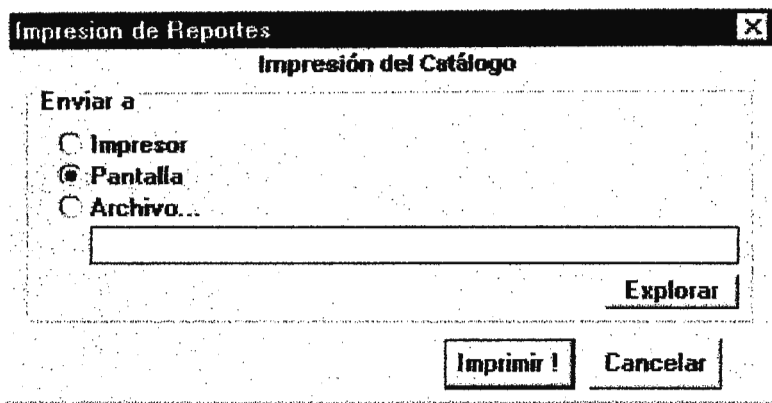


Ilustración 66: Ventana utilizada para especificar el destino de la impresión.

A continuación serán explicados de manera más detallada los pasos específicos a seguir para cada uno de los reportes contables :

Opción: Catálogo de Cuentas

A fin de imprimir el Catálogo de Cuentas únicamente hay que especificar en una pequeña ventana la Contabilidad cuyo Catálogo se desea imprimir. Esto será válido únicamente para aquellos usuarios con autoridad suficiente para acceder varias Contabilidades. Para la mayoría de usuarios, sin embargo, aparecerá únicamente la Contabilidad en la cual estén trabajando en ese momento, y no podrán seleccionar ninguna otra.

Puede también especificarse un Ejercicio en especial, ya que también pueden imprimirse los Catálogos Históricos existentes, o sea, los de los Ejercicios anteriores al actual (que hayan sido ingresados al Sistema). La casilla de Período está deshabilitada para este reporte, por lo que aparece una etiqueta con la frase “Todos los Períodos”. Esta casilla es utilizada para otros reportes.

Al presionar el botón de Aceptar, se mostrará en pantalla el Catálogo seleccionado, con la opción de imprimirlo posteriormente.

Ilustración 67: Ventana de Selección de Catálogo de Cuentas a imprimir.

Opción: Libro Auxiliar

El Libro Auxiliar muestra de forma detallada los distintos movimientos que una cuenta contable ha tenido en un período dado.

Se cuenta en el Sistema con dos tipos de Libro Auxiliar : uno que utiliza sólo Partidas No Mayorizadas, y el otro que utiliza sólo Partidas Mayorizadas. Para ambos el proceso de generación es el mismo :

1. Se selecciona primero cual de los dos tipos se desea imprimir.
2. Aparecerá una ventana en la cual se especificará qué datos se desea incluir en el reporte. Se define para ello un Rango de Cuentas Contables y un rango de Partidas. Aparecerá inicialmente los rangos totales para ambos casos : la primera y la última de las Cuentas en el Catálogo, y la fecha de la primera partida y la de la última del tipo seleccionado (Mayorizadas o No Mayorizadas). Podrá entonces especificar el rango que desee dentro de estos límites.

LIBRO AUXILIAR DE CUENTAS (Partidas No Mayorizadas)

Rango de cuentas a imprimir:

Desde Cuenta: 10-01-001
Primera cuenta de detalle del Catalogo

Hasta Cuenta: 80-02-001-003
Ultima cuenta de detalle en el Catalogo

Rango de Partidas:

Fecha Partida 1: 01-Feb-1997

Fecha Partida 2: 15-Mar-1997

Imprimir **Cancelar**

Ilustración 68: Ventana de Filtro de Datos a incluir en el Libro Auxiliar.

Opción: Libro Mayor

El Libro Mayor muestra el Saldo de todas las Cuentas para un período determinado, así como el Saldo Acumulado de los períodos anteriores y el Saldo total (Saldo anterior más el solicitado) por cada una de ellas.

En la ventana de filtro de datos se define :

- Rango de Partidas, especificando el rango de fechas en el cual se encuentran las partidas a incluir en el reporte.
- Si se desea incluir en los Saldos los valores correspondientes a los movimientos de las partidas aún no mayorizadas que se encuentren dentro del rango de fechas especificado. Por omisión se tomarán los valores sólo de las Partidas Mayorizadas.
- El nivel máximo de las Cuentas a incluir en el reporte. Se imprimirán únicamente los saldos de las Cuentas desde el nivel 1 hasta el nivel especificado en esta caja de texto.

LIBRO MAYOR

Rango de Partidas:

Fecha Partida 1:
01 Feb 1997

Fecha Partida 2:
27 Jun 1997

Incluir:

Partidas No Mayorizadas

Nivel <= 3

Imprimir

Cancelar

Ilustración 69: Ventana de Filtro de Datos a incluir en el Libro Mayor.

Opción: Balance de Comprobación

A fin de general el Balance de Comprobación, al especificársele un rango de fechas, el Sistema calculará los Saldos de los movimientos de las Cuentas contables entre esas fechas, así como los Saldos Anteriores al período especificado, es decir, del 1 de Enero hasta el día anterior a la fecha menor del rango.

En una ventana muy similar a la del Libro Auxiliar, para generar el Balance de Comprobación se solicita únicamente el rango de fechas para el cual se desea calcular los saldos, y si se desea incluir también las Partidas No Mayorizadas. Se especifican también qué Cuentas del Catálogo se desea utilizar, definiendo el rango de Cuentas correspondiente.

BALANCE DE COMPROBACION	
Rango de cuentas a imprimir:	
Desde Cuenta:	10-01-001 Primera cuenta de detalle del Catalogo
Hasta Cuenta:	80-02-001-003 Ultima cuenta de detalle en el Catalogo
Rango de Partidas:	
Fecha Partida 1:	01-Feb-1997
Fecha Partida 2:	15-Mar-1997
Incluir:	<input checked="" type="checkbox"/> Partidas No Mayorizadas
Imprimir	Cancelar

Ilustración 70: Filtros del Balance de Comprobación.

Opción: Balance General y Estado de Resultados.

El Balance General muestra la situación financiera de una empresa en un momento dado, mostrando los saldos de sus Cuentas de Activo, Pasivo y Capital.

El Estado de Resultados muestra los saldos de las Cuentas de Ingresos y Gastos de la empresa en un período dado, con lo cual se puede calcular la Ganancia o Pérdida de la Empresa en dicho período.

El formato utilizado en el Sistema para ambos Estados Financieros es el formato de Reporte.

Ambos reportes pueden ser impresos utilizando las Partidas Mayorizadas a cualquier fecha dentro del Ejercicio Actual. Puede utilizarse adicionalmente las Partidas No Mayorizadas si se desea.

Los Estados Financieros mostrarán los Saldos sólo de Cuentas de Mayor, o sea, aquellas cuentas cuyo nivel se encuentre entre 1 y el Nivel de Cuentas de Mayor especificado en el Mantenimiento de Contabilidades.

A fin de generar dichos reportes se utiliza una ventana similar a la del Libro Mayor. En ella se especificará el rango de fechas a utilizar para generar el correspondiente reporte, y si desea utilizar también las Partidas No Mayorizadas.

Ilustración 71: Ventana de filtros del Balance General.

Ilustración 72: Ventana de filtros del Estado de Resultados.

Opción: Saldos de Períodos Cerrados

Este reporte permite comparar los saldos de las cuentas contables entre dos períodos contables. Estos períodos pueden o no pertenecer al Ejercicio actual, lo cual permite comparar los saldos mensuales entre dos períodos del Ejercicio actual, o dos de Ejercicios Históricos, o uno del Ejercicio actual con otro de un Ejercicio pasado.

Para poder comparar un Período éste debe estar ya Cerrado, de lo contrario el Sistema no permitirá utilizarlo para este reporte.

Para generarlo únicamente se especifican los Ejercicios y códigos de los Períodos a comparar.

COMPARATIVO DE SALDOS DE PERIODOS CERRADOS

Seleccione los dos períodos a comparar

Período 1:	1	Descripción:	
Ejercicio 1:	1996		Enero 1996
Período 2:	3	Descripción:	
Ejercicio 2:	1997		Marzo 1997

Imprimir Cancelar

Ilustración 73: Ventana de filtros del Reporte Comparativo.

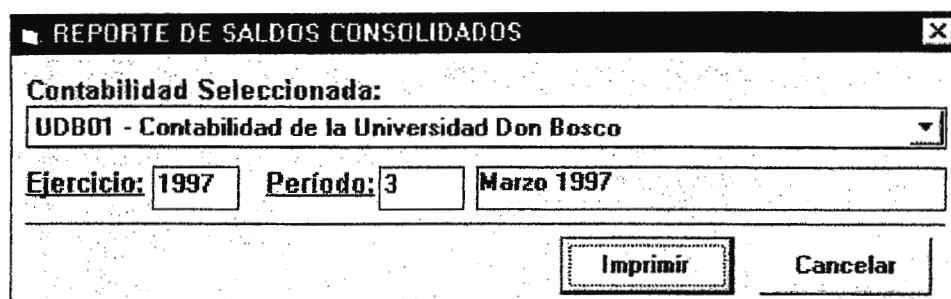
Menú: Reportes Consolidados

Este menú provee acceso a todos los reportes disponibles en el sistema cuya información refleja los saldos consolidados con otras contabilidades que han sido relacionadas a través del catálogo de cuentas, tal y como se explica en la opción del Mantenimiento del Catálogo de Cuentas.

Opción: Saldos Consolidados

Este reporte tiene la misma estructura y datos que el Libro Mayor, con la diferencia que mostrará los Saldos Consolidados a cierto período para una Contabilidad específica, no los saldos propios a dicha contabilidad.

Para generarlo se especifica en una ventana únicamente el período cerrado para el cual se desea el reporte, y para el caso de los usuarios con acceso a múltiples contabilidades, se puede seleccionar también para qué Contabilidad se desea.



REPORTE DE SALDOS CONSOLIDADOS

Contabilidad Seleccionada:
UDB01 - Contabilidad de la Universidad Don Bosco

Ejercicio: 1997 Período: 3 Marzo 1997

Imprimir Cancelar

Ilustración 74: Selección de Reporte de Saldos Consolidados para un Período Cerrado.

Opción: Balance General y Estado de Resultados Consolidados

Ambos reportes tiene la misma estructura que los correspondientes Balance General y Estado de Resultados normales, salvo que utilizan los saldos consolidados para una contabilidad específica, y únicamente pueden ser generados para períodos cerrados y no para cualquier fecha.

Utilizan la misma ventana y datos que el Reporte de Saldos Consolidados explicado anteriormente.

Opción: Estados Financieros Históricos

Pueden ser generados en el Sistema tanto el Balance General como el Estado de Resultados utilizando los datos (cuentas y saldos) para los Períodos de Ejercicios Históricos (anteriores al actual), para lo cual se utilizará la misma ventana y especificaciones que para los Reportes Consolidados.

Se mantiene en estos reportes la misma estructura que para los Estados Financieros normales (para el ejercicio actual), salvo que se indica en los títulos de los reportes que se trata de reportes históricos.

Menú: Reportes de Presupuesto

Se generan asimismo en el Sistema varios reportes respecto al control del Presupuesto. Para poder emitir estos reportes se utiliza un método distinto al de los reportes contables. La principal diferencia consiste en que para los reportes de Presupuesto NO es posible definir rangos (tanto de rubros o fechas o períodos) para imprimir sólo algunas partes de los datos existentes.

Debido a que el Sistema únicamente contempla el Control del Presupuesto, los reportes que se han incluido en el programa muestran los datos y valores absolutamente necesarios para llevar a cabo tal función. Es por ello que siempre son impresos los datos para todo el Catálogo de Presupuesto actual, y no se permite seleccionar sólo una parte de éste.

Para cada uno de los reportes presupuestales, únicamente podrá especificarse el destino de la impresión, es decir, enviar el reporte ya sea directamente al impresor, a un archivo, o a pantalla (con la posibilidad de imprimirse luego), utilizando la “Ventana de Selección del Destino de Impresión” explicada al inicio de la sección de Reportes del Sistema.

Los Reportes de Presupuesto incluidos en el Sistema son :

Opción: Rubros de Presupuesto

Este reporte consiste en el listado de todos los Rubros de Presupuesto para una Contabilidad o Unidad específica.

Incluye los siguientes datos :

- Código del Rubro Presupuestal
- Descripción o Nombre del Rubro
- Tipo del Rubro

Opción: Listado de Tipos de Rubro

Este reporte consiste en el listado de Tipos de Rubros Presupuestales utilizados en el Sistema. Sirve básicamente como información descriptiva de la clasificación de Rubros utilizada en el programa.

Opción: Montos Presupuestados

Este es el reporte principal para el Módulo de Control Presupuestal, ya que es el que muestra para cada uno de los Rubros la comparación de sus Saldos Presupuestados, Saldos Reales, y Saldos Reales Consolidados.

Reportes de Saldos Presupuestados

Contabilidad Seleccionada
COP03 Contabilidad General de la Universidad Don Bosco

Seleccione el Tipo de Reporte

Lista agrupada Columnar

Rango de Periodos

Periodo Inicial: 1996 1 Enero 1996

Periodo Final: 1996 12 Diciembre 1996

Rango de Rubros

Desde: EE

Egresos Extraordinarios

Hasta: 10.61.50.50

Varios

Imprimir Cancelar

Ilustración 75: Pantalla para la selección de los reportes de montos presupuestados.

Es con este reporte que período a período se podrá llevar el control de los ingresos y gastos de la Institución, a través de la comparación de los valores mencionados.

Debido a que los Saldo Reales provienen directamente de la Contabilidad (de las Partidas Mayorizadas), la información que se obtiene siempre estará actualizada al mismo nivel que la Contabilidad de la Unidad que emita el reporte.

Al seleccionar esta opción el sistema presentará una forma que permitirá al usuario seleccionar varios parámetros para delimitar o personalizar el reporte de montos presupuestales. Los elementos que componen dicha forma son los siguientes:

Contabilidad Seleccionada Esta caja de selección tipo combo, permite seleccionar la contabilidad de la cual se imprimirá el reporte, por omisión el sistema siempre seleccionará automáticamente la contabilidad actual de trabajo seleccionada por el usuario.

Tipo de Reporte Este recuadro presenta dos opciones: Lista Agrupada o Columnar. Permite seleccionar al usuario cual de los dos tipos de reporte quiere imprimir. En el primero de los casos, se imprimirán los montos presupuestados y se compararan con los montos reales y consolidados para cada período del ejercicio actual. La segunda opción, presentará un reporte de los montos presupuestados por período para cada rubro de presupuesto.

Rango de Períodos Permite seleccionar hasta que período del ejercicio actual se quiere imprimir los montos de cada rubro presupuestal.

Rango de Rubros Permite seleccionar un rango dentro del listado de rubros presupuestales a imprimir. Para facilidad de uso, el usuario puede digitar directamente los códigos del rubro inicial y final, o puede presionar los botones adyacentes para poder examinar una lista de rubros presupuestales de los cuales podrá seleccionar el deseado.

Imprimir Muestra la forma general de impresión, para que el usuario seleccione si lo quiere ver en pantalla o imprimir.

Cancelar Cancela la operación de imprimir y regresa al menú principal.

Opción: Listado de Cuentas asignadas a los Rubros de Presupuesto

Este reporte básicamente consiste en el listado de los Rubros de Presupuesto, incluyendo para cada uno de ellos un sub-listado de las Cuentas Contables que han sido asociadas a él, o sea, aquellas cuentas de las cuales obtiene sus Saldos Reales y Consolidados. Es sobre estas cuentas que se lleva el Control a través del Presupuesto.

Este no es un reporte que deba generarse muy a menudo, pero tiene una vital importancia, ya que sin él no podría saberse con exactitud qué cuentas del Catálogo están siendo controladas a través del Presupuesto, y si han sido correctamente asignadas a los Rubros adecuados.

Menú: Presupuesto

Este menú contiene todas las opciones necesarias para poder dar mantenimiento y hacer consultas del modulo de Control Presupuestal del Sistema COP. A continuación se especifica en detalle cada una de las opciones.

Opción: Rubros de Presupuesto

Al igual que en el Modulo contable, en donde El catálogo de Cuentas es su columna vertebral, el modulo de control presupuestal cuenta con el catálogo de rubros presupuestales. Este catálogo no difiere mucho con respecto al catálogo de cuentas, ya que posee la misma estructura jerárquica. La diferencia principal esta en la información que cada una de estas estructuras de datos guarda. Los rubros presupuestales, es un mecanismo que permite al usuario definir la manera en que quiere controlar su presupuesto. Cada uno de los rubros que se definen pueden ser subdivididos sucesivamente hasta “n” niveles, siempre y cuando no se exceda el número máximo de dígitos o caracteres del código de rubro (30 caracteres). Cada Contabilidad o Empresa definida en el sistema lleva su propio Catálogo de Rubros presupuestales, en donde cada uno de estos rubros se relaciona a una o más cuentas de su respectivo catálogo de cuentas. De esta manera, el sistema es lo suficientemente flexible como para manejar una amplia gama de escenarios presupuestales. Generalmente, los sistemas contables, permiten un nivel pobre para el control del presupuesto, que consiste en

asignar un monto presupuestado a cada cuenta contable para después compararlo con los saldos acumulados en el ejercicio. En el Modulo de Control Presupuestal del sistema COP, este soporte es más extenso, ya que el usuario es capaz de poder definir una estructura jerárquica de rubros presupuestales que no necesariamente refleja la misma estructura del catálogo de cuentas. Por este motivo, el sistema incluye un mantenimiento de asignación de cuentas a rubros, el cual permite definir las relaciones entre el catálogo de rubros y el de cuentas contables. Por otro lado, es posible asignar un monto presupuestado a cada rubro por período contable, (ejemplo: por cada mes del año, o como estén estructurados los periodos contables en el ejercicio) una vez asignados, el sistema automáticamente ira reflejando los cambios con respecto a los saldos reales que están basados en los saldos por período de cada cuenta contable relacionada al catálogo de presupuesto.

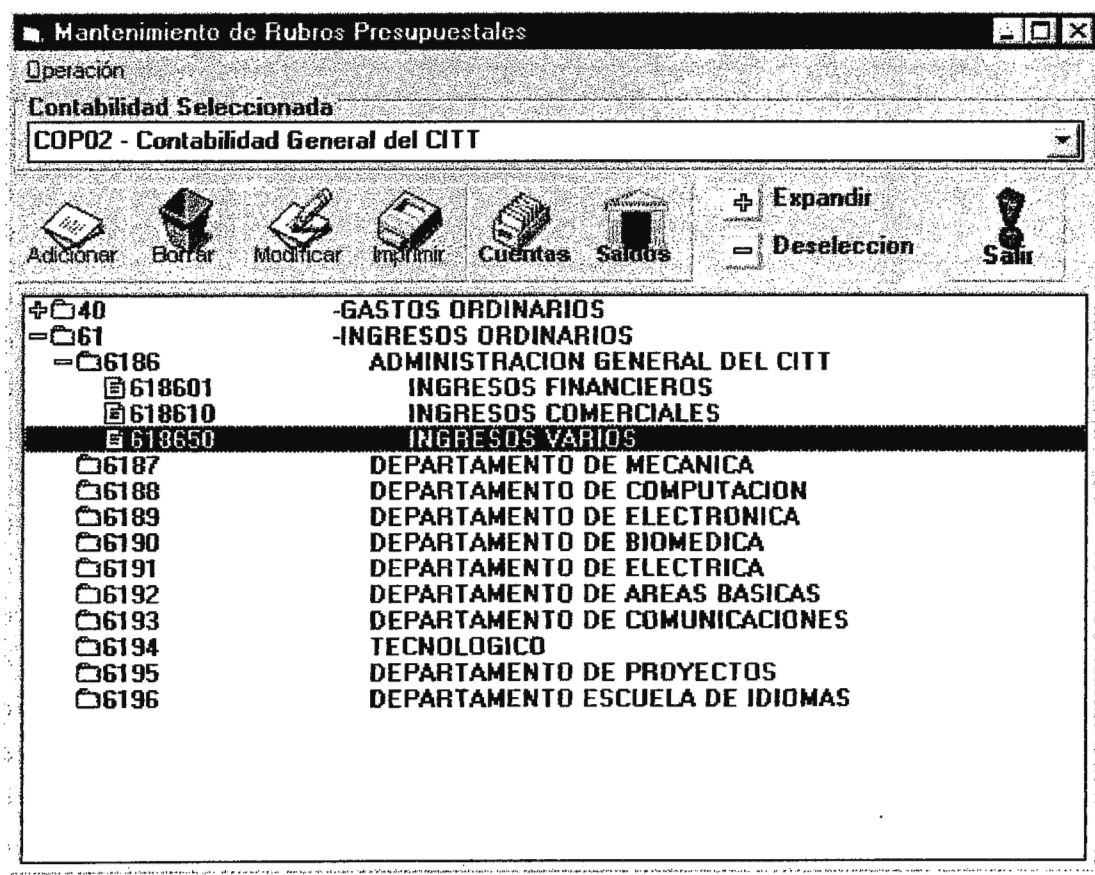


Ilustración 76: Mantenimiento Jerárquico del Catálogo de Rubros de Presupuesto.

La forma de mantenimiento del Catálogo de Rubros Presupuestales es muy similar a la del Catálogo de Cuentas Contables, tanto en apariencia como en funcionalidad. Esta forma presenta una vista de árbol para representar la estructura jerárquica de los rubros presupuestales, donde el usuario puede hacer Click en cada rubro para ir descubriendo los sub-rubros que se consolidan en el anterior. Este mantenimiento, resume muchas operaciones relacionadas al control presupuestal del sistema COP, pues, además de la capacidad de adicionar, borrar, y editar rubros, también es posible asignarles montos presupuestados por cada período contable en el ejercicio actual, además de proveer un mantenimiento para establecer las relaciones entre el rubro de presupuesto seleccionado y la(s) cuenta(s) contable(s) que reflejan el monto real para dicho rubro.

Al seleccionar cualquiera de las opciones Adicionar, Editar o Borrar, una forma de tipo MANSAB aparece en pantalla, la cual permite al usuario dar mantenimiento al rubro seleccionado o adicionar uno nuevo.

Mantenimiento de Catálogo de Rubros Presupuestales

Código Contabilidad: COP02 Nivel Jerárquico: 3

Código de Rubro: 618950

Código de Rubro Padre: 6196

Código de Tipo de Rubro: AD1

Descripción Rubro: INGRESOS VARIOS

Es Detalle?: 1 - Sí Controlar este Rubro?

Editando [Aceptar] [Cancelar]

Ilustración 77: Forma para la adición, edición o borrado de un Rubro de Presupuesto.

La forma de mantenimiento del rubro presupuestal se compone de los siguientes elementos:

Cod. Contabilidad Este campo contiene el código de la contabilidad actual seleccionada al iniciar el sistema, por tanto no es editable, pues sólo sirve para indicarle al usuario, a que contabilidad se le esta adicionando, borrando o editando un rubro de presupuesto.

Cod. De Rubro Este caja de texto, permite entrar el código que identificará de manera única al rubro que se esta adicionando. Si el rubro que se esta

adicionando es un sub-rubro, la aplicación, automáticamente llenará la primera parte del campo con el código del Rubro padre, para que el usuario sólo deba digitar el resto del código. El código puede estar compuesto por dígitos o caracteres alfanuméricos, siempre y cuando no exceda un máximo de 30 espacios. Este campo sólo puede ser introducido por el usuario en modo de Adición.

Cod. Rubro Padre Este campo es sólo para fines informativos, y presenta el código del rubro padre o de consolidación, por tanto no es editable. En el caso que el rubro que se esta adicionando es de primer nivel jerárquico, este campo aparecerá vacío.

Cod. Tipo de Rubro Contiene el código que clasifica al rubro actual. Al igual que en el catálogo de cuentas, estos tipos de rubro, son definidos por el usuario a través de otro mantenimiento, lo que provee de flexibilidad en el momento de clasificar los rubros de presupuesto. En el caso de que se este adicionando un sub-rubro, este campo automáticamente será seleccionado por la aplicación, pues el tipo de rubro se hereda del rubro padre. El usuario puede digitar el código de clasificación del rubro directamente o puede presionar el botón adyacente para invocar a una forma de tipo CONLIS que le muestra todos los tipos definidos en el sistema, de los cuales podrá seleccionar uno.

Descripción del Rubro

Permite entrar o editar el concepto del rubro, que lo identifica de una manera más lógica.

Es detalle?

Este elemento presenta una caja de selección tipo combo, en la cual el usuario puede definir si el rubro que se esta adicionando o editando, posee mas sub-rubros o no. Sin embargo, el sistema, a través de las validaciones en el servidor de bases de datos, mantiene este campo automáticamente actualizado.

Controlar este rubro?

Esta caja de chequeo, permite definir si el usuario desea que se le mantenga al tanto, cuando el monto real del rubro a sobrepasado el

monto presupuestado en un período contable dado. Esto se logra a través de mecanismos de validación en la base de datos, lo que asegura su efectividad.

Aceptar	Permite ejecutar la acción seleccionada y regresar a la forma anterior.
Cancelar	Cancela la operación y retorna a la forma anterior.

Es importante hacer ver, que debido a los alcances y limitaciones expuestos en el diseño de la aplicación, no se incluye la opción de poder consolidar los rubros de presupuesto en otros rubros ajenos a la contabilidad actual, de la misma forma como se hace en el Catálogo de Cuentas contables. Esto se debe principalmente a que el sistema proveerá un mecanismo para el control presupuestal y no para la asignación del presupuesto, ya que esta última actividad involucra decisiones administrativas basados en criterios lógicos y circunstanciales, lo cual es imposible de mecanizar.

Restricciones y Verificaciones

- El tipo de rubro es heredado del rubro padre, y no se puede modificar, a excepción de aquellos rubros que se adicionan en el primer nivel de la jerarquía.
- El código del rubro no puede exceder 30 caracteres, y se puede utilizar códigos diferentes a los utilizados en el catálogo de cuentas contables.
- No es posible borrar un rubro de presupuesto si este no es de detalle, es decir que no posea sub-rubros.

Adicionalmente a las opciones estándares de mantenimiento para el catálogo de rubros presupuestales, existen 2 opciones adicionales: Cuentas y SalDOS. La opción de Cuentas en la barra superior de botones, permite al usuario relacionar cada rubro de presupuesto con cuentas contables cuyos saldos reflejen el monto real del rubro en un período dado. Des esta manera, el sistema puede saber como calcular el monto real gastado o ingresado para luego poder compararlo con los montos de presupuesto. Debido a que el catálogo de rubros es una estructura jerárquica el sistema sólo permitirá relacionar a las cuentas un rubro que sea de detalle, para evitar comparaciones cíclicas o redundantes, ya que de lo contrario, se pudiera

dar el caso en que un rubro de nivel mayor esta relacionado a la misma cuenta que uno de sus rubros hijos, lo que crearía una duplicación del monto real para el rubro en cuestión.

Por otro lado, un rubro de presupuesto puede calcular su monto real, a partir de la suma (asignación) de 1 o más cuentas contables del mismo nivel jerárquico, pero no viceversa, dicho de otra manera, una cuenta contable no puede figurar en el calculo del monto real de varios rubros. Lo anterior se justifica por el hecho de que si una cuenta afecta a dos rubros de presupuesto distintos (que podrían pertenecer a la misma rama del árbol jerárquico) podrían causar redundancias de montos en el momento de comparar saldos reales y presupuestados.

La opción de asignación de cuentas al rubro seleccionado, presenta una forma standard SELNAV de mantenimiento en donde el usuario puede adicionar o borrar las cuentas asignadas al rubro presupuestal, por lo que sólo los botones de adicionar y borrar aparecerá habilitados. En el caso de que el rubro seleccionado no sea de detalle, el botón “Cuentas” aparecerá deshabilitado, debido a que no es posible asignar o relacionar cuentas a rubros que no son de detalle por las razones antes expuestas.

Opciones	
Adicionar	Borrar
Editar	Imprimir
Filtro	Buscar...

Contab.	COP02
Rubro	618650

Cod.cuenta	
618650	INGRESOS VARIOS

Ilustración 78: Forma para la asignación de cuentas contables a un rubro de presupuesto.

Al igual que cualquier forma estándar utilizada en el sistema, al presionar los botones de adicionar o borrar, una forma de tipo MANSAB aparecerá para que el usuario asigne o borre una cuenta contable a las relaciones del rubro seleccionado, dicho sea de paso, este se presenta en la parte superior del mantenimiento para informar al usuario, a que rubro de presupuesto le est asignando cuentas.

Ilustración 79: Forma para la asignación de una cuenta contable a un rubro presupuestal.

Los elementos que componen la forma de mantenimiento son los siguientes:

Cod. de Contabilidad	Indica el código de la empresa o contabilidad actual. No es editable.
Cod y Descripción del Rubro	Indica el código y concepto del rubro presupuestal al cual se le esta asignando o desligando una cuenta contable. No es editable.
Cod. y Descripción de Cuenta	Indica el Código y Concepto de la cuenta contable que se esta asignando, y es el único campo que es digitado por el usuario en el proceso de adición o asignación. Para facilidad de uso, el usuario puede presionar el botón adyacente, el cual le mostrará una forma que contiene todas las cuentas posibles para que seleccione una.
Aceptar	Ejecuta la acción y regresa a la forma anterior.
Cancelar	Cancelar la operación y regresa a la forma anterior.

Verificaciones y Restricciones

- Sólo se podrá asignar cuentas contables a rubros de detalle.
- Estas cuentas deben ser del mismo nivel jerárquico para un rubro dado.
- Las cuentas asignadas deben ser de la misma contabilidad de trabajo.
- Una cuenta contable no puede estar asignada a dos o más rubros diferentes.

El Mantenimiento de Rubros presupuestales también cuenta con una opción para asignar los montos presupuestados en cada uno de los periodos contables del ejercicio actual. Esta opción es accesada a través del botón “Saldos”, este además permite consultar la relación entre los saldos reales y los presupuestados en cualquier momento del ejercicio, siempre y cuando, ya se hayan asignado las cuentas contables a cada rubro previamente.

Al igual que la asignación de cuentas, esta opción sólo permite asignar montos presupuestados a aquellos rubros que sean de detalle, de esta manera se garantiza que los montos presupuestados en el rubro padre sean consistentes con los montos que se le asignan a sus rubros hijos o sub-rubros.

Ejercicio	Periodo	Saldo Presup	Saldo Real	Saldo Consolidado
1996	1	C40,000.00	(C2,000.00)	(C2,000.00)
1996	2	C50,000.00	C0.00	C0.00
1996	3	C25,000.00	C0.00	C0.00
1996	4	C52,500.00	C0.00	C0.00
1996	5	C47,000.00	C0.00	C0.00
1996	6	C12,356.00	C0.00	C0.00
1996	7	C26,354.00	C0.00	C0.00
1996	8	C41,650.00	C0.00	C0.00

Ilustración 80: Mantenimiento de montos asignados a los rubros de presupuesto.

A cada rubro de presupuesto se le asigna una cantidad para cada período del ejercicio, la cual será comparada con la sumatoria de saldos de las cuentas que tiene asignadas ese rubro durante todo el ejercicio. AL principio de ejercicio contable, el usuario deberá asignar únicamente los montos presupuestados para cada rubro de detalle, y el sistema se encargará de mantener actualizados los montos reales cada vez que se mayoriza una partida, de tal manera que esta consulta reflejará los últimos cambios. Con el fin desasignar los montos presupuestados, el usuario adiciona, edita o borra dichas asignaciones al rubro que ha seleccionado previamente, para ello utiliza los botones que aparecen en la parte superior de la forma, los cuales harán aparecer la forma MANSAB para permitir realizar dicha operación.

Mantenimiento de Saldos Presupuestados

Cod. Contabilidad: 10000

Cod. Rubro Presupuesto: 418600

GASTOS ADMINISTRATIVOS

Ejercicio Contable: 1996

Periodo Contable: 1

Saldo Presupuestado: 4000000

Editando: Aceptar Cancelar

Ilustración 81: Forma para la asignación, edición o borrado de un monto presupuestado a un rubro.

La forma resultante contiene los siguientes elementos:

Cod. Contabilidad	Indica el código de la contabilidad seleccionada actual y no es permitido su edición.
Cod. Rubro de Presupuesto	Contiene el código del rubro de presupuesto al cual se le esta asignando un monto presupuestado. No es editable.
Ejercicio y Período Contable	Estas dos cajas de texto permiten digitar el ejercicio y período al cual se le esta presupuestando el monto, si el usuario desea, puede presionar el botón adyacente para seleccionar un período valido del registro de periodos contables definidos en el sistema.
Saldo Presupuestado	Permite digitar el monto que se le asignará al rubro presupuestal en el período contable especificado.
Aceptar	Realiza la acción y regresa a la forma anterior.
Cancelar	Cancela la operación y regresa a la forma anterior.

Adicionalmente, existen 2 campos adicionales que el usuario no ve en este mantenimiento, debido a que son mantenidos automáticamente por el sistema, ellos son: Saldo Real y Saldo Real Consolidado. El primero, contiene el saldo que resulta de totalizar los saldos de las cuentas asignadas al rubro para el período indicado y el segundo contiene el resultado de sumar los saldos de las cuentas asignadas al rubro mas el de las cuentas que

pertenecen a otras contabilidades pero se consolidan en la cuenta en cuestión. Estos dos campos son actualizados por el sistema cada vez que se afecta los saldo contables a través del proceso de mayorización por partidas. Dichos campos servirán para que el usuario pueda realizar consultas sobre su estado presupuestario en cualquier momento del ejercicio, así como también para la generación de reportes de presupuesto.

Restricciones y Verificaciones

- Sólo es posible asignar montos presupuestados a rubros de detalle, el sistema se encargará automáticamente de reflejar los totales en los rubros de nivel mayor.
- La asignación del presupuesto es por periodos contables y no por el ejercicio.
- Si no se especifica el monto presupuestado para el rubro en un período determinado, el sistema comparará el saldo real contra 0 (cero).

Opción: Tipos de Rubro

En el mantenimiento anteriormente expuesto, se discutía sobre la clasificación de los rubros de presupuesto en diferentes tipos o clases, dichos tipos son definidos a través de esta opción por el usuario.

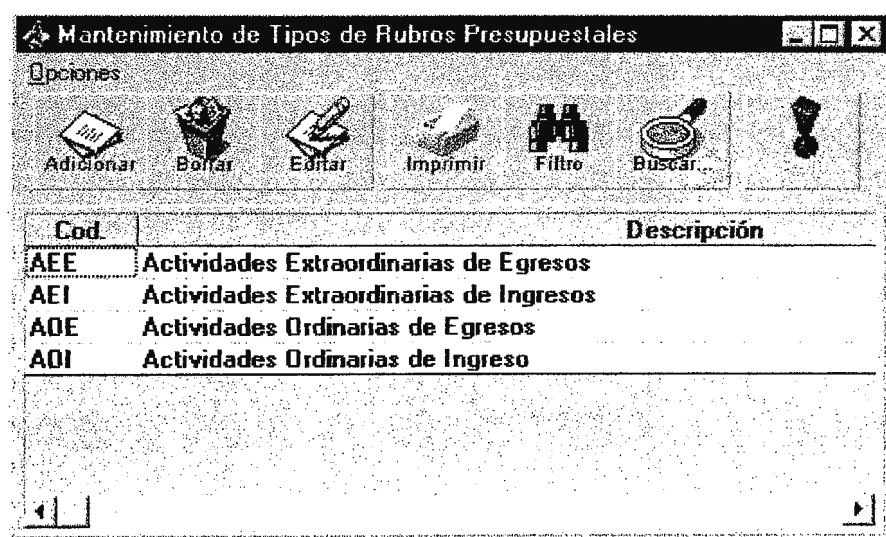


Ilustración 82: Mantenimiento de Tipos de Rubros Presupuestales.

La clasificación de cada rubro presupuestal esta dada por los tipos definidos en esta forma, para dicho efecto, el usuario puede agregar, borrar o editar los atributos de cada clase o tipo accedando los botones que se muestran en la parte superior de la forma. Al accesarlos, una forma de tipo MANSAB aparecerá para permitir realizar la opción seleccionada.

Ilustración 83: Forma para la adición, edición o borrado de tipos de rubros presupuestales.

Cada tipo o clase de rubro presupuestal se define por los siguientes atributos:

Cod. Tipo de Rubro	Contiene un código que identifica al tipo de rubro, y sólo se puede editar en el momento de adicionar un nuevo tipo.
Descripción	Permite definir una descripción lógica que conceptualiza al tipo de rubro.
Tipo General	Esta caja de selección, presenta dos opciones posibles: Ingreso o Egreso. Define si el tipo o clase de rubro que se esta adicionando es considerado como ingresos o egresos para efectos posteriores de calculo total del déficit o superávit.
Aceptar	Realiza la acción seleccionada y regresa a la forma anterior.
Cancelar	Cancela la operación si hacer ningún cambio y regresa a la forma anterior.

Restricciones y Verificaciones

- No es posible borrar un tipo de rubro si existen rubros presupuestales clasificados por ese tipo.

- Los tipos de rubros son parámetros globales de la aplicación por lo que todas las contabilidades definidas en el sistema se rigen por la misma clasificación, esto permite una consistencia a través de todas las unidades involucradas.

Menú: Ayuda

Permitirá el acceso a la ayuda en línea del sistema, aun esta en construcción.

Conclusiones

El Plan de Desarrollo Informático de la Ciudadela Don Bosco fue ambicioso desde un principio pero no imposible de realizar. Dado que Ciudadela Don Bosco es una gran obra aun en construcción y desarrollo, los procesos administrativos están cambiando constantemente, en un proceso continuo de prueba y error, prueba y error, y a la larga prueba y éxito. Pero mientras esto ocurre es imperante que la administración de la Ciudadela Don Bosco entienda que un proyecto de tipo Cliente/Servidor no se realiza de la noche a la mañana y con recursos prestados, definitivamente, es necesario mucho mas que eso. Debido a muchos obstáculos fuera de nuestro control, el proyecto quedara como un sistema mas sin implementar, esperando a que se de la oportunidad de ponerlo a funcionar. Pero antes de hacer cualquier evaluación subjetiva de este proyecto, es necesario reconocer que los sistemas fueron desarrollados hasta donde los recursos disponibles en Ciudadela permitieron.

Queremos dejar constancia de nuestro parecer de cómo un proyecto de este tipo puede ser implementado en Ciudadela Don Bosco pero no si antes tomar en cuenta los siguientes elementos:

- El rol de la administración de cada Unidad en el desarrollo de los proyectos, debe ser protagónico, lo que requeriría un proceso de aprendizaje de parte de estos usuarios para poder trabajar en conjunto con los analistas y desarrolladores.
- Debería de existir al menos un coordinador del proyecto, que tenga cierto poder de decisión para administrar y si es necesario adquirir mas recursos, ya sean estos humanos o informáticos.
- Considerar que el proyecto no puede ser implementado sin costo alguno, es decir, no sin antes hacer una inversión, que a la larga, generara muchos beneficios, y que de no ser implementado, generara grandes costos de operación y desorganización a corto plazo.
- Un proyecto de esta magnitud, no puede darse el lujo de ser implementado a medias, pues

es mejor una solución parcial que funcione bien a una solución completa que no lo hace, con esto queremos dar a entender, que sería óptimo comenzar a implementar los módulos del proyecto de una manera secuencial, desde el más básico al más auxiliar.

- Este proyecto podría durar más de un año por lo que aconseja que la institución trate de mantener a los empleados claves en el proyecto para no perder su continuidad.
- Definitivamente, las personas que desarrollen e implementen el sistema, deberán ser empleados de CDB, pues es de todos conocido que un sistema necesita cuidado y mantenimiento constante antes de que llegue a ser independiente y totalmente funcional.
- La red que interconecta a CDB deberá ser implementada antes que los sistemas que se produzcan de este proyecto, a menos que todos los sistemas de todas las unidades se centralicen en un solo centro de cómputo, lo cual es inconveniente e impráctico, por lo que consideramos al Proyecto de Interconexión de la CDB como un requisito indispensable.
- Y quizá la más importante de las recomendaciones, es que la CDB deberá (obligatoriamente) definir bien sus procedimientos o al menos sus objetivos administrativos, ponerlos en concordancia a lo largo y ancho de toda la Institución, para poder mecanizar cualquier proceso, por simple que este parezca. A lo largo del desarrollo de las aplicaciones, se notó un desacuerdo y falta de orientación en lo que a los objetivos y procedimientos administrativos se refería, por lo que nuestra aplicación no pudo adaptarse al final, a las necesidades actuales (o al menos a las necesidades que imperaban en el momento en que se escribió este documento).

A continuación y para concluir, presentamos las recomendaciones de los expertos en desarrollo de Proyectos Cliente/Servidor de la Empresa Microsoft y que podrían servir de mucho para futuros proyectos de esta naturaleza en la Ciudadela Don Bosco.

Microsoft ha vendido más de 3 millones de copias de Visual Basic en sólo unos pocos años. Este fenomenal récord debe atribuirse principalmente a la capacidad de Desarrollo rápido de aplicaciones (RAD; Rapid Application Development) de Visual Basic, pero la última versión, Visual Basic 4, también permite la creación de aplicaciones cliente/servidor.

Pero no se equivoque, ninguna herramienta RAD, ni siquiera Visual Basic, elimina la necesidad de una buena planificación. Antes de acometer el desarrollo de su primer proyecto, revise la siguiente lista. Esta lista dista mucho de ser completa, pero sirve para demostrar que en la construcción de una buena aplicación cliente/servidor hay que hacer algo más que escribir el código.

- Defina los objetivos de negocios para el proyecto.
- Determine las reglas de negocios para la organización.
- Desarrolle un plan general para el proceso.
- Diseñe cuidadosamente la estructura de datos subyacente.

Si no está familiarizado con alguna de las herramientas que va a utilizar, el riesgo de fallo aumenta enormemente. Existen varios métodos para mitigar ese riesgo. Controle los procesos en los que trabaja para reducir el número de incógnitas, reservando los proyectos más complicados para más adelante. Si no dispone de todo el tiempo que requiere este enfoque, plantéese la posibilidad de contratar a alguien con experiencia. Si el proyecto es algo excepcional, puede que tenga sentido contratarlo externamente. Los servicios de asesoría y consultoría son otra alternativa atractiva. El consultor adecuado puede ayudarle a minimizar el riesgo del proyecto, tanto con su ayuda en las decisiones de diseño como ofreciendo funciones de formación con las herramientas que usted todavía no domina. La mayor parte de los elementos de este capítulo no sólo tienen que ver con Visual Basic sino también con el desarrollo de sistemas cliente/servidor en general. Le animamos a leerlo porque la mayoría de estos temas son los que determinan el éxito o el fracaso de un sistema.

Funciones del cliente, de la administración y del desarrollo

La cultura corporativa es una realidad de la vida. Buena, mala o indiferente, juega un papel importante en todo lo que hacemos. En lugar de permitirle que dirija su proyecto, aprovéchese de ella a la hora de definir las funciones y las responsabilidades de las distintas partes implicadas en la planificación y el diseño de su sistema cliente/servidor, incluidos los usuarios finales, los desarrolladores y la administración.

Y un tópico real: las culturas pueden dar al traste con su proyecto cliente/servidor. Es necesario que entienda la cultura de su empresa.

Asociación con el cliente

Uno de los factores claves para asegurar el éxito de un proyecto consiste en crear una asociación con los usuarios finales, o clientes, para aunar unos conocimientos y atributos complementarios. El usuario final suele ser el que mejor entiende el proceso actual y puede ofrecer la perspectiva necesaria, que puede ser de gran ayuda durante el desarrollo de la aplicación.

Esta asociación puede aplicarse igualmente en aquellas situaciones en las que la experiencia en los sistemas cliente/servidor viene de fuera de la organización. De hecho, la asociación es incluso más vital con vendedores externos, porque rara vez conocen por completo los procedimientos, costumbres, objetivos y otros factores de la empresa que los empleados han asimilado a lo largo de los años. Disponer de personas externas implicadas en el proceso es también recomendable porque una persona externa a menudo se cuestiona los temas que las personas familiarizadas con el proceso actual pueden asumir como correctas de forma tácita. Si no puede disponer de un consultor para trabajar en el proyecto, puede utilizar a alguien de otro departamento para desempeñar la misma función. Pero asegúrese de que no está demasiado familiarizado con el proceso actual y de que no tenga miedo de hacer preguntas.

Asegúrese de implicar a los estratos superiores de la administración en el proceso de establecimiento de asociaciones. Es fundamental contar con el apoyo de los niveles altos de la organización, por la probabilidad de que un proyecto cliente/servidor cruce los límites departamentales o incluso plantee una reestructuración. La mayor parte de los proyectos que no consiguen ese apoyo nunca llegan a llevarse a cabo.

Función de la administración

Es casi imposible implementar con éxito un proyecto cliente/servidor sin contar con el apoyo de los sectores superiores de la administración para dar respaldo al proyecto y controlar la observancia de los planes del proyecto por parte de la organización. Conseguir el apoyo de la administración para el proyecto desde el principio es crucial porque los planes de los proyectos son dinámicos, cambian con el paso del tiempo según se va consiguiendo más información sobre el proyecto y sobre el entorno de negocios.

La administración de categoría superior también debe apoyar la estructura organizativa necesaria para conseguir los objetivos definidos para la organización, lo que no sólo supone la reestructuración de los departamentos relacionados con el tema, sino también la creación de equipos software para el desarrollo de aplicaciones y realización de otras tareas necesarias para el proyecto.

Cualquier proyecto de la magnitud del típico primer proyecto cliente/servidor exigirá a veces que se tomen decisiones desagradables. La administración de categoría superior debe tener la voluntad de tomar esas decisiones y de ejecutar los planes necesarios para ponerlas en práctica. También debe apoyar a los equipos del proyecto en su trabajo de descubrir los problemas y de recomendar soluciones a la administración.

Análisis de costes/beneficios y administración de proyectos

El principal objetivo de cualquier proyecto es el de conseguir una solución software que realice sus tareas con un coste razonable. (Que la empresa pueda pagar por el proyecto.) La efectividad del coste del trabajo adicional es cuestionable por encima de un cierto punto.

Recuerde la antigua regla del 80/20: el 80 por ciento de los beneficios pueden obtenerse con el 20 por ciento de la inversión. Esto es cierto casi para cualquier proyecto. Conseguir el último 20 por ciento del beneficio es lo que dispara el coste.

Los costes bajos y los beneficios altos suelen estar en los extremos opuestos del espectro. Para obtener más de uno de ellos es necesario sacrificar parte del otro. Las empresas suelen emprender el desarrollo de proyectos de bajo coste, que nunca se completan o que se archivan después de su terminación, porque no resuelven con eficiencia el problema definido originalmente. Por este y otros motivos, la tasa de fallo en los proyectos software de empresa se estima que es superior al 90 por ciento.

Por tanto, lo primero que debe hacer es determinar si puede conseguir los objetivos con los recursos disponibles. Si su objetivo es el de crear un sistema de entrada de pedidos que pueda integrarse en un sistema de contabilidad, deberá tener mucho cuidado para asegurar que los dos sistemas funcionen bien juntos. Las preguntas como “Cómo deben interaccionar los usuarios con los clientes para realizar un pedido?” pueden servir para comprender en detalle la facilidad de uso que ofrece la interfaz de usuario. Otras preguntas como “Se comunicará la aplicación directamente con el libro mayor general o con los subsistemas de cuentas por pagar?” tendrán que tener respuesta antes de que el proyecto avance demasiado.

Algunas veces se definen objetivos que parecen inalcanzables. Si no puede conseguir su objetivo con los recursos disponibles, no necesariamente tendría que tirar a la basura el proyecto. A menudo podrá proponer una solución útil que resuelva una parte importante del problema. Una solución parcial que funcione es infinitamente mejor que una solución completa que no funcione.

La administración del proyecto juega un papel primordial en la consecución de los objetivos definidos para el proyecto. Una parte del trabajo de ser un buen administrador de proyectos consiste en reconocer los problemas potenciales y ocuparse de ellos antes de que se produzcan. Aunque este es el caso ideal, no es posible predecir todos los problemas. Cuando se produce un problema no previsto, habrá que ocuparse de él inmediatamente o se pagará el precio en pérdida de tiempo. Si el proyecto se enfrenta a un problema irresoluble, es el administrador del proyecto el que tiene la responsabilidad de modificar o cancelar el proyecto. Puede que le sorprenda el resultado de decir “No!” a un problema irresoluble. Un administrador realmente bueno deberá recompensar a los que reconozcan y se ocupen de los

problemas potenciales, en lugar de a los que simplemente trabajen laboriosamente, profundizando en el pozo de los fallos. Utilice los fallos como oportunidades de aprendizaje. Si ofrece un entorno que admita los fallos menores, estará ofreciendo un entorno que anime las soluciones creativas.

Utilización de los especialistas de la plantilla

Puede aprovechar los conocimientos de su equipo de desarrollo de varias maneras. Una de las filosofías de la utilización de empleados sostiene que la especialización es la clave de la eficiencia. Este enfoque puede ofrecer una cierta eficiencia a corto plazo, pero tiene ciertas desventajas. Por ejemplo, si el especialista se va, también se van sus conocimientos especiales. Además, puede que el especialista quiera responsabilidades o experiencias adicionales. Si la especialidad de la persona tiene una gran demanda en la empresa y nadie tiene la suficiente formación como para ayudarle, puede que no tenga tiempo para dedicarse a otros temas de su interés.

El enfoque de los especialistas sería óptimo si las personas fueran máquinas. (Se utiliza la mejor para cada tarea una y otra vez hasta que se estropee y entonces se repara o se sustituye por otra.) Pero las personas no son así, y los especialistas pueden ser realmente difíciles de sustituir.

Hace algunos años, conocimos un ingeniero eléctrico de un importante fabricante de automóviles. Resultó que se trataba de un especialista. El siguiente es un extracto de la entrevista:

Autor: Entones, en qué tipo de proyectos trabaja?

Ingeniero: El módulo de control del motor (MCM).

Autor: Parece interesante.

Ingeniero: No especialmente.

Autor: Trabaja en proyectos interesantes?

Ingeniero: No

Autor: Cuánto tiempo ha estado en la empresa?

Ingeniero: Tres años

Autor: Ha trabajado en otros proyectos?

Ingeniero: No, sólo en el MCM.

Autor: Vaya, yo no podría trabajar en un sistema de control durante tres años.

Ingeniero: Realmente no trabajo en todo el sistema de control. Sólo en R17.

Autor: R17?

Ingeniero: Sí, la resistencia 17.

Autor: Ha trabajado en una única resistencia durante tres años?

Ingeniero: Sí, nadie sabe tanto de RI 7 como yo.

Autor: Le gusta trabajar sólo en RI 7?

Ingeniero: No, pero la empresa lo quería así. Por eso busco otro trabajo.

Por supuesto que éste es un caso extremo, pero piense en el enorme potencial de mejorar la situación tanto para la empresa como para el empleado. Si a este empleado se le hubiera permitido aumentar sus funciones con el transcurso del tiempo y aprender más del proyecto, habría sido más valioso para la organización y para sí mismo, y lo más probable es que se hubiera quedado mucho más tiempo, y hubiera tenido una importancia grande en la organización.

Como no puede permitirse un gran movimiento de personal ni tampoco tener empleados que no conozcan la tecnología actual, quedan pocas alternativas. Ofrezca un entorno de trabajo acogedor a estos empleados valiosos, y haga todo lo que pueda para conseguir empleados que encajen bien en la filosofía de la empresa. Desarrolle una filosofía en la que los empleados tengan sus retos, pero se sientan cómodos.

Responsabilidad del desarrollador

Los desarrolladores en un entorno cliente/servidor deben asumir responsabilidades superiores a las que tenían en el pasado. Muchos grupos de desarrollo en los entornos tradicionales están acostumbrados a tomar decisiones unilaterales con respecto a los sistemas. Los usuarios no cuentan demasiado, y la planificación se realiza a un nivel operativa o táctico, en lugar de a un nivel estratégico.

Algunas organizaciones promueven un enfoque que especifica que cada grupo de desarrollo debe tomar sus propias decisiones, lo que suele dar como resultado sistemas software que no pueden comunicarse entre ellos. A menudo estos sistemas están escritos en distintos lenguajes o utilizan distintas bases de datos como almacenamiento. Para la mayor parte de estas empresas la planificación estratégica es un idioma extranjero, mientras que quien dirige es el pensamiento táctico.

Una característica común en las organizaciones con tasas altas de éxito en implementaciones cliente/servidor es que aprovechan los cambios. Evalúan constantemente las nuevas tecnologías y determinan de que manera dichas tecnologías pueden ayudar a los objetivos de la organización. El desarrollador debe contribuir y participar en todos los niveles de la planificación e implementación (operativa, táctico y estratégico).

Un excelente libro que trata de la implicación de los desarrolladores en el proceso de reestructuración es *Debugging the Development Process* de Steve Maguire (Microsoft Press, 1995). En este libro no sólo se trata la función del desarrollador sino también la del equipo completo y la filosofía que debe aplicarse para conseguir un entorno de desarrollo que tenga éxito.

Dónde encajan los consultores?

La mayor parte de las organizaciones descubren que, sin una ayuda bien informada, es difícil o imposible llevar a cabo sus proyectos cliente/servidor. La programación cliente/servidor es una nueva forma de vida y exige nuevas herramientas y técnicas, lo que puede ser inabordable sin ayuda externa. Esta es una experiencia para la empresa, y una fuente de experiencia oportuna y fiable puede suponer la diferencia entre el éxito y el fracaso.

No puede esperarse que el personal de la casa, que han estado utilizando las mismas herramientas durante años, consiga acostumbrarse a una herramienta potente y complicada como Visual Basic con sólo uno o dos cursillos de formación. Los mejores desarrolladores serán realmente productivos sólo después de haber acabado su primer proyecto importante. Otros tardarán más en serlo. Asegúrese de que haya alguien que conozca los compromisos para ayudar al personal en su trabajo de aprendizaje de las herramientas y para asegurar que el proyecto se acabe con éxito en el plazo previsto.

Algunas consideraciones sobre la selección de consultores:

- Deben tener experiencia demostrable.
- Deben encajar con la filosofía de la empresa.
- Deben ofrecer referencias (que deberá comprobar).
- Además de ser competentes en el terreno técnico, deben ser buenos docentes.

A muchas empresas no les gustan las asesorías externas. Puede que hayan tenido malas experiencias en el pasado, o que prefieran controlar todo el proceso por sí mismas. No obstante, es casi imposible que el personal interno pueda construir un buen sistema cliente/servidor al primer intento. El personal debe seguir manteniendo las aplicaciones existentes además de aprender el nuevo entorno.

Cuando hablamos de asesorías no nos referimos a utilizar “programadores contratados”, que sólo suelen tener unos conocimientos y experiencia limitados. Un buen consultor, por el contrario, suele tener años de experiencia con un amplio abanico de proyectos.

Planificación y desarrollo concurrentes

Cómo es posible planificar y construir de forma concurrente? Es más natural de lo que puede parecerle. El aprendizaje natural no es un proceso lineal. Cómo aprendemos a hablar? Es un proceso de prueba y error, prueba y error, prueba y éxito. Es en la enseñanza superior donde por primera vez nos vemos forzados a utilizar un modelo de aprendizaje distinto. Es cierto que el desarrollo iterativo puede ser un proceso penoso con algunos retrocesos, pero los humanos están hechos para aprender así. El tiempo que transcurre entre el lanzamiento de un proyecto y su terminación puede minimizarse permitiendo ciertos tropiezos en su desarrollo.

Esto no quiere decir que no deba planificar. Durante la planificación, deberá tener previsto el aprendizaje en el proceso, y no castigar los fallos. Todos cometen errores, y a todos se les debe dejar espacio para algunos fallos para que puedan progresar en su carrera.

Naturalmente que, si ha desarrollado un sistema concreto, no debe ignorar las lecciones que ha aprendido en el proceso. El objetivo global del ejercicio de desarrollo iterativo es el de aprender durante el proceso. Tres cuartas partes del juego consisten en desarrollar la

capacidad de reconocer los procesos valiosos y construir sobre ellos. La cuarta parte restante consiste en reconocer los que no funcionan y desecharlos.

Como ocurre en la planificación arquitectónica, algunas cuestiones de la planificación de un sistema cliente/servidor son casi obvias de manera intuitiva. Por ejemplo, el bloque de cemento sobre el que pretende construir una casa debe ser lo suficientemente grande como para que quepa toda la casa. La red, en el mundo “conectado” actual, es el equivalente al bloque de cemento. Si no soporta todo el peso previsto, tendría serios problemas. Si ofrece servicios de red de capacidad limitada, no le sorprenda si su personal de soporte de la red tiene que hacer horas extras y, así y todo, no es capaz de resolver todos los problemas que surgen. Puede quejarse de los servidores, pasar a Ethernet a 100 Mb, y hacer particiones en la red. Pero hubiera sido mucho más sencillo y, a la larga, menos costoso, hacerlo bien desde el principio. Si tiene el tiempo y el dinero suficiente, puede utilizar un enfoque iterativo incluso aquí. No obstante, antes de hacerlo, piense si su organización puede permitirse el pagar el proceso de aprendizaje. Puede resultar rentable contratar un consultor experto o un empleado con experiencia demostrada.

Formación

Como lo habitual es que sea la primera vez que las herramientas software necesarias se utilicen en la organización, a menudo suele ser necesario formar a los desarrolladores, y puede que se encuentre con algunos obstáculos. Si el equipo de desarrollo ha estado utilizando las mismas herramientas durante varios años, es posible que no pueda contar con algunas personas del equipo, que prefieren seguir utilizando las viejas herramientas en lugar de aprender a manejar las nuevas.

Una técnica de formación eficaz consiste en utilizar mentores. Un mentor puede realizar dos funciones. En primer lugar, puede ayudar en las tareas de desarrollo, gracias a sus conocimientos de fondo. En segundo lugar, un mentor puede transmitir parte de ese conocimiento al personal de desarrollo. A pesar de todo, la transferencia del conocimiento es una cuestión complicada, ya que muchos programadores excelentes son incapaces de

comunicar sus conocimientos a sus colaboradores. Es obvio que un mentor debe ser tanto un profesor como un técnico.

Por qué fallan los métodos tradicionales de diseño?

El proceso de escritura de software siempre ha sido una cuestión de acierto y fallo. Algunos de los trabajos de desarrollo tienen éxito mientras que muchos otros resultan inútiles. Los programadores y los administradores del desarrollo no tardaron mucho en reconocer que nuestro conocimiento del proceso de desarrollo del software es incompleto. Como resultado, se crearon las metodologías del software (prescripciones paso a paso para ayudar a los desarrolladores de software) para refinar la comprensión del proceso.

Para los desarrolladores y los administradores, una lectura esencial sobre el proceso de desarrollo del software es *The Mythical Man-Month*, Anniversary Edition, de Frederick Brooks, Jr. (Addison-Wesley, 1995). Aunque las observaciones de Brooks son el resultado de sus experiencias en la administración del desarrollo del proyecto OS/360 de IBM hace más de 20 años, gran parte de lo que dice sigue siendo aplicable. El hecho de que la industria siga peleándose con los mismos problemas identificados por Brooks, indica que las metodologías del software son una solución que dista mucho de ser completa.

La mayor parte de las metodologías sólo se ocupan de una parte de un proceso de desarrollo que cada vez es más complejo, aunque algunas intentan abarcarlo todo. Una de las más útiles ha sido el Ciclo de vida del desarrollo de sistemas (SDLC; System Development Life Cycle), que funcionaba bien para la construcción de sistemas tradicionales, pero que ya no es suficiente. SDLC y otras tecnologías fracasan en el entorno de negocios actual, donde se espera un desarrollo rápido, modificaciones frecuentes y una funcionalidad elevada de forma habitual. No podemos esperar años por una aplicación o semanas por un sencillo informe.

Como nosotros proponemos que el proceso de desarrollo del software debe cambiar, puede que se pregunte si deben cambiar también las funciones tradicionales asociadas con el desarrollo del software, como el analista de sistemas.

En el pasado, el analista de sistemas solía reunirse con el usuario para conocer sus necesidades y después redactaba una especificación que los desarrolladores debían seguir. El resultado de

este proceso era una aplicación que automatizaba algunas tareas para el usuario. En la actualidad, las oportunidades de optimizar las operaciones organizativas superan con creces la automatización de las tareas sencillas; las herramientas disponibles hoy en día permiten una mayor interacción con los propios usuarios de la aplicación. En consecuencia, la función de los analistas de sistemas ha cambiado: ahora se encargan de buscar nuevas oportunidades para mejorar la organización, y sus encuentros con los usuarios se parecen más a sesiones de diseño interactivas. La nueva filosofía se ocupa de estos cambios en las metodologías y funciones tradicionales.

Filosofía de diseño e implementación recomendable

Una metodología de software es distinta de una filosofía de diseño, igual que un algoritmo software es distinto de un heurístico: un algoritmo es un conjunto específico de pasos; un heurístico es una descripción más general del proceso por el que puede llegarse a una respuesta. Se utiliza un heurístico siempre que no puede desarrollarse un algoritmo que se ocupe de todos los casos de un problema concreto, o cuando un algoritmo que cubriera todos los casos posibles sería demasiado complicado.

Una metodología suele emplear reglas y pasos inflexibles por los que transcurre un diseño. Además, el objetivo de las metodologías típicas suele ser una parte del proceso de desarrollo, como un modelo de datos o un diagrama del flujo de datos. Debido al alto contenido en datos de las aplicaciones de empresa, la mayor parte de las metodologías se concentran en la base de datos, aunque también están disponibles algunas herramientas para el “proceso de negocios”.

Por el contrario, una buena filosofía de diseño proporciona un marco que sirve de guía a la creación del software en lugar de gestionarla a bajo nivel. Las filosofías de diseño no excluyen la utilización de metodologías. De hecho, ciertas metodologías son recomendables (por ejemplo, la metodología de base de datos IDEFIX utilizada por ERwin). Piense en la filosofía de diseño como en un saco de técnicas que usted y su organización pueden utilizar de acuerdo con los requisitos del problema en cuestión. Su objetivo debe ser adoptar herramientas, metodologías y directrices que funcionen para resolver los problemas de su organización. Los elementos como listas de verificación, documentos de especificación o

incluso una base de datos para realizar el seguimiento de los proyectos en desarrollo o las reglas de negocios también pueden jugar un papel importante en la filosofía de diseño.

Un buen proceso de diseño es iterativo, evolutivo y ajustado al problema de negocios original.

Diseño iterativo

Es necesario un proceso de diseño iterativo cuando una organización cambia a una nueva tecnología. Uno de los pasos más importantes de este proceso es el constante cambio del paradigma de un usuario, o la ventana por la que una persona ve el mundo.

Los usuarios típicos de un entorno tradicional tienen un paradigma que concuerda con el entorno del sistema actual. Este entorno suele estar basado en caracteres, necesita de la pulsación de muchas teclas, no ofrece consejos, puede tener o no tener informes que se presenten en pantalla o se impriman, etc. Si le pregunta a los usuarios lo que quieren de un sistema nuevo, adivina lo que le responderán? Quieren un sistema que funcione como el antiguo, pero sin los problemas de éste y con algunas funciones nuevas. En este contexto, los usuarios están limitados por su paradigma porque sólo pueden imaginarse la solución actual con algunas mejoras.

Las personas suelen dar pasos cortos al cambiar de paradigmas. Los pasos cortos suelen ser más seguros y con frecuencia tienen más éxito que las grandes zancadas. El proceso de diseño iterativo asume esta característica humana fundamental, permitiendo a los usuarios y desarrolladores modificar su paradigma de forma incremental. Hace muy poco que un proceso de diseño iterativo eficiente es realmente posible gracias a las herramientas RAD (Rapid Application Development; Desarrollo rápido de aplicaciones). Las herramientas RAD, más que ninguna otra, permiten realizar cambios rápidos que permiten un diseño del tipo “¿Qué ocurriría si ...?”. Por ejemplo, un buen analista de sistemas intenta entender cómo puede mejorar el entorno de negocios y entonces crea un prototipo de un sistema que puede demostrar al usuario. El usuario y el analista trabajan juntos para mejorar el prototipo de forma incremental. Se produce una oportunidad para un leve cambio de paradigma cada vez que el usuario ve una nueva implementación. Obviamente, cuanto menor sea el tiempo de cambio entre implementaciones, más rápido se producirá el cambio de paradigma. Las

metodologías tradicionales son demasiado rígidas para permitir este tipo de realimentación de forma cotidiana, y las herramientas RAD actuales permiten que el proceso tenga lugar en semanas en lugar de en años.

Diseño evolutivo

El proceso de diseño también debe ser evolutivo, en el sentido de que en cada iteración se añadan detalles a las especificaciones de trabajo. (En la página 42 puede encontrar los detalles sobre cómo crear un documento de especificaciones.) Durante la evolución del proceso, la aplicación cada vez se aproxima más a la resolución de los problemas de negocios definidos como objetivo.

El propio documento de especificaciones también debe evolucionar durante este proceso, y contener tanto la información original como los refinamientos realizados durante el desarrollo. El documento también debe contener los enlaces con la aplicación (lugar de almacenamiento y nombre del proyecto) y con la base de datos utilizada por la aplicación (nombre de la base de datos, nombre del servidor y cualquier información que resulte pertinente).

Probablemente es una buena idea incluir también en el documento de especificaciones un enlace activo mediante OLE con el modelo de datos. La utilización de un enlace activo permitirá actualizar el documento de especificaciones de forma automática cuando se modifique el modelo de datos.

Diseño ajustado al problema de negocios original

Un elemento clave que suele dejarse al margen en la mayoría de las especificaciones de diseño es el problema de negocios original que la aplicación intenta resolver. Este elemento debe ser el primero del documento de especificaciones. Una formulación clara del problema de negocios le permitirá remitir la aplicación al problema original con facilidad y documentar los fundamentos lógicos de la aplicación para los desarrolladores, administradores y usuarios futuros.

GLOSARIO

Términos Contables

- **Activo:** Está constituido por los recursos económicos propiedad de una Empresa que se espera generen beneficios en períodos futuros.
- **Pasivo:** Se da el nombre de Pasivo a los importes que adeuda una Empresa.
- **Capital:** El Capital o aportaciones de los Propietarios representa el Capital Neto o "interés" de los Propietarios en el Activo de una Empresa, y es igual al Activo Neto de la misma, es decir, el activo menos el pasivo.
- **Balance General:** Estado Financiero en el que se presentan el activo, el pasivo y la participación de los propietarios, de una empresa, en una fecha dada.
- **Estado de Resultados:** Estado financiero en el que se resumen
- los resultados de la operación de una empresa en un período dado mediante revelación de los ingresos devengados y los gastos en los que se incurre.
- **Balanza de Comprobación:** Informe interno con el que se verifica la igualdad de los débitos y créditos del Mayor.
- **Cuentas:** Son los registros que se elaboran para cada elemento o partida de activo, pasivo y capital de los propietarios. La cuenta sirve para registrar los incrementos y disminuciones que resultan de transacciones.
- **Cargos y Abonos:** Los cargos y abonos (o débitos y créditos) son una herramienta empleada por los contadores para incrementar o disminuir los saldos de las cuentas.
- **Saldo:** El saldo de una cuenta es la diferencia entre la suma total de sus cargos y la suma total de sus abonos. Una cuenta tendrá saldo deudor cuando su total de cargos sea mayor que su total de abonos, y tendrá saldo acreedor cuando el total de sus abonos sea mayor que el total de sus cargos.
- **Catálogo de Cuentas:** Es una lista detallada de las cuentas de una compañía, cada una con su número o código correspondiente.
- **Libro Diario:** Registro cronológico que representa el punto de entrada de las

transacciones al sistema contable formal de una compañía.

- **Mayor General:** Es en esencia un libro que contiene listas separadas para cada cuenta que aparece en los estados financieros de una organización.
- **Mayorización:** Consiste en la actualización de los saldos de las cuentas en el Libro Mayor a partir de los valores registrados en las partidas.
- **Cierre Contable:** Consiste en el cúmulo de actividades y operaciones necesarias para dar por finalizados los procesos contables de un período determinado.
- **Consolidación:** es el proceso de integración de la información contable (cuentas y saldos) de varias Unidades, para efectos de determinar la situación financiera de todo el conjunto para un período determinado.
- **Saldar Cuentas:** Consiste en poner a cero el saldo de una cuenta, realizando esto a través de una partida con un movimiento que tenga un abono con el mismo valor que el saldo de la cuenta (en caso de que sea saldo deudor), o con un cargo con el valor del saldo (en caso que tenga saldo acreedor), de manera que la diferencia entre las sumas de cargos y abonos de la cuenta sean cero.
- **Presupuesto:** Es un documento donde son establecidas metas concretas de ingresos y gastos proyectados para un período determinado (usualmente un año) a las Unidades de la Ciudadela y sus distintos Centros de Costo.
- **Rubros Presupuestales:** Son las distintas entidades y actividades para las cuales se proyecta que generen una cierta cantidad de ingresos y gastos a la Ciudadela, en sus distintas Unidades y Centros de Costo, para un período determinado.

Términos Informáticos

- **Sistema:** también llamado sistema Informático, sistema mecanizado, aplicación o módulo y se entiende como conjunto de datos, procesos y programas en un Computador que interactúan con el usuario para satisfacer sus demandas de información.
- **Hardware:** Se refiere a los equipos, accesorios y materiales que conforman un Computador o un conjunto de computadores (Red).
- **Software:** La secuencia lógica de comandos que conforman un programa de Computador,

el cual realiza una o más tareas específicas.

- **Base de Datos:** Una gran colección de registros almacenados e integrados que pueden ser mantenidos y manipulados con gran flexibilidad y que representa toda la información relacionada a un negocio o empresa.
- **Manejador de Base de Datos Relacional:** Es una herramienta de Software que permite al usuario crear, manipular y mantener una base de datos integrada de un negocio para producir información administrativa relevante, también llamado RDBMS por sus siglas en inglés (Relational Data Base Management System).
- **Herramientas de Desarrollo de Cuarta Generación:** Son lenguajes de programación de tipo no procedimental, llamados así porque permite a los programadores y usuarios especificar lo que la computadora debe de hacer sin definir cómo lo va a hacer, también llamados lenguajes de alto nivel o 4GL por sus siglas en inglés (4th Generation Languages).
- **Red:** Conjunto de computadoras relacionadas entre sí (interconectadas), que se comunican para compartir y transferir información (conjunto de datos).
- **Visual Basic:** Lenguaje de Cuarta Generación orientado a objetos y eventos, el cual permite desarrollar aplicaciones para Windows de una forma visual, en la cual se construye el interfaz gráfico de la aplicación a partir de Ventanas (Formas) y objetos (controles), a los cuales se les asigna procedimientos en lenguaje BASIC que le indicará a cada elemento como reaccionar tanto a un suceso (evento) interno del sistema, como externo (por el usuario).
- **TCP/IP:** (Transmission control protocol /internet protocol) Un protocolo de comunicaciones basado ampliamente en los sistemas operativos UNIX. TCP/IP fue originalmente desarrollado por ARPA, una agencia del departamento de defensa de los E.U
- **Sybase SQL Server:** Es un Manejador de Bases de Datos Relacionales desarrollado por Sybase y que está diseñado para manejar bases de datos a nivel empresarial. Ha sido adaptado a diferentes plataformas, siendo UNIX una de las más aceptadas para el manejo de bases de datos. Permite el control de varias Bases de Datos simultáneamente y posee un control de seguridades centralizado, en donde los usuarios son definidos a nivel del Servidor de Bases de Datos. Sybase SQL Server ha sido desarrollado para trabajar en

arquitecturas cliente-servidor, no es un producto que pueda ser utilizado por si solo, sino más bien en ambientes de redes y multiusuarios.

- **SQL :** Un lenguaje de bases de datos que permite a los usuarios manipular la información utilizando un lenguaje parecido al inglés que es interpretado por un manejador de bases de datos con el propósito de ver la base de datos en una variedad de formas. SQL fue originalmente desarrollado por IBM para usarse con sus sistemas de bases de datos relacionales.
- **Conexión en línea:** Se refiere a la transmisión de los datos en tiempo real a través de un medio de comunicación.
- **Usuario:** Usualmente se refiere a la persona que interactúa con una aplicación o programa.
- **Topología de Red:** Diseño y estructura de una Red, definiendo la organización de los componentes y medios de comunicación en esta.
- **UNIX:** Un sistema operativo multiusuario y multitarea que corre en una amplia variedad de plataformas de hardware. Existen dos variantes principales de UNIX, una disponible para AT&T (System V), y la otra basada en las extensiones de Berkeley. UNIX fue primeramente desarrollado en los laboratorios Bell de AT&T a principios de los setenta y fue escrito en lenguaje de programación C.
- **Puerto Lógico (UNIX):** En los sistemas operativos UNIX, se refiere a líneas de comunicación que son asignados por el sistema operativo a un servicio de transmisión de datos, también son llamados TCP sockets.
- **Protocolo de Comunicación:** Un esquema usado para transmitir datos entre computadores o entre computadoras y sus periféricos. El esquema es definido por el hardware y el software usado para transmisión.
- **GUI (Interfaz Gráfico de Usuario):** Es un sistema de interface con el usuario ya sea para un sistema operativo o un programa que se basa en el uso de dibujos y objetos para la comunicación de la información. Es el contraste con los interfaces tradicionales basados en texto, que son manejados a través de comandos o menús.
- **Windows:** Un sistema operativo desarrollado por Microsoft, basado en un ambiente gráfico que permite la ejecución de programas y manipulación de los recursos de una PC.
- **ODBC (Open DataBase Connectivity):** Interfaz estándar definido por Microsoft para conectar aplicaciones cliente a sistemas manejadores de bases de datos en ambientes

Windows y Windows NT.

- **Control (Objeto):** Es cualquier cosa, real o abstracta, acerca de la cual almacenamos datos y los métodos que controlan dichos datos. Es un objeto que se dibuja en una forma y que permite o habilita la interacción del usuario con un programa o aplicación.
- **Forma:** Es una ventana que sirve de fondo para los controles y para los gráficos situados sobre la misma.
- **Ventana:** Una porción de la pantalla a la cual se limitan las acciones tomadas.
- **Evento:** Es una acción o un suceso reconocido por un objeto, como el click del mouse o presionar una tecla, y para el cual puede escribirse código para responder. Los eventos pueden ocurrir como resultado de acción del usuario o código de programación, o pueden ser ‘disparados’ por el sistema.
- **Tabla:** En una base de datos relacional toda la información es almacenada en tablas, las cuales se componen de filas y columnas. Cada tabla contiene una o más columnas, y a cada columna le es asignado un tipo de dato, como un número entero, una secuencia de caracteres (texto) o una fecha. Cada fila en la tabla tiene un valor para cada columna.
- **Vista:** Consiste en una tabla ‘calculada’ o hecha a través del requerimiento de una combinación de columnas de varias tablas relacionadas. Los datos contenidos en la vista son buscados y/o calculados cada vez que se accesa la vista, ya que es almacenada únicamente la definición de ésta y no los datos que contiene. Usualmente solo se puede consultar y no alterar sus datos.
- **Regla:** Condición que debe cumplir el valor a almacenar en una columna de una tabla a fin de mantener la integridad de los datos.
- **Trigger:** Procedimiento o Rutina que se ejecuta automáticamente cuando se insertan, borran o modifican los datos de la tabla a la cual está ligado.
- **Procedimiento Almacenado:** Procedimiento o rutina compleja que realiza alguna operación específica sobre los datos de la base de datos, que puede recibir parámetros y dar resultados según las especificaciones dadas por el sistema o quien lo creó.
- **Default:** Valor por omisión en el cual se especifica el dato que será ingresado en una columna si el usuario no le especifica al momento de insertar datos.
- **Base de Datos:** Colección de Tablas conteniendo información relacionada y que además puede contener objetos que rigen la forma en que se almacena y relaciona la información,

tales como reglas, Triggers, procedimientos almacenados, Defaults, etc.

- **Herramientas CASE:** (Siglas en inglés de Ingeniería de Sistemas apoyada por Computadora), proporciona formas orientadas gráficamente para expresar planos, modelos y diseños.

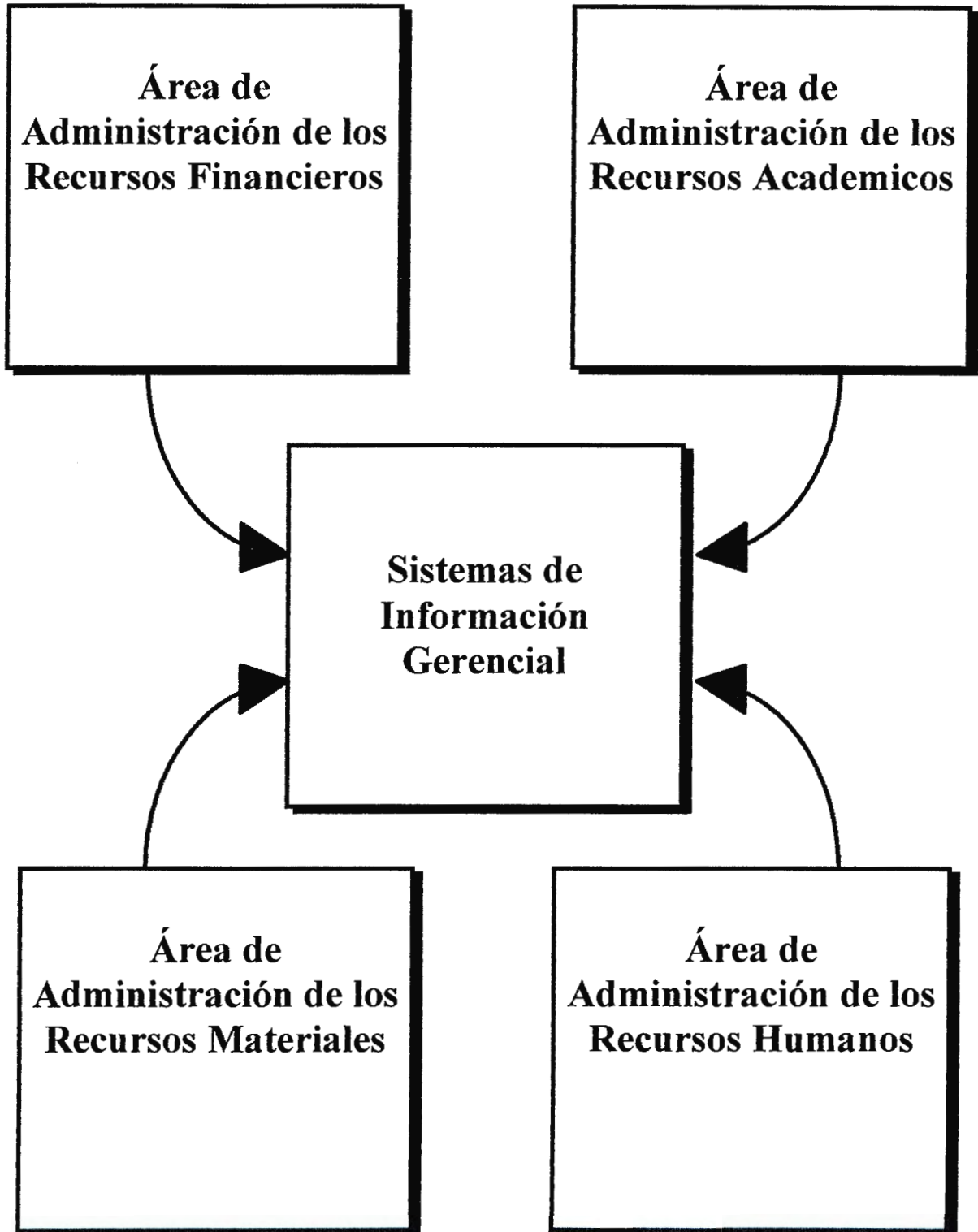
BIBLIOGRAFÍA

- ANAYA Gutiérrez Noël, Manual de Aplicación de Cuentas para la Institución Salesiana,
- FLORES Elías Lara, "Primer Curso de Contabilidad", 10a Edición.
- HUTCHINSON Sarah E. - SAWYER Stacey C., "Computers, The User perspective", 2a Edición, 1990.
- KROENKE David, "Management Information Systems", 1a Edición, 1989
- STONER James A.F. - WANKEL Charles, "Administración", 3a Edición., 1987.
- VASQUEZ Lic. Guillermo, "Plan de Desarrollo Informático de la UDB", Centro de Cómputo-CITT.
- WYATT Allen L. , "Computers Professional's Dictionary", 1a Edición, 1990
- Kenneth L. Spencer, Ken Miller, "Programación Cliente/Servidor con Microsoft Visual Basic", Microsoft Press, 1996.

Fuentes de Información en InterNet:

- Microsoft Web Site: <http://www.microsoft.com>
- Sybase Web Site: <http://www.sybase.com>

Anexo A: Areas Administrativas Consideradas en el Plan de Desarrollo Informático de la Ciudadela Don Bosco.



Anexo B: Aplicaciones a Desarrollar para el Área de Administración de los Recursos Académicos

Registro
Academico UDB

Asesoría e
Inscripción UDB

Administración de
Recursos (Horarios)

Registro Academico
Colegio Don Bosco

Administración de
Recursos Centro de
Computo

Control de Horas
Sociales

Registro Academico
Centro Formación
Prof.

Control de Usuarios
de Biblioteca

Anexo C: Aplicaciones a Desarrollar para el Área de Administración de los Recursos Financieros.

Cuentas por Pagar

Control del IVA

Cuentas por Cobrar

Colecturía y Control de Pagos

Contabilidad General

Control Presupuestal

Disponibilidad y Bancos

Planillas

Anexo D: Aplicaciones a Desarrollar para el Área de Administración de los Recursos Humanos.

Reclutamiento y
Preselección de
Personal

Expediente de
Empleados

Planillas y Control de
Pagos

Control de Asistencia
de Personal

Registro General de
Personas

Anexo E: Aplicaciones a Desarrollar para el Área de Administración de los Recursos Materiales.

Activos Fijos

Control de Bodegas

Inventario de
Recursos Consumibles

Inventario de Libros
(Biblioteca)

Inventario de Puntos
de Venta

Anexo F: Reportes Generados por el Actual Sistema de Contabilidad de la Ciudadela Don Bosco.

A Continuación se presentan ejemplos de los reportes financieros que son generados por el actual sistema contable de la Ciudadela Don Bosco. Las cifras presentadas en ellos han sido representadas por ceros (0.00) por razones de confidencialidad de los datos.

LIBRO MAYOR
ACTUALIZADO HASTA LA FECHA DEL : 01/01/96

PAG.

NOMBRE DE LA CUENTA	MES ANTERIOR	C A R G O S	A B O N O S	CAMBIO NETO	NUEVO SALDO	SALDO
DISPONIBLE	0.00				0.00	DEBI
CAJA	0.00				0.00	DEBI
CAJA CHICA	0.00				0.00	DEBI
ADMINISTRACION UDB	0.00				0.00	DEBI
ADMINISTRACION CITT	0.00				0.00	DEBI
OFICINA RECTORIA	0.00				0.00	DEBI
OFICINA HUMANIDADES Y TEOLOGIA	0.00				0.00	DEBI
PROYECTO UDB-MINED	0.00				0.00	DEBI
DIRECTOR CITT	0.00				0.00	DEBI
LIBRERIA UDB	0.00				0.00	DEBI
CAJA [CHEQUES RECHAZADOS]	0.00				0.00	DEBI
ANA LETICIA SOLORZANO	0.00				0.00	DEBI
FERROELECTRO	0.00				0.00	DEBI
ALEX FUNES	0.00				0.00	DEBI
CH#0938	0.00				0.00	DEBI
ADMN GENERAL CDB(PAGO COMPUTADORA)	0.00				0.00	DEBI
SANIPLAN	0.00				0.00	DEBI
INSTITUTO RICALDONE	0.00				0.00	DEBI
PADRE CESAR MONTENEGRO	0.00				0.00	DEBI
TELESIS	0.00				0.00	DEBI
JOSE MARIO MARTINEZ	0.00				0.00	DEBI
JOSE ANTONIO RODRIGUEZ	0.00				0.00	DEBI
DIDISA	0.00				0.00	DEBI
AZUCENA MARISOL ARIAS	0.00				0.00	DEBI
BANCOS	0.00				0.00	DEBI
BANCO CUSCATLAN	0.00				0.00	DEBI
CUENTA CORRIENTE	0.00				0.00	DEBI
1 UNIVERSIDAD DON BOSCO	0.00				0.00	DEBI
2 UNIVERSIDAD DON BOSCO	0.00				0.00	DEBI
CUENTA DE AHORRO	0.00				0.00	DEBI
1 UNIVERSIDAD DON BOSCO	0.00				0.00	DEBI
CUENTA A PLAZO FIJO	0.00				0.00	DEBI
1 FONDOS BECAS	0.00				0.00	DEBI
2 CUENTA PRESTACIONES	0.00				0.00	DEBI
3 CUENTA PRESTACIONES	0.00				0.00	DEBI
4 APORTES PROMIPE FAES PRESTACIONES	0.00				0.00	DEBI
5 CUENTA PRESTACIONES	0.00				0.00	DEBI
CERTIFICADO	0.00				0.00	DEBI
BANCASA	0.00				0.00	DEBI
CUENTA CORRIENTE	0.00				0.00	DEBI
CUENTA DE AHORRO	0.00				0.00	DEBI
1 UNIVERSIDAD DON BOSCO	0.00				0.00	DEBI
2 UNIVERSIDAD DON BOSCO	0.00				0.00	DEBI
CUENTA A PLAZO FIJO	0.00				0.00	DEBI
BANCO SALVADOREÑO	0.00				0.00	DEBI
CUENTA CORRIENTE	0.00				0.00	DEBI
CUENTA DE AHORRO	0.00				0.00	DEBI

BALANCE DE COMPROBACION

AL : 01/01/96

PAG. 1

T A	NOMBRE DE LA CUENTA	NIVEL 5	NIVEL 4	NIVEL 3	NIVEL 2	NIVEL 1
-001	ADMINISTRACION UDB		0.00			
-002	ADMINISTRACION CITT		0.00			
-003	OFICINA RECTORIA		0.00			
-004	OFICINA HUMANIDADES Y TEOLOGIA		0.00			
-005	PROYECTO UDB-MINED		0.00			
	CAJA CHICA			0.00		
-004	CH#0938		0.00			
-004	INSTITUTO RICARDONE		0.00			
	CAJA (CHEQUES RECHAZADOS)			0.00		
	CAJA		0.00		0.00	
-001-01	UNIVERSIDAD DON BOSCO	0.00				
-001-01	UNIVERSIDAD DON BOSCO	0.00				
-001	CUENTA CORRIENTE		0.00			
-002-01	UNIVERSIDAD DON BOSCO	0.00				
-002	CUENTA DE AHORRO		0.00			
-003-01	FONDOS BECAS	0.00				
-003-02	APORTE PROMIPE FAES (PRESTACIONES)	0.00				
-003	CUENTA A PLAZO FIJO		0.00			
	BANCO CUSCATLAN			0.00		
-002-01	UNIVERSIDAD DON BOSCO	0.00				
-002-02	UNIVERSIDAD DON BOSCO	0.00				
-002	CUENTA DE AHORRO		0.00			
	BANCASA			0.00		
	BANCOS				0.00	
	DISPONIBLE					0.00
-004	UNIVERSIDAD DON BOSCO		0.00			
	ESCOLARIDADES			0.00		
	CLIENTES MOROSOS				0.00	
	CUENTAS POR COBRAR					0.00
-001	PABLO ANTONIO RAMIREZ		0.00			

UNIVERSIDAD DON BOSCO '95
BALANCE GENERAL

AL 31 DE JULIO DE 1995

ACTIVO		PASIVO		
	0.00	EXIGIBLES A CORTO PLAZO		0.00
	0.00	CUENTAS POR PAGAR	0.00	
	0.00	CASAS SALESIANAS	0.00	
	-----	PRESTAMOS BANCARIOS	0.00	
	0.00		-----	
POR COBRAR	0.00			
ARIOS	0.00	PRESTAMOS A LARGO PLAZO		0.00
SIANAS	0.00			
S	0.00	OTRAS CUENTAS DE PASIVO		0.00
EN CAMINO	0.00	TRANSITORIOS (PASIVO)	0.00	
	-----	RESERVA LABORAL	0.00	
	0.00	OTRAS RESERVAS	0.00	
	0.00	CREDITOS DIFERIDOS	0.00	
	-----		-----	
	0.00	PATRIMONIO		0.00
	0.00	CAPITAL	0.00	
PARA DEPRECIACION	0.00			
	-----	EXCEDENTES DE OPERACIONES	0.00	
	0.00	EXCEDENTES ACUMULADOS	0.00	
IONES	0.00	EXCEDENTES DEL EJERCICIO	0.00	
EN TRANSITO	0.00		-----	

NTAS DE ACTIVO	0.00			
Y TRANSITORIOS	0.00			
ES	0.00			

ACTIVO	0.00	TOTAL DEL PASIVO		0.00
	=====			=====

DIRECTOR

F. _____
P. JOSE MANUEL SOLANO
ADMON. GRAL.

F. _____
LIC. MANUEL DE J. VELASCO
ADMINISTRADOR U.D.B.

F. _____
VICTOR JAVIER SERRANO
CONTADOR

Anexo G: Reportes del Sistema COP.

Listado de Periodos Contables Definidos

		Fecha	Fecha	
	periodo	Inicio	Fin	Nombre del Periodo
5	12	1/12/1995	31/12/1995	Diciembre 1995
5	1	1/01/1996	31/01/1996	Enero 1996
5	2	1/02/1996	28/02/1996	Febrero 1996
5	3	1/03/1996	31/03/1996	Marzo 1996
5	4	1/04/1996	30/04/1996	Abril 1996
6	5	1/05/1996	31/05/1996	Mayo 1996
6	6	1/06/1996	30/06/1996	Junio 1996
6	7	1/07/1996	31/07/1996	Julio 1996
6	8	1/08/1996	31/08/1996	Agosto 1996
6	9	1/09/1996	30/09/1996	Septiembre 1996
6	10	1/10/1996	31/10/1996	Octubre 1996
6	11	1/11/1996	30/11/1996	Noviembre 1996
6	12	1/12/1996	31/12/1996	Diciembre 1996
7	1	1/01/1997	31/01/1997	Enero 1997
7	2	1/02/1997	28/02/1997	Febrero 1997
7	3	1/03/1997	31/03/1997	Marzo 1997
7	4	1/04/1997	30/04/1997	Abirl 1997
7	5	1/05/1997	31/05/1997	Mayo 1997
7	6	1/06/1997	30/06/1997	Junio 1997
7	7	1/07/1997	31/07/1997	Julio de 1997

Listado de Tipos de Cuentas Contables Definidos

Descripcion del Tipo	Clase	Tipo Generico
Activo	D	A
Capital	C	C
Egresos	D	G
Ingreso	C	I
Cuentas de Orden	C	A
Pasivo	C	P

Listado de Tipos de Partidas

Tipo	Descripcion del Tipo	Indica Partida de Cierre
	Cheque	No
	Partida de Diario	No
	Partida de Egresos	No
	Partida de Ingresos	No

Catalogo de Cuentas Contables

Contabilidad General del CITT

e Tipo: ACT Activo

Cuenta	Concepto	Det Nivel	Consolida en	
			Contat	Cod. Cuenta
	DISPONIBLE			1
	CAJA			2
	CAJA GENERAL	X		3
	CAJA CHICA			3
	PROYECTO MINED(DEMANDA SOCIAL)	X		4
	ADMINISTRACION CITT	X		4
	PADRE PEDRO JOSE GARCIA	X		4
	PROYECTO HNNBB-UDB	X		4
	DIRECTOR CITT	X		4
	CAJA (CHEQUES RECHAZADOS)			3
	BANCO CUSCATLAN			4
	CUENTA CORRIENTE			5
	CUENTA # 1509-00849	X		6
	BANCOS			2
	BANCO CUSCATLAN			3
	CUENTA CORRIENTE			4
	CTA. # 1509-00823/PROYECTO	X		5
	CTA. # 1509-00849/CITT	X		5
	CUENTA DE AHORRO			4
	CTA.# 15-30289-3	X		5
	CUENTA A PLAZO FIJO	X		4
	CERTIFICADO	X		4
	BANCASA			3
	CUENTA CORRIENTE	X		4
	CUENTA DE AHORRO			4
	CTA.# 19-1-189292/CITT	X		5
	CUENTA A PLAZO FIJO	X		4
	BANCO SALVADORENO			3
	CUENTA CORRIENTE	X		4
	CUENTA DE AHORRO	X		4
	CUENTA A PLAZO FIJO	X		4
	jjkaksd	X		5
	CUENTAS POR COBRAR			1
	CLIENTES			2
	DIVERSOS			3
	CAJAS Y BOLSAS	X		4
	SIEMENS	X		4
	INDUSTRIAS DE CENTROAMERICA S.A.	X		4
	PNUD	X		4
	HOSPITAL DE NIÑOS BENJAMIN BLOOM	X		4
	SANIPLAN	X		4
	TALLER DE MECANICA			3
	INSINCA	X		4
	GBM	X		4
	COLIMA	X		4
	CORCHO Y LATA	X		4
	PROYECTO GTZ/ISRI	X		4
	EL DORADO SA	X		4
	AGROPECUARIA	X		4
	COMPANIA HOTELERA	X		4
	MEISA	X		4
	PROFESORES ASOCIADOS	X		4
	ACM INGENIEROS	X		4
	PLANTA DE TORREFACCION DE CAFE,S.A.	X		4
	DIDISA	X		4
	DEPTO. DE ADMINISTRACION			3
	P.PEDRO GARCIA	X		4
	TALLER DE ELECTRICIDAD			3
	NESTLE	X		4
	PLOSEMSA	X		4
	ALMACENES SIMAN	X		4
	ANTEL	X		4
	TALLER DE COMPUTACION			3
	NESTLE	X		4
	COLEGIO SANTA CECILIA	X		4
	PROMIPE BID	X		4
	UNIVERSIDAD DON BOSCO	X		4
	BIMBO	X		4
	REVIT	X		4
	M.A.V.	X		4
	DISTRIBUIDORA ELECTRICA	X		4
	PRODUCTOS M.	X		4
	PRODUCTOS BOCADELLY	X		4
	LABORATORIO TERAPEUTICOS MEDICOS	X		4
	INST.NAC.EMPLEADOS PUBLICOS	X		4
	DR. EDUARDO	X		4
	CAMARA DE COMERCIO	X		4
	EMPRESA AGOAGEI DE R.L.	X		4
	COLEGIO DON BOSCO	X		4
	EDITORIAL INTEGRADA, S.A.	X		4
	TALLER DE ELECTRONICA			3
	CIGARRERIA MORAZAN	X		4

Catalogo de Cuentas Contables

Contabilidad General del CITT

F.A DALTON	X	4
LA CONSTANCIA	X	4
TELSINCA	X	4
INDUSTRIAS CRISTAL	X	4
LABORATORIOS LOPEZ	X	4
LA SALUD	X	4
SISTEMAS FINANCIEROS,S.A.	X	4
CEL	X	4
LAB.TERAPEUTICOS MEDICOS	X	4
DEPARTAMENTO DE BIOMEDICA	X	3
DEPARTAMENTO DE AREAS BASICAS	X	3
DPTO. ESCUELA DE IDIOMAS		3
KIMBERLY CLARK	X	4
NABISCO	X	4
ADMON.CITT		3
COCA-COLA	X	4
PROYECTOS		3
HNNBB-UDB	X	4
SANIPLAN	X	4
CLIENTES MOROSOS		2
TALLER DE COMPUTACION		3
VIRGLIO MARQUINA	X	4
DEUDORES VARIOS		1
EMPLEADOS		2
ADMINISTRATIVO		3
VIOLETA CHACON	X	4
MANUEL MARENCO	X	4
SANDRA MANZON	X	4
CRISTHIAN BUENOA	X	4
LINDA SAYONARA	X	4
JOSE ARMANDO FUENTES)	X	4
MERCEDES NOLASCO	X	4
ROSALIDIA ARTIGA	X	4
NORA GIRON	X	4
FAUSTINO VASQUEZ	X	4
VIRGINIA FUENTES	X	4
JOSE SANTOS MARAVILLA	X	4
JOSE AMILGAR CRUZ	X	4
SANBINA ALVAREZ	X	4
NOE MARTINEZ	X	4
JOSE LEONSO LOPEZ	X	4
JOSE LUIS CATILLO ULLOA	X	4
JOSE ANTONIO MEJIA	X	4
PATRICIA MENJIVAR	X	4
LUIS ROBERTO GALINDO	X	4
JOSE ORTIZ VAQUIS	X	4
WILLIAM ARMANDO RODRIGUEZ	X	4
MARCO ANTONIO VILLALTA	X	4
FEDERICO MENDEZ MENDEZ	X	4
ISMAEL GOMEZ	X	4
LEONEL GUZMAN RODRIGUEZ	X	4
CARLOS GUILLERMO VASQUEZ	X	4
ROGOBERTO OSORIO	X	4
ROBERTO CARLOS NAVAS	X	4
CARLOS ROBERTO LOPEZ	X	4
WILLIAM ALEXIS REBOLONE QUINTANILLA	X	4
GILBERTO ANTONIO LARA SOSA	X	4
MANUEL DE JESUS ORELLANA	X	4
NELSON QUINTANILLA	X	4
GERBER OMAR PINEDA	X	4
PABLO ANTONIO QUINTANILLA	X	4
OSCAR VILLALTA	X	4
MILTON LANDOS VASQUEZD	X	4
NESTOR LOZANO LEIVA	X	4
VICTOR MANUEL CALLEJAS	X	4
WILBER MARTINEZ ABARCA	X	4
FRANCISCO ANTONIO RODRIGUEZ	X	4
RICARDO GONZALEZ	X	4
JOAQUIN ANTONIO HERNANDEZ	X	4
JOSE MAURICIO MARQUEZ	X	4
GUZTAVO ENRIQUE RUIZ CERNA	X	4
JUAN RUBERN BONILLA	X	4
DANIEL CRUZ OCHOA	X	4
ANA MARIA SORIANO	X	4
MARIO EDURADO GALDAMEZ	X	4
ANGEL ERNESTO SORIANO	X	4
JORGE ERNESTO LEMUS	X	4
FRANCISCO JOSE RIVERA	X	4
NESTOR MENDEZ FUENTES	X	4
HAROLD JHONSON	X	4
SONIA	X	4
CANDELARIO ANSELMO	X	4
LUIS ORLANDO ALFARO	X	4
CARLOS AGUILAR	X	4
CARLOS ALBERTO NAJERA PEREZ	X	4
ENEYDA DEL CARMEN GUARDADO	X	4
ALEXANDER TORRES	X	4
VIRGINIA TORRES	X	4
COMUNIDAD	X	3
DEUDORES PERSONALES(ANTICIPOS SUELDOS)	X	2

Catalogo de Cuentas Contables

Contabilidad General del CITT

CASAS SALESIANAS		2
COLEGIO DON BOSCO	X	3
UNIVERSIDAD DON BOSCO	X	3
COMUNIDAD	X	3
COMEDOR MAMA MARGARITA	X	3
OTROS DEUDORES		2
OTROS	X	3
CREDITO FISCAL - I V A	X	3
ACTIVO FIJO		1
BIENES INMUEBLES		2
TERRENOS	X	3
EDIFICIOS	X	3
INSTALACIONES		3
TALLER MECANICA		4
AREA SOLDADURA	X	5
AREAS BASICAS		4
AREA DE FISICA	X	5
BIENES MUEBLES		2
MOBILIARIO Y EQUIPO		3
MOBILIARIO	X	4
EQUIPO		4
OFICINA DE PROYETCO	X	5
CENTRO DE COMPUTO	X	5
MAQUINARIA Y EQUIPO		3
MAQUINARIA		4
MECANICA	X	5
CONSTRUCCIONES	X	1
INVERSIONES	X	1
DIFERIDOS Y TRANSITORIOS		1
DIFERIDOS		2
PAPELERIA Y UTILES		3
PAPELERIA		4
CINTAS IMPRESORAS FX850	X	5
LIBRETAS DE TAQUIMECANOGRAFIA	X	5
CAJA DE ACETATOS	X	5
PAPEL BOND TAMAYO CARTA	X	5
PAPEL BOND TAMAYO OFICIO	X	5
LAPICES	X	5
BOLIGRAFOS	X	5
ADHESIVOS	X	5
PAPEL PARA FAX	X	5
CINTAS IMPRESORAS PANASONIC KXP115	X	5
FOLDERS	X	5
BORRADOR LAPIZ	X	5
CORRECTORES	X	5
DICCIONARIOS	X	5
GRAPAS	X	5
PAPEL PERIODICO	X	5
DICCIONARIOS WORD BY WORD	X	5
CINTAS IMPRESORAS LQ 2500	X	5
FOLDERS SUPENSORES	X	5
CLIPS	X	5
PAPEL CONTINUO	X	5
FASTENERS	X	5
THONER	X	5
CINTAS IMPRESORAS CITIZEN	X	5
SOBRES MANILA	X	5
CINTAS IMPRESORAS 1050	X	5
FOLDERS TAMAYO CARTA	X	5
CAJAS DISKETS	X	5
POST-IT'S	X	5
CARTULINA	X	5
ARCHIVADORES	X	5
PAPEL CARBON	X	5
CINTAS IMPRESORAS 1060	X	5
FECHEROS	X	5
PILAS	X	5
LIJAS	X	5
S	X	5
PAPEL MEMBRETADO	X	5
CINTAS IMPRESORAS IBM # 111/XL	X	5
CINTAS IMPRESORAS IBM # 4234	X	5
CINTAS EPSON 77621 SERVER	X	5
AGENDAS	X	5
TINTA	X	5
CHINCHES	X	5
PLUMONES FLOURECENTES	X	5
BORRADORES DE PIZARRA	X	5
PLUMONES DE PIZARRA	X	5
ROLLOS DE TIRRO	X	5
UTILES		4
REGLAS	X	5
TJERAS	X	5
ENGRAPADORAS	X	5
DISPENSADORES	X	5
SACAPUNTAS	X	5
PAPELERAS	X	5
PERFORADORAS	X	5
PLUMONES	X	5
ALMOADILLAS	X	5

Catalogo de Cuentas Contables

Contabilidad General del CITT

	SACAGRAPAS	X	5	
	OTROS GASTOS ANTICIPADOS	X	3	
	TRANSITORIOS		2	
	CUENTAS POR LIQUIDAR		3	
	PROYECTO MINED	X	4	
	UNIVERSIDAD DON BOSCO	X	4	
	PROYECTO EDUCACION MEDIA	X	4	
	MERCADERIA EN CAMINO	X	3	
	MAQUINARIA Y EQUIPO EN CAMINO	X	3	
	ASEO Y HIGIENE		3	
	ASEO	X	4	
	COSTOS DE PRODUCCION		2	
	COSTOS DIRECTOS		3	
	MATERIA PRIMA	X	4	
	INVENTARIOS		1	
	LIBRERIA		2	
	LIBROS		3	
	LIBROS SPECTRUM	X	4	
	FOLLETOS SERVICIOS SOCIAL	X	4	
	VIRGINIA FUENTES	X	5	
	LUIS ESCOBAR (BLOOM)	X	5	
Cuentas	283			

Tipo: CAP Capital

Cuenta	Concepto	Det Nive	Consolida en	
			Contat	Cod. Cuenta
	CAPITAL		1	
	DONACIONES		2	
	OTROS DONANTES NACIONALES	X	3	
Cuentas	3			

Tipo: EGR Egresos

Cuenta	Concepto	Det Nive	Consolida en	
			Contat	Cod. Cuenta
	CUENTA DE GASTOS		1	
	ADMINISTRACION GENERAL DEL CITT		2	
	GASTOS ADMINISTRATIVOS		3	
	DIRECCION CITT		4	
	ALIMENTACION	X	5	
	ENERGIA ELECTRICA	X	5	
	COMBUSTIBLE		5	
	COMBUSTIBLE PARA VEHICULO	X	6	
	COMBUSTIBLE PARA VEHICULO	X	6	
	PAPELERIA Y UTILES		5	
	PAPELERIA	X	6	
	MATERIAL PARA COMPUTADORA	X	6	
	ASISTENCIA MEDICA	X	5	
	TELEFONO-CORREO		5	
	TELEFONO	X	6	
	CORREO	X	6	
	VIAJES Y VIATICOS		5	
	VIAJES	X	6	
	VIATICOS	X	6	
	MTTO. MOBILIARIO Y EQUIPO	X	5	
	MTTO. DE VEHICULOS	X	5	
	SUELDOS		5	
	SUELDOS DE OFICINA	X	6	
	VACACIONES		5	
	VACACIONES DE OFICINA	X	6	
	AGUINALDO		5	
	AGUINALDOS DE OFICINA	X	6	
	INDEMNIZACIONES		5	
	INDEMNIZACIONES DE OFICINA	X	6	
	HONORARIOS		5	
	HONORARIOS DE OFICINA	X	6	
	ISSS Y FSV (CUOTA PATRONAL)	X	5	
	ATENCIONES DIVERSAS	X	5	
	BONIFICACIONES		5	
	BONIFICACIONES DE OFICINA	X	6	
	HORAS EXTRAS		5	
	HORAS EXTRAS OFICINA	X	6	
	SUSCRIPCIONES	X	5	
	GTSO DE REPRESENTACION	X	5	
	CAPACITACION DE PERSONAL	X	5	
	CUOTA PATRONAL DE INSAFORP	X	5	
	VARIOS	X	5	
	ATENCIONES		5	
	ATENCION A PERSONAL	X	6	
	ATENCION A PADRES SALESIANOS	X	6	
	SUB-DIRECCION TECNICA CITT		4	
	ENERGIA ELECTRICA	X	5	

LIBRO AUXILIAR DE CUENTAS

Desde cuenta 10-01-02-001 hasta cuenta 8

Del 1 de Enero al 3 de Marzo de 1996

CHA MOVIMIENTOS	CARGOS	ABONOS
1 Administración UDB		
01/19 CARGO POR 1000 A ADMINISTRACION UDB	1,000.00	0.00
01/19 no se	0.00	1,500.00
01/19 cargo a administracion udb por 100000.00	100,000.00	0.00
02/19 Abono por 1000 a caja administracion udb	0.00	1,000.00
02/19 cargo por 1000 a caja admon udb	1,000.00	0.00
03/19 cargo de 1000 en cuenta 10-01-02-001 (administracion udb)	1,000.00	0.00
03/19 cargo de 1000	1,000.00	0.00
Total Cuenta :	104,000.00	2,500.00
2 Administración CITT		
01/19 CARGO POR 1000 A ADMINISTRACION CITT	1,000.00	0.00
01/19 CARGO POR 1000 EN ADMINISTRACION DEL CITT	1,000.00	0.00
01/19 sdfasdasdv asd adsf	1,500.00	0.00
02/19 cargo por 1000 a admon citt	1,000.00	0.00
03/19 cargo en cuenta 10-01-02-002 que pertenece a la Administracion del CITT	1,000.00	0.00
03/19 abono de 1000	0.00	1,000.00
Total Cuenta :	5,500.00	1,000.00
3 Oficina Rectoría (Ing. Huguet)		
01/19 CARGO POR 500 POR OFICINA RECTORIA	2,500.00	0.00
02/19 cargo por 1000 a oficina rectoria	1,000.00	0.00
03/19 Cargo a la cuenta 10-01-02-003 asignada a la oficina de rectoría (o sea al ing. Huguet)	1,000.00	0.00
Total Cuenta :	4,500.00	0.00
4 Cheque # 98989898		
01/19 CARGO POR 700 A CHEQUE #98989898	700.00	0.00
01/19 no se	0.00	100.00
01/19 cargo a cheque número 12030192301	1,000.00	0.00
02/19 cargo por 1000 a cheque #98989898	1,000.00	0.00
Total Cuenta :	2,700.00	100.00
7 Instituto Ricaldone		
01/19 ABONO POR INSTITUTO RICALDONE	0.00	3,000.00
02/19 cargo por 1000 a instituto ricaldone	1,000.00	0.00
Total Cuenta :	1,000.00	3,000.00
10-01 Cuenta Corriente Universidad Don Bosco #81293819208192391231		
01/19 ABONO POR 2000 A CUENTA CORRIENTE UDB	0.00	2,000.00
01/19 CARGO POR 10,000.00 A UNA DE LAS CUENTAS CORRIENTE	10,000.00	0.00
01/19 cargo por 1000 en cuenta corriente universidad don bosco	1,000.00	0.00
01/19 Cargo a cuenta corriente UDB por 10,000.00	10,000.00	0.00
01/19 Retiro	0.00	100.00
01/19 Cargo a cuenta corriente udb	1,000.00	0.00
01/19 Cargo a cuenta corriente udb	1,000.00	0.00
01/19 Pago de varias cosas con cheque de la cuenta corriente 1 de la udb	0.00	2,000.00
01/19 dajlkdsjfkldas	0.00	20,000.00
01/19 abono por 1000	0.00	1,000.00
01/19 KKLAJSDLKFJALSJFLKASDF	1,000.00	0.00
01/19 cargo por 10000 a cuenta corriente udb	10,000.00	0.00
02/19 cargo por 1000 a cuenta corriente udb	1,000.00	0.00
02/19 cargo a cuenta corriente udb 1	1,000.00	0.00
02/19 Cargo por depósito de 1000 en cuenta corriente universidad don bosco	1,000.00	0.00
03/19 movimiento de cuenta 10-02-01-001-01	1,000.00	0.00
Total Cuenta :	38,000.00	25,100.00
10-02 Cuenta Corriente Universidad Don Bosco #98989898989898989898		
01/19 ABONO POR 200 A CUENTA CORRIENTE UDB	0.00	200.00
01/19 CARGO POR 5000 A LA OTRA CUENTA CORRIENTE	5,000.00	0.00
01/19 ABONO POR 1000 A CUENTA CORRIENTE DE LA UNIVERSIDAD DON BOSCO EN	0.00	1,000.00

LIBRO AUXILIAR DE CUENTAS

Desde cuenta 10-01-02-001 hasta cuenta 8

Del 1 de Enero al 3 de Marzo de 1996

FECHA	MOVIMIENTOS	CARGOS	ABONOS
	LA CIUDADELA DON BOSCO		
01/19	cargo por 10000	10,000.00	0.00
01/19	otro cargo para cuenta corriente universidad don bosco	1,000.00	0.00
01/19	Cargo a otra cuenta corriente udb	1,000.00	0.00
01/19	Cargo a otra cuenta corriente udb el día de hoy	1,000.00	0.00
02/19	cargo por 1000 a cuenta udb 2	1,000.00	0.00
03/19	Movimiento de la cuenta número 10-02-01-001-02 (cuenta corriente universidad don bosco)	0.00	1,000.00
	Total Cuenta :	19,000.00	2,200.00
	2-01 Fondo de Becas		
01/19	cargo a la cuenta del fondo de becas	5,000.00	0.00
01/19	Para que cuadre	200.00	0.00
02/19	Saqué 1000 del fondo de becas para abonarlo a cuenta corriente udb	0.00	1,000.00
	Total Cuenta :	5,200.00	1,000.00
	2-02 Aporte PROMIPE FAES (Prestaciones)		
01/19	Cargo a cuenta de Aporte promipe faes	1,000.00	0.00
	Total Cuenta :	1,000.00	0.00
	3 Cuenta a Plazo Fijo		
01/19	Cargo a cuenta contable de cuenta a plazo fijo udb	1,000.00	0.00
01/19	kdlkkaldskfasdf	515.00	0.00
	Total Cuenta :	1,515.00	0.00
	2-01 Cta.Ahorro Bancasa UDB #982938298392832839283928392839232323		
01/19	cargo por 1000	1,000.00	0.00
01/19	ALKSDJFKLASDFA	500.00	0.00
	Total Cuenta :	1,500.00	0.00
	2-02 Cta. Ahorro Bancasa de la UDB #98392839182038190283098390283		
01/19	abono en cuenta ahorro bancasa de la universidad don bosco	0.00	1,000.00
01/19	abono en cuenta bancasa udb	0.00	1,000.00
	Total Cuenta :	0.00	2,000.00
	Cliente Deudor 1		
01/19	kjjkjkilkj	0.00	515.00
	Total Cuenta :	0.00	515.00
	Administración General		
03/19	abono a la cuenta 14-03-07 que fue asignada a la administracion general	0.00	1,000.00
	Total Cuenta :	0.00	1,000.00
	1 Padre César Montenegro		
01/19	abono del padre césar	0.00	2,500.00
	Total Cuenta :	0.00	2,500.00
	2 Padre Pierre		
01/19	abono del padre pierre	0.00	2,500.00
	Total Cuenta :	0.00	2,500.00
	Proyecto Ciudadela Don Bosco		
03/19	Abono a la cuenta 14-03-09, la cual fue asignada al Proyecto Ciudadela Don Bosco	0.00	1,000.00
	Total Cuenta :	0.00	1,000.00
	5 Papelería y Útiles		
01/19	Cargo por compra de papelería y útiles	1,000.00	0.00
	Total Cuenta :	1,000.00	0.00

LIBRO AUXILIAR DE CUENTAS

Desde cuenta 10-01-02-001 hasta cuenta 8

Del 1 de Enero al 3 de Marzo de 1996

FECHA	MOVIMIENTOS	CARGOS	ABONOS
7	Teléfono y Correo		
01/19	alksdjfklajsdlfasd	20,000.00	0.00
	Total Cuenta :	20,000.00	0.00
7	ISSS y FSV (CUOTA PATRONAL)		
01/19	ABONO POR 3000 POR ISSS Y FSV (CUOTA PATRONAL)	0.00	3,000.00
	Total Cuenta :	0.00	3,000.00
5-03	Enseres		
01/19	Cargo por compra de enseres varios	1,000.00	0.00
	Total Cuenta :	1,000.00	0.00
	Retenciones ISSS y FSV		
01/19	CARGO DE 100 EN RETENCIONES ISSS Y FSV	1,000.00	0.00
01/19	abono a retenciones ISSS Y FSV	0.00	1,000.00
	Total Cuenta :	1,000.00	1,000.00
	Cuota patronal ISSS y FSV		
01/19	CARGO DE 2000 EN CUOTA PATRONAL ISSS Y FSV	2,000.00	0.00
01/19	AKJDFKLAJSKLDJFKALSDF	0.00	500.00
	Total Cuenta :	2,000.00	500.00
0	Ingresos Varios (Biblioteca)		
01/19	INGRESO POR VARIOS (BIBLIOTECA)	0.00	1,000.00
	Total Cuenta :	0.00	1,000.00
1	Librería		
01/19	ABONO POR 2000 A LIBRERIA	0.00	2,000.00
01/19	abono a librería por 2000	0.00	2,000.00
	Total Cuenta :	0.00	4,000.00
4	Fotocopias		
01/19	Abono por pago de fotocopias	0.00	100.00
03/19	Abono a cuenta 61-19-10-004 , la cual dice que es para fotocopias	0.00	1,000.00
	Total Cuenta :	0.00	1,100.00
0	Varios		
01/19	Pagos varios a la udb	0.00	500.00
	Total Cuenta :	0.00	500.00
1-01	Matrícula		
01/19	Abono a ingresos por matrícula por 5,000.00	0.00	5,000.00
01/19	Abono por pago matrículas	0.00	1,000.00
	Total Cuenta :	0.00	6,000.00
1-07	Uso de Biblioteca		
02/19	abono por 1000 a noseque	0.00	1,000.00
	Total Cuenta :	0.00	1,000.00
1-08	Libreta y Carnet		
01/19	abono a libreta y carnet por 2000	0.00	2,000.00
02/19	abono por 1000 a libreta y carnet	0.00	1,000.00
	Total Cuenta :	0.00	3,000.00
1-09	Derechos de Exámenes		
01/19	Abono por pago de derechos de exámenes	0.00	500.00
	Total Cuenta :	0.00	500.00
1-10	Equivalencias		

LIBRO AUXILIAR DE CUENTAS

Desde cuenta 10-01-02-001 hasta cuenta 8

Del 1 de Enero al 3 de Marzo de 1996

FECHA	MOVIMIENTOS	CARGOS	ABONOS
01/19	ABONO POR EQUIVALENCIAS	0.00	4,000.00
	Total Cuenta :	0.00	4,000.00
01-13	Interciclos		
01/19	ABONO POR INTERCICLOS POR 3000 COLONES	0.00	3,000.00
01/19	Abono por pago interciclos	0.00	500.00
	Total Cuenta :	0.00	3,500.00
01-14	Cursillo Pre-Universitario		
01/19	ABONO POR CURSILLO PRE-UNIVERSITARIOS	0.00	2,000.00
01/19	Abono por 5000 a cursillo pre-universitario	0.00	5,000.00
	Total Cuenta :	0.00	7,000.00
2-01	Matrícula		
01/19	Abono por más matrículas	0.00	1,000.00
01/19	abono a matrícula por 2000	0.00	2,000.00
02/19	abono por 2000 a matrícula de no se donde	0.00	2,000.00
	Total Cuenta :	0.00	5,000.00
2-02	Mensualidades		
01/19	abono por 10000	0.00	10,000.00
02/19	abono a mensualidades de no se donde	0.00	2,000.00
	Total Cuenta :	0.00	12,000.00
2-07	Uso de Biblioteca		
01/19	Abono por pagos de uso de biblioteca	0.00	400.00
	Total Cuenta :	0.00	400.00
2-09	Derechos de Exámenes		
02/19	abono por 1000 a derechos de exámenes	0.00	1,000.00
	Total Cuenta :	0.00	1,000.00
2-10	Equivalencias		
01/19	Pago por equivalencias	0.00	500.00
	Total Cuenta :	0.00	500.00
2-13	Interciclos		
01/19	Abono por pago interciclos	0.00	500.00
01/19	abono por 2000 a interciclos	0.00	2,000.00
	Total Cuenta :	0.00	2,500.00
2-18	Retiros de materias		
01/19	abono por pago de retiros de materias	0.00	500.00
	Total Cuenta :	0.00	500.00
2-19	Tutorías		
01/19	pago por tutorías	0.00	500.00
	Total Cuenta :	0.00	500.00
2-20	Constancias		
01/19	ABONO POR IMPRESION DE CONSTANCIAS DE NOTAS	0.00	3,000.00
01/19	Pago por constancias	0.00	500.00
	Total Cuenta :	0.00	3,500.00
2-21	Material de Graduación		
01/19	Pago por material de graduación	0.00	500.00
01/19	abono por 2000 a material de graduación	0.00	2,000.00
	Total Cuenta :	0.00	2,500.00

LIBRO AUXILIAR DE CUENTAS

Desde cuenta 10-01-02-001 hasta cuenta 8

Del 1 de Enero al 3 de Marzo de 1996

FECHA	MOVIMIENTOS	CARGOS	ABONOS
04	Profesorado en Letras		
5/01/1996	abono por 100,000 en profesorado en letras	0.00	100,000.00
	Total Cuenta :	0.00	100,000.00

LIBRO MAYOR

Del 1 de Enero al 3 de Marzo de 1996

CUENTA	NOMBRE DE LA CUENTA	SALDO ANT.	CAMBIO NETO	NUEVO SALD
	pagar	<u>0.00</u>	<u>-1,500.00</u>	<u>-1,500.</u>
	arios	<u>0.00</u>	<u>-1,500.00</u>	<u>-1,500.</u>
	SSS y FSV	0.00	0.00	0.
	ISSS y FSV	<u>0.00</u>	<u>-1,500.00</u>	<u>-1,500.</u>
		0.00	-1,500.00	

ESTADO DE RESULTADOS

Del 1 de Enero al 3 de Marzo de 1996

CUENTA	NOMBRE DE LA CUENTA	SALDO
	Renta e Ingresos	160,000.00
	Ingresos Biblioteca	1,000.00
	Ingresos por Otras Actividades (Biblioteca)	1,000.00
0	Ingresos Varios (Biblioteca)	1,000.00
	Librería	5,600.00
	Actividad Comercial	5,600.00
1	Librería	4,000.00
4	Fotocopias	1,100.00
0	Varios	500.00
	Facultad de Humanidades	153,400.00
	Actividades Educativas	153,400.00
1	Ciencias de la Comunicación	25,000.00
1-01	Matrícula	6,000.00
1-07	Uso de Biblioteca	1,000.00
1-08	Libreta y Carnet	3,000.00
1-09	Derechos de Exámenes	500.00
1-10	Equivalencias	4,000.00
1-13	Interciclos	3,500.00
1-14	Cursillo Pre-Universitario	7,000.00
2	Ciencias de la Educación	28,400.00
2-01	Matrícula	5,000.00
2-02	Mensualidades	12,000.00
2-07	Uso de Biblioteca	400.00
2-09	Derechos de Exámenes	1,000.00
2-10	Equivalencias	500.00
2-13	Interciclos	2,500.00
2-18	Retiros de materias	500.00
2-19	Tutorías	500.00
2-20	Constancias	3,500.00
2-21	Material de Graduación	2,500.00
4	Profesorado en Letras	100,000.00
	TOTAL INGRESO:	160,000.00
	Cuenta de Gastos UDB	19,000.00
	Gastos Rectoría	18,000.00
	Administración	18,000.00
5	Papelería y Útiles	1,000.00
7	Teléfono y Correo	20,000.00
7	ISSS y FSV (CUOTA PATRONAL)	-3,000.00
	Dirección General Administrativa UDB	1,000.00
	Administración	1,000.00
5	Papelería y Útiles	1,000.00
6-03	Enseres	1,000.00
	TOTAL EGRESOS:	19,000.00
	GANANCIA:	141,000.00

RECTOR	F.	ADMON. GRAL.	F.	ADMINISTRADOR	F.	CONTADOR
--------	----	--------------	----	---------------	----	----------

BALANCE GENERAL

Del 1 de Enero al 3 de Marzo de 1996

CUENTA	NOMBRE DE LA CUENTA	SALDO
	Disponible	147,015.00
	Caja	111,100.00
	Caja Chica	110,500.00
01	Administración UDB	101,500.00
02	Administración CITT	4,500.00
03	Oficina Rectoría (Ing. Huguet)	4,500.00
	Caja (Cheques Rechazados)	600.00
04	Cheque # 98989898	2,600.00
07	Instituto Ricaldone	-2,000.00
	Bancos	35,915.00
	Banco Cuscatlán	36,415.00
01	Cuenta Corriente	29,700.00
01-01	Cuenta Corriente Universidad Don Bosco #81293819208192391231	12,900.00
01-02	Cuenta Corriente Universidad Don Bosco #98989898989898989898	16,800.00
02	Cuenta de Ahorro	5,200.00
02-01	Fondo de Becas	4,200.00
02-02	Aporte PROMIPE FAES (Prestaciones)	1,000.00
03	Cuenta a Plazo Fijo	1,515.00
	Bancasa	-500.00
02	Cuenta de Ahorro en Bancasa	-500.00
02-01	Cta.Ahorro Bancasa UDB #982938298392832839283928392839232323	1,500.00
02-02	Cta. Ahorro Bancasa de la UDB #98392839182038190283098390283	-2,000.00
	Cuentas por Cobrar	-515.00
	Cientes Morosos	-515.00
	Cliente Deudor 1	-515.00
	Deudores Varios	-7,000.00
	Casas salesianas	-7,000.00
	Administración General	-1,000.00
	Comunidad	-5,000.00
01	Padre César Montenegro	-2,500.00
02	Padre Pierre	-2,500.00
	Proyecto Ciudadela Don Bosco	-1,000.00
	TOTAL ACTIVO:	139,500.00
	Excedentes de Operación	141,000.00
	Excedentes Acumulados	141,000.00
	TOTAL CAPITAL:	141,000.00
	Cuentas por Pagar	-1,500.00
	Acreedores Varios	-1,500.00
	Cuota patronal ISSS y FSV	-1,500.00
	TOTAL PASIVO:	-1,500.00
	TOTAL PASIVO MAS CAPITAL:	139,500.00

RECTOR F. ADMON.GRAL. F. ADMINISTRADOR F. CONTADOR

Contabilidad General de la Universidad Don Bosco
 REPORTE DE SALDOS COMPARATIVO

Pag.

CONTABLE	SALDO 1	SALDO 2	DIFERENCIA
o			
01			
Administración UDB	99,500.00	0.00	-99,500.00
02			
Administración CITT	3,500.00	1,000.00	-2,500.00
03			
na Rectoría (Ing. Huguet)	2,500.00	1,000.00	-1,500.00
04			
que # 98989898	1,600.00	1,000.00	-600.00
07			
uto Ricaldone	-3,000.00	1,000.00	4,000.00
01-01			
nta Corriente Universidad Don Bosco #81293819208192391231	8,900.00	3,000.00	-5,900.00
01-02			
nta Corriente Universidad Don Bosco #989898989898989898	16,800.00	1,000.00	-15,800.00
02-01			
do de Becas	5,200.00	-1,000.00	-6,200.00
02-02			
nte PROMIPE FAES (Prestaciones)	1,000.00	0.00	-1,000.00
03			
nta a Plazo Fijo	1,515.00	0.00	-1,515.00
02-01			
Ahorro Bancasa UDB #982938298392832839283928392839232323	1,500.00	0.00	-1,500.00
02-02			
Ahorro Bancasa de la UDB #98392839182038190283098390283	-2,000.00	0.00	2,000.00
nte Deudor 1	-515.00	0.00	515.00
01			
o Antonio Ramírez	0.00	0.00	0.00
02			
riz Marisol Valle	0.00	0.00	0.00
dores personales (anticipos sueldos)	0.00	0.00	0.00
PROMIPE BID	0.00	0.00	0.00
Administración General	0.00	0.00	0.00
01			
re César Montenegro	-2,500.00	0.00	2,500.00
02			
re Pierre	-2,500.00	0.00	2,500.00
15			
dklfajskdjfalskdjflasdjkl	0.00	0.00	0.00
recto Ciudadela Don Bosco	0.00	0.00	0.00
s Standar "B"	0.00	0.00	0.00
01			
oría y Administración	0.00	0.00	0.00
02			
s Standar "A"	0.00	0.00	0.00
illa UDB	0.00	0.00	0.00
torium UDB	0.00	0.00	0.00
16			
nte Construcción de Cancha BKB	0.00	0.00	0.00
25			
oción y Publicidad Universidad Don Bosco	0.00	0.00	0.00
01			
s Spectrum	0.00	0.00	0.00

Contabilidad General de la Universidad Don Bosco
 REPORTE DE SALDOS COMPARATIVO

Pag.

CONTABLE	SALDO 1	SALDO 2	DIFERENCIA
al			
de orden	0.00	0.00	0.00
Asos			
5			
ería y Utiles	1,000.00	0.00	-1,000.00
7			
ono y Correo	20,000.00	0.00	-20,000.00
2			
los	0.00	0.00	0.00
7			
y FSV (CUOTA PATRONAL)	-3,000.00	0.00	3,000.00
5-01			
ería	0.00	0.00	0.00
5-03			
res	1,000.00	0.00	-1,000.00
9-02			
os	0.00	0.00	0.00
2-01			
a	0.00	0.00	0.00
5-01			
ería	0.00	0.00	0.00
5-03			
res	0.00	0.00	0.00
so			
os Varios (Biblioteca)	1,000.00	0.00	-1,000.00
ía	4,000.00	0.00	-4,000.00
opias	100.00	0.00	-100.00
s	500.00	0.00	-500.00
-01			
cula	6,000.00	0.00	-6,000.00
-07			
e Biblioteca	0.00	1,000.00	1,000.00
-08			
a y Carnet	2,000.00	1,000.00	-1,000.00
-09			
hos de Exámenes	500.00	0.00	-500.00
-10			
alencias	4,000.00	0.00	-4,000.00
-13			
clos	3,500.00	0.00	-3,500.00
-14			
o Pre-Universitario	7,000.00	0.00	-7,000.00
-01			
cula	3,000.00	2,000.00	-1,000.00
-02			
ualidades	10,000.00	2,000.00	-8,000.00
-07			
e Biblioteca	400.00	0.00	-400.00
-09			
hos de Exámenes	0.00	1,000.00	1,000.00
-10			
alencias	500.00	0.00	-500.00

Contabilidad General de la Universidad Don Bosco
 REPORTE DE SALDOS COMPARATIVO

Pag.

CONTABLE	SALDO 1	SALDO 2	DIFERENCIA
02-13			
ciclos	2,500.00	0.00	-2,500.00
02-18			
ros de materias	500.00	0.00	-500.00
02-19			
rías	500.00	0.00	-500.00
02-20			
stancias	3,500.00	0.00	-3,500.00
02-21			
erial de Graduación	2,500.00	0.00	-2,500.00
Subtotal			
ciones ISSS y FSV	0.00	0.00	0.00
a patronal ISSS y FSV	-1,500.00	0.00	1,500.00
FORD	0.00	0.00	0.00
gio Don Bosco	0.00	0.00	0.00

Listado de Tipos de Rubros Presupuestales

Descripción del Tipo

Tipo Genérico

Actividades Extraordinarias de Egresos
Actividades Extraordinarias de Ingresos
Actividades Ordinarias de Egresos
Actividades Ordinarias de Ingreso

E
I
E
I

Reporte del Catalogo de Rubros Presupuestales

**de Rubros Presupuestales para:
idad de Consolidacion para la CDB**

Descripcion Rubro	Tipo Rt
CUENTA DE GASTOS	AOE
EGRESOS DEL CITT	AOE
EGRESOS DE LA UDB	AOE
EGRESOS DEL CFP	AOE
EGRESOS DEL ORATORIO	AOE
EGRESOS DE LA PARROQUIA	AOE
EGRESOS DEL COLEGIO	AOE
EGRESOS DE LA LIBRERIA	AOE
GASTOS KFW	AOE
GASTOS PROMIPE	AOE
GASTOS GTZ	AOE
GASTO DE LA COMUNIDAD	AOE
CUENTA DE INGRESOS	AOI
INGRESOS CITT	AOI
INGRESOS UDB	AOI
INGRESOS CFP	AOI
INGRESOS ORATORIO	AOI
INGRESOS PARROQUIA	AOI
INGRESOS COLEGIO	AOI
INGRESOS VENTAS DE LIBRERIA	AOI
INGRESOS KFW	AOI
INGRESOS DE PROMIPE	AOI
INGRESOS GTZ	AOI
INGRESOS DE LA COMUNIDAD	AOI

ros: 24

**de Rubros Presupuestales para:
idad General del CITT**

Descripcion Rubro	Tipo Rt
GASTOS ORDINARIOS	AOE
ADMINISTRACION GENERAL DEL CITT	AOE
GASTOS ADMINISTRATIVOS	AOE
GASTOS FINANCIEROS	AOE
DEPARTAMENTO DE MECANICA	AOE
GASTOS DE VENTAS	AOE
ADMINISTRACION DE MECANICA	AOE
DEPARTAMENTO DE COMPUTACION	AOE
GASTOS DE VENTAS	AOE
ADMINISTRACION DE COMPUTACION	AOE
DEPARTAMENTO DE ELECTRONICA	AOE
GASTOS DE VENTAS	AOE
ADMINISTRACION DE ELECTRONICA	AOE
DEPARTAMENTO DE BIOMEDICA	AOE
GASTOS DE VENTAS	AOE
ADMINISTRACION DE BIOMEDICA	AOE
DEPARTAMENTO DE ELECTRICA	AOE
GASTOS DE VENTAS	AOE
ADMINISTRACION DE ELECTRICIDAD	AOE
DEPARTAMENTO DE AREAS BASICAS	AOE
GASTOS DE VENTAS	AOE
ADMINISTRACION DE AREAS BASICAS	AOE
DEPARTAMENTO DE COMUNICACIONES	AOE
GASTOS DE VENTAS	AOE
ADMINISTRACION DE COMUNICACIONES	AOE
TECNOLOGICO	AOE
GASTOS DE VENTAS	AOE
ADMINISTRACION DE TECNOLOGICO	AOE
DEPARTAMENTO DE PROYECTOS	AOE
GASTOS ADMINISTRATIVOS	AOE
DEPARTAMENTO ESCUELA DE IDIOMAS	AOE
GASTOS DE VENTAS	AOE
ADMINISTRACION DE ESCUELA DE IDIOMAS	AOE
INGRESOS ORDINARIOS	AOI
ADMINISTRACION GENERAL DEL CITT	AOI
INGRESOS FINANCIEROS	AOI
INGRESOS COMERCIALES	AOI
INGRESOS VARIOS	AOI
DEPARTAMENTO DE MECANICA	AOI
INGRESOS EDUCATIVOS	AOI
INGRESOS INDUSTRIALES	AOI

Reporte del Catalogo de Rubros Presupuestales

DEPARTAMENTO DE COMPUTACION	AOI
INGRESOS EDUCATIVOS	AOI
OTROS INGRESOS	AOI
DEPARTAMENTO DE ELECTRONICA	AOI
INGRESOS EDUCATIVOS	AOI
INGRESOS INDUSTRIALES	AOI
DEPARTAMENTO DE BIOMEDICA	AOI
INGRESOS EDUCATIVOS	AOI
INGRESOS INDUSTRIALES	AOI
DEPARTAMENTO DE ELECTRICA	AOI
INGRESOS EDUCATIVOS	AOI
INGRESOS INDUSTRIALES	AOI
DEPARTAMENTO DE AREAS BASICAS	AOI
INGRESOS EDUCATIVOS	AOI
DEPARTAMENTO DE COMUNICACIONES	AOI
INGRESOS EDUCATIVOS	AOI
INGRESOS INDUSTRIALES	AOI
TECNOLOGICO	AOI
INGRESOS EDUCATIVOS	AOI
DEPARTAMENTO DE PROYECTOS	AOI
OTROS INGRESOS	AOI
DEPARTAMENTO ESCUELA DE IDIOMAS	AOI
INGRESOS EDUCATIVOS	AOI
OTROS INGRESOS	AOI

ros:

65

Asignación de Cuentas Contables por Rubros de Presupuesto

Presupuesto para: COP01

Contabilidad de Consolidación para la CDB

Presupuesto	Cuenta(s) Contable(s)
EL CITT	4086 CENTRO DE INVESTIGACIONES Y TRANSFERENCIA DE TECNOLOGIA
E LA UDB	4087 UNIVERSIDAD DON BOSCO
EL CFP	4088 CENTRO DE FORMACION PROFESIONAL
EL ORATORIO	4089 ORATORIO JUVENIL
E LA PARROQUIA	4090 PARROQUIA SAN JUAN BOSCO
EL COLEGIO	4091 COLEGIO DON BOSCO
E LA LIBRERIA	4092 LIBRERIA DON BOSCO
	4093 PROYECTO KFW
OMIPE	4094 PROMIPE
	4095 PROYECTO GTZ
A COMUNIDAD	4096 COMUNIDAD SALESIANA
ITT	6186 CENTRO DE INVESTIGACIONES Y TRANSFERENCIA DE TECNOLOGIA
DB	6187 UNIVERSIDAD DON BOSCO
FP	6188 CENTRO DE FORMACION PROFESIONAL
RATORIO	6189 ORATORIO JUVENIL
ARROQUIA	6190 PARROQUIA SAN JUAN BOSCO
OLEGIO	6191 COLEGIO DON BOSCO
ENTAS DE LIBRERIA	6192 LIBRERIA DON BOSCO
FW	6193 PROYECTO KFW
E PROMIPE	6194 PROMIPE
TZ	6195 PROYECTO GTZ
E LA COMUNIDAD	6196 COMUNIDAD SALESIANA
Cuentas Asignadas:	22

Presupuesto para: COP02

Contabilidad General del CITT

Presupuesto	Cuenta(s) Contable(s)
ADMINISTRATIVOS	408603 GASTOS ADMINISTRATIVOS
FINCIEROS	408604 GASTOS FINANCIEROS
VENTAS	408702 GASTOS DE VENTAS
ACION DE MECANICA	408703 ADMINISTRACION DE MECANICA
VENTAS	408802 GASTOS DE VENTAS
ACION DE COMPUTACION	408803 ADMINISTRACION DE COMPUTACION
VENTAS	408902 GASTOS DE VENTAS
ACION DE ELECTRONICA	408903 ADMINISTRACION DE ELECTRONICA
VENTAS	409002 GASTOS DE VENTAS

Asignación de Cuentas Contables por Rubros de Presupuesto

ADMINISTRACION DE BIOMEDICA	409003
ADMINISTRACION DE BIOMEDICA	ADMINISTRACION DE BIOMEDICA
GASTOS DE VENTAS	409102
GASTOS DE VENTAS	GASTOS DE VENTAS
ADMINISTRACION DE ELECTRICIDAD	409103
ADMINISTRACION DE ELECTRICIDAD	ADMINISTRACION DE ELECTRICIDAD
GASTOS DE VENTAS	409202
GASTOS DE VENTAS	GASTOS DE VENTAS
ADMINISTRACION DE AREAS BASICAS	409203
ADMINISTRACION DE AREAS BASICAS	ADMINISTRACION DE AREAS BASICAS
GASTOS DE VENTAS	409302
GASTOS DE VENTAS	GASTOS DE VENTAS
ADMINISTRACION DE COMUNICACIONES	409303
ADMINISTRACION DE COMUNICACIONES	ADMINISTRACION DE COMUNICACIONES
GASTOS DE VENTAS	409402
GASTOS DE VENTAS	GASTOS DE VENTAS
ADMINISTRACION DE TECNOLOGICO	409403
ADMINISTRACION DE TECNOLOGICO	ADMINISTRACION DE TECNOLOGICO
GASTOS ADMINISTRATIVOS	409503
GASTOS ADMINISTRATIVOS	GASTOS ADMINISTRATIVOS
GASTOS DE VENTAS	409602
GASTOS DE VENTAS	GASTOS DE VENTAS
ADMINISTRACION DE ESCUELA DE IDIOMAS	409603
ADMINISTRACION DE ESCUELA DE IDIOMAS	ADMINISTRACION DE ESCUELA DE IDIOMAS
INGRESOS FINANCIEROS	618601
INGRESOS FINANCIEROS	INGRESOS FINANCIEROS
INGRESOS COMERCIALES	618610
INGRESOS COMERCIALES	INGRESOS COMERCIALES
INGRESOS VARIOS	618650
INGRESOS VARIOS	INGRESOS VARIOS
INGRESOS EDUCATIVOS	618702
INGRESOS EDUCATIVOS	INGRESOS EDUCATIVOS
INGRESOS INDUSTRIALES	618712
INGRESOS INDUSTRIALES	INGRESOS INDUSTRIALES
INGRESOS EDUCATIVOS	618802
INGRESOS EDUCATIVOS	INGRESOS EDUCATIVOS
OTROS INGRESOS	618850
OTROS INGRESOS	OTROS INGRESOS
INGRESOS EDUCATIVOS	618902
INGRESOS EDUCATIVOS	INGRESOS EDUCATIVOS
INGRESOS INDUSTRIALES	618912
INGRESOS INDUSTRIALES	INGRESOS INDUSTRIALES
INGRESOS EDUCATIVOS	619002
INGRESOS EDUCATIVOS	INGRESOS EDUCATIVOS
INGRESOS INDUSTRIALES	619012
INGRESOS INDUSTRIALES	INGRESOS INDUSTRIALES
INGRESOS EDUCATIVOS	619102
INGRESOS EDUCATIVOS	INGRESOS EDUCATIVOS
INGRESOS INDUSTRIALES	619112
INGRESOS INDUSTRIALES	INGRESOS INDUSTRIALES
INGRESOS EDUCATIVOS	619202
INGRESOS EDUCATIVOS	INGRESOS EDUCATIVOS
INGRESOS EDUCATIVOS	619302
INGRESOS EDUCATIVOS	INGRESOS EDUCATIVOS
INGRESOS INDUSTRIALES	619312
INGRESOS INDUSTRIALES	INGRESOS INDUSTRIALES
INGRESOS EDUCATIVOS	619402
INGRESOS EDUCATIVOS	INGRESOS EDUCATIVOS
OTROS INGRESOS	619550
OTROS INGRESOS	OTROS INGRESOS
INGRESOS EDUCATIVOS	619602
INGRESOS EDUCATIVOS	INGRESOS EDUCATIVOS
OTROS INGRESOS	619650
OTROS INGRESOS	OTROS INGRESOS

Cuentas Asignadas: 41

Reporte de Montos Presupuestados

Presupuestados para:
idad de Consolidación para la CDB

Descripción		T A						
CUENTA DE GASTOS								
Mes	Presupuesto	Real	Diferencia		Real Consolidado		Diferencia	
Enero 1996	2,200,000.00		-500.00	2,200,500.00	100.02%	-2,500.00	2,202,500.00	100.00%
	Valores Acumulados	2,200,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Febrero 1996	2,200,000.00		0.00	2,200,000.00	100.00%	0.00	2,200,000.00	100.00%
	Valores Acumulados	4,400,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Marzo 1996	2,200,000.00		0.00	2,200,000.00	100.00%	0.00	2,200,000.00	100.00%
	Valores Acumulados	6,600,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Abril 1996	2,200,000.00		0.00	2,200,000.00	100.00%	0.00	2,200,000.00	100.00%
	Valores Acumulados	8,800,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Mayo 1996	2,200,000.00		0.00	2,200,000.00	100.00%	0.00	2,200,000.00	100.00%
	Valores Acumulados	11,000,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Junio 1996	2,200,000.00		0.00	2,200,000.00	100.00%	0.00	2,200,000.00	100.00%
	Valores Acumulados	13,200,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Julio 1996	2,200,000.00		0.00	2,200,000.00	100.00%	0.00	2,200,000.00	100.00%
	Valores Acumulados	15,400,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Agosto 1996	2,200,000.00		0.00	2,200,000.00	100.00%	0.00	2,200,000.00	100.00%
	Valores Acumulados	17,600,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Septiembre 1996	2,200,000.00		0.00	2,200,000.00	100.00%	0.00	2,200,000.00	100.00%
	Valores Acumulados	19,800,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Octubre 1996	2,200,000.00		0.00	2,200,000.00	100.00%	0.00	2,200,000.00	100.00%
	Valores Acumulados	22,000,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Noviembre 1996	2,200,000.00		0.00	2,200,000.00	100.00%	0.00	2,200,000.00	100.00%
	Valores Acumulados	24,200,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Diciembre 1996	2,200,000.00		0.00	2,200,000.00	100.00%	0.00	2,200,000.00	100.00%
	Valores Acumulados	26,400,000.00	-500.00	26,400,500.00	100.00%	-2,500.00	26,402,500.00	100.00%
Totales		26,400,000.00	-500.00			-2,500.00		

Descripción		T A						
EGRESOS DEL CITT								
Mes	Presupuesto	Real	Diferencia		Real Consolidado		Diferencia	
Enero 1996	200,000.00		-500.00	200,500.00	100.25%	-2,500.00	202,500.00	101.25%
	Valores Acumulados	200,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Febrero 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	400,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Marzo 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	600,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Abril 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	800,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Mayo 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	1,000,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Junio 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	1,200,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Julio 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	1,400,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Agosto 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	1,600,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Septiembre 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	1,800,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Octubre 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	2,000,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Noviembre 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	2,200,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Diciembre 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	2,400,000.00	-500.00	2,400,500.00	100.02%	-2,500.00	2,402,500.00	100.00%
Totales		2,400,000.00	-500.00			-2,500.00		

Descripción		T A						
EGRESOS DE LA UDB								
Mes	Presupuesto	Real	Diferencia		Real Consolidado		Diferencia	
Enero 1996	200,000.00		0.00	200,000.00	100.00%	0.00	200,000.00	100.00%
	Valores Acumulados	200,000.00	0.00	2,400,000.00	100.00%	0.00	2,400,000.00	100.00%

Anexo H: Triggers y Procedimientos Almacenados del Sistema COP.

Repositorio de Procedimientos Almacenados y Triggers del Sistema

Procedimientos Almacenados de Uso General

- sp_cop_copiar_cuentas
Copiar cuentas de una Contabilidad a otra
- sp_cop_mayoriza
Traslado de Partidas No Mayorizadas a tabla de Partidas Mayorizadas
- sp_cop_cierre_periodo
Cierre Contable del Período Actual
- sp_cop_cierre_ejercicio
Cierre Contable del Ejercicio Actual

Procedimientos Almacenados para elaboración de Reportes

- sp_cop_calc_saldo
Procedimiento para el Cálculo de Saldos Reales y/o Estimados de Cuentas de Detalle entre dos fechas
- sp_saldopnom
Cálculo de los Saldos de las Partidas No Mayorizadas
- sp_cop_comparativo
Prepara Datos p/Reportes Comparativos
- sp_diamayor
Cálculo de los saldos de las cuentas de mayor para un rango de cuentas, limitadas además por un rango de partidas (fechas).
- sp_est_financieros
Elaboración de los Estados Financieros

Scripts Complementarios para los Triggers generados por OpenSnap

- trins!!cop_contabilidad
Complemento Antes del Trigger de Inserción de cop_contabilidad
- trins!!cop_cuenta_contab
Complemento Antes del Trigger de Inserción de cop_cuenta_contab
- trins::cop_cuenta_contab
Complemento Después del Trigger de Inserción de cop_cuenta_contab
- trdel!!cop_cuenta_contab
Complemento Antes del Trigger de Borrado de cop_cuenta_contab

- `trdel::cop_cuenta_contab`
Complemento Después del Trigger de Borrado de `cop_cuenta_contab`
- `trdel!!pre_rubro`
Complemento Antes del Trigger de Borrado de `pre_rubro`
- `trins!!cop_partenc_2`
Complemento Antes del Trigger de Inserción de `cop_partenc_2`
- `trins!!pre_rubro`
Complemento Antes del Trigger de Inserción de `pre_rubro`
- `trins!!pre_rubro`
Complemento Antes del Trigger de Inserción de `pre_rubro`
- `trins::cop_partdet_2`
Complemento Después del Trigger de Inserción de `cop_partdet_2`
- `trins::cop_saldo`
Complemento Después del Trigger de Inserción de `cop_saldo`
- `trins::pre_rubro`
Complemento Después del Trigger de Inserción de `pre_rubro`
- `trins::pre_saldo`
Complemento Después del Trigger de Inserción de `pre_saldo`
- `trinscop_periodo`
Complemento Antes del Trigger de Inserción de `cop_periodo`
- `trupdcop_saldo_pnom`
Trigger de Actualización de `cop_saldo_pnom`

Procedimientos Almacenados de uso General

Tipo:	Procedimiento Almacenado
Nombre:	sp_cop_copiar_cuentas
Código:	COPYCOP
Descripción:	Copiar cuentas de una Contabilidad a otra

```

create proc sp_cop_copiar_cuentas
    (@vcod_contab1 char(5),
    @vcod_contab2 char(5),
    @vnivel int,
    @vcod_cuenta1 char(30),
    @vcod_cuenta2 char(30)
    )
as
/* Copiar cuentas de una Contabilidad a otra */
declare @error_msg varchar(100)
declare @i int

if exists(select * from cop_cuenta_contab
    where cod_contab=@vcod_contab1)
begin
    if (exists(select * from cop_cuenta_contab
        where cod_contab=@vcod_contab1 and
        cod_cuenta=@vcod_cuenta1 and
        nivel<=1) and
        exists(select * from cop_cuenta_contab
            where cod_contab=@vcod_contab1 and
            cod_cuenta=@vcod_cuenta2 and
            nivel<=1))
        begin
            if not exists(select * from cop_cuenta_contab c1, cop_cuenta_contab c2
                where c1.cod_contab=@vcod_contab1 and
                c2.cod_contab=@vcod_contab2 and
                (c1.cod_cuenta between @vcod_cuenta1 and @vcod_cuenta2)
                and
                c1.cod_cuenta=c2.cod_cuenta)
                begin
                    select @i = 0
                    while (@i<=@vnivel)
                    begin
                        insert into cop_cuenta_contab
(cod_contab,cod_cuenta,cod_cuenta_p,nivel,cod_t_cuenta,descrip_cta,cod_contab_c,cod_cue
nta_c,detalle,dependencia)

```

```

                select @vcod_contab2,c1.cod_cuenta, c1.cod_cuenta_p,
c1.nivel, c1.cod_t_cuenta, c1.descripcion, c1.cod_contab_c, c1.cod_cuenta_c, c1.detalle,
c1.dependencia
                from cop_cuenta_contab c1
                where c1.cod_contab=@vcod_contab1 and
                    (c1.cod_cuenta between @vcod_cuenta1 and
@vcod_cuenta2)
                    and c1.nivel=@i
                order by c1.nivel

                select @i = @i+1
            end
        end
    else
    begin
        select @error_msg= 'Existen Cuentas en la Contabilidad Fuente que ya
existen en
                la Destino!'
                raiserror 40000 @error_msg
                return
            end
    end
    else
    begin
        select @error_msg= 'Las cuentas en el rango seleccionado, deben ser de primer
nivel!'
                raiserror 40000 @error_msg
                return
            end
    end
    else
    begin
        select @error_msg= 'No existen Cuentas Contables en la Contabilidad: ' +
convert(varchar,
                @vcod_contab1)
                raiserror 40000 @error_msg
                return
            end
    end
end

```

Tipo:	Procedimiento Almacenado
Nombre:	sp_cop_mayoriza
Código:	SPCOP01
Descripción:	Traslado de Partidas No Mayorizadas a tabla de Partidas Mayorizadas

```

create proc sp_cop_mayoriza (      @v_cod_contab char(5),
                                @v_fecha_partida datetime,
                                @v_num_corr_d int)

as
/* Traslado de Partidas */
declare @v_num_corr_ejerc int

if @v_fecha_partida < ( select max(fecha_partida)
                        from cop_partenc_2
                        where cod_contab = @v_cod_contab)

begin
    raiserror 40000 "La partida es anterior a la ultima fecha mayorizada. No puede
mayorizarse con esa
    fecha"
    return
end
if (      select count(*)
    from cop_partenc_1
    where
    cod_contab=@v_cod_contab and
    fecha_partida=@v_fecha_partida and
    num_corr_d=@v_num_corr_d) = 1
begin
    if (      select count(*)
        from cop_partenc_1 pa,cop_contabilidad co,cop_periodo pe
        where pa.cod_contab=@v_cod_contab and
              pa.fecha_partida=@v_fecha_partida and
              pa.num_corr_d=@v_num_corr_d and
              pa.cod_contab=co.cod_contab and
              pa.ejercicio=pe.ejercicio and
              pa.periodo=pe.periodo and
              pa.periodo=co.periodo and
              pa.ejercicio=co.ejercicio) = 1

begin
    begin transaction
    select @v_num_corr_ejerc=isnull(max(num_corr_ejerc),0) + 1
    from cop_partenc_2
    where cod_contab=@v_cod_contab

```

```

insert into
cop_partenc_2(fecha_partida,num_corr_ejerc,cod_contab,periodo,ejercicio,con
cepto_part,
cod_t_partida,fecha_proc,cod_app,cod_unidad)
select
pa.fecha_partida,@v_num_corr_ejerc,pa.cod_contab,pa.periodo,pa.ejercicio,
pa.concepto_part,pa.cod_t_partida,pa.fecha_proc,pa.cod_app,pa.cod_unidad
from cop_partenc_1 pa
where pa.fecha_partida=@v_fecha_partida and
      pa.num_corr_d=@v_num_corr_d and
      pa.cod_contab=@v_cod_contab

```

```

insert into
cop_partdet_2(cod_contab,num_corr_ejerc,num_corr_movim,cod_cuenta,cargo
s,
abonos,descrip_det)
select
pa.cod_contab,@v_num_corr_ejerc,pa.num_corr_movim,pa.cod_cuenta,pa.car
gos,
pa.abonos,pa.descrip_det
from cop_partdet_1 pa
where pa.fecha_partida=@v_fecha_partida and
      pa.num_corr_d=@v_num_corr_d and
      pa.cod_contab=@v_cod_contab

```

```

delete cop_partdet_1
where fecha_partida=@v_fecha_partida and
      num_corr_d=@v_num_corr_d and
      cod_contab=@v_cod_contab

```

```

delete cop_partenc_1
where fecha_partida=@v_fecha_partida and
      num_corr_d=@v_num_corr_d and
      cod_contab=@v_cod_contab
commit transaction

```

```

end
else
begin
raiserror 40000 "No es posible mayorizar la partida. Revise que pertenezca al
periodo y
ejercicio actual para esta contabilidad."
return
end
end
else

```

```
begin
  raiserror 40000 "No existe la Partida No Mayorizada: "
  return
end
return
```

Tipo:	Procedimiento Almacenado
Nombre:	sp_cop_cierre_periodo
Código:	CIERRE_P
Descripción:	Cierre Contable del Período Actual

```

create proc sp_cop_cierre_periodo
    (@contabilidad char(5),
    @vperiodo_actual int,
    @vejercicio_actual int)
as
/* Cierre Contable del Periodo Actual */

/* Declaracion de variables */
declare @vnuevo_periodo int, @vnuevo_ejercicio int,
        @vfecha_fin datetime, @vfecha_ini datetime, @mensaje varchar(100)

/* Definicion de nuevo período (el siguiente cronologicamente)*/
select @vnuevo_ejercicio = @vejercicio_actual

select @vfecha_fin = (select fecha_fin from cop_periodo
where ejercicio = @vejercicio_actual and periodo=@vperiodo_actual)

select @vnuevo_periodo = (select periodo from cop_periodo
where ejercicio = @vejercicio_actual and
fecha_ini = (select min(fecha_ini) from cop_periodo
              where ejercicio = @vejercicio_actual and
              fecha_ini > @vfecha_fin))
/* Si el periodo actual es el ultimo del ejercicio, entonces se deja el mismo */
if @vnuevo_periodo is null
begin
    select @vnuevo_periodo = @vperiodo_actual
end

/* Si no es el ultimo... */
else
begin

    /* Actualizacion de Contabilidad con nuevo Periodo actual */
    update cop_contabilidad
    set    ejercicio = @vnuevo_ejercicio,
          periodo = @vnuevo_periodo
    where cod_contab = @contabilidad

end
select @mensaje = "Periodo1: " + convert(varchar, @vperiodo_actual) +
        " Ejercicio1:" + convert(varchar, @vejercicio_actual)

```

```

print @mensaje
select @mensaje = "Periodo2: " + convert(varchar, @vnuevo_periodo) +
    " Ejercicio2:" + convert(varchar, @vnuevo_ejercicio)
print @mensaje

/* ----- */

/* Mayorizar la cuenta de Pérdidas/Ganancias del Ejercicio */

/* CALCULO */
declare @cargos_pg money, @abonos_pg money, @cargos_ant money, @abonos_ant
money, @vcuenta_superavit varchar(30)

select @cargos_pg = sum(isnull(cargos,0)), @abonos_pg=sum(isnull(abonos,0))
from cop_saldo s, cop_cuenta_contab c, cop_t_cuenta t
where c.cod_contab = @contabilidad and
    c.cod_contab = s.cod_contab and
    c.cod_cuenta = s.cod_cuenta and
    s.ejercicio = @vejercicio_actual and
    s.periodo = @vperiodo_actual and
    c.cod_t_cuenta= t.cod_t_cuenta and
    c.detalle = 1 and
    t.t_cuenta_gral = "I"

select @cargos_pg = isnull(@cargos_pg, 0), @abonos_pg = isnull(@abonos_pg, 0)
select @cargos_ant = @cargos_pg, @abonos_ant = @abonos_pg

select @mensaje = "Ingresos: C:" + convert(varchar, @cargos_pg) + " A:" +
convert(varchar, @abonos_pg)
print @mensaje

select @cargos_pg = @cargos_pg + sum(isnull(cargos,0)), @abonos_pg =
@abonos_pg + sum(isnull(abonos,0))
from cop_saldo s, cop_cuenta_contab c, cop_t_cuenta t
where c.cod_contab = @contabilidad and
    c.cod_contab = s.cod_contab and
    c.cod_cuenta = s.cod_cuenta and
    s.ejercicio = @vejercicio_actual and
    s.periodo = @vperiodo_actual and
    c.cod_t_cuenta= t.cod_t_cuenta and
    c.detalle = 1 and
    t.t_cuenta_gral = "G"

select @cargos_pg = isnull(@cargos_pg, @cargos_ant), @abonos_pg =
isnull(@abonos_pg, @abonos_ant)
select @cargos_ant = @cargos_pg, @abonos_ant = @abonos_pg

```

```
select @mensaje = "Ganancia/Perdida: C:" + convert(varchar, @cargos_pg) + " A:" +
convert(varchar, @abonos_pg)
print @mensaje
```

```
/* Actualizacion del Saldo para Cuenta Perdidas/Ganancias */
```

```
select @vcuenta_superavit from cop_contabilidad where cod_contab = @contabilidad
```

```
if not exists(select * from cop_saldo
  where cod_contab = @contabilidad and
        cod_cuenta = @vcuenta_superavit and
        ejercicio = @vejercicio_actual and
        periodo = @vperiodo_actual)
begin
  insert into cop_saldo(cod_contab, cod_cuenta, ejercicio, periodo, abonos, cargos,
    abonos_cons, cargos_cons, cerrado)
  values (@contabilidad, @vcuenta_superavit,
    @vejercicio_actual, @vperiodo_actual,
    @abonos_pg, @cargos_pg, 0, 0, 0)
end
else
begin
  update cop_saldo
  set   abonos = abonos + @abonos_pg,
        cargos = cargos + @cargos_pg
  where cod_contab = @contabilidad and
        cod_cuenta = @vcuenta_superavit and
        ejercicio = @vejercicio_actual and
        periodo = @vperiodo_actual
end
```

```
/* ----- */
```

```
/* Define el periodo actual como cerrado (en los saldos de las cuentas) */
```

```
update cop_saldo
set cerrado = 1
where   cod_contab = @contabilidad and
        ejercicio = @vejercicio_actual and
        periodo = @vperiodo_actual
```

Tipo:	Procedimiento Almacenado
Nombre:	sp_cop_cierre_ejercicio
Código:	CIERRE_E
Descripción:	Cierre Contable del Ejercicio Actual

```

create proc sp_cop_cierre_ejercicio
    (@contabilidad char(5),
    @vejercicio_actual int)
as
/* Cierre Contable del Ejercicio Actual */

/* Declaracion de variables */
declare @vnuevo_periodo int, @vnuevo_ejercicio int,
    @vfecha_fin datetime, @vfecha_ini datetime, @mensaje varchar(100),
    @hay_ingresos bit

/* ... VALIDACIONES PREVIAS AL CIERRE DEL EJERCICIO .....*/

/* Verificar si hay cuentas definidas */

if not exists(select * from cop_cuenta_contab) begin
    select @mensaje="No se han definido Cuentas Contables aun! No puede ejecutar el
Cierre."
    print @mensaje
    return
end

/* Verificar si ya se realizo el Cierre del Ejercicio */

if exists(select * from cop_cuenta_contab_h where ejercicio=@vejercicio_actual) begin
select @mensaje="Ya se realizo el Cierre del Ejercicio " + convert(varchar,
@vejercicio_actual)
    print @mensaje
    return
end

/* Verificar si ya se definio la cuenta de Superavit del Ejercicio */

if not exists(    select cuenta_superavit from cop_contabilidad
                where cuenta_superavit is not null and cod_contab=@contabilidad)
begin
    select @mensaje="Aun no se ha definido la Cuenta de Superavit/Deficit del Ejercicio
para esta contabilidad!"
    print @mensaje

```

```

return
end

```

```

/* Verificar si ya se cerro el ultimo periodo del ejercicio actual */

```

```

select @vnuevo_periodo =
(select periodo from cop_periodo
where ejercicio=@vejercicio_actual
and fecha_fin=(      select max(fecha_fin) from cop_periodo
                    where ejercicio=@vejercicio_actual))

```

```

if not exists(      select * from cop_saldo where cod_contab=@contabilidad
and ejercicio=@vejercicio_actual and periodo=@vnuevo_periodo
and cerrado=1) begin

```

```

    select @mensaje= "No se puede realizar el Cierre del Ejercicio. Aun no se ha cerrado
el Periodo "+

```

```

        convert(varchar, @vnuevo_periodo) + " : " +
        (select descrip_periodo from cop_periodo where
        periodo=@vnuevo_periodo and ejercicio=@vejercicio_actual)

```

```

    print @mensaje
    return
end

```

```

/* ..... TRASPASO DE TABLAS DE TRABAJO A TABLAS HISTORICAS .....
*/

```

```

begin transaction

```

```

/* CATALOGO DE CUENTAS */

```

```

insert into cop_cuenta_contab_h(
    ejercicio, cod_contab, cod_cuenta, cod_cuenta_p, nivel, cod_t_cuenta,
    cod_contab_c, cod_cuenta_c, detalle, dependencia, descrip_cta)
select @vejercicio_actual, cod_contab, cod_cuenta, cod_cuenta_p, nivel,
cod_t_cuenta,
    cod_contab_c, cod_cuenta_c, detalle, dependencia, descrip_cta
from cop_cuenta_contab
where cod_contab = @contabilidad

delete cop_cuenta_contab
where cod_contab = @contabilidad

```

```
/* SALDOS DE LAS CUENTAS */
```

```
insert into cop_saldo_h(
    cod_contab, cod_cuenta, ejercicio, periodo, abonos, cargos,
    cargos_cons, abonos_cons, cerrado)
select cod_contab, cod_cuenta, ejercicio, periodo, abonos, cargos,
    cargos_cons, abonos_cons, cerrado
from cop_saldo
where cod_contab = @contabilidad and
    ejercicio = @vejercicio_actual and
    cerrado = 1
```

```
delete cop_saldo
where cod_contab = @contabilidad and
    ejercicio = @vejercicio_actual and
    cerrado = 1
```

```
/* ENCABEZADO Y DETALLE DE PARTIDAS MAYORIZADAS */
```

```
insert into cop_partenc_h(
    cod_contab, ejercicio, periodo, num_corr_ejerc, fecha_partida,
    concepto_part, cod_t_partida, fecha_proc, cod_app, cod_unidad)
select cod_contab, ejercicio, periodo, num_corr_ejerc, fecha_partida,
    concepto_part, cod_t_partida, fecha_proc, cod_app, cod_unidad
from cop_partenc_2
where cod_contab = @contabilidad
```

```
insert into cop_partdet_h(
    cod_contab, ejercicio, num_corr_ejerc, num_corr_movim,
    cod_cuenta, cargos, abonos, descrip_det)
select cod_contab, @vejercicio_actual, num_corr_ejerc, num_corr_movim,
    cod_cuenta, cargos, abonos, descrip_det
from cop_partdet_2
where cod_contab = @contabilidad
```

```
delete cop_partenc_2
where cod_contab = @contabilidad
```

```
commit transaction
```

```
/* .....CREACION PARTIDAS DE CIERRE DEL EJERCICIO..... */
```

```
/* VARIABLES A UTILIZAR EN EL PROCESO DE CREACION DE LAS
PARTIDAS */
```

```
declare @ult_part smallint, @concepto varchar(120), @ult_fecha datetime,
@vfecha_proc datetime,
    @tipo_part char(3), @aplicacion char(3), @unidad char(5), @corr_movim tinyint
declare @tot_cargos money, @tot_abonos money, @debe_pg money, @haber_pg money
declare @sintesis varchar(250), @ctaperdigan varchar(30), @nomctaperdigan
varchar(60)
```

```
select    @ult_part = max(num_corr_ejerc)
from      cop_partenc_h
where     cod_contab = @contabilidad and
         ejercicio = @vejercicio_actual
```

```
select    @ult_fecha = (      select fecha_fin from cop_periodo
                             where ejercicio = @vejercicio_actual and periodo=@vnuevo_periodo)
```

```
select    @tipo_part = cod_t_partida
from      cop_t_partida
where     partida_cierre = 1
```

```
select    @aplicacion = cod_app, @unidad = cod_unidad
from      cop_contabilidad
where     cod_contab = @contabilidad
```

```
select    @ctaperdigan= c.cod_cuenta, @nomctaperdigan = c.descripcion_cta
from      cop_cuenta_contab_h c, cop_contabilidad b
where     c.cod_contab = @contabilidad and
         c.ejercicio = @vejercicio_actual and
         c.cod_contab = b.cod_contab and
         c.cod_cuenta = b.cuenta_superavit
```

```
/* CALCULAR SALDOS DEL EJERCICIO PARA TODAS LAS CUENTAS DE
DETALLE */
```

```
create table #tmp_saldos(
    ejercicio      int,
    cod_contab     char(5),
    cod_cuenta     varchar(30),
    cargos         money,
    abonos         money,
    num_corr_movim smallint)
```

```

insert into #tmp_saldos(ejercicio, cod_contab, cod_cuenta, cargos, abonos)
select  s.ejercicio, @contabilidad, s.cod_cuenta, sum(s.cargos), sum(s.abonos)
from    cop_saldo_h s, cop_cuenta_contab c
where   s.cod_contab = @contabilidad and
        s.ejercicio = @vejercicio_actual and
        s.cod_contab = c.cod_contab and
        s.cod_cuenta = c.cod_cuenta and
        c.detalle = 1
group by s.cod_contab, s.cod_cuenta

```

```

/* .....PARTIDA DE CIERRE DE CUENTAS DE RESULTADOS..... */

```

```

/* VERIFICAR SI HAY GASTOS O INGRESOS */

```

```

select @tot_cargos = sum(s.cargos), @tot_abonos = sum(s.abonos)
from #tmp_saldos s, cop_cuenta_contab_h c, cop_t_cuenta t
where  s.cod_contab = c.cod_contab and
        s.ejercicio = c.ejercicio and
        s.cod_cuenta = c.cod_cuenta and
        c.cod_t_cuenta = t.cod_t_cuenta and
        (t.t_cuenta_gral="I" or t.t_cuenta_gral="G")

```

```

if @tot_cargos=0 and @tot_abonos=0 begin
    select @sintesis = "No se generó Partida de Cierre de Cuentas de Resultados debido a
        que no existen saldos para tales cuentas."
end

```

```

/* HAY SALDOS DE CUENTAS DE RESULTADOS, POR LO QUE SE CREARA LA
PARTIDA */

```

```

else begin

```

```

    select @ult_part = @ult_part + 1
    select @concepto = "Traspaso de SalDOS de Cuentas de Resultados a Cuenta de
        Superavit/Deficit del Ejercicio."

```

```

begin transaction

```

```

/* CREAR EL ENCABEZADO..... */

```

```

insert into cop_partenc_h(
        cod_contab, fecha_partida, num_corr_ejerc, ejercicio, periodo,
        concepto_part, cod_t_partida, fecha_proc, cod_app, cod_unidad)

```

```

values(
    @contabilidad, @ult_fecha, @ult_part, @vejercicio_actual, @vnuevo_periodo,

```

@concepto, @tipo_part, @vfecha_proc, @aplicacion, @unidad)

/* CREAR EL DETALLE..... */

select @corr_movim = 1

/* AGREGAR LA PERDIDA A CUENTA PERDIDAS O GANANCIAS DEL EJERCICIO */

```
if @tot_cargos > @tot_abonos begin
    insert into cop_partdet_h(
        cod_contab, ejercicio, num_corr_ejerc, cod_cuenta,
        num_corr_movim, cargos, abonos, descrip_det)
    values(@contabilidad, @vejercicio_actual, @ult_part, @ctaperdigan,
        @corr_movim, (@tot_cargos-@tot_abonos), 0, @nomctaperdigan)

    select @corr_movim = @corr_movim + 1
end
```

/* AGREGAR CUENTAS CON SALDO ACREEDOR */

```
insert into cop_partdet_h(
    cod_contab, ejercicio, num_corr_ejerc, cod_cuenta,
    num_corr_movim, cargos, abonos, descrip_det)
select @contabilidad, s.ejercicio, @ult_part, s.cod_cuenta,
    s.num_corr_movim, (s.abonos-s.cargos), 0, c.descripcion_cta
from #tmp_saldos s, cop_cuenta_contab_h c, cop_t_cuenta t
where s.cod_contab = c.cod_contab and
    s.cod_cuenta = c.cod_cuenta and
    s.ejercicio = c.ejercicio and
    c.cod_t_cuenta = t.cod_t_cuenta and
    (t.t_cuenta_gral="I" or t.t_cuenta_gral="G")
```

/* AGREGAR CUENTAS CON SALDO DEUDOR */

```
insert into cop_partdet_h(
    cod_contab, ejercicio, num_corr_ejerc, cod_cuenta,
    num_corr_movim, cargos, abonos, descrip_det)
select @contabilidad, s.ejercicio, @ult_part, s.cod_cuenta,
    s.num_corr_movim, 0, (s.cargos-s.abonos), c.descripcion_cta
from #tmp_saldos s, cop_cuenta_contab_h c, cop_t_cuenta t
```

```

where s.cod_contab = c.cod_contab and
      s.cod_cuenta = c.cod_cuenta and
      s.ejercicio = c.ejercicio and
      c.cod_t_cuenta = t.cod_t_cuenta and
      (t.t_cuenta_gral="I" or t.t_cuenta_gral="G")

```

```

/* AGREGAR LA GANANCIA A CUENTA PERDIDAS O GANANCIAS DEL
EJERCICIO */

```

```

if @tot_cargos < @tot_abonos begin
    insert into cop_partdet_h(
        cod_contab, ejercicio, num_corr_ejerc, cod_cuenta,
        num_corr_movim, cargos, abonos, descrip_det)
    values(@contabilidad, @vejercicio_actual, @ult_part, @ctaperdigan,
        @corr_movim, 0, (@tot_abonos-@tot_cargos), @nomctaperdigan)

    select @corr_movim = @corr_movim + 1
end

end      /* FIN CREACION PARTIDA CIERRE CUENTAS RESULTADOS */

```

```

/* ..... CREAR PARTIDA DE CIERRE DE CUENTAS DEL BALANCE GENERAL
..... */

```

```

/* VERIFICAR SI HAY SALDOS DE CUENTAS QUE NO SEAN GASTOS O
INGRESOS */

```

```

select @tot_cargos = sum(s.cargos), @tot_abonos = sum(s.abonos)
from #tmp_saldos s, cop_cuenta_contab_h c, cop_t_cuenta t
where s.cod_contab = c.cod_contab and
      s.ejercicio = c.ejercicio and
      s.cod_cuenta = c.cod_cuenta and
      c.cod_t_cuenta = t.cod_t_cuenta and
      (t.t_cuenta_gral <> "I" and t.t_cuenta_gral <> "G")

```

```

if @tot_cargos=0 and @tot_abonos=0 begin
    select @sintesis = "No se generó Partida de Cierre de Cuentas del Balance debido a
que no existen saldos para tales cuentas."
end

```

```

/* HAY SALDOS DE CUENTAS DEL BALANCE, POR LO QUE SE CREARA LA
PARTIDA */
else begin

select @ult_part = @ult_part + 1
select @concepto = "Partida de Cierre del Ejercicio."

begin transaction

/* CREAR EL ENCABEZADO..... */

insert into cop_partenc_h(
    cod_contab, fecha_partida, num_corr_ejerc, ejercicio, periodo,
    concepto_part, cod_t_partida, fecha_proc, cod_app, cod_unidad)

values(@contabilidad, @ult_fecha, @ult_part, @vejercicio_actual, @vnuevo_periodo,
    @concepto, @tipo_part, @vfecha_proc, @aplicacion, @unidad)

/* CREAR EL DETALLE..... */

select @corr_movim = 1

/* AGREGAR CUENTAS CON SALDO ACREEDOR */

insert into cop_partdet_h(
    cod_contab, ejercicio, num_corr_ejerc, cod_cuenta,
    num_corr_movim, cargos, abonos, descrip_det)
select @contabilidad, s.ejercicio, @ult_part, s.cod_cuenta,
    s.num_corr_movim, (s.abonos-s.cargos), 0, c.descripcion_cta
from #tmp_saldos s, cop_cuenta_contab_h c, cop_t_cuenta t
where s.cod_contab = c.cod_contab and
    s.cod_cuenta = c.cod_cuenta and
    s.ejercicio = c.ejercicio and
    c.cod_t_cuenta = t.cod_t_cuenta and
    (t.t_cuenta_gral <> "I" and t.t_cuenta_gral <> "G")

/* AGREGAR CUENTAS CON SALDO DEUDOR */

insert into cop_partdet_h(
    cod_contab, ejercicio, num_corr_ejerc, cod_cuenta,
    num_corr_movim, cargos, abonos, descrip_det)
select @contabilidad, s.ejercicio, @ult_part, s.cod_cuenta,
    s.num_corr_movim, 0, (s.cargos-s.abonos), c.descripcion_cta

```

```

from #tmp_saldos s, cop_cuenta_contab_h c, cop_t_cuenta t
where s.cod_contab = c.cod_contab and
      s.cod_cuenta = c.cod_cuenta and
      s.ejercicio = c.ejercicio and
      c.cod_t_cuenta = t.cod_t_cuenta and
      (t.t_cuenta_gral <> "I" or t.t_cuenta_gral <> "G")

end      /* FIN CREACION PARTIDA CIERRE BALANCE */

/* ..... PREPARACION DE DATOS PARA EL SIGUIENTE EJERCICIO ..... */

/* DEFINICION DEL NUEVO PERIODO (EL SIGUIENTE
CRONOLOGICAMENTE)*/

select @vnuevo_ejercicio = @vejercicio_actual

select @vfecha_fin = (select fecha_fin from cop_periodo
where ejercicio = @vejercicio_actual and periodo=@vnuevo_periodo)

select @vnuevo_periodo = (select periodo from cop_periodo
where ejercicio = @vejercicio_actual and
fecha_ini = (select min(fecha_ini) from cop_periodo
              where ejercicio = @vejercicio_actual and
              fecha_ini > @vfecha_fin))

```

Procedimientos Almacenados utilizados para el Cálculo de Datos de los Reportes del Sistema.

Tipo:	Procedimiento Almacenado para elaboración de Reportes
Nombre:	sp_cop_calc_saldo
Código:	SALD_ANT
Descripción:	Procedimiento para el Cálculo de Saldos Reales y/o Estimados de Cuentas de Detalle entre dos fechas

```

create proc sp_cop_calc_saldo
    (@contab varchar(5),
    @minfecha datetime, @maxfecha datetime,
    @mincuenta varchar(30), @maxcuenta varchar(30),
    @t_part tinyint =1)
    /* @t_part: 1=Mayorizadas, 2=No Mayorizadas, 3=Ambos */
as
/* Cálculo de Saldos Reales y/o Estimados de Cuentas de Detalle entre dos fechas */
/* Creado por: sa en SIS04 */
/* Fecha: 19-Mar-1997 */

/* LIMPIAR LA TABLA TEMPORAL */
delete cop_saldo_pnom

/* INSERTAR TODAS LAS CUENTAS DE DETALLE PARA EL RANGO
ESPECIFICADO */

insert into cop_saldo_pnom
    (fecha_saldo, cod_contab, cod_cuenta, cod_cuenta_p, es_detalle, abonos, cargos,
    abonos_anterior, cargos_anterior, cod_t_cuenta)
select getdate(), cod_contab, cod_cuenta, null, detalle, 0, 0, 0, 0, cod_t_cuenta
from cop_cuenta_contab
where cod_contab = @contab and
    (cod_cuenta >= @mincuenta and cod_cuenta <= @maxcuenta) and
    detalle = 1

/* ACTUALIZAR LOS SALDOS CON CALCULOS EN EL RANGO DE CUENTAS Y
PARTIDAS */

```

```
/* SALDOS DE PARTIDAS MAYORIZADAS */
```

```
if @t_part = 1 or @t_part = 3 begin
```

```
    update cop_saldo_pnom
```

```
    set
```

```
    /* CARGOS ACTUALES */
```

```
    cop_saldo_pnom.abonos = cop_saldo_pnom.abonos +
```

```
    (select sum(isnull(d.abonos,0))
```

```
    from cop_partdet_2 d, cop_partenc_2 e
```

```
    where d.cod_contab = cop_saldo_pnom.cod_contab and
```

```
          d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
```

```
          d.cod_contab = e.cod_contab and
```

```
          d.num_corr_ejerc = e.num_corr_ejerc and
```

```
          (e.fecha_partida>=@minfecha and e.fecha_partida<=@maxfecha)),
```

```
    /* ABONOS ACTUALES */
```

```
    cop_saldo_pnom.cargos = cop_saldo_pnom.cargos +
```

```
    (select sum(isnull(d.cargos,0))
```

```
    from cop_partdet_2 d, cop_partenc_2 e
```

```
    where d.cod_contab = cop_saldo_pnom.cod_contab and
```

```
          d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
```

```
          d.cod_contab = e.cod_contab and
```

```
          d.num_corr_ejerc = e.num_corr_ejerc and
```

```
          (e.fecha_partida>=@minfecha and e.fecha_partida<=@maxfecha))
```

```
end
```

```
/* UTILIZAR PARTIDAS NO MAYORIZADAS */
```

```
if @t_part = 1 or @t_part = 3 begin
```

```
    update cop_saldo_pnom
```

```
    set
```

```
    /* ABONOS ACTUALES */
```

```
    cop_saldo_pnom.abonos = cop_saldo_pnom.abonos +
```

```
    (select sum(isnull(d.abonos,0))
```

```
    from cop_partdet_1 d, cop_partenc_1 e
```

```
    where d.cod_contab = cop_saldo_pnom.cod_contab and
```

```
          d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
```

```
          d.cod_contab = e.cod_contab and
```

```
          d.fecha_partida = e.fecha_partida and
```

```
          d.num_corr_d = e.num_corr_d and
```

```
          (e.fecha_partida>=@minfecha and e.fecha_partida<=@maxfecha)),
```

```
    /* CARGOS ACTUALES */
```

```
    cop_saldo_pnom.cargos = cop_saldo_pnom.cargos +
```

```
(select sum(isnull(d.cargos,0))  
from cop_partdet_1 d, cop_partenc_1 e  
where d.cod_contab = cop_saldo_pnom.cod_contab and  
d.cod_cuenta = cop_saldo_pnom.cod_cuenta and  
d.cod_contab = e.cod_contab and  
d.fecha_partida = e.fecha_partida and  
d.num_corr_d = e.num_corr_d and  
(e.fecha_partida >= @minfecha and e.fecha_partida <= @maxfecha))  
  
end
```

Tipo:	Procedimiento Almacenado para elaboración de Reportes
Nombre:	sp_saldopnom
Código:	SALPNOM
Descripción:	Cálculo de los SalDOS de las Partidas No Mayorizadas

```

create proc sp_saldopnom
    (@contab varchar(5),
    @minfecha datetime, @maxfecha datetime,
    @mincuenta varchar(30), @maxcuenta varchar(30),
    @pnom tinyint =0)
as
/* SalDOS Part. No Mayorizadas */

/* IDENTIFICAR EL PERIODO ANTERIOR AL RANGO ESPECIFICADO */

declare @minfecha2 datetime, @maxfecha2 datetime

select @minfecha2 =
    (select p.fecha_ini from cop_periodo p, cop_contabilidad c
    where p.ejercicio = c.ejercicio and
        p.periodo = 1 and c.cod_contab = @contab)

select @maxfecha2 = dateadd(dd, -1, @minfecha)

/* LIMPIAR LA TABLA TEMPORAL */
delete cop_saldo_pnom

/* INSERTAR TODAS LAS CUENTAS PARA EL RANGO ESPECIFICADO */

insert into cop_saldo_pnom
    (fecha_saldo, cod_contab, cod_cuenta, cod_cuenta_p, es_detalle, abonos, cargos,
    abonos_anterior, cargos_anterior)
select getdate(), cod_contab, cod_cuenta, cod_cuenta_p, detalle, 0, 0, 0, 0
from cop_cuenta_contab
where cod_contab = @contab and
    (cod_cuenta >= @mincuenta and cod_cuenta <= @maxcuenta)

/* ACTUALIZAR LOS SALDOS CON CALCULOS EN EL RANGO DE CUENTAS Y
PARTIDAS */

/* SALDOS DE PARTIDAS MAYORIZADAS */
update cop_saldo_pnom

```

set

```

/* CARGOS ACTUALES */
cop_saldo_pnom.abonos = cop_saldo_pnom.abonos +
(select sum(isnull(d.abonos,0))
from cop_partdet_2 d, cop_partenc_2 e
where d.cod_contab = cop_saldo_pnom.cod_contab and
      d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
      d.cod_contab = e.cod_contab and
      d.num_corr_ejerc = e.num_corr_ejerc and
      (e.fecha_partida>=@minfecha and e.fecha_partida<=@maxfecha)),

```

```

/* ABONOS ACTUALES */
cop_saldo_pnom.cargos = cop_saldo_pnom.cargos +
(select sum(isnull(d.cargos,0))
from cop_partdet_2 d, cop_partenc_2 e
where d.cod_contab = cop_saldo_pnom.cod_contab and
      d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
      d.cod_contab = e.cod_contab and
      d.num_corr_ejerc = e.num_corr_ejerc and
      (e.fecha_partida>=@minfecha and e.fecha_partida<=@maxfecha))

```

update cop_saldo_pnom

set

```

/* ABONOS ANTERIORES */
cop_saldo_pnom.abonos_anterior = cop_saldo_pnom.abonos_anterior +
(select sum(isnull(d.abonos,0))
from cop_partdet_2 d, cop_partenc_2 e
where d.cod_contab = cop_saldo_pnom.cod_contab and
      d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
      d.cod_contab = e.cod_contab and
      d.num_corr_ejerc = e.num_corr_ejerc and
      (e.fecha_partida>=@minfecha2 and e.fecha_partida<=@maxfecha2)),

```

```

/* CARGOS ANTERIORES */
cop_saldo_pnom.cargos_anterior = cop_saldo_pnom.cargos_anterior +
(select sum(isnull(d.cargos,0))
from cop_partdet_2 d, cop_partenc_2 e
where d.cod_contab = cop_saldo_pnom.cod_contab and
      d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
      d.cod_contab = e.cod_contab and
      d.num_corr_ejerc = e.num_corr_ejerc and
      (e.fecha_partida>=@minfecha2 and e.fecha_partida<=@maxfecha2))

```

/* UTILIZAR PARTIDAS NO MAYORIZADAS */

```

if @pnom = 1
begin
update cop_saldo_pnom
set
  /* ABONOS ACTUALES */
  cop_saldo_pnom.abonos = cop_saldo_pnom.abonos +
  (select sum(isnull(d.abonos,0))
  from cop_partdet_1 d, cop_partenc_1 e
  where d.cod_contab = cop_saldo_pnom.cod_contab and
        d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
        d.cod_contab = e.cod_contab and
        d.fecha_partida = e.fecha_partida and
        d.num_corr_d = e.num_corr_d and
        (e.fecha_partida >= @minfecha and e.fecha_partida <= @maxfecha)),

  /* CARGOS ACTUALES */
  cop_saldo_pnom.cargos = cop_saldo_pnom.cargos +
  (select sum(isnull(d.cargos,0))
  from cop_partdet_1 d, cop_partenc_1 e
  where d.cod_contab = cop_saldo_pnom.cod_contab and
        d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
        d.cod_contab = e.cod_contab and
        d.fecha_partida = e.fecha_partida and
        d.num_corr_d = e.num_corr_d and
        (e.fecha_partida >= @minfecha and e.fecha_partida <= @maxfecha))

update cop_saldo_pnom
set
  /* ABONOS ANTERIORES */
  cop_saldo_pnom.abonos_anterior = cop_saldo_pnom.abonos_anterior +
  (select sum(isnull(d.abonos,0))
  from cop_partdet_1 d, cop_partenc_1 e
  where d.cod_contab = cop_saldo_pnom.cod_contab and
        d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
        d.cod_contab = e.cod_contab and
        d.fecha_partida = e.fecha_partida and
        d.num_corr_d = e.num_corr_d and
        (e.fecha_partida >= @minfecha2 and e.fecha_partida <= @maxfecha2)),

  /* CARGOS ANTERIORES */
  cop_saldo_pnom.cargos_anterior = cop_saldo_pnom.cargos_anterior +
  (select sum(isnull(d.cargos,0))
  from cop_partdet_1 d, cop_partenc_1 e
  where d.cod_contab = cop_saldo_pnom.cod_contab and
        d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
        d.cod_contab = e.cod_contab and

```

```
d.fecha_partida = e.fecha_partida and  
d.num_corr_d = e.num_corr_d and  
(e.fecha_partida >= @minfecha2 and e.fecha_partida <= @maxfecha2))
```

```
end /* PARTIDAS NO MAYORIZADAS */
```

Tipo:	Procedimiento Almacenado para elaboración de Reportes
Nombre:	sp_cop_comparativo
Código:	CÓMPAL2
Descripción:	Prepara Datos para los Reportes Comparativos

```

create proc sp_cop_comparativo
    (@contab varchar(5),
    @v_periodo1 int,
    @v_ejercicio1 int,
    @v_periodo2 int,
    @v_ejercicio2 int)
as
/* Prepara los Datos para los Reportes Comparativos */

/* DECLARACION DE VARIABLES A UTILIZAR */
declare @tabla1 char(1), @tabla2 char(2), @mensaje varchar(100)

delete cop_saldo_comp
where cod_contab=@contab

/* VERIFICAR SI EXISTEN SALDOS DE TALES PERIODOS, Y QUE ESTEN
CERRADOS */

/* REVISAR EL PERIODO 1 */
if exists(select * from cop_saldo where cod_contab=@contab and
    ejercicio=@v_ejercicio1 and periodo=@v_periodo1 and cerrado=1)
begin
    /* PERIODO 1 ESTA EN COP_SALDO */
    select @tabla1 = "A"
end
else begin
    if exists(select * from cop_saldo_h where cod_contab=@contab and
        ejercicio=@v_ejercicio1 and periodo=@v_periodo1 and cerrado=1)
    begin
        /* PERIODO 1 ESTA EN COP_SALDO_H */
        select @tabla1 = "H"
    end
    else begin
        /* DE PERIODO 1 NO HAY SALDOS O NO ESTA CERRADO */
        select @tabla1 = "N"
        select @mensaje = "No existen SalDOS para el Periodo:" + convert(varchar,
@v_periodo1) +
            " del Ejercicio " + convert(varchar, @v_ejercicio1) +
            ", o el Período NO está cerrado aún."
    end
end

```

```

        print @mensaje
        return
    end
end

/* REVISAR EL PERIODO 2 */
if exists(select * from cop_saldo where cod_contab=@contab and
        ejercicio=@v_ejercicio2 and periodo=@v_periodo2 and cerrado=1)
begin
    /* PERIODO 2 ESTA EN COP_SALDO */
    select @tabla2 = "A"
end
else begin
    if exists(select * from cop_saldo_h where cod_contab=@contab and
            ejercicio=@v_ejercicio2 and periodo=@v_periodo2 and cerrado=1)
        begin
            /* PERIODO 2 ESTA EN COP_SALDO_H */
            select @tabla2 = "H"
        end
    else begin
        /* DE PERIODO 2 NO HAY SALDOS O NO ESTA CERRADO */
        select @tabla2 = "N"
        select @mensaje = "No existen SalDOS para el Periodo:" + convert(varchar,
            @v_periodo2) +
            " del Ejercicio " + convert(varchar, @v_ejercicio2) +
            ", o el Período NO está cerrado aún."
        print @mensaje
        return
    end
end

/* AMBOS PERIODOS ESTAN CERRADOS Y TIENEN SALDOS */

/* AGREGAR SALDOS SEGUN LA TABLA EN LA QUE SE ENCUENTRE EL
PERIODO 1 */

if @tabla1 = "A" begin
    insert into cop_saldo_comp(cod_contab, cod_cuenta, descrip_cta, cod_t_cuenta,
        cargos1, abonos1, cargos2, abonos2)
    select @contab, c.cod_cuenta, c.descrip_cta, c.cod_t_cuenta, sum(s.cargos),
sum(s.abonos), 0, 0
    from cop_saldo s, cop_cuenta_contab c
    where s.cod_contab = @contab and
        s.ejercicio = @v_ejercicio1 and
        s.periodo = @v_periodo1 and
        s.cod_contab = c.cod_contab and

```

```

        s.cod_cuenta = c.cod_cuenta and
        c.detalle = 1
    group by c.cod_contab, c.cod_cuenta, c.descripcion_cta, c.cod_t_cuenta
end
else begin
    insert into cop_saldo_comp(cod_contab, cod_cuenta, cod_t_cuenta, descripcion_cta,
    cargos1, abonos1, cargos2, abonos2)
    select @contab, c.cod_cuenta, c.descripcion_cta, c.cod_t_cuenta, sum(s.cargos),
sum(s.abonos), 0, 0
    from cop_saldo_h s, cop_cuenta_contab_h c
    where s.cod_contab = @contab and
        s.ejercicio = @v_ejercicio1 and
        s.periodo = @v_periodo1 and
        s.cod_contab = c.cod_contab and
        s.cod_cuenta = c.cod_cuenta and
        c.detalle = 1
    group by c.cod_contab, c.cod_cuenta, c.descripcion_cta, c.cod_t_cuenta
end

create table #tmp_sal2(
    cod_contab char(5),
    cod_cuenta varchar(30),
    cargos money,
    abonos money)

/* AGREGAR SALDOS SEGUN LA TABLA EN LA QUE SE ENCUENTRE EL
PERIODO 2 */
if @tabla2 = "A" begin

    /* SI NO EXISTEN, INSERTARLAS */
    insert into #tmp_sal2(cod_contab, cod_cuenta, cargos, abonos)
    select @contab, c.cod_cuenta, sum(s.cargos), sum(s.abonos)
    from cop_saldo s, cop_cuenta_contab c
    where s.cod_contab = @contab and
        s.ejercicio = @v_ejercicio2 and
        s.periodo = @v_periodo2 and
        s.cod_contab = c.cod_contab and
        s.cod_cuenta = c.cod_cuenta and
        c.detalle = 1
    group by c.cod_contab, c.cod_cuenta, c.descripcion_cta, c.cod_t_cuenta

    /* ACTUALIZARLAS */
    update cop_saldo_comp
    set cargos2=t.cargos, abonos2=t.abonos
    from cop_saldo_comp s, #tmp_sal2 t
    where s.cod_contab=t.cod_contab and

```

```
s.cod_cuenta=t.cod_cuenta
```

```
insert into cop_saldo_comp(cod_contab, cod_cuenta, descrip_cta, cod_t_cuenta,
cargos1, abonos1, cargos2, abonos2)
select @contab, c.cod_cuenta, c.descripcion_cta, c.cod_t_cuenta, 0, 0, sum(s.cargos),
sum(s.abonos)
from cop_saldo s, cop_cuenta_contab c
where s.cod_contab = @contab and
s.ejercicio = @v_ejercicio2 and
s.periodo = @v_periodo2 and
s.cod_contab = c.cod_contab and
s.cod_cuenta = c.cod_cuenta and
c.detalle = 1 and
c.cod_cuenta not in (select distinct cod_cuenta from cop_saldo_comp)
group by c.cod_contab, c.cod_cuenta, c.descripcion_cta, c.cod_t_cuenta
```

```
end
else begin
```

```
/* SI NO EXISTEN, INSERTARLAS */
insert into cop_saldo_comp(cod_contab, cod_cuenta, descrip_cta, cod_t_cuenta,
cargos1, abonos1, cargos2, abonos2)
select @contab, c.cod_cuenta, c.descripcion_cta, c.cod_t_cuenta, 0, 0, sum(s.cargos),
sum(s.abonos)
from cop_saldo_h s, cop_cuenta_contab_h c
where s.cod_contab = @contab and
s.ejercicio = @v_ejercicio2 and
s.periodo = @v_periodo2 and
s.cod_contab = c.cod_contab and
s.cod_cuenta = c.cod_cuenta and
c.detalle = 1 and
c.cod_cuenta not in (select distinct cod_cuenta from cop_saldo_comp)
group by c.cod_contab, c.cod_cuenta, c.descripcion_cta, c.cod_t_cuenta
```

```
end
```

Tipo:	Procedimiento Almacenado para elaboración de Reportes
Nombre:	sp_diamayor
Código:	DIAMAYOR
Descripción:	Cálculo de los saldos de las cuentas de mayor para un rango de cuentas, limitadas además por un rango de partidas (fechas).

```

create proc sp_diamayor
    (@contab varchar(5),
    @minfecha datetime, @maxfecha datetime,
    @mincuenta varchar(30), @maxcuenta varchar(30),
    @pnom tinyint = 0)
as
/* Cálculo Saldos Libro Diario Mayor */

/* Variables temporales para el procedimiento */
declare @fecha_actual datetime, @contador smallint

/* Limpiar tabla temporal*/
delete cop_saldo_pnom

/* Insertar cuentas que hayan tenido movimientos y están dentro del rango
(se asume que sólo hay de detalle en los detalles de partidas)*/

/* UTILIZAR PARTIDAS NO MAYORIZADAS */
if @pnom = 1
begin
    insert into cop_saldo_pnom
    (fecha_saldo, cod_contab, cod_cuenta, cod_cuenta_p, es_detalle, abonos, cargos)
    select d.fecha_partida, d.cod_contab, d.cod_cuenta, c.cod_cuenta_p, c.detalle, 0, 0
    from cop_partdet_1 d, cop_cuenta_contab c
    where d.cod_cuenta = c.cod_cuenta and
        c.cod_contab = d.cod_contab and
        d.cod_contab = @contab and
        (d.fecha_partida >= @minfecha and d.fecha_partida <= @maxfecha) and
        (d.cod_cuenta >= @mincuenta and d.cod_cuenta <= @maxcuenta)
    group by
        d.fecha_partida, d.cod_contab, d.cod_cuenta, c.cod_cuenta_p, c.detalle
    order by
        d.fecha_partida, d.cod_contab, d.cod_cuenta
end

/* Se buscará e insertará todas las cuentas padres, una copia por cada fecha, y hasta
después se actualizan los saldos*/

```

```

select @fecha_actual = @minfecha
while @fecha_actual<=@maxfecha
begin
    select @contador = 1
    /* Si hay cuentas hijas pero no cuentas padre, agregar estas últimas */
    while exists (select * from cop_saldo_pnom
        where cod_cuenta_p is not null
        and abonos=0 and cargos=0) and @contador<17
    begin

        /* UTILIZAR PARTIDAS NO MAYORIZADAS */
        if @pnom = 1
        begin
            insert into cop_saldo_pnom
            (fecha_saldo, cod_contab, cod_cuenta, cod_cuenta_p, es_detalle,
abonos, cargos)
            select distinct
            @fecha_actual, s.cod_contab, c.cod_cuenta, c.cod_cuenta_p, c.detalle,
0,0
            from cop_saldo_pnom s, cop_cuenta_contab c
            where s.cod_cuenta_p = c.cod_cuenta and
            s.cod_contab = c.cod_contab and
            c.cod_cuenta not in
            (select cod_cuenta from cop_saldo_pnom where
            fecha_saldo=@fecha_actual)
        end /* PARTIDAS NO MAYORIZADAS */

        select @contador = @contador + 1

    end /* DEL WHILE */

    select @fecha_actual = dateadd(dd, 1, @fecha_actual)

end

/* Actualización de saldos para cada cuenta en cada fecha */

/* UTILIZAR PARTIDAS NO MAYORIZADAS */
if @pnom = 1
begin

    update cop_saldo_pnom
    set cop_saldo_pnom.abonos = cop_saldo_pnom.abonos +
    (select sum(cop_partdet_1.abonos)
    from cop_partdet_1

```

```
where cop_partdet_1.cod_contab = cop_saldo_pnom.cod_contab and  
cop_partdet_1.cod_cuenta = cop_saldo_pnom.cod_cuenta and  
cop_partdet_1.cod_contab = @contab and  
cop_partdet_1.fecha_partida=cop_saldo_pnom.fecha_saldo),
```

```
cop_saldo_pnom.cargos = cop_saldo_pnom.cargos +  
(select sum(cop_partdet_1.cargos)  
from cop_partdet_1  
where cop_partdet_1.cod_contab = cop_saldo_pnom.cod_contab and  
cop_partdet_1.cod_cuenta = cop_saldo_pnom.cod_cuenta and  
cop_partdet_1.cod_contab = @contab and  
cop_partdet_1.fecha_partida=cop_saldo_pnom.fecha_saldo)
```

```
end      /* PARTIDAS NO MAYORIZADAS */
```

Tipo:	Procedimiento Almacenado para elaboración de Reportes
Nombre:	sp_est_financieros
Código:	EST_FIN1
Descripción:	Elaboración de los Estados Financieros (Balance General y Estado de Resultados)

```

create proc sp_est_financieros
    (@contab varchar(5),
    @minfecha datetime, @maxfecha datetime,
    @tipo_reporte tinyint, @pnom tinyint = 0)
as
/* Elaboración de Estado Financieros */

/* Limpiar tabla temporal*/
delete cop_saldo_pnom

/* Insertar todas las cuentas según el reporte */

/* ESTADO DE RESULTADOS */
if @tipo_reporte = 1
begin
    insert into cop_saldo_pnom
    (fecha_saldo, cod_contab, cod_cuenta, cod_cuenta_p, es_detalle, abonos, cargos,
    abonos_anterior, cargos_anterior)
    select getdate(), cod_contab, cod_cuenta, cod_cuenta_p, detalle, 0, 0, 0, 0
    from cop_cuenta_contab c, cop_t_cuenta t
    where c.cod_contab = @contab and
           c.cod_t_cuenta = t.cod_t_cuenta and
           (t.t_cuenta_gral = "I" or t.t_cuenta_gral="G")
end

/* BALANCE GENERAL */
else
begin
    insert into cop_saldo_pnom
    (fecha_saldo, cod_contab, cod_cuenta, cod_cuenta_p, es_detalle, abonos, cargos,
    abonos_anterior, cargos_anterior)
    select getdate(), cod_contab, cod_cuenta, cod_cuenta_p, detalle, 0, 0, 0, 0
    from cop_cuenta_contab c, cop_t_cuenta t
    where c.cod_contab = @contab and
           c.cod_t_cuenta = t.cod_t_cuenta and
           (t.t_cuenta_gral = "A" or t.t_cuenta_gral="P" or t.t_cuenta_gral="C")
end

```

```
/* Actualizar los saldos con cálculos del rango de cuentas y partidas */
```

```
/* UTILIZAR PARTIDAS MAYORIZADAS */
```

```
update cop_saldo_pnom
set cop_saldo_pnom.abonos = cop_saldo_pnom.abonos +
(select sum(isnull(d.abonos,0))
from cop_partdet_2 d, cop_partenc_2 e
where d.cod_contab = cop_saldo_pnom.cod_contab and
d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
d.cod_contab = e.cod_contab and
d.num_corr_ejerc = e.num_corr_ejerc and
(e.fecha_partida>=@minfecha and e.fecha_partida<=@maxfecha)),
```

```
cop_saldo_pnom.cargos = cop_saldo_pnom.cargos +
(select sum(isnull(d.cargos,0))
from cop_partdet_2 d, cop_partenc_2 e
where d.cod_contab = cop_saldo_pnom.cod_contab and
d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
d.cod_contab = e.cod_contab and
d.num_corr_ejerc = e.num_corr_ejerc and
(e.fecha_partida>=@minfecha and e.fecha_partida<=@maxfecha))
```

```
/* UTILIZAR PARTIDAS NO MAYORIZADAS */
```

```
if @pnom = 1
```

```
update cop_saldo_pnom
set cop_saldo_pnom.abonos = cop_saldo_pnom.abonos +
(select sum(isnull(d.abonos,0))
from cop_partdet_1 d, cop_partenc_1 e
where d.cod_contab = cop_saldo_pnom.cod_contab and
d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
d.cod_contab = e.cod_contab and
d.fecha_partida = e.fecha_partida and
d.num_corr_d = e.num_corr_d and
(e.fecha_partida>=@minfecha and e.fecha_partida<=@maxfecha)),
```

```
cop_saldo_pnom.cargos = cop_saldo_pnom.cargos +
(select sum(isnull(d.cargos,0))
from cop_partdet_1 d, cop_partenc_1 e
where d.cod_contab = cop_saldo_pnom.cod_contab and
d.cod_cuenta = cop_saldo_pnom.cod_cuenta and
d.cod_contab = e.cod_contab and
d.fecha_partida = e.fecha_partida and
d.num_corr_d = e.num_corr_d and
```

```
(e.fecha_partida>=@minfecha and e.fecha_partida<=@maxfecha))
```

```
/* CALCULAR PERDIDAS Y GANANCIAS EN EL EJERCICIO PARA  
COMPLETAR EL BALANCE GENERAL */
```

```
if @tipo_reporte = 2
```

```
begin
```

```
/* CALCULO */
```

```
declare @cargos_pg money, @abonos_pg money, @cargos_ant money, @abonos_ant  
money
```

```
/* UTILIZAR PARTIDAS MAYORIZADAS */
```

```
select @cargos_pg = sum(isnull(cargos,0)), @abonos_pg=sum(isnull(abonos,0))  
from cop_partdet_2 d, cop_partenc_2 e, cop_cuenta_contab c, cop_t_cuenta t  
where d.cod_contab = c.cod_contab and  
d.cod_cuenta = c.cod_cuenta and  
c.cod_t_cuenta = t.cod_t_cuenta and  
d.cod_contab=@contab and  
d.cod_contab=e.cod_contab and  
d.num_corr_ejerc=e.num_corr_ejerc and  
t_cuenta_gral = "I" and  
(e.fecha_partida>=@minfecha and e.fecha_partida<=@maxfecha)
```

```
select @cargos_pg = isnull(@cargos_pg, 0), @abonos_pg = isnull(@abonos_pg, 0)  
select @cargos_ant = @cargos_pg, @abonos_ant = @abonos_pg
```

```
select @cargos_pg = @cargos_pg + sum(isnull(cargos,0)), @abonos_pg =  
@abonos_pg + sum(isnull(abonos,0))  
from cop_partdet_2 d, cop_partenc_2 e, cop_cuenta_contab c, cop_t_cuenta t  
where d.cod_contab = c.cod_contab and  
d.cod_cuenta = c.cod_cuenta and  
c.cod_t_cuenta = t.cod_t_cuenta and  
d.cod_contab=@contab and  
d.cod_contab=e.cod_contab and  
d.num_corr_ejerc=e.num_corr_ejerc and  
t_cuenta_gral = "G" and  
(e.fecha_partida>=@minfecha and e.fecha_partida<=@maxfecha)
```

```
select @cargos_pg = isnull(@cargos_pg, @cargos_ant), @abonos_pg =  
isnull(@abonos_pg, @abonos_ant)  
select @cargos_ant = @cargos_pg, @abonos_ant = @abonos_pg
```

```
/* UTILIZAR PARTIDAS NO MAYORIZADAS */
```

```
if @pnom = 1
```

```
begin
```

```

select @cargos_pg=@cargos_pg+sum(isnull(cargos,0)),
@abonos_pg=@abonos_pg+sum(isnull(abonos,0))
from cop_partdet_1 d, cop_cuenta_contab c, cop_t_cuenta t
where d.cod_contab = c.cod_contab and
      d.cod_cuenta = c.cod_cuenta and
      c.cod_t_cuenta = t.cod_t_cuenta and
      d.cod_contab=@contab and
      t_cuenta_gral = "I" and
      (d.fecha_partida>=@minfecha and d.fecha_partida<=@maxfecha)

```

```

select @cargos_pg = isnull(@cargos_pg, @cargos_ant), @abonos_pg =
isnull(@abonos_pg, @abonos_ant)
select @cargos_ant = @cargos_pg, @abonos_ant = @abonos_pg

```

```

select @cargos_pg = @cargos_pg + sum(isnull(cargos,0)), @abonos_pg =
@abonos_pg + sum(isnull(abonos,0))
from cop_partdet_1 d, cop_cuenta_contab c, cop_t_cuenta t
where d.cod_contab = c.cod_contab and
      d.cod_cuenta = c.cod_cuenta and
      c.cod_t_cuenta = t.cod_t_cuenta and
      d.cod_contab=@contab and
      t_cuenta_gral = "G" and
      (d.fecha_partida>=@minfecha and d.fecha_partida<=@maxfecha)

```

```

select @cargos_pg = isnull(@cargos_pg, @cargos_ant), @abonos_pg =
isnull(@abonos_pg, @abonos_ant)

```

```

end /* FIN PARTIDAS NO MAYORIZADAS */

```

```

/* ASIGNACION */

```

```

update cop_saldo_pnom
set cargos = @cargos_pg, abonos = @abonos_pg
from cop_saldo_pnom, cop_contabilidad
where cop_saldo_pnom.cod_contab = cop_contabilidad.cod_contab and
      cop_saldo_pnom.cod_cuenta = cop_contabilidad.cuenta_superavit and
      cop_saldo_pnom.cod_contab = @contab

```

```

end

```

Scripts complementarios para los Triggers generados por OpenSnap

Tipo:	Complemento Antes del Trigger de Inserción de cop_contabilidad
Nombre:	trins!!cop_contabilidad
Código:	COP_TR07

```

/* Antes del trins de cop_contabilidad */
if @rowcount != 0
begin /* inicio de validaciones */

/* Validar que Cuenta Superavit del Ejercicio exista */
if update(cuenta_superavit)
begin
if (select count(*) from inserted, cop_cuenta_contab
where cop_cuenta_contab.cod_cuenta = inserted.cuenta_superavit and
cop_cuenta_contab.cod_contab = inserted.cod_contab and
inserted.cuenta_superavit is not null) <> ( select count(*) from inserted
where cuenta_superavit is not null)

begin
set rowcount 1
select @error_message = "No existe Cuenta Superavit del Ejercicio " +
convert(varchar, inserted.cod_contab) + ":" + convert(varchar, inserted.cuenta_superavit)
from inserted
where inserted.cuenta_superavit is not null and
not exists(select * from cop_cuenta_contab
where inserted.cuenta_superavit = cop_cuenta_contab.cod_cuenta and
inserted.cod_contab = cop_cuenta_contab.cod_contab)

if @error_message is null
begin
select @error_message="Error en Cuenta Superavit"
end

set rowcount 0
raiserror 40000 @error_message
rollback transaction
return
end
end
end

```

Tipo:	Complemento Antes del Trigger de Inserción de cop_cuenta_contab
Nombre:	trins!!cop_cuenta_contab
Código:	cop_TR01

```

/* antes del trins de cop_cuenta_contab */
if @rowcount != 0
begin /* inicio de validaciones */

/* Validar si la cuenta padre existe y es correcta */
if update(cod_cuenta_p)
begin
if (select count(*) from inserted, cop_cuenta_contab
where inserted.cod_cuenta_p = cop_cuenta_contab.cod_cuenta and
inserted.cod_contab = cop_cuenta_contab.cod_contab and
inserted.cod_cuenta != inserted.cod_cuenta_p and
inserted.cod_cuenta_p is not null) != (select count(*) from inserted where
cod_cuenta_p is not null)
begin
set rowcount 1
select @error_message = "No existe Cuenta Padre " + convert(varchar,
inserted.cod_contab) + ":" + convert(varchar, inserted.cod_cuenta_p)
from inserted
where inserted.cod_cuenta_p is not null and
not exists(select * from cop_cuenta_contab
where inserted.cod_cuenta_p = cop_cuenta_contab.cod_cuenta and
inserted.cod_contab = cop_cuenta_contab.cod_contab)

set rowcount 0
raiserror 40000 @error_message
rollback transaction
return
end
end

/* Validar que la cuenta padre no sea la Cuenta de Superavit */
if update(cod_cuenta_p)
begin
if exists(select * from inserted, cop_contabilidad
where inserted.cod_cuenta_p = cop_contabilidad.cuenta_superavit and
inserted.cod_contab = cop_contabilidad.cod_contab)
begin

set rowcount 1
select @error_message = "No puede subdividirse la Cuenta de Superavit " +
convert(varchar, inserted.cod_contab) + ":" + convert(varchar, inserted.cod_cuenta_p)
from inserted

```

```

where inserted.cod_cuenta_p is not null and
  exists(select * from cop_contabilidad
    where inserted.cod_cuenta_p = cop_contabilidad.cuenta_superavit and
      inserted.cod_contab = cop_contabilidad.cod_contab)

set rowcount 0
raiserror 40000 @error_message
rollback transaction
return
end
end

/* Verificar que cuenta padre no se haya utilizado en partidas */
if update(cod_cuenta_p)
begin
  if exists(select * from inserted, cop_partdet_1
    where inserted.cod_cuenta_p = cop_partdet_1.cod_cuenta and
      inserted.cod_contab = cop_partdet_1.cod_contab)
  begin

    set rowcount 1
    select @error_message = "La cuenta "+ inserted.cod_cuenta_p + " en " +
      inserted.cod_contab + " es utilizada en las partidas no mayorizadas. No puede
      subdividirse."
    from inserted, cop_partdet_1
    where inserted.cod_contab = cop_partdet_1.cod_contab and
      inserted.cod_cuenta_p = cop_partdet_1.cod_cuenta

    set rowcount 0
    raiserror 40000 @error_message
    rollback transaction
    return
  end

  if exists(select * from inserted, cop_partdet_2
    where inserted.cod_cuenta_p = cop_partdet_2.cod_cuenta and
      inserted.cod_contab = cop_partdet_2.cod_contab)
  begin

    set rowcount 1
    select @error_message = "La cuenta "+ inserted.cod_cuenta_p + " en " +
      inserted.cod_contab + " es utilizada en las partidas mayorizadas. No puede
      subdividirse."
    from inserted, cop_partdet_2
    where inserted.cod_contab = cop_partdet_2.cod_contab and
      inserted.cod_cuenta_p = cop_partdet_2.cod_cuenta
  end
end

```

```

        set rowcount 0
        raiserror 40000 @error_message
        rollback transaction
        return
    end
end

/* Verificar si el nivel es el correcto */
if update(nivel)
begin
    if (select count(*) from inserted, cop_cuenta_contab
        where inserted.cod_cuenta_p = cop_cuenta_contab.cod_cuenta and
            inserted.cod_contab=cop_cuenta_contab.cod_contab and
            inserted.cod_cuenta != inserted.cod_cuenta_p and
            inserted.nivel = (cop_cuenta_contab.nivel + 1) and
            inserted.cod_cuenta_p is not null) != (select count(*) from inserted where
            cod_cuenta_p is not null)
    begin
        set rowcount 1
        select @error_message = "Nivel erroneo en Cuenta " + convert(varchar,
            inserted.cod_contab) + ":" + convert(varchar,
            inserted.cod_cuenta)+"."+convert(varchar, inserted.nivel)
        from inserted
        where inserted.cod_cuenta_p is not null and
            not exists(select * from cop_cuenta_contab
                where inserted.cod_cuenta_p = cop_cuenta_contab.cod_cuenta and
                    inserted.cod_contab = cop_cuenta_contab.cod_contab and
                    inserted.nivel = (cop_cuenta_contab.nivel + 1))

        set rowcount 0
        raiserror 40000 @error_message
        rollback transaction
        return
    end

    if (select count(*) from inserted
        where inserted.cod_cuenta_p is null and
            inserted.nivel != 1) > 1
    begin
        set rowcount 1
        select @error_message = "Nivel erroneo en cuenta " + convert(varchar,
            inserted.cod_contab) + ":" + convert(varchar,
            inserted.cod_cuenta)+"."+convert(varchar, inserted.nivel)
        from inserted
        where inserted.cod_cuenta_p is null and inserted.nivel != 1
    end
end

```

```

set rowcount 0
raiserror 40000 @error_message
rollback transaction
end
end

/* Verificar si el tipo es el correcto */
if update(cod_t_cuenta)
begin
if (select count(*) from inserted, cop_cuenta_contab
where inserted.cod_cuenta_p = cop_cuenta_contab.cod_cuenta and
inserted.cod_contab=cop_cuenta_contab.cod_contab and
inserted.cod_cuenta != inserted.cod_cuenta_p and
inserted.cod_t_cuenta = cop_cuenta_contab.cod_t_cuenta and
inserted.cod_cuenta_p is not null) != (select count(*) from inserted where
cod_cuenta_p is not null)
begin

set rowcount 1
select @error_message = "Tipo distinto de Cuenta Padre en Cuenta " +
convert(varchar, inserted.cod_contab) + ":" + convert(varchar, inserted.cod_cuenta)+"
Tipo:"+convert(varchar, inserted.cod_t_cuenta)
from inserted
where inserted.cod_cuenta_p is not null and
not exists(select * from cop_cuenta_contab
where inserted.cod_cuenta_p = cop_cuenta_contab.cod_cuenta and
inserted.cod_contab = cop_cuenta_contab.cod_contab and
inserted.cod_t_cuenta = cop_cuenta_contab.cod_t_cuenta)

if @error_message is null
begin
select @error_message="Error: mensaje nulo"
end

set rowcount 0
raiserror 40000 @error_message
rollback transaction
return
end
end
/*Verificar que Cuenta de Consolidacion es correcta*/
if update(cod_contab_c) or update(cod_cuenta_c)
begin
if (select count(*) from cop_cuenta_contab c, inserted i
where (i.cod_contab_c=c.cod_contab and
i.cod_cuenta_c=c.cod_cuenta_c and

```

```
        i.cod_contab!=c.cod_contab and
        i.cod_contab_c is not null and i.cod_cuenta_c is not null) or
        (i.cod_contab_c is null and i.cod_cuenta_c is null))!=@rowcount
begin
    select @error_message="Se trato de asignar una cuenta de consolidacion
    inexistente o no valida!"
    raiserror 40000 @error_message
    rollback trigger
    return
end
end
end /* Fin de Validaciones */
```

Tipo:	Complemento Después del Trigger de Inserción de cop_cuenta_contab
Nombre:	trins::cop_cuenta_contab
Código:	cop_TR02

```
/* luego del trins de cop_cuenta_contab */
if @rowcount != 0
begin /* los padres de las nuevas cuentas ya no son de detalle */

if update(cod_contab) or update(cod_cuenta)
begin
    update cop_cuenta_contab
    set     detalle = 0
    from inserted
    where  cop_cuenta_contab.cod_contab = inserted.cod_contab and
           cop_cuenta_contab.cod_cuenta = inserted.cod_cuenta_p
end
end /* Fin del Proceso */
```

Tipo:	Complemento Antes del Trigger de Borrado de cop_cuenta_contab
Nombre:	trdel!!cop_cuenta_contab
Código:	cop_TR03

```

/* antes del trdel de cop_cuenta_contab */
if @rowcount != 0 begin

/* Verificar si tiene sub-cuentas */
if exists(select * from deleted d, cop_cuenta_contab c

        where d.cod_contab = c.cod_contab and
              d.cod_cuenta = c.cod_cuenta_p)
begin

raiserror 41000 "Existen Cuentas de Detalle para esa Cuenta"

rollback transaction

return

end
/* Verificar si tiene sub-cuentas */
if exists(select * from deleted d, cop_cuenta_contab c

        where d.cod_contab = c.cod_contab_c and
              d.cod_cuenta = c.cod_cuenta_c)
begin

raiserror 41000 "Existen Cuentas de Detalle en otras Contabilidades para esa Cuenta"

rollback transaction

return

end
end

```

Tipo:	Complemento Después del Trigger de Borrado de cop_cuenta_contab
Nombre:	trdel::cop_cuenta_contab
Código:	COP_TR08

```
/* luego del trdel de cop_cuenta_contab */
```

```
/* Cambia detalle a 1 para cuentas que ya no tienen hijos */
```

```
if @rowcount != 0
```

```
begin
```

```
  if exists(select * from deleted, cop_cuenta_contab
```

```
    where deleted.cod_cuenta_p= cop_cuenta_contab.cod_cuenta and
```

```
    deleted.cod_contab= cop_cuenta_contab.cod_contab and
```

```
    not exists(select * from cop_cuenta_contab , deleted
```

```
      where deleted.cod_cuenta_p= cop_cuenta_contab.cod_cuenta_p
```

```
and
```

```
        deleted.cod_contab= cop_cuenta_contab.cod_contab))
```

```
begin
```

```
  update cop_cuenta_contab
```

```
  set detalle = 1
```

```
  from deleted
```

```
  where deleted.cod_cuenta_p= cop_cuenta_contab.cod_cuenta and
```

```
  deleted.cod_contab= cop_cuenta_contab.cod_contab and
```

```
  not exists(select * from cop_cuenta_contab , deleted
```

```
    where deleted.cod_cuenta_p= cop_cuenta_contab.cod_cuenta_p
```

```
and
```

```
      deleted.cod_contab= cop_cuenta_contab.cod_contab)
```

```
end
```

```
end
```

Tipo:	Complemento Antes del Trigger de Borrado de pre_rubro
Nombre:	trdel!!pre_rubro
Código:	PRETR05

```
/* antes del trdel de pre_rubro */
if @rowcount != 0
begin
    if exists(
        select * from pre_rubro, deleted
        where pre_rubro.cod_contab=deleted.cod_contab and
              pre_rubro.cod_rubro_p = deleted.cod_rubro)
    begin
        set rowcount 1
        select @error_message= "No se puede Borrar el rubro: " +
            convert(varchar,deleted.cod_contab + " " + deleted.cod_rubro) + " porque posee
            rubros de detalle!"
        from deleted
        where exists( select * from pre_rubro
                    where pre_rubro.cod_contab = deleted.cod_contab and
                          pre_rubro.cod_rubro_p = deleted.cod_rubro)

        set rowcount 0
        raiserror 40000 @error_message
        rollback trigger
        return
    end
end
```

Tipo:	Complemento Después del Trigger de Borrado de pre_rubro
Nombre:	trdel::pre_rubro
Código:	PRETR04

```
/* luego del trdel de pre_rubro */
if @rowcount != 0
begin
    if (    select count(*) from deleted
        where not exists (    select * from pre_rubro
                            where pre_rubro.cod_contab=deleted.cod_contab and
                                pre_rubro.cod_rubro_p=deleted.cod_rubro)) > 0
    begin
        update pre_rubro
        set pre_rubro.detalle=1
        from deleted
        where deleted.cod_contab=pre_rubro.cod_contab and
            deleted.cod_rubro_p=pre_rubro.cod_rubro
    end
end
```

Tipo:	Complemento Antes del Trigger de Inserción de cop_partenc_2
Nombre:	trins!!cop_partenc_2
Código:	COP_TR04

```

/* antes del trins de cop_partenc_2 */
if @rowcount != 0
begin
if update(fecha_partida) or update(ejercicio) or update(periodo)
begin
if (select count(*) from inserted i,cop_contabilidad c,cop_periodo p
where i.cod_contab=c.cod_contab and
p.ejercicio=c.ejercicio and
p.periodo=c.periodo and
(i.fecha_partida>=p.fecha_ini and i.fecha_partida<=p.fecha_fin) and
i.ejercicio=p.ejercicio and
i.periodo=p.periodo)!=@rowcount
begin
set rowcount 0
print "La fecha de la partida no corresponde al periodo actual"
raiserror 40000 "La fecha de la partida no corresponde al periodo actual!"
rollback transaction
return
end
end
end

```

Tipo:	Complemento Antes del Trigger de Inserción de pre_rubro
Nombre:	trins!!pre_rubro
Código:	PRETR01

```

/* antes del trins de pre_rubro */
declare @rowcountnonull int
select @rowcountnonull = count(*) from inserted where inserted.cod_rubro_p is not null

if (@rowcount != 0)
begin
    if update(cod_rubro_p)
    begin
        /* Verificar si existe el Rubro padre especificado */
        if (    select count(*) from pre_rubro r, inserted i
              where i.cod_contab = r.cod_contab and
                    i.cod_rubro_p = r.cod_rubro and
                    i.cod_rubro_p is not null) != @rowcountnonull
        begin
            set rowcount 1
            select @error_message =    "No Existe El Rubro Padre: " +
                convert(varchar,inserted.cod_rubro_p) +
                " para el Rubro: " + convert(varchar,inserted.cod_rubro)
            from inserted
            where not exists(    select * from pre_rubro
                               where inserted.cod_rubro_p =
pre_rubro.cod_rubro and
                               inserted.cod_rubro_p is not null)

            set rowcount 0
            raiserror 40000 @error_message
            rollback trigger
            return
        end
    end
    if update(nivel)
    begin
        /* Verificar si el nivel jerarquico es correcto */
        if (    select count(*) from pre_rubro r, inserted i
              where i.cod_contab = r.cod_contab and
                    i.cod_rubro_p = r.cod_rubro and
                    i.cod_rubro_p is not null and
                    i.nivel - 1 = r.nivel) != @rowcountnonull
        begin
            set rowcount 1
            select @error_message = "El Nivel Jerarquico del Rubro: " +
                convert(varchar, inserted.cod_rubro) + " es incorrecto!"
        end
    end
end

```

```
from inserted
where not exists(
    select * from pre_rubro
    where inserted.cod_rubro_p =
    pre_rubro.cod_rubro and
    inserted.cod_rubro_p is not null and
    inserted.nivel - 1 = pre_rubro.nivel)

set rowcount 0
raiserror 40000 @error_message
rollback trigger
return

end
end
end
```

Tipo:	Complemento Antes del Trigger de Inserción de pre_rubro
Nombre:	trins!!pre_rubro
Código:	PRETR01

```

procedimiento 'PRETR03' {
  nombre 'trins!!pre_rubro_cuenta'
  descripción 'antes del trins de pre_rubro_cuenta'
  usuario 'sa'
  fecha 'Apr 28 1997 5:13PM'
  tabla 'pre_rubro_cuenta'
  tipo '17'
  /* antes del trins de pre_rubro_cuenta */
  /* Creado por: sa en */
  /* Fecha: 12-Dec-1996 */
  if @rowcount!=0
  begin
    /* Revisar porque el error_message regresa nulo */
    if update(cod_contab) or update(cod_rubro)
    begin
      /* Verificar que el rubro asignado es de detalle*/
      if ( select count(*) from pre_rubro r, inserted i
          where r.cod_contab=i.cod_contab and
                r.cod_rubro=i.cod_rubro and
                r.detalle = 1) != @rowcount

      begin
        set rowcount 1
        select @error_message= "No es posible asignarle Cuentas contables al
        rubro: " + convert(varchar, inserted.cod_contab + " - " +
        inserted.cod_rubro) + " porque no es de detalle!"
        from inserted, pre_rubro
        where inserted.cod_contab=pre_rubro.cod_contab and
              inserted.cod_rubro=pre_rubro.cod_rubro and
              pre_rubro.detalle!=1

        set rowcount 0
        raiserror 40000 @error_message
        rollback trigger
        return
      end
    end
  end

  /* Verificar que el rubro asignado controla solamente una cuenta */
  if exists(select count(*) from pre_rubro_cuenta rc, inserted i
           where rc.cod_contab=i.cod_contab and rc.cod_cuenta=i.cod_cuenta
  )

```

```
group by rc.cod_contab,rc.cod_cuenta having count(*)>1 )
```

```
begin
  set rowcount 1
  select @error_message = "Se esta asignando una cuenta, que ya habia sido
  asignada a otro rubro!"
  set rowcount 0
  raiserror 40000 @error_message
  rollback trigger
  return
end
end
```

Tipo:	Complemento Después del Trigger de Inserción de cop_partdet_2
Nombre:	trins::cop_partdet_2
Código:	COP_TR05

```

/* luego del trins de cop_partdet_2 */
if @rowcount != 0
begin
    /*Hacer con Cursores, solo inserto una fila de las afectadas!*/
    if update(cod_cuenta)
    begin
        /* Crear los registros en Saldos, cuando no existan aun */
        if (select count(*) from inserted i, cop_partenc_2 e, cop_saldo s
            where i.cod_contab=e.cod_contab and
                  i.num_corr_ejerc=e.num_corr_ejerc and
                  e.periodo=s.periodo and
                  e.ejercicio=s.ejercicio and
                  i.cod_cuenta=s.cod_cuenta) != @rowcount
        begin
            insert into
            cop_saldo(cod_contab,periodo,ejercicio,cod_cuenta,abonos,cargos,abonos_cons,cargos_cons,cerrado)
            select
            i.cod_contab,e.periodo,e.ejercicio,i.cod_cuenta,0.00,0.00,0.00,0.00,0
            from inserted i, cop_partenc_2 e
            where i.cod_contab=e.cod_contab and
                  i.num_corr_ejerc=e.num_corr_ejerc and
                  not exists (select *
                              from cop_saldo s
                              where e.periodo = s.periodo and
                                     e.ejercicio = s.ejercicio and
                                     i.cod_cuenta = s.cod_cuenta)
        end
    end
end
/* Actualizar los Saldos */
if update(abonos) or update(cargos)
begin
    update cop_saldo
    set     cop_saldo.abonos=cop_saldo.abonos + (select sum(i.abonos)
                                                  from inserted i,cop_partenc_2 e
                                                  where i.cod_contab=e.cod_contab and
                                                        i.num_corr_ejerc=e.num_corr_ejerc and
                                                        cop_saldo.cod_contab=i.cod_contab and
                                                        cop_saldo.cod_cuenta=i.cod_cuenta and

```

```
cop_saldo.ejercicio=e.ejercicio and  
cop_saldo.periodo=e.periodo ),
```

```
cop_saldo.cargos=cop_saldo.cargos + (select sum(i.cargos)  
from inserted i,cop_partenc_2 e  
where i.cod_contab=e.cod_contab and  
i.num_corr_ejerc=e.num_corr_ejerc and  
cop_saldo.cod_contab=i.cod_contab and  
cop_saldo.cod_cuenta=i.cod_cuenta and  
cop_saldo.ejercicio=e.ejercicio and  
cop_saldo.periodo=e.periodo)
```

```
from inserted i,cop_partenc_2 e  
where i.cod_contab=e.cod_contab and  
i.num_corr_ejerc=e.num_corr_ejerc and  
cop_saldo.cod_contab=i.cod_contab and  
cop_saldo.cod_cuenta=i.cod_cuenta and  
cop_saldo.ejercicio=e.ejercicio and  
cop_saldo.periodo=e.periodo
```

```
end  
end
```

Tipo:	Complemento Después del Trigger de Inserción de cop_saldo
Nombre:	trins::cop_saldo
Código:	COP_TR06

```
/*Trigger de Insercion-Actualizacion de cop_saldo (Mayorizacion)*/
```

```
/*Si se afectaron filas*/
```

```
if @rowcount!=0
```

```
begin /* Si se afectaron filas */
```

```
    declare cur_inserted cursor
```

```
    for
```

```
    select
```

```
cod_contab,cod_cuenta,periodo,ejercicio,abonos,cargos,cargos_cons,abonos_cons,cerrado
```

```
from inserted
```

```
where cargos<>0.00 or
```

```
    abonos<>0.00 or
```

```
    cargos_cons<>0.00 or
```

```
    abonos_cons<>0.00
```

```
/*Variables para el Cursor*/
```

```
declare @v_cod_contab char(5)
```

```
declare @v_cod_cuenta char(30)
```

```
declare @v_ejercicio int
```

```
declare @v_periodo int
```

```
declare @v_cargos money
```

```
declare @v_abonos money
```

```
declare @v_cargos_cons money
```

```
declare @v_abonos_cons money
```

```
declare @v_cerrado bit
```

```
/*Variables para valores antes del cambio*/
```

```
declare @v_abonos_ant money
```

```
declare @v_cargos_ant money
```

```
/*Variables del Catalogo de Cuentas*/
```

```
declare @v_cod_cuenta_p char(30)
```

```
declare @v_cod_contab_c char(5)
```

```
declare @v_cod_cuenta_c varchar(30)
```

```
/*Variables Temporales*/
```

```
declare @v_total_cargos money
```

```
declare @v_total_abonos money
```

```
declare @v_total_cargos_cons money
```

```
declare @v_total_abonos_cons money
```

```
/*Variables para actualizacion de Presupuesto*/
```

```

declare @v_cod_rubro varchar(30)
declare @v_saldo_real money
declare @v_saldo_real_cons money

```

```

if ( select count(*)
      from inserted i,
           cop_cuenta_contab c,
           cop_contabilidad co
      where c.cod_contab=i.cod_contab and
            c.cod_cuenta=i.cod_cuenta and
            c.cod_contab=co.cod_contab and
            co.ejercicio=i.ejercicio and
            co.periodo=i.periodo ) = @rowcount
begin /* Si las Insertadas estan Bien */
  open cur_inserted
  while (@@sqlstatus=0)
  begin /*Mientras no sea fin del cursor*/

    fetch cur_inserted into
    @v_cod_contab,@v_cod_cuenta,@v_periodo,@v_ejercicio,@v_abonos
    ,@v_cargos,@v_cargos_cons,@v_abonos_cons,@v_cerrado
    if (@@sqlstatus=0)
    begin /*Si no es fin de cursor*/
      /*Determinar cuenta padre y de consolidacion*/
      select @v_cod_cuenta_p=rtrim(cod_cuenta_p),
             @v_cod_contab_c=rtrim(cod_contab_c),
             @v_cod_cuenta_c=rtrim(cod_cuenta_c)
      from cop_cuenta_contab
      where cod_contab=@v_cod_contab and
            cod_cuenta=@v_cod_cuenta
      /*Determinar los Cargos y Abonos que tenia antes de la
actualizacion*/

      select @v_abonos_ant=isnull(abonos,0.00),
             @v_cargos_ant=isnull(cargos,0.00)
      from deleted
      where cod_contab=@v_cod_contab and
            cod_cuenta=@v_cod_cuenta and
            ejercicio=@v_ejercicio and
            periodo=@v_periodo
      if @v_abonos_ant is null select @v_abonos_ant=0.00
      if @v_cargos_ant is null select @v_cargos_ant=0.00
      /*Determinar al Rubro de Presupuesto que esta asociada la
cuenta*/

      select @v_cod_rubro=cod_rubro
      from pre_rubro_cuenta

```

```

where cod_contab=@v_cod_contab and
      cod_cuenta=@v_cod_cuenta
if (@@rowcount>1)
begin /*Si hay mas de un rubro asociado a la cuenta*/
  select @error_message="ERROR: Hay mas de un rubro
asociado a la cuenta: "+@v_cod_contab+" :
"+@v_cod_cuenta
  set rowcount 0
  raiserror 40000 @error_message
  rollback trigger
  return
end /*Si hay mas de un rubro asociado a la cuenta*/

if update(cargos)
begin /*Si Se Actualizo Cargos*/
  if (@v_cod_cuenta_p is not null)
  begin /*Si hay Cuenta padre local, actualizarla*/
    /*Calcular Total de Cargos para cuenta Padre*/
    select
@v_total_cargos=isnull(sum(s.cargos),0.00)
    from cop_saldo s, cop_cuenta_contab c
    where s.cod_contab=c.cod_contab and
          s.cod_cuenta=c.cod_cuenta and
          c.cod_contab=@v_cod_contab and
          c.cod_cuenta_p=@v_cod_cuenta_p

    if not exists(select * from cop_saldo
                  where cod_contab=@v_cod_contab and
                        cod_cuenta=@v_cod_cuenta_p and
                        ejercicio=@v_ejercicio and
                        periodo=@v_periodo)
    begin /* Si no existe el registro de saldo para la
cuenta padre*/
      set self_recursion on
      insert into
cop_saldo(cod_contab,cod_cuenta,ejercicio,period
o,abonos,cargos,abonos_cons,cargos_cons,cerrado
)
values
(@v_cod_contab,@v_cod_cuenta_p,@v_ejercicio
,
@v_periodo,@v_abonos,@v_total_cargos,
@v_abonos_cons,@v_cargos_cons,@v_cerrado)
    end
  else
  begin

```

```

set self_recursion on
update cop_saldo
set   cargos=@v_total_cargos
where cod_contab=@v_cod_contab and
      cod_cuenta=@v_cod_cuenta_p and
      ejercicio=@v_ejercicio and
      periodo=@v_periodo
end /*Fin si no existe registro de saldo para cuenta
padre*/
end /* Si hay cuenta padre local*/

if (@v_cod_contab_c is not null and @v_cod_cuenta_c is
not
null)
Actualizarla*/
begin /*Si hay Cuenta de Consolidacion Padre,

/*Calcular total de Cargos Consolidados para
cuenta padre remota*/
select
@v_total_cargos_cons=isnull(sum(s.cargos),0.00)
from cop_saldo s, cop_cuenta_contab c
where s.cod_contab=c.cod_contab and
      s.cod_cuenta=c.cod_cuenta and
      c.cod_contab_c=@v_cod_contab_c and
      c.cod_cuenta_c=@v_cod_cuenta_c
if not exists(select * from cop_saldo
where cod_contab=@v_cod_contab_c and
      cod_cuenta=@v_cod_cuenta_c and
      ejercicio=@v_ejercicio and
      periodo=@v_periodo)
begin /* Si no existe registro de saldo de cuenta
padre remota, crearlo*/
set self_recursion on
insert into

cop_saldo(cod_contab,cod_cuenta,
ejercicio,periodo,abonos,cargos,abonos_co
ns,
cargos_cons,cerrado)
values
(@v_cod_contab_c,@v_cod_cuenta_c,@v
_ejercicio,@v_periodo,@v_abonos,@v_ca
rgos,@v_abonos_cons,@v_total_cargos_c
ons,@v_cerrado)

end
else
begin
set self_recursion on

```

```

update cop_saldo
set   cargos_cons=cargos +
      @v_total_cargos_cons
where cod_contab=@v_cod_contab_c and
      cod_cuenta=@v_cod_cuenta_c and
      ejercicio=@v_ejercicio and
      periodo=@v_periodo
end /*Fin si no existe registro de saldo cuenta
padre remota*/
end /* Fin si hay cuenta de consolidacion Padre*/

/*Actualizar el consolidado de (esta) cuenta*/
set self_recursion on
update cop_saldo
set   cargos_cons=cargos_cons + (@v_cargos -
      @v_cargos_ant)
where cod_contab=@v_cod_contab and
      cod_cuenta=@v_cod_cuenta and
      ejercicio=@v_ejercicio and
      periodo=@v_periodo
end /* Fin Actualizacion Cargos*/

if update(abonos)
begin /*Si Se Actualizo Abonos*/
  if (@v_cod_cuenta_p is not null)
  begin /*Si hay Cuenta padre local, actualizarla*/
    /*Calcular Total de Cargos para cuenta Padre*/
    select
@v_total_abonos=isnull(sum(s.abonos),0.00)
    from   cop_saldo s, cop_cuenta_contab c
    where  s.cod_contab=c.cod_contab and
          s.cod_cuenta=c.cod_cuenta and
          c.cod_contab=@v_cod_contab and
          c.cod_cuenta_p=@v_cod_cuenta_p

    if not exists(select * from cop_saldo
where cod_contab=@v_cod_contab and
      cod_cuenta=@v_cod_cuenta_p and
      ejercicio=@v_ejercicio and
      periodo=@v_periodo)
begin /* Si no existe el registro de saldo para la
cuenta padre*/
  set self_recursion on
  insert into
  cop_saldo(cod_contab,cod_cuenta,ejercici

```

```

o,periodo,abonos,cargos,abonos_cons,carg
os_cons,cerrado)
values
(@v_cod_contab,@v_cod_cuenta_p,
@v_ejercicio,@v_periodo,@v_total_abon
os,@v_cargos,@v_abonos_cons,
@v_cargos_cons,@v_cerrado)
end
else
begin
set self_recursion on
update cop_saldo
set abonos=@v_total_abonos
where cod_contab=@v_cod_contab and
cod_cuenta=@v_cod_cuenta_p and
ejercicio=@v_ejercicio and
periodo=@v_periodo
end /*Fin si no existe registro de saldo para cuenta
padre*/
end /* Si hay cuenta padre local*/

if (@v_cod_contab_c is not null and @v_cod_cuenta_c is
not null)
begin /*Si hay Cuenta de Consolidacion Padre,

/*Calcular total de Abonos Consolidados para
cuenta padre remota*/
select
@v_total_abonos_cons=isnull(sum(s.abonos),0.00
)
)
from cop_saldo s, cop_cuenta_contab c
where s.cod_contab=c.cod_contab and
s.cod_cuenta=c.cod_cuenta and
c.cod_contab_c=@v_cod_contab_c and
c.cod_cuenta_c=@v_cod_cuenta_c
if not exists(select * from cop_saldo
where cod_contab=@v_cod_contab_c and
cod_cuenta=@v_cod_cuenta_c and
ejercicio=@v_ejercicio and
periodo=@v_periodo)
begin /*Si no existe registro de saldo de cuenta
padre remota, crearlo*/
set self_recursion on
insert into
cop_saldo(cod_contab,cod_cuenta,ejercici
o,

```

Actualizarla*/

```

        periodo,abonos,cargos,abonos_cons,
        cargos_cons,cerrado)
        values
        (@v_cod_contab_c,@v_cod_cuenta_c,
        @v_ejercicio,@v_periodo,@v_abonos,
        @v_cargos,@v_total_abonos_cons,
        @v_cargos_cons,@v_cerrado)
    end
    else
    begin
        set self_recursion on
        update cop_saldo
        set    abonos_cons=abonos +
            @v_total_abonos_cons
        where cod_contab=@v_cod_contab_c and
            cod_cuenta=@v_cod_cuenta_c and
            ejercicio=@v_ejercicio and
            periodo=@v_periodo
        end /*Fin si no existe registro de saldo cuenta
        padre remota*/
    end /* Fin si hay cuenta de consolidacion Padre*/

    /*Actualizar el consolidado de (esta) cuenta*/
    set self_recursion on
    update cop_saldo
    set    abonos_cons=abonos_cons + (@v_abonos -
        @v_abonos_ant)
    where cod_contab=@v_cod_contab and
        cod_cuenta=@v_cod_cuenta and
        ejercicio=@v_ejercicio and
        periodo=@v_periodo

    end /*Fin Actualizacion Abonos*/

    if update(cargos_cons)
    begin /*Si Se Actualizo Cargos Consolidados*/
        if (@v_cod_cuenta_p is not null)
        begin /*Si hay Cuenta padre local, actualizar cargos
        consolidados*/
            /*Calcular Total de Cargos Consolidados para
            cuenta Padre local*/
            select
            @v_total_cargos_cons=isnull(sum(s.cargos_cons),0.00)
            from  cop_saldo s, cop_cuenta_contab c
            where s.cod_contab=c.cod_contab and
                s.cod_cuenta=c.cod_cuenta and

```

```

c.cod_contab=@v_cod_contab and
c.cod_cuenta_p=@v_cod_cuenta_p

if not exists(select * from cop_saldo
              where cod_contab=@v_cod_contab and
                    cod_cuenta=@v_cod_cuenta_p and
                    ejercicio=@v_ejercicio and
                    periodo=@v_periodo)
begin /*Si no existe el registro de saldo para la
cuenta padre*/
  set self_recursion on
  insert into
  cop_saldo(cod_contab,cod_cuenta,ejercicio,
            periodo,abonos,cargos,abonos_cons,
            cargos_cons,cerrado)
  values
  (@v_cod_contab,@v_cod_cuenta_p,
  @v_ejercicio,@v_periodo,@v_abonos,
  @v_cargos,@v_abonos_cons,
  @v_total_cargos_cons,@v_cerrado)
end
else
begin
  set self_recursion on
  update cop_saldo
  set
cargos_cons=@v_total_cargos_cons
  where cod_contab=@v_cod_contab and
        cod_cuenta=@v_cod_cuenta_p and
        ejercicio=@v_ejercicio and
        periodo=@v_periodo
end /*Fin si no existe registro de saldo para cuenta
padre*/
end /* Si hay cuenta padre local*/
end /*Fin Actualizacion Cargos Consolidados*/

if update(abonos_cons)
begin /*Si Se Actualizo Abonos Consolidados*/
  if (@v_cod_cuenta_p is not null)
  begin /*Si hay Cuenta padre local, actualizar cargos
consolidados*/
    /*Calcular Total de Abonos Consolidados para
cuenta Padre local*/
    select
    @v_total_abonos_cons=isnull(sum(s.abonos_cons),0.00)

```

```

from cop_saldo s, cop_cuenta_contab c
where s.cod_contab=c.cod_contab and
      s.cod_cuenta=c.cod_cuenta and
      c.cod_contab=@v_cod_contab and
      c.cod_cuenta_p=@v_cod_cuenta_p

if not exists(select * from cop_saldo
              where cod_contab=@v_cod_contab and
                    cod_cuenta=@v_cod_cuenta_p and
                    ejercicio=@v_ejercicio and
                    periodo=@v_periodo)
begin /*Si no existe el registro de saldo para la
cuenta
padre*/
    set self_recursion on
    insert into
    cop_saldo(cod_contab,cod_cuenta,ejercicio,
    periodo,abonos,cargos,abonos_cons,
    cargos_cons,cerrado)
    values
    (@v_cod_contab,@v_cod_cuenta_p,
    @v_ejercicio,@v_periodo,@v_abonos,
    @v_cargos,@v_total_abonos_cons,
    @v_cargos_cons,@v_cerrado)
end
else
begin
    set self_recursion on
    update cop_saldo
    set
    abonos_cons=@v_total_abonos_cons
    where cod_contab=@v_cod_contab and
          cod_cuenta=@v_cod_cuenta_p and
          ejercicio=@v_ejercicio and
          periodo=@v_periodo
end /*Fin si no existe registro de saldo para cuenta
padre*/
end /* Si hay cuenta padre local*/
end /*Fin Actualizacion Abonos Consolidados*/

if update(cargos) or update(abonos) or update(cargos_cons) or
update(abonos_cons)
begin /*Si se ha actualizado cualquier monto*/
    if (@v_cod_rubro is not null)

```

Presupuesto*/

begin /*Si hay rubro relacionado, Actualizar

```
select @v_saldo_real=0.00,
       @v_saldo_real_cons=0.00
/*Calcular suma de cargos y abonos para cuentas
de saldo debito c-a*/
```

```
select @v_total_cargos=isnull(sum(s.cargos),0.00),
       @v_total_abonos=isnull(sum(s.abonos),0.00),
       @v_total_cargos_cons=isnull(sum(s.cargos_cons),0.00),
       @v_total_abonos_cons=isnull(sum(s.abonos_cons),0.00)
from   cop_saldo s, cop_cuenta_contab c,
       cop_t_cuenta t,
```

```
pre_rubro_cuenta r
where  s.cod_contab=c.cod_contab and
       s.cod_cuenta=c.cod_cuenta and
       c.cod_contab=r.cod_contab and
       c.cod_cuenta=r.cod_cuenta and
       c.cod_t_cuenta=t.cod_t_cuenta and
       t.tipoc="D" and
       r.cod_rubro=@v_cod_rubro
```

((@v_total_cargos -

```
select @v_saldo_real=@v_saldo_real +
       @v_total_abonos),
       @v_saldo_real_cons=@v_saldo_real_cons +
       (@v_total_cargos_cons - @v_total_abonos_cons)
```

de

```
/*Calcular suma de cargos y abonos para cuentas
saldo credito a-c*/
```

```
select
@v_total_cargos=isnull(sum(s.cargos),0.00),
       @v_total_abonos=isnull(sum(s.abonos),0.00),
```

```
@v_total_cargos_cons=isnull(sum(s.cargos_cons),
                               0.00),
```

```
@v_total_abonos_cons=isnull(sum(s.abonos_cons),
                              0.00)
```

```
from   cop_saldo s, cop_cuenta_contab c,
       cop_t_cuenta t, pre_rubro_cuenta r
where  s.cod_contab=c.cod_contab and
       s.cod_cuenta=c.cod_cuenta and
       c.cod_contab=r.cod_contab and
```

```

c.cod_cuenta=r.cod_cuenta and
c.cod_t_cuenta=t.cod_t_cuenta and
t.tipoc="C" and
r.cod_rubro=@v_cod_rubro

```

```

select @v_saldo_real=@v_saldo_real +
(@v_total_abonos - @v_total_cargos),
@v_saldo_real_cons=@v_saldo_real_cons

(@v_total_abonos_cons - @v_total_cargos_cons)

```

```

/* Actualizar Saldos de Presupuesto */
if not exists(select * from pre_saldo
where cod_contab=@v_cod_contab and
cod_rubro=@v_cod_rubro and
ejercicio=@v_ejercicio and
periodo=@v_periodo)

```

```

begin /*Si no existe el registro de saldo para el

```

```

presupuesto, crearlo*/
set self_recursion on
insert into
pre_saldo(cod_contab,cod_rubro,ejercicio,
periodo,saldo_presup,saldo_real,
saldo_real_cons)
values (@v_cod_contab,@v_cod_rubro,
@v_ejercicio,@v_periodo,0.00,
@v_saldo_real,@v_saldo_real_cons)

```

```

end
else
begin

```

```

set self_recursion on
update pre_saldo
set saldo_real=@v_saldo_real,
saldo_real_cons=@v_saldo_real_cons
where cod_contab=@v_cod_contab and
cod_rubro=@v_cod_rubro and
ejercicio=@v_ejercicio and
periodo=@v_periodo

```

```

end /*Fin si no existe registro de saldo para rubro
presupuesto*/

```

```

end /*Si existe rubro que actualizar*/

```

```

end /*Fin Actualizacion del Presupuesto*/

```

+

rubro de

```
        end /*Fin si no se ha terminado de leer cursor*/

        end /*Fin While cursor*/
        close cur_inserted
    end
    else
    begin
        select @error_message="Se esta intentando actualizar saldos de Periodos
Incorrectos o Hay
        un error en el Catalogo de Cuentas!"
        set rowcount 0
        raiserror 40000 @error_message
        rollback trigger
        return
    end /* Fin Si todas las insertadas estan bien */
    deallocate cursor cur_inserted
end /*Fin Si se afectaron filas*/
```

Tipo:	Complemento Después del Trigger de Inserción de pre_rubro
Nombre:	trins::pre_rubro
Código:	PRETR02

```
/* luego del trins de pre_rubro */
```

```
/* Creado por: sa en */
```

```
/* Fecha: 12-Dec-1996 */
```

```
if (@rowcount != 0)
```

```
begin
```

```
if update(detalle)
```

```
begin
```

```
if exists(select * from inserted i, pre_rubro r
          where r.cod_contab = i.cod_contab and
                r.cod_rubro_p = i.cod_rubro and
                r.cod_rubro_p is not null and
                i.detalle = 1)
```

```
begin
```

```
set rowcount 1
```

```
select @error_message = "El Rubro: " + convert(varchar,i.cod_rubro) +
```

```
detalle!"
```

```
from inserted i, pre_rubro r
```

```
where r.cod_contab = i.cod_contab and
```

```
      r.cod_rubro_p = i.cod_rubro and
```

```
      r.cod_rubro_p is not null and
```

```
      i.detalle = 1
```

```
set rowcount 0
```

```
raiserror 40000 @error_message
```

```
rollback trigger
```

```
return
```

```
end
```

```
else
```

```
begin
```

```
update pre_rubro
```

```
set detalle = 0
```

```
from inserted
```

```
where pre_rubro.cod_contab=inserted.cod_contab and
```

```
      pre_rubro.cod_rubro=inserted.cod_rubro_p and
```

```
      inserted.detalle = 1
```

```
end
```

```
end
```

```
end
```

" NO es de


```

s.periodo=@v_periodo and
r.cod_contab=@v_cod_contab and
r.cod_rubro_p=@v_cod_rubro_p
if not exists(select * from pre_saldo
              where cod_contab=@v_cod_contab and
                    cod_rubro=@v_cod_rubro_p and
                    ejercicio=@v_ejercicio and
                    periodo=@v_periodo)
begin
  set self_recursion on
  insert into
  pre_saldo(cod_contab,cod_rubro,ejercicio,periodo
            ,
            saldo_presup,saldo_real,saldo_real_cons)
  values
  (@v_cod_contab,@v_cod_rubro_p,@v_ejercicio,
  @v_periodo,@v_total_presup,@v_saldo_real,
  @v_saldo_real_cons)
end
else
begin
  set self_recursion on
  update pre_saldo
  set   saldo_presup=@v_total_presup
  where cod_contab=@v_cod_contab and
        cod_rubro=@v_cod_rubro_p and
        ejercicio=@v_ejercicio and
        periodo=@v_periodo
end
end
end
if update(saldo_real)
begin
  if @v_cod_rubro_p is not null
  begin
    select @v_total_real=isnull(sum(s.saldo_real),0.00)
    from   pre_saldo s, pre_rubro r
    where  s.cod_contab=r.cod_contab and
          s.cod_rubro=r.cod_rubro and
          s.ejercicio=@v_ejercicio and
          s.periodo=@v_periodo and
          r.cod_contab=@v_cod_contab and
          r.cod_rubro_p=@v_cod_rubro_p
    if not exists(select * from pre_saldo
                  where cod_contab=@v_cod_contab and
                        cod_rubro=@v_cod_rubro_p and

```

```

                                ejercicio=@v_ejercicio and
                                periodo=@v_periodo)
begin
    set self_recursion on
    insert into
    pre_saldo(cod_contab,cod_rubro,ejercicio,periodo
    ,
    saldo_presup,saldo_real,saldo_real_cons)
    values
    (@v_cod_contab,@v_cod_rubro_p,@v_ejercicio,
    @v_periodo,@v_saldo_presup,@v_total_real,
    @v_saldo_real_cons)
end
else
begin
    set self_recursion on
    update pre_saldo
    set    saldo_real=@v_total_real
    where cod_contab=@v_cod_contab and
           cod_rubro=@v_cod_rubro_p and
           ejercicio=@v_ejercicio and
           periodo=@v_periodo
end
end
end
if update(saldo_real_cons)
begin
    if @v_cod_rubro_p is not null
    begin
        select
        @v_total_real_cons=isnull(sum(s.saldo_real_cons),0.00)
        from  pre_saldo s, pre_rubro r
        where s.cod_contab=r.cod_contab and
              s.cod_rubro=r.cod_rubro and
              s.ejercicio=@v_ejercicio and
              s.periodo=@v_periodo and
              r.cod_contab=@v_cod_contab and
              r.cod_rubro_p=@v_cod_rubro_p
        if not exists(select * from pre_saldo
                      where cod_contab=@v_cod_contab and
                             cod_rubro=@v_cod_rubro_p and
                             ejercicio=@v_ejercicio and
                             periodo=@v_periodo)
        begin
            set self_recursion on

```

```

insert into
pre_saldo(cod_contab,cod_rubro,ejercicio,periodo
,
saldo_presup,saldo_real,saldo_real_cons)
values
(@v_cod_contab,@v_cod_rubro_p,@v_ejercicio,
@v_periodo,@v_saldo_presup,@v_saldo_real,
@v_total_real_cons)
end
else
begin
set self_recursion on
update pre_saldo
set saldo_real_cons=@v_total_real_cons
where cod_contab=@v_cod_contab and
cod_rubro=@v_cod_rubro_p and
ejercicio=@v_ejercicio and
periodo=@v_periodo
end
end
end
end
end
close cur_inserted
deallocate cursor cur_inserted
end /*Fin si se afectaron filas*/

```

Tipo:	Complemento Antes del Trigger de Inserción de cop_periodo
Nombre:	trinscop_periodo
Código:	COP_TR09

```
/* antes del trins de cop_periodo */
```

```
if @rowcount != 0 begin
```

```
/* Validar el rango de fechas especificado */
```

```
if update(fecha_ini) or update(fecha_fin) begin
```

```
if exists(select * from inserted where fecha_ini > fecha_fin) begin
```

```
select @error_message = "Rango de fechas inválido. Las fechas estan  
invertidas."
```

```
set rowcount 0
```

```
raiserror 40000 @error_message
```

```
rollback transaction
```

```
return
```

```
end
```

```
end
```

```
/* Validar que el rango de fechas no se traslape con otro existente */
```

```
if update(fecha_ini) or update(fecha_fin) begin
```

```
if exists(select * from cop_periodo, inserted
```

```
where ((cop_periodo.fecha_ini between inserted.fecha_ini and  
inserted.fecha_fin) or
```

```
(cop_periodo.fecha_fin between inserted.fecha_ini and
```

```
inserted.fecha_fin) or
```

```
(inserted.fecha_ini between cop_periodo.fecha_ini and
```

```
cop_periodo.fecha_fin) or
```

```
(inserted.fecha_fin between cop_periodo.fecha_ini and
```

```
cop_periodo.fecha_fin))
```

```
and convert(varchar, inserted.ejercicio) + convert(varchar,  
inserted.periodo) <>
```

```
convert(varchar, cop_periodo.ejercicio) + convert(varchar,  
cop_periodo.periodo))
```

```
begin
```

```

set rowcount 1
select @error_message = "El rango de fechas definido para el Periodo " +
                        convert(varchar, inserted.periodo) +
                        " y Ejercicio " +
                        convert(varchar, inserted.ejercicio) +
                        " se traslapa con el de otro periodo ya existente!"
from inserted
where exists(select * from cop_periodo, inserted
            where ((cop_periodo.fecha_ini between inserted.fecha_ini and
                    inserted.fecha_fin) or
                    (inserted.fecha_ini between cop_periodo.fecha_ini and
                    cop_periodo.fecha_fin) or
                    (inserted.fecha_fin between cop_periodo.fecha_ini and
                    cop_periodo.fecha_fin))
            and convert(varchar, inserted.ejercicio) + convert(varchar,
                    inserted.periodo) <>
            convert(varchar, cop_periodo.ejercicio) + convert(varchar,
                    cop_periodo.periodo))

if @error_message is null begin
    select @error_message = "Error: mensaje nulo"
end
set rowcount 0

raiserror 40000 @error_message

rollback transaction

return

end
end
end

```

```

begin
  if exists (select * from inserted i
            where i.cod_cuenta_p is not null and
                  (i.abonos<>0 or i.cargos<>0 or i.abonos_anterior<>0 or
                   i.cargos_anterior<>0))
  begin
    open cur_salpnom
    while (@@sqlstatus = 0)
    begin
      fetch cur_salpnom
      into  @v_fecha_saldo, @v_cod_contab, @v_cod_cuenta,
           @v_cod_cuenta_p, @v_es_detalle, @v_abonos,
           @v_abonos_anterior, @v_cargos_anterior,
           @v_cargos,
           @v_cod_t_cuenta

      if (@@sqlstatus=0)
      begin

        /* Sumar cargos y abonos para todas las cuentas
           cuya cuenta padre es la recién encontrada */

        if @v_cod_cuenta_p is not null
        begin

          select  @v_total_cargos=sum(s.cargos),
                  @v_total_abonos=sum(s.abonos)
          from    cop_saldo_pnom s
          where  s.cod_contab=@v_cod_contab and
                 s.cod_cuenta_p=@v_cod_cuenta_p and
                 s.fecha_saldo=@v_fecha_saldo

          select  @v_tot_cargos_ant=0, @v_tot_abonos_ant=0

          select  @v_tot_cargos_ant=sum(s.cargos_anterior),
                  @v_tot_abonos_ant=sum(s.abonos_anterior)
          from    cop_saldo_pnom s
          where  s.cod_contab=@v_cod_contab and
                 s.cod_cuenta_p=@v_cod_cuenta_p and
                 s.fecha_saldo=@v_fecha_saldo

          set self_recursion on

          /* Actualizar el Saldo */
          update cop_saldo_pnom
          set    abonos=@v_total_abonos,

```